European Community on
Computational Methods in
Applied Sciences

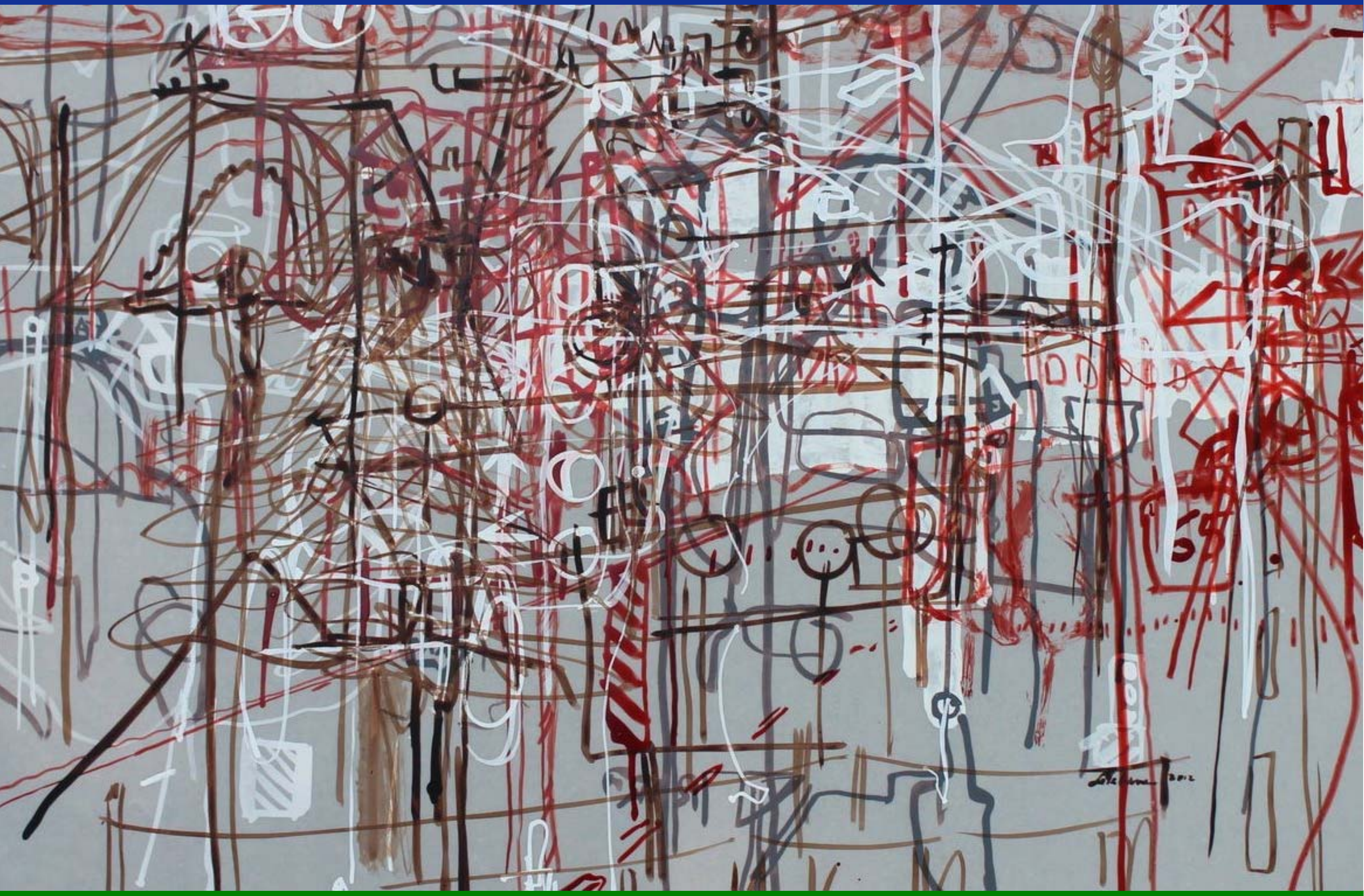
ECCOMAS
Thematic Conference

UNCECOMP 2019

3rd International Conference on Uncertainty Quantification in Computational Sciences and Engineering

PROCEEDINGS

M. Papadrakakis, V. Papadopoulos, G. Stefanou (Eds)



An IACM Special Interest Conference

UNCECOMP 2019
Uncertainty Quantification in Computational Sciences and Engineering

Proceedings of the 3rd International Conference on Uncertainty
Quantification in Computational Sciences and Engineering
Held in Crete, Greece
24-26 June 2019

Edited by:

M. Papadrakakis
National Technical University of Athens, Greece

V. Papadopoulos
National Technical University of Athens, Greece

G. Stefanou
Aristotle University of Thessaloniki, Greece

A publication of:

Institute of Structural Analysis and Antiseismic Research
School of Civil Engineering
National Technical University of Athens (NTUA)
Greece

UNCECOMP 2019
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (Eds)

First Edition, October 2019

ISBN: **978-618-82844-9-4**

PREFACE

This volume contains the full-length papers presented in the 3rd International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP 2019) that was held on June 24-26, 2019 in Crete, Greece.

UNCECOMP 2019 is also a Thematic Conference of ECCOMAS, with the objective to reflect the recent research progress in the field of analysis and design of engineering systems under uncertainty, with emphasis in multiscale simulations. The aim of the conference is to enhance the knowledge of researchers in stochastic methods and the associated computational tools for obtaining reliable predictions of the behavior of complex systems. The UNCECOMP conference series, held in conjunction with the COMPDYN conferences, gives the opportunity to the participants to interact with the Computational Dynamics community for their mutual benefit.

The UNCECOMP 2019 Conference is supported by the National Technical University of Athens (NTUA), the Greek Association for Computational Mechanics (GRACM).

The editors of this volume would like to thank all authors for their contributions. Special thanks go to the colleagues who contributed to the organization of the Minisymposia and to the reviewers who, with their work, contributed to the scientific quality of this e-book.

M. Papadrakakis
National Technical University of Athens, Greece

V. Papadopoulos
National Technical University of Athens, Greece

G. Stefanou
Aristotle University of Thessaloniki, Greece

ACKNOWLEDGEMENTS

The conference organizers acknowledge the support towards the organization of the “3rd International Conference on Uncertainty Quantification in Computational Sciences and Engineering”, to the following organizations:

- European Community on Computational Methods in Applied Sciences (ECCOMAS)
- School of Civil Engineering, National University of Athens (NTUA)
- Greek Association for Computational Mechanics (GRACM)
- Hellenic Republic – Region of Crete
- Municipality of Herakion

Plenary Speakers and Invited Session Organizers

We would also like to thank the Plenary and Semi-Plenary Speakers and the Minisymposia Organizers for their help in the setting up of a high standard Scientific Programme.

Plenary Speakers: Isaac Elishakoff, Charbel Farhat, Roger Ghanem, George Karniadakis, Christian Soize, Alexander Vakakis

Semi-Plenary Speakers: Hermann G. Matthies, Xavier Oliver, Costas Papadimitriou, Michael D. Shields, Daniel Straub, Bruno Sudret

MS Organizers: Michael Beer, Jean-Marc Bourinet, Matteo Broggi, Eleni Chatzi, Alice Cicirello, Vasilis Dertimanis, Matthias Faes, Dimitrios Giovanis, Wolfgang Graf, Michael Hanss, Dionissis Hristopoulos, Michael Kaliske, Anna Kucerova, Sigrid Leyendecker, Alexander Litvinenko, Sankaran Mahadevan, Stefano Marelli, David Moens, Costas Papadimitriou, Vissarion Papadopoulos, Manolis Papadrakakis, Iason Papaioannou, Edoardo Patelli, Nicola Pedroni, Paris Perdikaris, Dirk Pflüger, Dmytro Pivovarov, Stefanie Reese, Bojana Rosic, Benjamin Sanderse, Dimitrios Savvas, Kheirollah Sepahvand, Michael Shields, Giovanni Stabile, George Stavroulakis, George Stefanou, Paul Steinmann, Daniel Straub, Bruno Sudret, Alex Taflanidis, I.C. Tsantili, Laurent van den Bos, Yous van Halder, Wolfgang Wall

SUMMARY

Minisymposia

Regular Sessions

CONTENTS

Minisymposia

MS 1: COMPUTATIONAL MULTISCALE MODELLING UNDER UNCERTAINTY

MS 2: BAYESIAN ANALYSIS OF NUMERICAL MODELS

MS 3: UNCERTAINTY QUANTIFICATION IN VIBRATION BASED MONITORING AND STRUCTURAL DYNAMICS SIMULATIONS

MS 4: INVERSE METHODS FOR UNCERTAINTY QUANTIFICATION INLARGE-SCALE APPLICATIONS

MS 5: SURROGATE AND REDUCED-ORDER MODELING FOR STOCHASTIC SIMULATION OF PHYSICAL SYSTEMS

MS 6: MINISYMPOSIUM IN HONOR OF PROF. HERMANN MATTHIES: "UNCERTAINTY COMPUTATIONS WITH REDUCED ORDER MODELS AND LOW-RANK REPRESENTATIONS"

MS 7: SURROGATE MODELS: BENCHMARK PROBLEMS AND SOLUTIONS

MS 9: UNCERTAINTY PROPAGATION AND QUANTIFICATION WITH COMPUTATIONALLY EXPENSIVE MODELS

MS 10: SOFTWARE FOR UNCERTAINTY QUANTIFICATION

MS 11: POLYMORPHIC UNCERTAIN DATA FOR NUMERICAL ANALYSIS AND DESIGN OF STRUCTURES

MS 12: UNCERTAINTY QUANTIFICATION USING EXPERIMENTAL DATA

MS 13: MULTISCALE ANALYSIS AND DESIGN OF RANDOM HETEROGENEOUS MEDIA

MS 15: MACHINE LEARNING APPROACHES TO UNCERTAINTY QUANTIFICATION

MS 16: FAILURE PREVENTION USING PHYSICAL-BASED MODELS AND DATA-BASED MODELS

Regular Sessions

RS 4: LARGE-SCALE STOCHASTIC FINITE ELEMENT APPLICATIONS

RS 15: STOCHASTIC FRACTURE AND DAMAGE

RS 16: SYSTEM RELIABILITY ANALYSIS, DESIGN AND RISK ASSESSMENT

RS 17: UNCERTAINTY QUANTIFICATION

RS 19: VALIDATION OF STOCHASTIC MODELING TECHNIQUES

RS 20: NUMERICAL SIMULATION METHODS FOR STOCHASTIC PROBLEMS

RS 21: MONTE CARLO SIMULATION

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

AGGREGATE CLUSTERING AND CASTING DIRECTION EFFECTS IN LATTICE DISCRETE PARTICLE MODEL SIMULATIONS

Jan Podroužek[1,2], **Marco Marcon**[1], **Jan Vorel**[1,3], **and Roman Wan-Wendner**[1,4]

[1]Christian Doppler Laboratory LiCRoFast, Department of Civil Engineering and Natural Hazards,
University of Natural Resources and Life Sciences (BOKU)
1190 Vienna, Austria
e-mail: marco.marcon@boku.ac.at

[2]Faculty of Civil Engineering, Brno University of Technology
602 00 Brno, Czech Republic
e-mail: podrouzek.j@vutbr.cz

[3]Faculty of Civil Engineering, Czech Technical University, Prague (CTU)
166 29 Praha 6, Czech Republic
e-mail: jan.vorel@fsv.cvut.cz

[4]Department of Structural Engineering, Faculty of Engineering and Architecture, Ghent University
9052 Ghent, Belgium
e-mail: roman.wanwendner@ugent.be

Keywords: Spatial Variability, Concrete, Autocorrelation length, Concrete Failure, Monte Carlo

Abstract. *The paper aims at characterizing the influence of particle placement and clustering in lattice discrete particle model (LDPM) simulations of concrete on structural response. The presented spatial variability package for LDPM enables for the first time to influence the previously independent and random particle placement. The proposed scheme correlates the particle placement to an initial random or gradient-based fields in order to mimic some of the experimentally observed phenomena such as aggregate clustering or the effect of casting direction. The study is based on high-dimensional Monte Carlo (MC) LDPM simulations of three classical concrete tests in which the inherent variability and production process are represented by the proposed particle placement schemes with varying parameters. The material property fields are kept constant at this phase of the investigation in order to isolate and quantify the potential effect of the proposed particle placement schemes on structural response. This investigation is based on a comparison of stress and strain values at peak for different tests against the case of independent and random placement. The coefficients of variation of the above-mentioned outputs are also evaluated. This research aims at evaluating the importance influencing the particle placement according to experimentally measurable phenomena before initiating research on the spatial variability of material properties and the respective correlation structure.*

1 Introduction

According to classical theories and some already outdated codes [1, 2] the nominal strength of geometrically similar structures made from non-random materials is independent of the structural size. The application of random heterogeneous media, however, such as concrete, require higher order theories [3] for a safe and economical design, especially when large or slender structures are considered. The scattering of physical experiments [4] and occasional structural failures [5], however, do not always comply with such theories, even if current probabilistic approaches are introduced. While the energetic size effect is relevant regardless if the material is heterogeneous or not there is also a statistical size effect linked to the spatial variability of heterogenous materials.

In Monte Carlo (MC) based probabilistic simulations [6], the random material properties are typically considered spatially uniform and as such are assigned to structural members or entire structures. This simplification is important for the formulation of MC sampling schemes [6, 7], required for feasible tail characterization of structural response (engineering failure probabilities).

On the other hand, neglecting the spatial variability means that the most fundamental concepts in structural reliability cannot be directly captured by simulations. Clearly, if random fields are introduced to MC based probabilistic simulations, the established sampling schemes cannot be applied, and the outcome of applying random fields is limited to answering what-if scenarios. The reason can be found in the extreme dimensionality of the problem combined with unknown sensitivity.

Recent developments in sampling schemes for spatial or temporal variability are discussed by [7, 8, 4], where a feasible sample selection strategy for spatial variability is proposed. With the introduction of random (spatially variable) fields in the MC simulations the question of autocorrelation quickly emerges as the amount of response scattering, e.g. in terms of COV (coefficient of variation) of load capacity, becomes sensitive to particular parameters of the random field. These are most typically the functional form of the autocorrelation function and its coefficients, which can be linked to a characteristic length, both mimicking the micro-structural features of a particular (composite) material and production processes [9, 10, 11, 12]. The introduction of spatial variability to discrete meso-scale simulations of concrete has been previously done by [13], who experimented with randomized material property fields and thin, nearly 2D, specimens.

This paper is unique in directly linking the micro-structural features of the random heterogeneous material to the COV of response by the investigated particle placement schemes, which are governed by the initial spatial field, while maintaining the material property fields constant. Moreover, the investigated classical concrete specimens are investigated fully in 3D, which results in a number of qualitatively different failure modes and crack paths, aside from the ability to capture the relationship between the COV of response and the particular correlation structure. In particular, gradient-based fields and random fields are introduced into the stochastic framework of the lattice discrete particle model (LDPM) to account for the inherent variability and production processes of several concrete specimen test series (Figure 4), such as cylinder and cube compression tests, and unnotched three point bending tests. As a consequence, the lattice models become sensitive to a particular particle placement concept, which is no longer independent and random [4], and the scattering of the response can, thus, be controlled and associated with the physical meaning of an auto-correlation length (6 values tested) and particular form of the spectral function (1 value tested in this paper). The aim of this contribution is

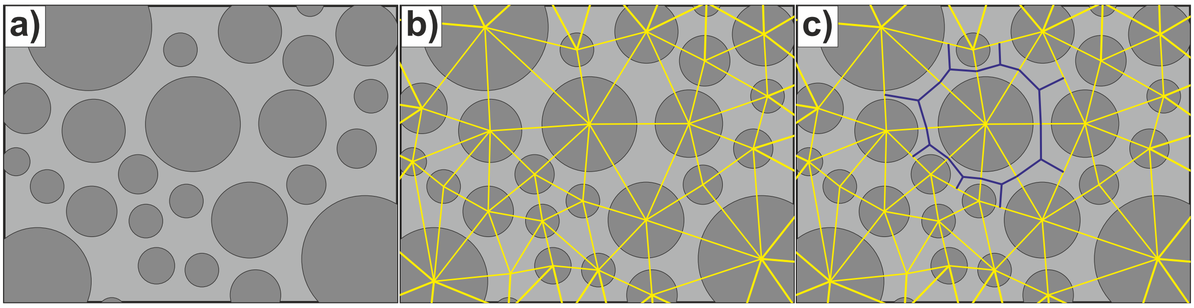


Figure 1: 2D representation of the LDPM polyhedral cell construction. a) Particle placement, b) Lattice mesh and tetrahedralisation, and c) Domain tessellation.

to evaluate the influence of such particle placement schemes on the structural response having constant material property fields. Only in a second step also the spatial variability in material properties and the respective correlation structure will be investigated. Due to complex nature of the investigated problem it is essential to first isolate and quantify the importance of mimicking clustering and directional effects by the later proposed particle placement schemes.

2 Lattice Discrete Particle Model (LDPM)

A well-established member of the discrete framework, the lattice discrete particle model (LDPM), has been extensively calibrated and validated. It has shown superior capabilities in reproducing and predicting concrete behaviour [14, 15, 16, 17, 18] in a number of practically relevant applications. It simulates the meso-structure of concrete by a three-dimensional (3D) assemblage of particles that are generated randomly according to a given grain size distribution. Figure 1 shows a 2D representation of the LDPM polyhedral cell construction. After the particles are randomly placed in the concrete domain from the biggest to the smallest (Figure 1a), the lattice mesh is generated connecting the centers of the spheres (Figure 1b). Delaunay tetrahedralization and 3D domain tessellation (Figure 1c) are used to generate the system of cells interacting through triangular facets (blue lines in the 2D representation). Note that even though spherical aggregates are assumed for the purpose of generating the particle distribution the final discretization consists of polyhedral cells as sketched in Figure 1c) for the 2D case.

Displacements and rotations of such adjacent particles form the discrete compatibility equations in terms of rigid body kinematics. At each cell facet the meso-scale constitutive law is formulated such that it simulates cohesive fracture, compaction due to pore collapse, frictional slip and rate effect. For each single particle equilibrium equations are finally formulated. An extended version of LDPM is currently developed and simulates various deterioration mechanisms, such as e.g. the Alkali-Silica reaction (ASR) [16], creep and shrinkage. Creep analyses are performed in a rate type form based on code models [19] or by utilizing the Micro-Prestress Solidification Theory MPS [20, 21, 22, 23, 24]. A further development is the age-dependent LDPM framework in which the local material properties are derived by chemo-mechanical coupling from a chemo-hygro-thermal model [25, 26].

3 Characterization of internal structure

If studied at a particular scale and quantity, random heterogeneous materials, such as concrete, exhibit clustering features, which cannot be mimicked by the current LDPM version, where particles are placed randomly and independently. Literature offers a number of approaches on how to describe, quantify, reproduce and compare observed and artificially generated spatially variable structures (spatial arrangement and heterogeneity of micro-structural features). Since the scope of this paper does not allow to cover all classes of statistical descriptors, please refer e.g. to [27, 28, 29] for review. Among the classical approaches allowing for the inference of correlation length (or in general the length scale parameters that characterize spatial heterogeneity and clustering) is the two-point, three-point, and higher order microstructural correlation functions [30], the Ripley's functions and its derivatives [31], the Lineal-Path Function, Chord-Length Density Function, Pore-Size Functions, or the Two-Point Cluster Function [29]. The increasingly available computing resources enabled the practical utilization of morphological-based analyses, which may be more suitable for smaller domain to feature size ratios, i.e. instances where the system size is not sufficiently larger that the correlation length of interest.

The authors briefly introduce here an original approach, which is both computationally efficient and robust, and is based on the Mean value of Minimum Euclidean Distance between centres of fitted circles (MMED). This approach requires that the originally continuous random field is binarized. By using standard image processing, object recognition and morphological analysis algorithms, the boundary components and possibly small components are deleted and subsequently the circular objects (circles) are detected (see Figure 3). Then, for each circle center coordinate a nearest (in euclidean space) neighbour is identified and the correlation length then corresponds to the mean value of such (nearest neighbour) distances. The detection and measurements of objects is based on a local feature detector and descriptor SURF algorithm [32], where the circular objects can be replaced by fitted ellipses or any other parametric shapes. The binarization process is governed by the Otsus cluster variance maximization method [33]. Please note that the correlation lengths in this paper are understood as relative measure, since the random field realizations are self-similar at various scales and independent on the resolution. In fact, this corresponds to various power spectral function parameter sets.

This way, it is ensured that although each specimen has different physical and discretization size, the same patters for the same power spectral function parameters emerge relative to the size of the specimen. The absolute values of the correlation length can be obtained by simply multiplying the relative value of the correlation length with the maximum size of the bounding box of the specimen, d. This is due to the fact that the random fields are initially generated in a $d \times d \times d$ box lattice.

A large number of paradigms related to the generation of correlated spatially variable structures (random fields) exists, including classical algebraic approaches, such as e.g. Fourier transformation [34], Karhunen-Love approximation [35], Polynomial chaos decomposition [36], or evolutionary algorithms, such as cellular automata [37], offering various levels of control in the achieved correlation, variance or stationariness, to name a few. Various production artefacts, such as casting process, can also be simulated by the classical or mesh-less particle-based computational fluid dynamics, possibly also coupled with discrete element method (DEM) [38]. The material structure can be also There is still an ongoing debate concerning the optimal model, mostly from a mathematical and philosophical perspective, since very little is yet known at the required (statistically relevant) level on structural materials.

Given the dimensionality of the problem and the aformentioned arguments, a simple model for the generation of random field has been adopted. It is based on a discrete inverse Fourier transform of a product of noise and amplitude. The noise is defined as a discrete Fourier transform of a pseudo-random variate from a symbolic (Gaussian) distribution and the amplitude is defined by an arbitrarily chosen spectral (autocorrelation) function [34].

In the literature, several functional forms are proposed for autocorrelation functions (Fig-

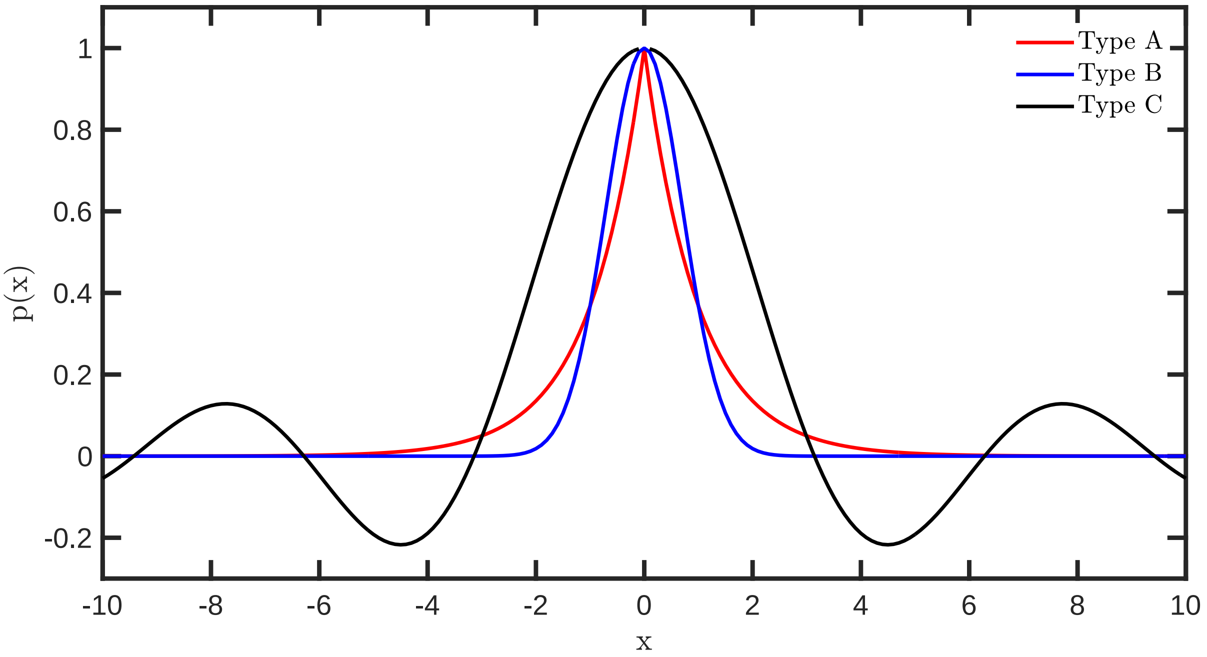


Figure 2: Comparison of the structurally relevant autocorrelation functions from the literature, scale of fluctuation equals to 2.

Table 1: Power spectral function exponents ($pfse$) and related relative autocorrelation length (RAL). The RAL is relative to the maximum specimen size.

a	3.5	3.0	2.5	2.0	1.5	1.0
RAL	18.36 %	7.81 %	4.69 %	3.52 %	3.13 %	2.73 %

ure 2). However, the proposed estimates for their coefficients vary in the order of magnitudes [39, 11, 40, 41] due to their different (physical) interpretation and due to the fact that random heterogeneous materials in general can be considered fractal, i.e. statistically self-similar on a range of length scales. Also, often different equations or names can be found describing the same functional form. Therefore, similarly to the question of generating random fields, the simplest form has been assumed, which corresponds to Type A from Figure 2 which presents the spectral functions $p(x)$ for continuous distance, x. Note that by definition, the spectral density function must be non-negative. The investigated power spectral function with exponent a reads:

$$p(x) = 1/(x^a) \tag{1}$$

The herein adopted meaning of autocorrelation length should follow from Figure 3a (MMED applied to periodic field), while the basis for quantification of a particular autocorrelation length for random fields using the proposed MMED is depicted in Figure 3b. The particular values of relative autocorrelation lengths (RAL) compared to size d of the bounding box depend on the power spectral function exponent (coefficient a in Equation 1) and are listed in Table 1.

The absolute values of the correlation length depend on the maximum values of the bounding box of the specimens, i.e. 400, 300 and 150 mm (beam, cylinder and cube, respectively) and can be computed by simply multiplying the latter sizes by the relative correlation lengths, i.e. ranging from 73 mm for the beam with a = 3.5, to 4 mm for the cube with a = 1.0.

The proposed particle placement schemes may influence the scattering and asymptotic prop-

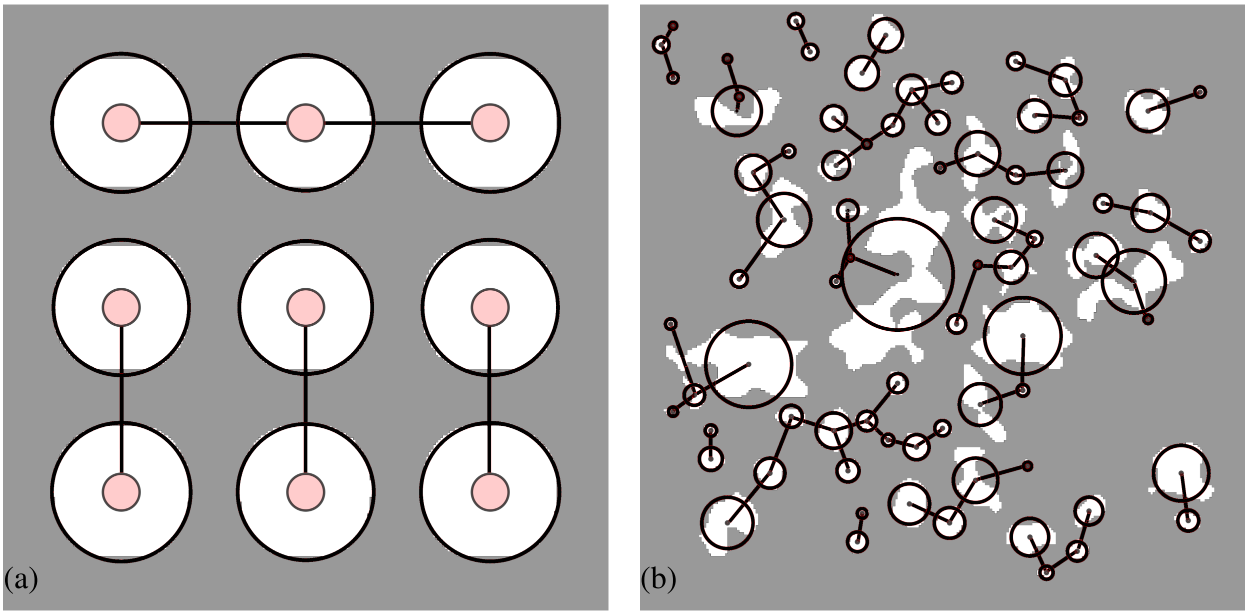


Figure 3: a) MMED applied to periodic field to illustrated the meaning of correlation length (mean distance between nearest neighbours), and b) MMED applied to binarised random field to illustrated the meaning of correlation length.

erties of the spatially variable models and, thus, contribute to the general understanding of the physics and reliability of spatial variability [42, 43]. The abstraction levels for LDPM are categorized as following [43]:

- Independent and random particle placement (IRPP);
- IRPP combined with random or gradient-based field for material characterization only;
- Particle generation governed by a field (PGGF).

Independent and random particle placement (IRPP) Independent and random particle placement and random diameter according to the size distribution curve and required volume fraction, as is currently implemented in the LDPM [14]. No conflicting requirements are to be solved. Overlapping or less than minimum distance particles are re-sampled.

IRPP combined with random or gradient-based field for material characterization only The second abstraction level assumes the original particle placement scheme, i.e. the IRPP, combined with one or more random fields, which are used to describe local fluctuations of material properties resulting from the inherent variability (random field) and construction or transport processes (gradient-based fields). Similarly to the previous case, there are no geometry-related conflicting requirements. Overlapping or less than minimum distance particles are re-sampled. Boundary regions may be normally populated by adopting a simple modification to the re-sampling algorithm.

Material characterizations derived from random fields must be verified for inadmissible values, such as negative strength, modulus, etc. This may lead to a conflict if the governing probability distribution used for generating the random field is to be maintained. Otherwise, truncated distributions may be used or the realizations of random field can be rescaled to fit the admissible range [13, 4, 44].

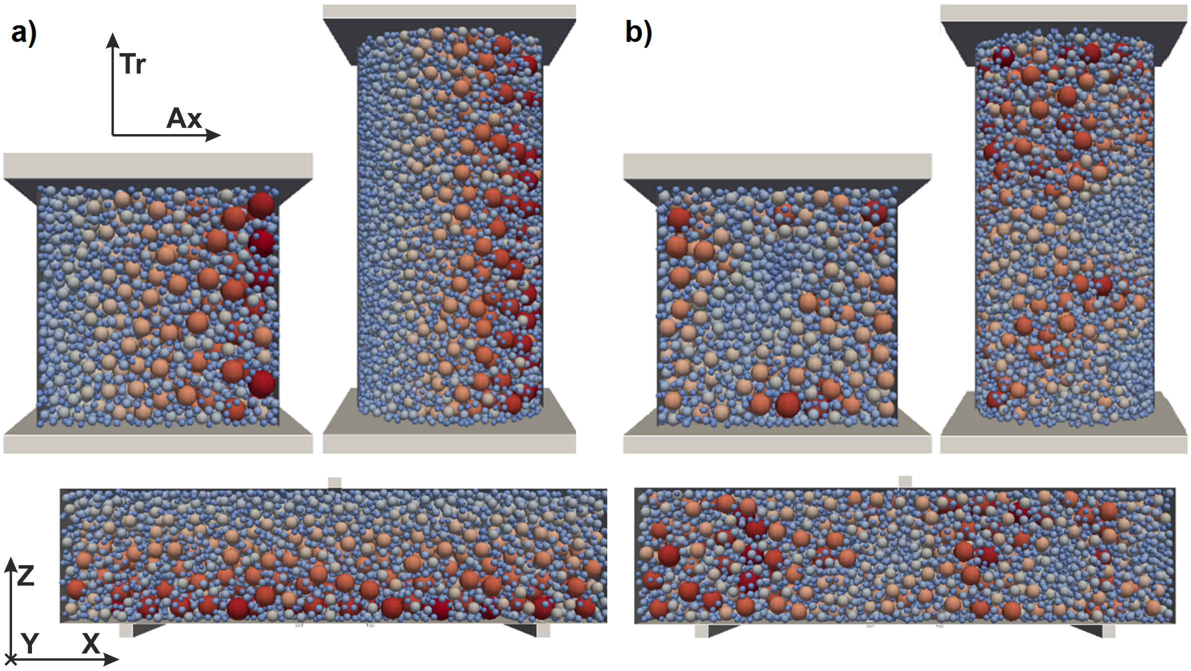


Figure 4: Visual representation of particle placement governed a) by gradient field (PGGF-G) and b) by a random field (PGGF-R).

Particle generation governed by a field (PGGF) Here it is assumed that an initial random (PGGF-R) or gradient-based (PGGF-G) field of choice (or their arbitrary combination) is governing not the material properties (optional), but the particle generation process (i.e. the position and/or the size of each particle). If the particle generation is to be governed not only by granulometric distributions, but also by a gradient-based field (PGGF-G, Figure 4a) or an initial random field (PGGF-R, Figure 4b), the particle generation becomes a complex problem and has to be approached by balancing trade-offs between conflicting goals.

Clearly, the global requirement to follow a particular size distribution can lead to a local conflict with the initial random field, the role of which can be further ambiguous if we consider it to affect both the position and size of the particles (clustering of large particles). Details regarding the associated steps/choices for random fields were published by [43, 45] and are detailed in 4.1 PGGF implementation.

For higher volume fractions this becomes a computationally expensive procedure, however local conflicts can be resolved in parallel and terminate with the first valid particle. The advantage of the approach lies in the compatibility of the mimicked meso-structure (lattice geometry) with any considered material property field (via governing random field) which otherwise cannot be maintained. This enables to verify the relationship between spatial variability, autocorrelation length of the random fields, type of spectral function and meso/micro-structure of the material which is an open research question. Ultimately, it is the goal to investigate the interaction of meso-structure and material property fields derived from the same or related random fields.

However, in a first step the statistical consequences of different particle generation schemes are investigated and compared to each other and the reference, the IRPP. For this purpose, the material properties remain spatially constant and are not derived from fields.

Table 2: LDPM mix design and main LDPM mesoscale properties. The parameters explanation can be found in [14, 15].

Mix Design LDPM parameters			Mesoscale LDPM parameters		
Cement content	240	kg/m^3	Elastic modulus	41000	MPa
Water/Cement	0.83	-	Poissons ratio	0.18	-
Aggregate/Cement	8.83	-	Tensile strength	2.54	MPa
Fullers coefficient	0.5	-	Softening exponent	1	-
Min. aggregate size	4	mm	Shear/Tensile strength	1.85	-
Max. aggregate size	18	mm	Tensile charact. length	200	mm

4 Numerical models

In this section, the numerical models of classical concrete experiments are introduced. Important inputs for the models are the maximum and minimum aggregate sizes. The higher bound of the sieve curve is defined by the maximum aggregate size (d_a) while the minimum aggregate size (d_0) defines its arbitrary lower cut-off, i.e. the diameter under which no particles are discretely generated and placed. Thus, the minimum aggregate size affects the refinement of the discrete mesh and consequently also the computational cost. The concrete parameters used for the LDPM in this contribution are taken from [18] since they were calibrated and validated on an experimental dataset. The main LDPM parameters along with the mix design parameters are defined in Table 2. Their explanation can be found in the original LDPM papers by Cusatis at al.[14, 15].

The simulations include cubes and cylinders loaded in compression, and unnotched beams loaded in a three point bending configuration. Cubes with an edge length of 150 mm and cylinders with a length of 300 mm and a diameter of 150 mm are considered. For both cubes and cylinders, the loading platens are modeled as rigid bodies. The unnotched beam has dimensions of $100 \times 100 \times 400$ mm and a span length between the supports of 300 mm. A visual representation of the specimens is shown in Figure 4. Figure 4a shows the three specimens' geometry having the particle placement distorted with a gradient based field while Figure 4b illustrates the particle placement according to a random field.

In the compression tests, friction between the concrete specimens and the steel platens is considered by a constraint algorithm implemented in the numerical framework MARS [46]. This algorithm constrains the surface nodes of the LDPM domain to the surface of the steel platens based on the friction coefficient $\mu(s)$, according to a contact algorithm described in a previous work by Cusatis et al. [47]. Such friction coefficient is dependent on the contact cumulative slippage s, on the static friction coefficient μ_s, on the dynamic friction coefficient μ_d, and on a characteristic length s_0 derived from fitting available test data (see [48]). The relation is described by: $\mu(s) = \mu_d + (\mu_s - \mu_d)s0/(s + s_0)$. For the cubes, aiming at simulating the contact between concrete and smooth steel, parameters $\mu_s = 0.13$, $\mu_d = 0.015$, and $s_0 = 1.3$ mm were used. For the cylinders, aiming at simulating the contact between concrete and a Teflon sheet on the steel platens, parameters $\mu_s = 0.03$, $\mu_d = 0.0084$, and $s_0 = 0.0195$ mm were used.

The loading speed for the beams was 2 mm/s, 8 mm/s for the cubes, and 5 mm/s for the cylinders. All simulations are run using a dynamic explicit solver implemented in MARS which ensures convergence at the price of computational cost due to the small stable time steps required. For all the models the kinetic energy has been monitored and limited to acceptable levels. As already mentioned, for the PGGF-R analyses of all the geometries, the power spectral function exponent a used in the simulations was chosen to be between 1.0 and 3.5 with a discrete step of 0.5 which means having a relative autocorrelation length range between 2.7% and 18.4% of the maximum specimen dimension.

For the PGGF-G compression specimens, two directions were chosen for the gradient based field. The direction along which the top plate moves to compress the specimen is identified as Ax direction, while the direction transversal to the loading is identified as Tr. For the PGGF-G beams, their dimensions, considering the three Cartesian orthogonal directions, are: 100 mm in the Z and Y directions and 400 mm in the X direction. The load and the two supports are acting along the Z direction. In this case, one gradient field along the Y direction (Y^+) and two gradient based fields in the Z directions are analysed. The two directional fields which are along the Z directions have opposite orientation (Z^+ has the same orientation of the load, and Z^- has the opposite orientation).

Each of the simulations was run in the Vienna Scientific Cluster which consists of 2020 nodes, (8 cores with 2.6 GHz) using one node each for about 3 hours.

4.1 PGGF implementation

Particle generation governed by a field is a modified version of a standard geometrical characterization of the concrete mesostructure presented in [14].

In the present study, the generated mesostructure has to follow both the particle distribution curve and the distribution of a given (random, directional, etc.) field. In the first step, particles represented by spheres are generated following the defined concrete granulometric distribution, the interested reader is referred to [14]. The main difference between the standard and the new procedure lies in the particle placement phase during which the particle centers are placed throughout the volume of the specimen one by one (from the largest to the smallest). Assuming that N_0 particles have to be placed, N_0 random particle positions are generated and the intensity for each of them is evaluated based on the prescribed field. The positions are then ordered following the given intensity (from the highest to the lowest) and the position with the highest intensity is assigned to the largest particle. The largest particle is then placed at this position (assuming that it does not cross the border of the domain) and both the particle and the position are deleted from their lists. Next, the new position with the highest intensity is utilized to place the new largest particle (previously second in the particle list). If there is no conflict with the previously placed particle(s) and the boundary of the domain, the particle is placed and again deleted from the list. However, if it exceeds the domain boundary or overlaps with the previously placed particle(s), this position is discarded, a new random position is generated and the intensity for it is evaluated. Then the positions are again ordered based on the given intensity and the particle placing procedure continues as described before. To minimize the geometrical bias of the discretization, a minimum distance between two adjacent particles is defined as $\delta_s\,(r_1 + r_2)$, where $r_{1,2}$ stand for the radii of the particles and $\delta_s \geq 0$ is the non-dimensional scaling parameter. The utilized minimum distance rule allows a smaller distance between small and large particles compared to the distance between two large particles. $\delta_s = 0.1$ is utilized in the current study.

Specific examples and alternative choices regarding the particle placement algorithm are presented in [45].

Table 3: IRPP results of different geometries with 20 repetitions. The COV is expressed in $\%$.

Type	Property	Beam	Cylinder	Cube
IRPP	F@P (MPa,kN)	11.12±2.82$\%$	21.50±0.41$\%$	26.20±0.55$\%$
	D@P ($-$,mm)	0.0389±4.51$\%$	0.0012±1.18$\%$	0.0017±3.58$\%$

5 Results discussion

The presented observations are based on an unique and extensive computational campaign involving in total 600 simulations. Given the dimensionality of the problem, it is hard to separate physically or mechanically relevant sources of response scattering from the noise components, owing to model uncertainties (solution and discretiztion artefacts) [49].

Along with simulations in which the particle generation is governed by a field (PGGF), also the independent and random particle placement (IRPP) simulations were run for direct comparison. In all cases, 20 repetitions per specimen configuration were run. The results used in the comparison are: the mean stress or the mean force at peak (mean $F@P$) for compression specimens and beams respectively, and the mean strain or the mean displacement at peak (mean $D@P$) for compression specimens and beams respectively. Also their coefficient of variations were computed for the comparison. Table 3 shows the IRPP results for the three geometries.

Table 3 shows that, for the 20 repetitions done for the three geometries, the $COVs$ of $D@P$ are generally higher than the COV of the $F@P$. Also, it can be noticed that the compressive tests results have lower COV than the test results of the beams. This can be explained by different failure mechanism in tension and in compression. In case of flexural failure (tension) there is just one main crack that propagates and ultimately leads to failure, while in case of compression, there are many small cracks that together lead to failure. Each small crack finds its own preferential (least energy) path but in the process causes local stress redistributions affecting the other cracks so that in the end, effects average out and the response for individual realisations stay quite close to the overall mean.

The PGGF results are presented with the same nomenclature as introduced for the IRPP. Also for the PGGF simulations, 20 repetitions per configuration were run. Figure 5 shows the results for different geometries of PGGF-R with $a = 3$ (which means that the RAL is 7.8$\%$ of the maximum size of the specimen). The solid line represents the average numerical result; the numerical results' envelope is plotted as grey area. As can be seen, the 20 repetitions lead to relatively small scatter both in terms of $F@P$ and $D@P$. Note that the simulated unnotched beams experience a snap-down instability in the early post-peak, as expected in displacement control. Nevertheless, the explicit simulations up to this point fully converge with an acceptable amount of kinetic energy in the explicit simulations. Therefore, load and displacement values at peak can be considered correct and serve for this investigation.

The figure shows for the beam case, in comparison with the IRPP results, that the mean values increase while the $COVs$ decrease. For the compression specimens the mean $F@P$ decreases while its scatter increases.

All the PGGF results are shown in Figure 6. On the left Y axes of each figure, the coefficients of variation of $D@P$ and $F@P$ are plotted in red empty square markers, while the right Y axes represent the mean $D@P$ and $F@P$ in black full diamonds. The straight lines represents the IRPP related results. Figures 6(a-b) present the beams results, figures 6(c-d) the cylinders results, and figures 6(e-f) the cube results. The left figures show the PGGF-G results while the right ones show the PGGF-R results.

Regarding the directional field beam (Figure 6a), the results show very consistent trends

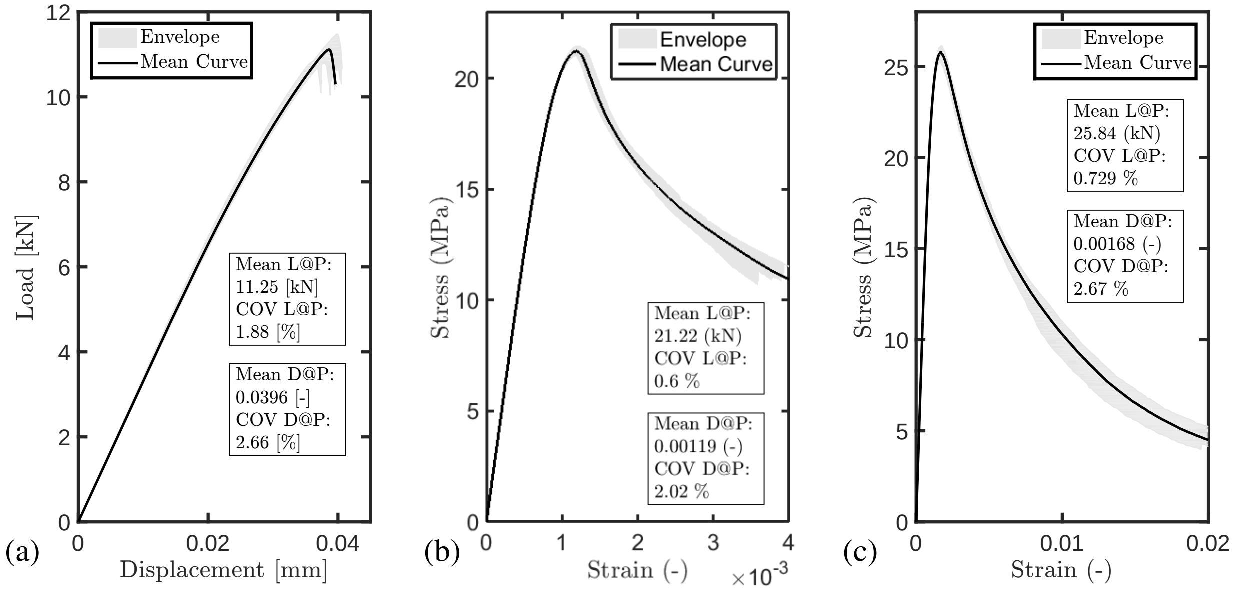


Figure 5: Results of a) Beams, b) Cylinders and c) Cubes for PGGF-R with $a = 3$

among the three different directions both in terms of mean values and in terms of COV. Regarding the random field results (Figure 6b), some weak trends can be noticed. The mean values tend to increase with the autocorrelation length, while the COV tends to stay constant. In comparison with the IRPP beam results, for the PGGF-R both the mean $F@P$ and $D@P$ increase while their $COVs$ decrease. This can be explained by the clustering of big and small particles during the particles placement. With the clustering, some weaker and stronger areas are created, causing the fracture surface to deviate from the nominal path in order to follow the least energy consuming path for its propagation.

The cylinder results are shown in Figure 6(c-d). The directional field results (Figure 6c) show noticeable differences between the axial direction and the tangential direction field. Even though it can be noticed that the transversal direction has higher mean values and COV (compared with the longitudinal one), this doesn't happen for the COV of the peak load which is smaller for the transversal direction. The PGGF-R cylinder results (Figure 6d) show stronger trends compared with the beam case. The trend of the mean $D@P$ and $F@P$ is decreasing with the autocorrelation length while, the trend of the COV is increasing with it. In comparison with the IRPP results, the $D@P$ mean and COV tend to increase; the mean $F@P$ decreases while the COV keeps similar values.

Figure 6(e-f) present the cube results. As for the cylinder, the PGGF-G show that one of the curves, (in this case the mean $D@P$) deviates from the main trend. The PGGF-G results (Figure 6f) show no clear trends for the mean values or the COV.

In comparison with the IRPP cube results, the mean values of $D@P$ and $F@P$ decrease while their COV stays approximately constant. As could be seen from this summary, for the compression geometries the mean $F@P$ tends to be smaller for the $PGGF$ field. In case of the beam only one macro-crack forms and ultimately causes failure, depending on its path across the specimen. For the compression specimens, multiple cracks lead to failure and in this case the clustering lead to a reduction of the peak load.

Please also note the purpose of this study was not to generate (increased) scatter in LDPM simulations, but to quantify scattering caused by the particle placement algorithm independent

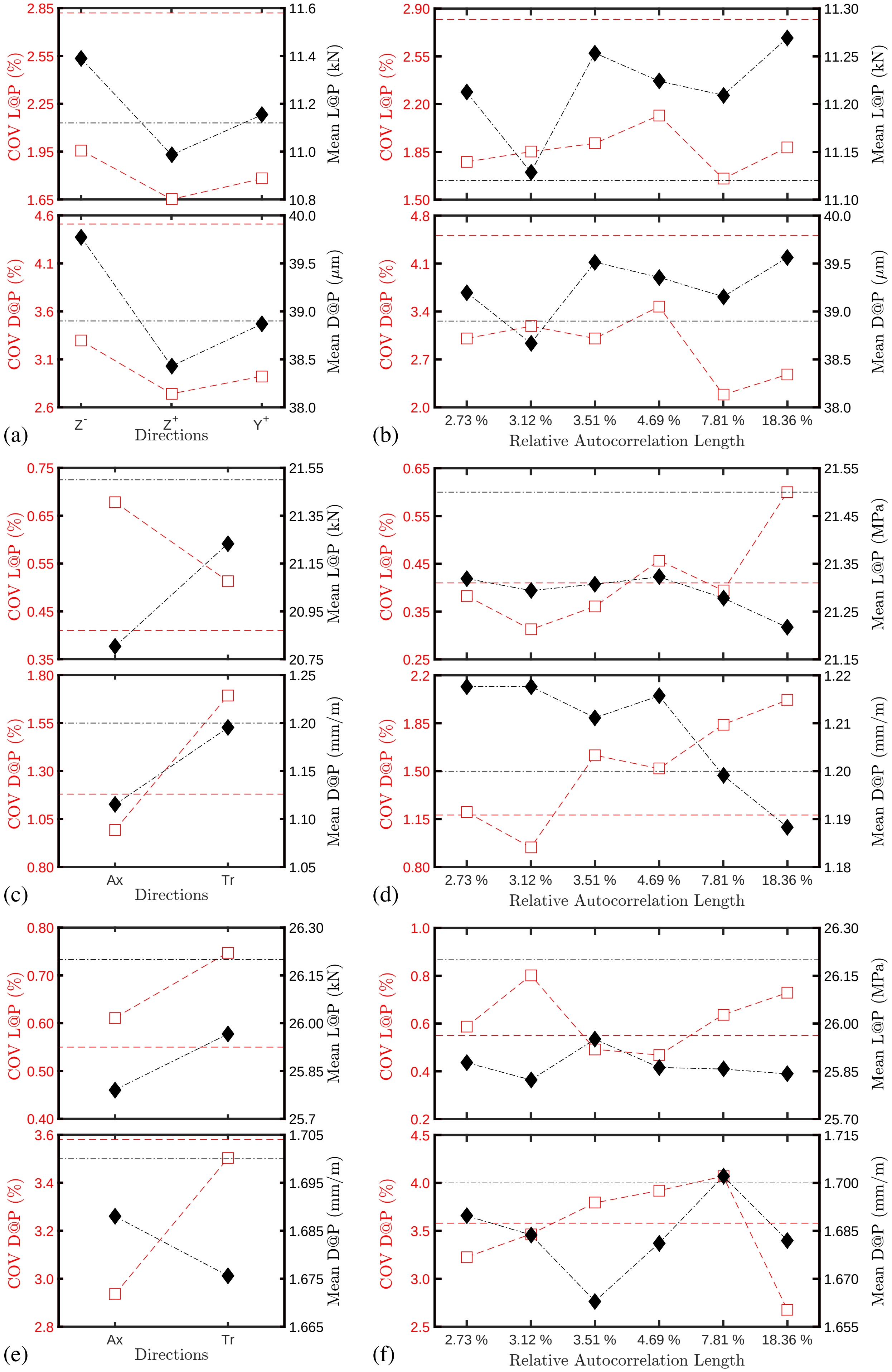


Figure 6: Normalized results for PGGF: (a-b) Beams, (c-d) cylinders, and (e-f) cubes. The straight lines represents the IRPP related results.

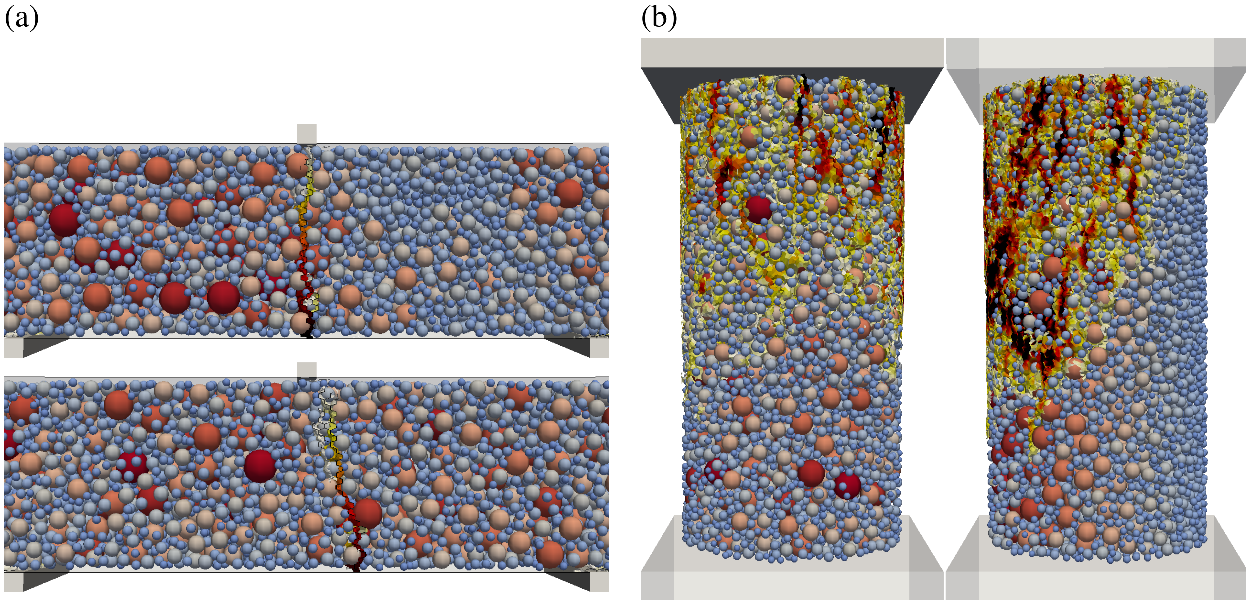


Figure 7: Different failure mechanisms for two repetitions of PGGF-R for a) beams and b) cylinders.

of any variability in the material property fields. The influence on the $L@P$ and the $D@P$ is limited in comparison with the experimental scatter. On the other hand the failure mechanisms are influenced, especially in the case of cylinders with different directional fields (see Figure 7).

5.1 Sample size

A subsampling-based analysis has been performed to evaluate the uncertainty in estimates of mean value and standard deviation with respect to sample size. For the analysis, a set of 40 simulations were run on cylinders having particles placed according to a gradient field perpendicular to the loading direction. Their result is assumed to be an independent and identically distributed sequence. From this sequence, subsamples with sizes from 2 to 38 have been randomly and non-repetitively drawn 700 times in order to ensure non-repetitiveness in boundary subsets containing exactly 2 or 38 elements, where only 780 combinations exist. The mean values and standard deviations of each subsample were calculated. Figure 8a shows the confidence bound on the mean value, while Figure 8b illustrates the uncertainty in the standard deviation. In Figure 8a, the solid line is the mean value of the original sequence. The circles are the 5% fractile of the 700 samples mean values and converge to the mean. The t-Student test was performed on each of the subsamples, in order to obtain the subsamples mean value with a 95% confidence interval. Maximum and mean values of the confidence intervals higher bound, and minimum and mean values of the confidence intervals lower bound, were recorded. The diamonds show mean higher and mean lower bounds of the confidence intervals for all the sample sizes. The squares show maximum higher and minimum lower bounds of the confidence intervals for all the sample sizes. From the results, 5 realizations of LDPM simulations appear to be a reasonable compromise in order to obtain a good approximation of the real mean value while limiting the computational cost.

A similar analysis was performed for the standard deviation as shown in Figure 8b. The estimation of the standard deviation confidence intervals is based on the χ-square test. It becomes clear that the confidence bounds converge much slower. Thus, approximately 15-20 realizations of LDPM simulations appear to be a reasonable compromise in order to obtain a good

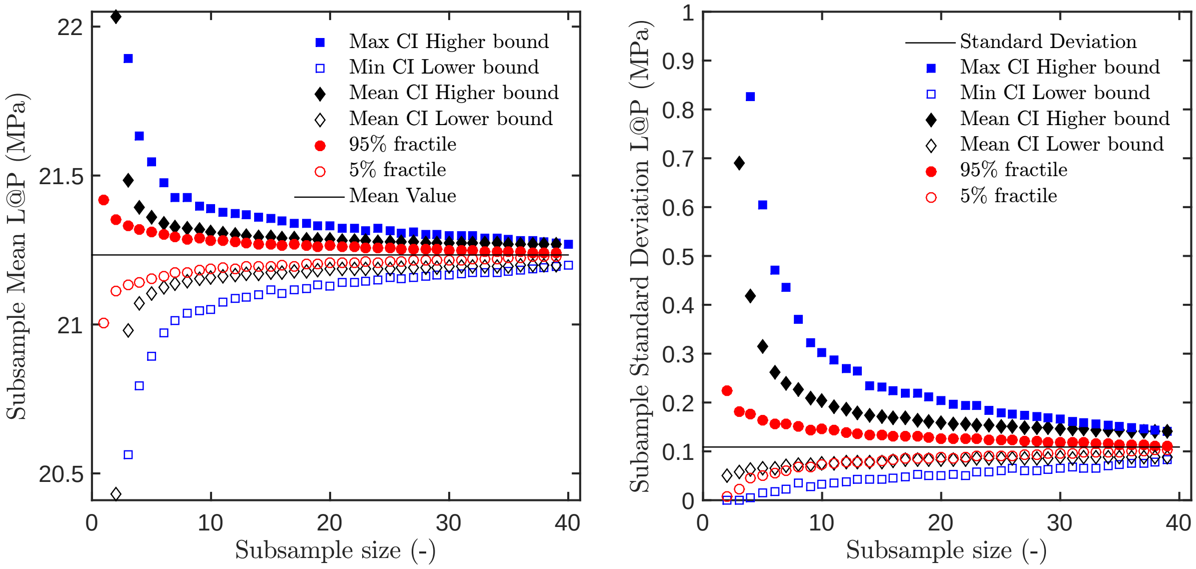


Figure 8: Subsampling-based analysis performed in order to evaluate the uncertainty of mean value and standard deviation predictions.

approximation of the real standard deviation.

6 Influence on failure mechanisms

In order to verify the observed effect of particle clustering on the failure mechanism, the three point bending tests (Figure 7a) are analyzed in detail. The compression simulations (as can be seen in Figure 7b) also show an indication of differences in the failure mechanism. The failure of the beams, on the other hand, shows a unique main crack, the end position of which can be measured and compared. The PGGF-G results are not analyzed because the three directions are symmetric along the main crack path. The results of this analysis are the crack initiation points for different autocorrelation lengths (Figure 9a) and the mean distances between the initiation points and the midspan of the beam (Figure 9b). It can be seen that the scatter in the crack initiation points is basically identical in all the cases.

Figure 9b shows a slight indication that increasing the autocorrelation length, the crack is more likely to reach larger distances from the beam center. Nevertheless, it appears that the only the placement of particles is not enough to perturbate such an output. A stronger trend was observed by Elias et al. [50], where the lattice geometry was independent and random but the material property fields were governed by random fields. From these results it can be concluded that also the material properties need to be influenced by the random field in order to reproduce realistic amounts of scatter and variability in failure modes. A stronger influence of the particle placement is visible for the case pf directional fields.

7 Conclusions

A spatial variability package for LDPM has been presented, including two new abstraction levels for the discrete framework, where particle generation are governed by an initial random field or directional filed. The presented work is a first step of a larger investigation in which modeling concepts for and different sources of spatial variability in concrete are being investigated including spatially variable material property fields.

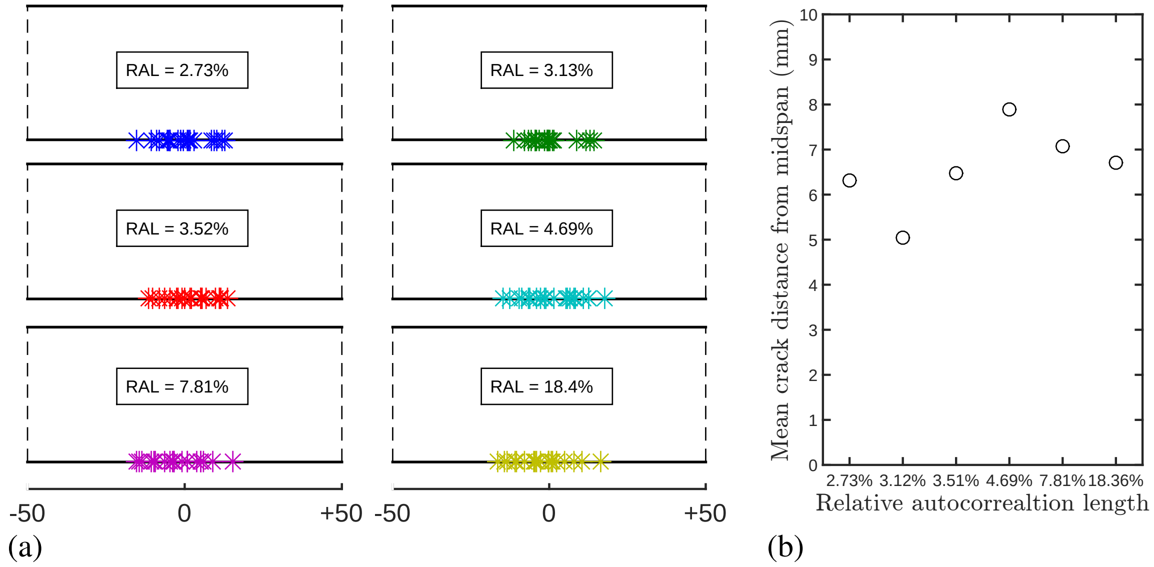


Figure 9: a) Distance between crack and beam center for different $pfse$, and b) Mean distance between crack and beams center for different autocorrelation lengths.

In order to separate the effects of particle generation process governed by random or gradient based field from randomized material property fields governed by random or gradient fields, the material property fields have been kept constant for all of the presented analyses. Thus, by considering constant material property fields, the presented results show how:

- Directional effects, mimicking production processes (concrete casting) and represented by gradient based fields, may affect the mean values of force at peak, displacement at peak, and the respective coefficients of variation;
- Correlated spatial variability models (random fields) governing the particle generation process influence the response and failure mode compared to the independent and random generation of particles;
- No clear functional dependence exists between COV of the structural response and autocorrelation length of the random field determining the particle placement and clustering, at least for the investigated geometries and chosen number of realizations.
- The investigated particle placement schemes with constant material property fields enhance the realisms of simulations but are insufficient to reproduce realistic amounts of experimental scatter.

Acknowledgments

The financial support by the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged, as well as the support of the project No. LO1408 ”AdMaS UP and FAST-S-18-5327. The computational results presented have been achieved using the Vienna Scientific Cluster (VSC).

REFERENCES

[1] R. Hill, *The mathematical theory of plasticity*, vol. 11. Oxford university press, 1998.

[2] Z. P. Bažant, “Size effect in blunt fracture: concrete, rock, metal,” *Journal of Engineering Mechanics*, vol. 110, no. 4, pp. 518–535, 1984.

[3] Q. Yu, J.-L. Le, M. H. Hubler, R. Wendner, G. Cusatis, and Z. P. Bažant, “Comparison of main models for size effect on shear strength of reinforced and prestressed concrete beams,” *Structural Concrete*, vol. 17, no. 5, pp. 778–789, 2016.

[4] J. Podroužek, J. Vorel, I. Boumakis, G. Cusatis, and R. Wendner, “Implications of spatial variability characterization in discrete particle models,” in *Proceedings of the 9th international conference on fracture mechanics of concrete and concrete structures. Presented at the FraMCoS-9, Berkeley, CA, USA Google Scholar*, 2016.

[5] Z. P. Bazant and J.-L. Le, *Probabilistic Mechanics of Quasibrittle Structures: Strength, Lifetime, and Size Effect*. Cambridge University Press, 2017.

[6] R. E. Melchers and A. T. Beck, *Structural reliability analysis and prediction*. John Wiley & Sons, 2017.

[7] J. Podrouzek, C. Bucher, and G. Deodatis, “Identification of critical samples of stochastic processes towards feasible structural reliability applications,” *Structural Safety*, vol. 47, pp. 39–47, 2014.

[8] J. Podroužek, A. Strauss, and D. Novák, “Spatial degradation in reliability assessment of ageing concrete structures,” in *1st ECCOMAS thematic conference on international conference on uncertainty quantification in computational sciences and engineering. Presented at the UNCECOMP*, 2015.

[9] B. Sudret, “Probabilistic models for the extent of damage in degrading reinforced concrete structures,” *Reliability Engineering & System Safety*, vol. 93, no. 3, pp. 410–422, 2008.

[10] P. Grassl and Z. P. Bažant, “Random lattice-particle simulation of statistical size effect in quasi-brittle structures failing at crack initiation,” *Journal of engineering mechanics*, vol. 135, no. 2, pp. 85–92, 2009.

[11] P. Grassl and M. Jirásek, “Meso-scale approach to modelling the fracture process zone of concrete subjected to uniaxial tension,” *International Journal of Solids and Structures*, vol. 47, no. 7-8, pp. 957–968, 2010.

[12] J. Podrouzek, M. Marcon, J. Vorel, and R. Wan-Wendner, “Response scatter control for discrete element models,” in *Proceedings of EURO-C 2018*, 2018.

[13] J. Eliáš, M. Vořechovskỳ, J. Skoček, and Z. P. Bažant, “Stochastic discrete meso-scale simulations of concrete fracture: Comparison to experimental data,” *Engineering fracture mechanics*, vol. 135, pp. 1–16, 2015.

[14] G. Cusatis, D. Pelessone, and A. Mencarelli, “Lattice discrete particle model (ldpm) for failure behavior of concrete. i: Theory,” *Cement and Concrete Composites*, vol. 33, no. 9, pp. 881–890, 2011.

[15] G. Cusatis, A. Mencarelli, D. Pelessone, and J. Baylot, “Lattice discrete particle model (ldpm) for failure behavior of concrete. ii: Calibration and validation,” *Cement and Concrete composites*, vol. 33, no. 9, pp. 891–905, 2011.

[16] M. Alnaggar, G. Cusatis, and G. Di Luzio, “Lattice discrete particle modeling (LDPM) of alkali silica reaction (ASR) deterioration of concrete structures,” *Cement and Concrete Composites*, vol. 41, pp. 45–59, 2013.

[17] I. Boumakis, G. D. Luzio, M. Marcon, J. Vorel, and R. Wan-Wendner, “Discrete element framework for modeling tertiary creep of concrete in tension and compression,” *Enginnering Fracture Mechanics (accepted)*, 2018.

[18] M. Marcon, J. Vorel, K. Ninevi, and R. Wan-Wendner, “Modeling adhesive anchors in a discrete element framework,” *Materials*, vol. 10, no. 8, 2017.

[19] Q. Yu, Z. P. Bazant, and R. Wendner, “Improved algorithm for efficient and realistic creep analysis of large creep-sensitive concrete structures,” *ACI Structural Journal*, vol. 109, no. 5, p. 665, 2012.

[20] Z. P. Bažant, A. B. Hauggaard, S. Baweja, and F.-J. Ulm, “Microprestress-solidification theory for concrete creep. i: Aging and drying effects,” *Journal of Engineering Mechanics*, vol. 123, no. 11, pp. 1188–1194, 1997.

[21] Z. P. Bažant and S. Prasannan, “Solidification Theory for Concrete Creep. I: Formulation,” *Journal of Engineering Mechanics*, vol. 115, pp. 1691–1703, 1989.

[22] I. Boumakis, M. Marcon, K. Ninevi, L.-M. Czernuschka, and R. Wan-Wendner, “Concrete creep and shrinkage effect in adhesive anchors subjected to sustained loads,” *Engineering Structures*, vol. 175, pp. 790 – 805, 2018.

[23] I. Boumakis, M. Marcon, K. Ninčević, L.-M. Czernuschka, and R. Wan-Wendner, “Concrete creep effect on bond stress in adhesive fastening systems,” in *Proceedings of the 3rd International Symposium on Connections between Steel and Concrete, ConSC 2017*, (Stuttgart, Germany), pp. 396–406, 2017.

[24] I. Boumakis, M. Marcon, L. Wan, and R. Wendner, “Creep and shrinkage in fastening systems,” in *CONCREEP 2015: Mechanics and Physics of Creep, Shrinkage, and Durability of Concrete and Concrete Structures - Proceedings of the 10th International Conference on Mechanics and Physics of Creep, Shrinkage, and Durability of Concrete and Concrete Structures*, pp. 657–666, 2015.

[25] L. Wan, R. Wendner, B. Liang, and G. Cusatis, “Analysis of the behavior of ultra high performance concrete at early age,” *Cement and Concrete Composites*, vol. 74, pp. 120–135, 2016.

[26] L. Wan-Wendner, R. Wan-Wendner, and G. Cusatis, “Age dependent size effect and fracture characteristics of ultra high performance concrete,” *Cement and Concrete Composites*, vol. 84, pp. 67–82, 2018.

[27] P. B. Corson, “Correlation functions for predicting properties of heterogeneous materials. i. experimental measurement of spatial correlation functions in multiphase solids,” *Journal of Applied Physics*, vol. 45, no. 7, pp. 3159–3164, 1974.

[28] B. D. Ripley, *Spatial statistics*, vol. 575. John Wiley & Sons, 2005.

[29] S. Torquato, *Random heterogeneous materials: microstructure and macroscopic properties*, vol. 16. Springer Science & Business Media, 2013.

[30] A. Tewari, A. Gokhale, J. Spowart, and D. Miracle, "Quantitative characterization of spatial clustering in three-dimensional microstructures using two-point correlation functions," *Acta Materialia*, vol. 52, no. 2, pp. 307–319, 2004.

[31] V. Lefort, G. Pijaudier-Cabot, and D. Grgoire, "Analysis by ripleys function of the correlations involved during failure in quasi-brittle materials: Experimental and numerical investigations at the mesoscale," *Engineering Fracture Mechanics*, vol. 147, pp. 449 – 467, 2015.

[32] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[33] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[34] G. Christakos, *Random field models in earth sciences*. Courier Corporation, 2012.

[35] C. Schwab and R. A. Todor, "Karhunen–loève approximation of random fields by generalized fast multipole methods," *Journal of Computational Physics*, vol. 217, no. 1, pp. 100–122, 2006.

[36] S. Sakamoto and R. Ghanem, "Polynomial chaos decomposition for the simulation of non-gaussian nonstationary stochastic processes," *Journal of engineering mechanics*, vol. 128, no. 2, pp. 190–201, 2002.

[37] E. Hernández Zubeldia, L. C. de SM Ozelim, A. Luís Brasil Cavalcante, and S. Crestana, "Cellular automata and x-ray microcomputed tomography images for generating artificial porous media," *International Journal of Geomechanics*, vol. 16, no. 2, p. 04015057, 2015.

[38] C. Kloss and C. Goniva, "Liggghts–open source discrete element simulations of granular materials based on lammps," *Supplemental Proceedings: Materials Fabrication, Properties, Characterization, and Modeling*, vol. 2, pp. 781–788, 2011.

[39] J. G. Teigen, D. M. Frangopol, S. Sture, and C. A. Felippa, "Probabilistic fem for nonlinear concrete structures. i: Theory," *Journal of Structural Engineering*, vol. 117, no. 9, pp. 2674–2689, 1991.

[40] E. Syroka-Korol, J. Tejchman, and Z. Mróz, "Fe investigations of the effect of fluctuating local tensile strength on coupled energetic–statistical size effect in concrete beams," *Engineering Structures*, vol. 103, pp. 239–259, 2015.

[41] B. L. Karihaloo, "A new approach to the design of rc structures based on concrete mix characteristic length," *International Journal of Fracture*, vol. 191, no. 1-2, pp. 147–165, 2015.

[42] M. Marcon, J. Podrouzek, J. Vorel, and R. Wan-Wendner, "Characterisation of spatial variability in lattice models," in *COMUS17 - Eccomas Thematic Conference - COMPUTATIONAL MODELLING OF MULTI-UNCERTAINTY AND MULTI-SCALE PROBLEMS , ECCOMAS*, 2017.

[43] J. Podrouzek, J. Vorel, and R. Wan-Wendner, "Discrete particle placement schemes for LDPM," in *2017 EMI International Conference*, 2017.

[44] M. Vořechovskỳ and D. Novák, "-efficient random fields simulation for stochastic fem analyses," in *Computational Fluid and Solid Mechanics 2003*, pp. 2383–2386, Elsevier, 2003.

[45] J. Podroužek, J. Vorel, G. Cusatis, and R. Wendner, "Imposed correlation between random field and discrete particle placement," in *14th International Probabilistic Workshop*, pp. 245–252, Springer, 2017.

[46] D. Pelessone, "MARS: Modeling and analysis of the response of structuresUsers manual," *ES3, Beach (CA), USA*, 2009.

[47] G. Cusatis, Z. P. Bažant, and L. Cedolin, "Confinement-shear lattice model for concrete damage in tension and compression: I. theory," *Journal of Engineering Mechanics*, vol. 129, no. 12, pp. 1439–1448, 2003.

[48] R. A. Vonk, "Softening of concrete loading in compression," *Technische Universiteit Eindhoven*, 1992.

[49] S. Prabhu, S. Atamturktur, and S. Cogan, "Model assessment in scientific computing: Considering robustness to uncertainty in input parameters," *Engineering Computations*, vol. 34, no. 5, pp. 1700–1723, 2017.

[50] J. Eliáš, M. Vořechovskỳ, and J.-L. Le, "Lattice modeling of concrete fracture including material spatial randomness," *Eng Mech*, vol. 20, pp. 413–426, 2013.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

APPLYING BAYESIAN INVERSION WITH MARKOV CHAIN MONTE CARLO TO PEDESTRIAN DYNAMICS

Marion Gödel[1,2], Rainer Fischer[1], and Gerta Köster[1]

[1]Munich University of Applied Sciences
Lothstrasse 64, 80335 Munich, Germany
e-mail: {marion.goedel, rainer.fischer, gerta.koester}@hm.edu

[2] Technical University of Munich
Boltzmannstraße 3, 85748 Garching, Germany

Keywords: uncertainty quantification, Bayesian inversion, Markov chain Monte Carlo, Metropolis algorithm, pedestrian dynamics, microscopic simulation.

Abstract. *Pedestrian simulations serve to predict the movement of a crowd. These simulations have become an important tool for building planners, event managers, crowd managers and many more. In microscopic simulations, one simulates individual virtual pedestrians (agents). The behavior models for the agents range from differential-equation-based over step-based to rule-based systems. Independent of the underlying systematics, all models depend on a set of parameters. It is crucial to analyze the parameters and their impact on the results in order to know how much trust we can put in the simulations.*

In this work, we present first results of an application of Bayesian inversion to a step-based model in our simulation framework Vadere as a proof of concept. More specifically, we focus on preferred walking speeds of pedestrians. Distributions of these free-flow speeds are a necessary input for most microscopic pedestrian simulations and can typically not be measured directly. We consider a simple scenario of pedestrians walking in a hallway. We apply the Metropolis algorithm to sample from the posterior.

If we provide a prior that differs from the actual distribution of the uncertain parameter, we expect the method to inform the posterior so that it is closer to the actual distribution. This serves as a first proof of concept and motivates the incorporation of experimental data. The results serve as a basis for the development of a framework with uncertainty quantification (UQ) methods customized for our open-source simulation framework Vadere.

1 INTRODUCTION

Pedestrian dynamics is a rather young research field that focuses on the movement of an aggregation of pedestrians. It comprises several areas of research: Observing pedestrian behavior, conducting experiments, modeling of observed behavioral patterns, implementing computer models. In addition, it is an interdisciplinary field of study: Psychologists and sociologists are interested in the topic as well as engineers, computer scientists and mathematicians. Even though they utilize different methods and focus on different aspects, their motivation is the same: Enhancing the safety of crowds.

There are several typical use-cases of pedestrian crowd simulation: Nowadays, the evaluation of emergency routes and closeout concepts is supported by simulation results. In general, a simulation of a venue can help to identify potential bottlenecks. Case studies can reveal the benefit of possible improvements. In addition, simulations help to estimate the capacity at events, in public spaces, or in infrastructural facilities. Another field of application is the optimization of pedestrian facilities like train stations or airports. Consequently, these simulations are handy for organizers of all kinds of events as well as building and infrastructure planners.

1.1 Microscopic crowd simulation

Our research group focuses on the microscopic simulation of pedestrian crowds. The underlying models are based on observations of pedestrians. The most popular locomotion models are cellular automata [8] and force-based models [10]. Cellular automata discretize the space into cells which are either occupied or free. A set of rules is defined in order to describe how each pedestrian moves from one cell to another. On the other hand, force-based models define social forces acting on pedestrians similar to Newtonian forces acting on particles. Each pedestrian is attracted by his / her target and repulsed from obstacles and other pedestrians. The most famous force-based model is the social force model, which is formulated as a system of ordinary differential equations. The positions of pedestrians are calculated at discrete time steps.

In contrast to the types of models listed above, in which time and / or space are discretized in an artificial manner, we consider a model that utilizes a natural discretization, the so-called stepping procedure. The optimal steps model [18, 19] finds, for each pedestrian in each time step, the next optimal position within a disc around his / her current position. Instead of modelling the impacts by forces, we use utility functions to encode the value of a position to a pedestrian.

1.2 Limitations of UQ for pedestrian dynamics

Independent of the chosen locomotion model, pedestrian crowd simulations typically have a rather large set of input parameters. The UQ literature distinguishes between two types of parameters: Physical parameters and non-physical parameters. Physical parameters can be measured in controlled experiments or inferred from videos. On the contrary, non-physical parameters are model parameters which cannot be measured.

Regarding crowd simulations, examples for physical parameters are the number of pedestrians in the scenario and all parameters concerning the description of the location that is simulated. Many parameters fall in the second category, non-physical parameters, due to different reasons: First, there are computational parameters such as the parameters of the optimizer. Second, all parameters that are introduced by mathematical modeling. For example, utility dips are used to model the disadvantage of positions close to an obstacle. There are certain parameters that define how large the influence radius of an obstacle is and how large the utility dip is. These parameters are part of the mathematical model which is motivated by the observation that pedestrians keep a natural distance from walls and other obstacles. Third, there are parameters which technically could be measured easily, but for which the measurement process would change the behavior of the human participants. Thus, these parameters elude measurement. Other examples are parameters influenced by psychology such as the percentage of people who would help others in emergencies or the speed of movement when carrying or helping injured people. We consider the high number of non-physical parameters a central challenge when applying methods of uncertainty quantification in pedestrian dynamics.

Another challenge are stochastic simulations. Most, if not all, methods of uncertainty quantification are designed to handle stochastic inputs to the system under investigation. Nevertheless, the system in focus is considered a deterministic system. This is not generally true for crowd simulators. In case of our simulation framework, Vadere [1], some attributes such as preferred speeds or starting positions are usually assigned randomly within the simulator.

In addition, the UQ methods assume that the relation between input parameters and quantity of interest is continuous. Again, this requirement cannot be fulfilled by all crowd simulators. It is important to be aware of these limitations. To our knowledge, these challenges have not been tackled so far, but they need to be addressed in the future.

1.3 Benefits of UQ for pedestrian dynamics

Despite these challenges when applying UQ methods to pedestrian dynamics, it is crucial to find ways to examine the impact of the parameters on the simulation output in order to know how much trust one can put into the results. Applying methods of uncertainty quantification is a promising approach to reach this goal.

One set of methods aiming in this direction are forward propagation and sensitivity analysis. When using forward propagation, for each uncertain parameter, a probability density function needs to be provided. For physical parameters, these distributions may be known from the application, but for non-physical parameters, they are typically unknown. One way to find the distribution of an uncertain input parameter is utilizing Bayesian inversion. Another field of application is the choice of parameter values for non-physical parameters. Here, inversion methods can be applied to infer the parameter values based on empirical data (parameter calibration).

1.4 State-of-the-Art

To our knowledge, there are currently only a few publications available in which methods of uncertainty quantification are applied to pedestrian dynamics. They can be divided into two groups: Applications of forward propagation and of Bayesian inversion.

Von Sivers *et al.* utilize forward propagation with stochastic collocation as intrusive method to investigate the impact of parameters of a certain sub-model of a pedestrian simulator [20]. In addition, Dietrich *et al.* apply forward propagation to the simulation of a train station [6]. To speed up the method, a surrogate model is constructed that approximates the simulator. This combination allows what they call real time uncertainty quantification.

In this contribution, we focus on inversion methods. Since pedestrian crowd simulators typically have many non-physical parameters, inferring parameter distributions for the forward propagation is an important task. To our knowledge, there are only two applications of inversion techniques to pedestrian crowd simulations: Corbetta *et al.* demonstrate an application of Bayesian inversion [5]. Based on experimental data, they infer non-physical model parameters. Finally, Bode utilizes approximate Bayesian computation, a likelihood-free inversion method to infer parameters and compare two different movement models [2].

All of these applications demonstrate first successful applications of uncertainty quantification concepts to pedestrian dynamics. Nevertheless, each one of them is an individual application of one method on a specific problem. In our contribution, we concentrate on the method of inversion itself and its general applicability to pedestrian dynamics models. We aim to provide a proof of concept of Bayesian inversion utilizing Markov chain Monte Carlo (MCMC) methods based on our model, the optimal steps model. Our goal is to provide a framework of uncertainty quantification methods for users of crowd simulators.

1.5 Paper Outline

In chapter 2 we will give a brief introduction to microscopic simulation of crowd behavior and the most commonly used terms, describe the simulation setup and the inversion method. In addition, we give an overview on the proposed framework. Chapter 3 focuses on the results of the proof of concept. We show that the inversion corrects the incorrect prior, which proves that it extracts information from the model. Finally, in chapter 4 we provide a conclusion on the performed work and an outlook.

2 METHODS

This chapter starts with a brief description of microscopic crowd simulations, which needs some specific terminology. We introduce commonly used terms in the pedestrian dynamics community.

Agent We refer to the simulated, virtual pedestrians as agents.

Origin Area in which pedestrians are spawned (generated) in the simulation.

Destination Physical target of pedestrians. In microscopic crowd simulators, individual agents move from origins to destinations.

Topography Description of the location that is simulated. This could be a building or a venue. The topography contains origins and destinations.

Scenario A scenario contains all information for the simulation. That means, it contains a topography and the configuration of the model including all parameters.

Trajectory A list of positions that an agent occupies during its movement to its origin.

Free-flow speed The free-flow speed is the speed with which a pedestrian moves unhindered towards his / her target, meaning in absence of other pedestrians or obstacles. This parameter is common to almost all locomotion models.

When it comes to the distinction between physical and non-physical parameters, there is a great difference between observing humans and technical systems. While parameters such as speed are easy to measure regarding the technical difficulty, humans behave differently when observed. In many experiments, the goal is to capture the free-flow speed by measuring the time that pedestrians need to walk through a hallway and then by deriving the speed [22]. Nevertheless, this can only serve as an approximation to the actual intrinsic free-flow speed of a pedestrian, if existent.

Even though the free-flow speed is a parameter that is necessary for most microscopic simulator types, it cannot be measured directly. Consequently, Bayesian inversion is particularly useful to obtain a distribution for this quantity. In addition, even though the free-flow speed is a non-physical parameter, it can be estimated from experiments and therefore we roughly know the size of it from studies [21].

2.1 Configuration of the simulation scenario

We focus on a simple but relevant scenario, a single agent walking unhindered from origin to destination (see Fig. 1). This scenario is also known as the first test case in the guidelines for pedestrian simulations [16]. In particular, the scenario shown in Fig. 1 is the test case for the free-flow speed. That is why we choose this scenario.

As described before, in most crowd simulators, pedestrians move from origins to destinations. Nevertheless, origin and destination are modelling constructs which may lead to artifacts. In reality, pedestrians do not just appear and disappear. That is why we only observe the pedestrian within the measurement area.

As quantity of interest, we choose the service time, which is the time the pedestrian needs to travel to his / her target. Instead of using the travel time between origin and destination, we use the travel time within the measurement area.

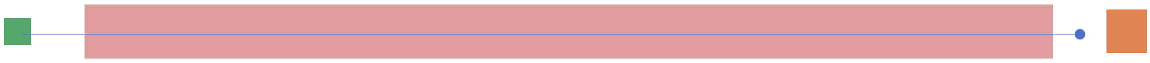


Figure 1: Single pedestrian scenario: The agent moves from the origin (green) to the destination (orange). The agent is shown as a blue circle and its trajectory is depicted in blue. The measurement area is shown in red. This scenario is also known as RiMEA 1 test case [16].

2.2 Bayesian inversion and MCMC

Now, we briefly introduce the concept of Bayesian inversion and Markov chain Monte Carlo methods. A detailed description of this approach can be found, for example, in [4, 11]. Bayesian inversion is an approach to solve the inverse problem

$$d = m(x) + e \tag{1}$$

for the random parameters $x \in \mathbb{R}^n$. Here, $m(\cdot) : \mathbb{R}^n \to \mathbb{R}^m$ is a deterministic map from parameters to observables. In our case, the map is the simulator including the evaluation of the chosen quantity of interest. It is handled as a black box since the model is rule-based, not equation-based. $d \in \mathbb{R}^m$ are the random data from which the parameters x will be inferred and $e \in \mathbb{R}^m$ is a random, additive noise. In the described setup, all terms are scalars ($n = m = 1$) and hence $m : \mathbb{R} \to \mathbb{R}$. We assume that $e \sim \mathcal{N}(0, \sigma)$, that is, e is a zero-mean Gaussian noise. This assumption leads to the likelihood

$$\rho_{lik}(d|x) = \text{exp}\left(\frac{- \parallel d - m(x) \parallel_2^2}{2\sigma^2}\right). \tag{2}$$

Bayes Theorem implies

$$\rho_{pos}(x) = \frac{\rho_{lik}(d|x)\rho_{pri}(x)}{\rho(d)} \propto \rho_{lik}(d|x)\rho_{pri}(x) \tag{3}$$

where $\rho_{pri}(x)$ is the prior of the parameter x and $\rho_{pos}(x|d)$ is the posterior of the parameters x given the data d. $\rho(d)$ is the so called evidence, a normalization constant.

One standard approach to access the posterior distribution is Markov chain Monte Carlo. All methods of this type construct a Markov chain whose stationary distribution is the posterior.

We use the well-known Metropolis algorithm [14]. Since our work addresses both the UQ and the pedestrian dynamics community, which is not familiar with the algorithm, we will shortly describe how the Metropolis algorithm works.

The parameters for the Metropolis algorithm are:

Initial point The starting point of the algorithm. Typically, the center of the prior distribution is chosen as initial point.

Proposal function The proposal function is a distribution used to generate a new candidate. This parameter contains a type of distribution and distribution parameters. A typical choice is a normal distribution.

Number of iterations The number of iterations that are performed with the Metropolis algorithm. The size depends on the number of uncertain parameters.

Burn-in The number of iterations before the algorithm reaches a steady state. These iterations are usually not considered for further evaluations.

The algorithm starts at the initial point. In each step, a new candidate

$$x' = x_t + z, z \sim \mathcal{N}(0, \tau). \tag{4}$$

is created based on the previous candidate x_t and a random value z drawn from the proposal distribution. For each candidate, the model is evaluated to calculate the posterior

$$\rho_{pos}(x') = \rho_{lik}(d|x')\rho_{pri}(x') = \exp\left(\frac{-\parallel d - m(x') \parallel_2^2}{2\sigma^2}\right)\rho_{pri}(x'). \tag{5}$$

Based on the posterior, it is decided if the candidate is accepted or rejected.

$$x_{t+1} = \begin{cases} x' & \text{if } \rho_{pos}(x') \geq \rho_{pos}(x_t) \text{ or } \frac{\rho_{pos}(x')}{\rho_{pos}(x_t)} \leq u \quad \text{(accepted)} \\ x_t & \text{otherwise} \quad \text{(rejected)} \end{cases}$$

where $u \sim \mathcal{U}(0, 1)$. After the chosen burn-in period, the accepted candidates can be used as a sample of the posterior distribution.

2.2.1 Adaptive regulation of the proposal distribution

The main parameter for configuring the Metropolis algorithm is in our case τ, i. e. the variance of the proposal distribution, also called the jump width. The jump width is a crucial parameter because a too large jump width deters the candidates from concentrating around the true parameter value. On the other hand, if the jump width is chosen too small, it takes long to converge, especially if the prior guess is inaccurate.

In order to find an appropriate jump width, we choose to adapt the jump width to the acceptance rate. The acceptance rate α is the ratio between the number of accepted candidates to the total number of candidates. In [7, 17] optimal acceptance rates for Gaussian posterior distributions are presented. They propose to monitor the acceptance ratio and to scale the jump width accordingly to obtain an optimal acceptance rate. According to [9], this method is widely used in practice. Gelman *et al.* state that the optimal acceptance rate for a one-dimensional problem is 0.44 [7]. Based on their findings, we modify the jump width only when it is outside the interval $[0.3; 0.5]$:

$$\tau = \begin{cases} \alpha \leq 0.3 & \tau/\alpha \\ \alpha \geq 0.5 & \tau \cdot \alpha. \end{cases} \tag{6}$$

To avoid too frequent adaption of the jump width, we allow the correction only in every 10^{th} step, similar to [15], but we make use of all candidates in order to compute the acceptance rate.

2.2.2 Measures for MCMC performance

There are two common measures to evaluate the quality of the samples obtained from the MCMC method: autocorrelation of samples and effective sample size. We calculate the effective sample size according to [13, p. 184] as

$$\text{ESS} = \frac{N}{1 + 2\sum_{i=1}^{\infty} \text{ACF}(i)} \approx \frac{N}{1 + 2\sum_{i=1}^{m} \text{ACF}(i)}$$

where N is the number of samples (after burn-in) and ACF is the autocorrelation function (ACF). The effective sample size is approximated by limiting the infinite sum when $\text{ACF}(m + 1) \leq 0.05$ according to [13, p. 184].

In general, the acceptance rate is also a measure which provides information about the performance of the algorithm. Nevertheless, as described in the previous chapter, we alter the jump width to obtain a certain acceptance rate. Consequently, the acceptance rate provides less information than in the regular case. Despite the manipulation, the acceptance rate can still be used to get an indication for the size of the burn-in. While the autocorrelation function is known in many disciplines to analyze the correlation of a time series, the effective sample size is a measure explicitly for Markov chain Monte Carlo methods. It gives an estimate of the number of uncorrelated samples within the samples of the posterior. Due to the generation manner of candidates, consecutive samples are highly correlated.

Both measures are strongly impacted by the jump width: While a small jump width leads to a high acceptance rate, because a new candidate close to the previous sample (accepted candidate) is very likely to be accepted, it also leads to highly correlated samples. Consequently, the effective sample size is small.

2.3 Proposed framework

The Bayesian inversion described in this manuscript is foreseen to be one module of a framework that we plan to provide. It will be designed in a way that users of pedestrian simulation can carry out parameter studies. Figure 2 shows the scheme of the framework and displays the building blocks. The uncertainty quantification framework will use the so-called SUQ-controller to communicate with the Vadere simulation framework. The SUQ-controller sends queries to Vadere, obtains the results and stores them. The uncertainty quantification framework will have three main building blocks: Forward propagation, Bayesian inversion and (global) sensitivity analysis. Some methods can be applied in multiple contexts. For example, active subspaces can be used for Bayesian inversion to reduce the dimensionality of the parameter space, but they can also be utilized to derive sensitivity indices [3]. It is important to design the framework in a modular way.

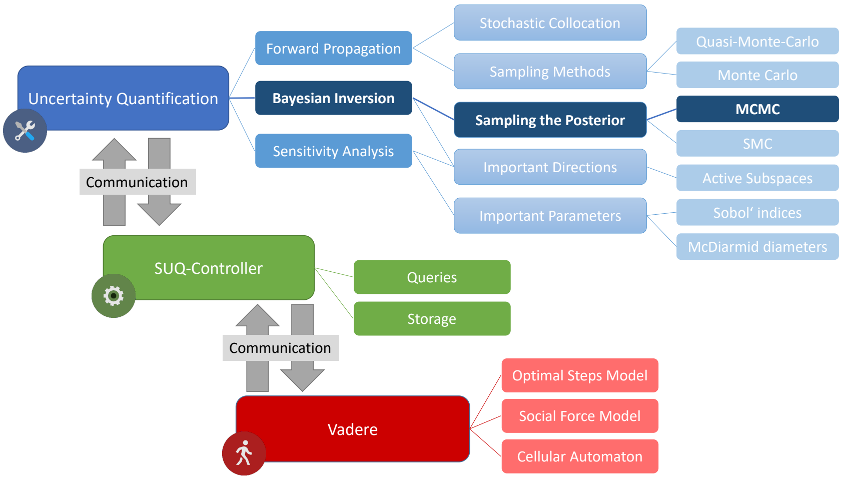


Figure 2: Scheme of the framework that is foreseen. The building blocks and pipeline part that we focus on in this work are highlighted in a darker blue. The uncertainty quantification framework will communicate via the SUQ-Controller with our Vadere simulation framework.

3 RESULTS AND DISCUSSION

In this section, we present numerical experiments. To carry out a proof of concept we start with the simple scenario described in Chapter 2.

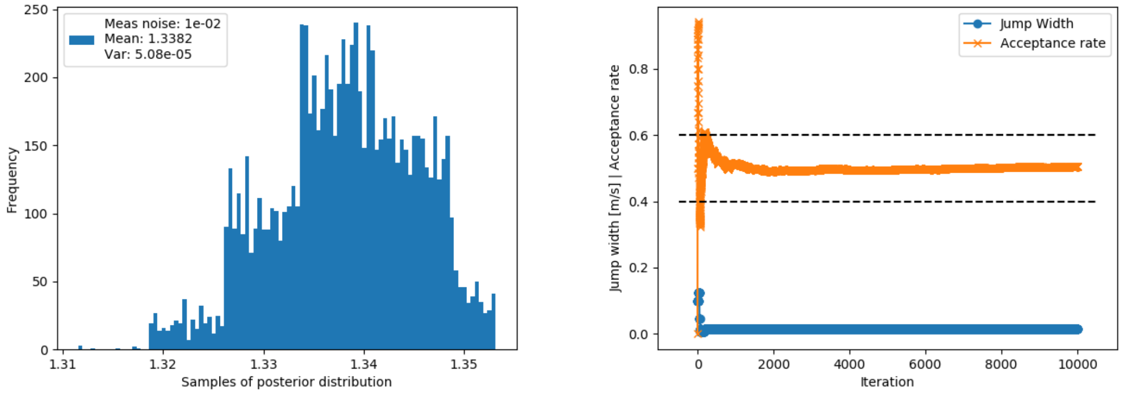
We observe that running one simulation of the proposed scenario takes about 1 second, that means 10^4 iterations of the Metropolis algorithm take about 2.8 hours. As a consequence, we perform only 10^4 iterations of the chain at first. Since the relationship between uncertain parameter and quantity of interest is strong and only one parameter is inferred, we can already see the chain converge to the true parameter even with the rather small number of iterations.

We start the Metropolis algorithm with the prior $\rho_{prior} \sim \mathcal{N}(2.5, 1.0)$ and a measurement noise of 10^{-2}. In Figure 3, we can see that the posterior distribution is no longer centered around the prior ($\mathcal{N}(2,1)$), but around the true parameter value of $1.34\,\frac{m}{s}$. The samples have a mean of 1.3382 and a variance of $5.0807 \cdot 10^{-5}$. This is the proof of concept that the method works sufficiently. In addition, the evolution of the acceptance rate shows that the adaptive regulation of the jump width succeeds at keeping the acceptance rate within the predefined limits.

3.1 Surrogate model

The amount of time spent on the model evaluation motivates the use of a surrogate model. There are two advantages regarding the speed when using MCMC with a surrogate model: First, the evaluations of the model can be replaced by evaluations of the surrogate model, which is typically much faster. Second, even though it takes time to generate the data points necessary for the model, this step can be easily parallelized (embarrassingly parallel).

In this explanatory setup, we only have one uncertain parameter and in addition, its relation to the quantity of interest is known. The service time is the quotient of travelled distance and speed. Therefore, it is clear which function should be fitted to the data. In addition, [21] states that free-flow speed is typically located between 0.5 and $2.2\,\frac{m}{s}$. Therefore, we evaluate the



(a) Histogram of posterior samples (without burn-in).

(b) Evolution of jump width and acceptance rate over time.

Figure 3: Proof of concept of applying Bayesian inversion to RiMEA test case 1: The inaccurate prior, centered around 2 was corrected over 10^4 iterations of the Metropolis algorithm. The resulting posterior is centered around 1.3382, close to the true parameter value of 1.34 m/s. The effective sample size is 1943.71.

model at equidistant candidates within this interval to find the base for the surrogate model. Figure 4 shows the evaluations of the model for different parameter values together with the fitted surrogate model. The high coefficient of determination shows that the surrogate is a good fit.

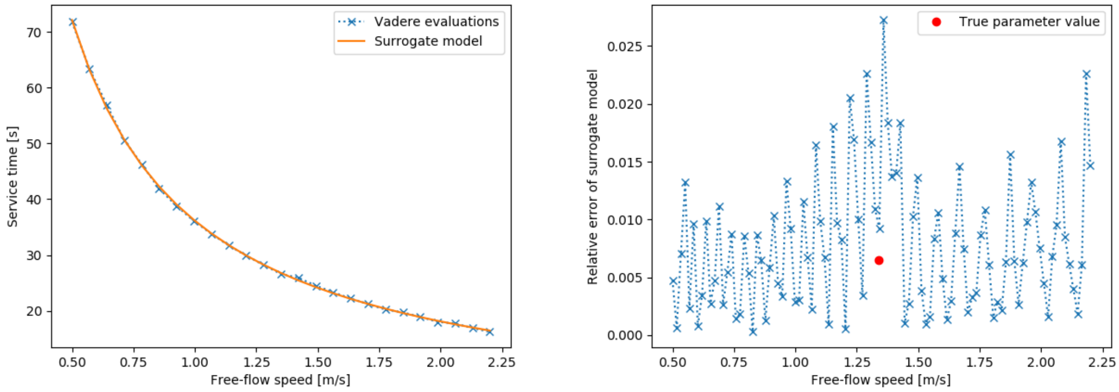
In our example, the usage of a surrogate model leads to another advantage: Due to the normal measurement noise, ρ_{prior} is conjugate prior for the likelihood. As a result, the posterior can be calculated analytically. In Figure 5, the results of the Bayesian inversion based on the surrogate model are presented. In addition to the posterior obtained from the samples in form of the histogram, we also show the analytical posterior. The histogram of the posterior samples obtained by the Metropolis algorithm fits the shape of the theoretical posterior perfectly. This is another indication that our sampling works and serves as verification of the code. When analyzing the mean of the posterior samples, one can observe that it is close to the true parameter value but even closer to the parameter value that produces a surrogate model output equal to the Vadere result for the true parameter. Here, one can see the impact of evaluating the surrogate model instead of the simulator itself. Nevertheless, the error is small enough to be neglected.

In our case, the surrogate model is simple. In higher dimensions, however, the construction of a surrogate model can be challenging. In [6] the application of forward propagation with a more elaborate surrogate model is presented.

3.2 Impact of measurement noise

Since the first proof of concept was successful, we take a look at the parameters of the inversion. One parameter is the measurement noise of the data provided. While this parameter cannot be varied when empirical data is used, in our exemplary setup, we are able to investigate the impact of the measurement noise on the obtained posterior. This variation serves as a plausibility check of the method.

In Figure 6, the histograms of the posterior samples are depicted for different levels of measurement noise. As expected, the width of the posterior distribution decreases with decreasing



(a) Surrogate model constructed from 25 simulations.

(b) Relative error of surrogate model evaluated at 100 reference points.

Figure 4: A surrogate model for the described configuration of Vadere. The coefficient of determination of the fit is $R^2 = 0.99975$. The relative error at the true parameter value is $6.5070 \cdot 10^{-3}$. Without any measurement noise, for the parameter of 1.34875 the result of the surrogate model is identical to the Vadere evaluation at the true parameter.

measurement noise. The measurement noise level provided to the algorithm can be seen as the level of trust that the algorithm puts into the data provided. If the data has only a low noise level, the estimate of the uncertain parameter is very accurate, hence the small posterior width.

3.3 Impact of Jump Width

Now we take a look at the impact of the jump width on two measures for MCMC performance described in chapter 2.2.2: acceptance rate and effective sample size. For the evaluations presented in this chapter, we have deactivated the adaptive jump width regulation and instead used a fixed jump width. We apply a measurement noise σ of 10^{-2}.

In Figure 7, the results of varying the jump width are laid out. As expected, the acceptance rate decreases with increasing jump width. In addition, the effective sample size shows that neither too large nor too small jump widths are favorable since both extremes lead to small effective sample sizes. While for too small jump widths the reasons of the low ESS results from the proximity of the candidates, for large jump widths, most candidates are rejected and therefore the number of unique samples is low. Both results are line with our expectations are verify our results as well as the implementation.

In addition, one run was performed with the adaptive regulation of the jump width. An effective sample size of 21020.32 is observed with a mean jump width of $1.04904 \cdot 10^{-2}$. The effective sample size obtained with this approach is better than with all fixed jump widths that we tried. This motivates altering the jump width based on the acceptance rate.

4 CONCLUSION AND OUTLOOK

In this work, we presented a first proof of concept application of Bayesian inversion with a Markov chain Monte Carlo method to a pedestrian crowd simulator. We applied the well-known Metropolis algorithm as a Markov chain Monte Carlo method and we chose the implementation of the optimal steps model within our framework Vadere as a crowd simulator. The reference

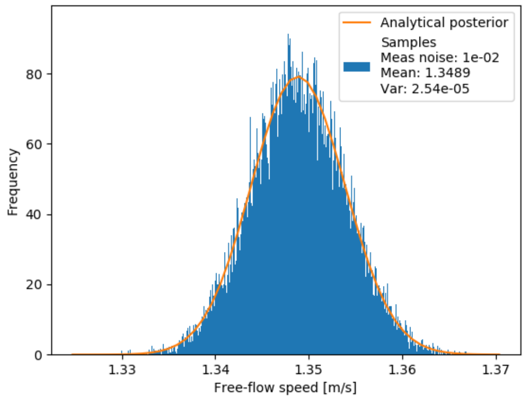


Figure 5: Results obtained with the surrogate model. The incorrect prior was centered around 2 ($\mathcal{N}(2, 1)$]). The center of the posterior is close to the true parameter value. In addition, the analytically derived posterior is depicted.

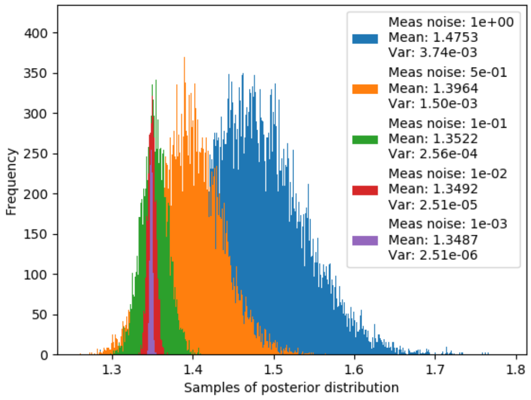
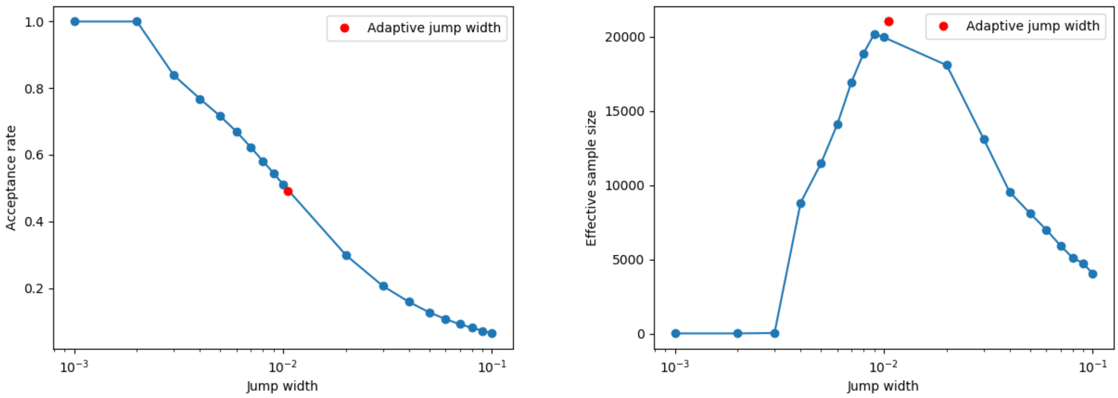


Figure 6: Histogram of the samples of the posterior obtained from the Metropolis algorithm for different levels of measurement noise. Surrogate model was created from 25 data points. For each noise level 10^5 iterations were performed.

scenario is simple but essential because it is the reference test case of the RiMEA guidelines for free-flow speed. A single pedestrian is walking through a hallway. We based the proof of concept on data obtained from the simulator itself instead of empirical data. Then, we started the inversion with an inaccurate prior. Our results show that the inversion is able to correct the information provided at the beginning by using information from model evaluations. The posterior is centered around the true parameter value. That means, this method is applicable to a typical parameter of pedestrian crowd simulations, the preferred speed (free-flow speed). In addition to the proof of concept application, we performed plausibility checks to verify the code.

Pedestrian crowd simulations become quickly computationally expensive when the number of pedestrian and / or the size of the scenario increases. Since Markov chain Monte Carlo methods are iterative methods that cannot be parallelized easily, the inversion can take a long time for a larger scenario. One approach to reduce the computation time is to construct a surrogate and evaluate the surrogate instead of the actual model in each iteration of the chain.



(a) Relation between jump width and acceptance rate.

(b) Relation between jump width and effective sample size.

Figure 7: Impact of the jump width on the effective sample size and the acceptance rate. In red, the results obtained in the same setup with the adaptive regulation of the jump width are shown.

Alternatively, another method can be used. We have demonstrated the usage of a surrogate model with our setup. However, with increasing dimensions, the construction of a surrogate becomes more challenging.

The work presented here serves as a first step towards a framework of uncertainty quantification methods for pedestrian dynamics. In the next step, the framework needs to be designed and built from modules such as Bayesian inversion, forward propagation and sensitivity analysis. During this process, it is crucial to pertain a modular structure to allow all useful combinations of building blocks. One module that can be linked to multiple other modules is the active subspaces module which can be combined either with the sensitivity analysis or the Bayesian inversion. The framework is aimed to provide support for users of pedestrian crowd simulations. It is planned to be an easy starting point for parameter analysis or sensitivity studies which should help users when calibrating parameters or comparing different models

ACKNOWLEDGEMENTS

The authors would like to thank Mario Teixeira Parente for discussions on Bayesian inversion and Markov chain Monte Carlo methods and Daniel Lehmberg for providing the SUQ-controller as a part of the Vadere framework.

This work was funded by the German Federal Ministry of Education and Research through the projects S2UCRE (grant no. 13N14464) and OPMOPS (grant no. 13N14562). The authors acknowledge the support by the Faculty Graduate Center CeDoSIA of TUM Graduate School at Technical University of Munich and the research office FORWIN at Munich University of Applied Sciences.

REFERENCES

[1] Vadere crowd simulation. online, 2016. https://gitlab.lrz.de/vadere/vadere.

[2] Nikolai Bode. Parameter calibration in crowd simulation models using approximate bayesian computation. In *Proceedings of Pedestrian and Evacuation Dynamics 2018*.

Springer (in press).

[3] Paul G. Constantine and Paul Diaz. Global sensitivity metrics from active subspaces. *Reliability Engineering & System Safety*, 162:1–13, 2017.

[4] Paul G. Constantine, Carson Kent, and Tan Bui-Thanh. Accelerating markov chain monte carlo with active subspaces. *SIAM Journal on Scientific Computing*, 38(5):A2779–A2805, 2016.

[5] Alessandro Corbetta, Adrian Muntean, and Kiamars Vafayi. Parameter estimation of social forces in pedestrian dynamics models via a probabilistic method. *Mathematical Biosciences and Engineering*, 12(2), 2015.

[6] Felix Dietrich, Florian Künzner, Tobias Neckel, Gerta Köster, and Hans-Joachim Bungartz. Fast and flexible uncertainty quantification through a data-driven surrogate model. *International Journal for Uncertainty Quantification*, 8:175–192, 2018.

[7] A. Gelman, G. O. Roberts, and W. R. Gilks. Efficient metropolis jumping rules. *Bayesian Statistics*, 5:599–607, 1996.

[8] P. G. Gipps and B. Marksjö. A micro-simulation model for pedestrian flows. *Mathematics and Computers in Simulation*, 27(2–3):95–105, 1985.

[9] Heikki Haario, Eero Saksman, and Johanna Tamminen. Adaptive proposal distribution for random walk metropolis algorithm. *Computational Statistics*, 14:375–395, 1999.

[10] Dirk Helbing and Péter Molnár. Social Force Model for pedestrian dynamics. *Physical Review E*, 51(5):4282–4286, 1995.

[11] Jari Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems*. Springer, Dordrecht, 2005.

[12] John K. Kruschke. *Doing Bayesian Data Analysis: A Tutorial with R, JAGS and Stan*. Academic Press, Inc., 2 edition, 2015.

[13] Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21:1087–1092, 1953.

[14] Peter Müller. A generic approach to posterior integration and gibbs sampling. Technical report, Purdue University, 1991.

[15] RiMEA. *Guideline for Microscopic Evacuation Analysis*. RiMEA e.V., 3.0.0 edition, 2016.

[16] G. O. Roberts, A. Gelman, and W. R. Gilks. Weak convergence and optimal scaling of random walk metropolis algorithms. *The Annals of Applied Probability*, 7(1):110–120, 1997.

[17] Michael J. Seitz and Gerta Köster. Natural discretization of pedestrian movement in continuous space. *Physical Review E*, 86(4):046108, 2012.

[18] Isabella von Sivers and Gerta Köster. Realistic stride length adaptation in the optimal steps model. In *Traffic and Granular Flow'13*, Jülich, Germany, 2013.

[19] Isabella von Sivers, Anne Templeton, Florian Künzner, Gerta Köster, John Drury, Andrew Philippides, Tobias Neckel, and Hans-Joachim Bungartz. Modelling social identification and helping in evacuation simulation. *Safety Science*, 89:288–300, 2016.

[20] Ulrich Weidmann. *Transporttechnik der Fussgänger*, volume 90 of *Schriftenreihe des IVT*. Institut für Verkehrsplanung, Transporttechnik, Strassen- und Eisenbahnbau (IVT) ETH, Zürich, 2nd edition, 1992.

[21] J. Zhang, Wolfram Klingsch, Andreas Schadschneider, and Armin Seyfried. Transitions in pedestrian fundamental diagrams of straight corridors and t-junctions. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(06):P06004, 2011.

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

UNCERTAINTY QUANTIFICATION OF NONLINEAR STOCHASTIC DYNAMIC PROBLEM USING A KRIGING-NARX SURROGATE MODEL

Biswarup Bhattacharyya[1], Eric Jacquelin[1], and Denis Brizard[1]

[1]Univ Lyon, Université Claude Bernard Lyon 1, IFSTTAR, LBMC UMR_T9406
F69622, Lyon, France
e-mail: {biswarup.bhattacharyya,denis.brizard}@ifsttar.fr
eric.jacquelin@univ-lyon1.fr

Keywords: Nonlinear stochastic dynamical system, Kriging, KNARX model, LARS, Uncertainty Quantification.

Abstract. *Uncertainty quantification of nonlinear stochastic dynamic problem is always a challenging task due to the complexity of the systems. In this paper, a hybrid surrogate modelling approach is proposed for the uncertainty quantification of nonlinear stochastic dynamical systems in the time domain. The proposed hybrid surrogate model is constructed using a nonlinear system identification tool, the Nonlinear AutoRegressive with eXogenous (NARX) input model, and the Kriging approach for uncertainty propagation. Further, to increase the computational efficiency, least angle regression (LARS) is utilized in the hybrid framework. The method is applied on a nonlinear stochastic dynamic oscillator to check its applicability. The time dependent mean and standard deviation are predicted using the proposed approach, and all the results are compared with the Monte Carlo simulation (MCS) results. A high-level accuracy is noticed using the proposed approach as compared to other state-of-the-art methods. This accuracy is achieved using a very limited number of model evaluations which is suggesting the efficiency of the proposed approach. Moreover, an excellent accuracy and efficiency is achieved using the proposed approach in predicting the probability density function (PDF) at several time instances for the nonlinear stochastic dynamic oscillator.*

1 INTRODUCTION

Uncertainty Quantification (UQ) of dynamical system has always been a great interest of research in the scientific community [1, 2, 3]. For the nonlinear stochastic dynamical systems, the stochastic response depends on the level of nonlinearity with respect to the uncertain variables. Therefore, an appropriate prediction of the stochastic response for the nonlinear dynamical systems is the main objective of this paper. Monte Carlo simulation (MCS) [1] is one of the main tools in state-of-the-art literature for UQ of any complex problem such as nonlinear dynamical system. However, the level of accuracy of the prediction gets increased with the increase of number of model evaluations which is the main limitation of this method. This issue has already been solved by several researchers in the past few decades with different surrogate models [4, 5, 3]. The main objective of most of the surrogate models is the reduction of the computation cost without compromising with the level of accuracy. Most of the surrogate models proposed for the stochastic dynamical systems have been based on polynomial chaos expansion (PCE) [6]. Initially, PCE was utilized to solve the stochastic partial differential equation (SPDE). Later on, PCE has been used for UQ in different class of problems including the dynamical systems [2, 7]. Therefore, several improvements have been made to apply the PCE for studying random dynamical systems.

The main computational burden in PCE for the dynamical system is the computation of the polynomial bases at each time-step. On the other hand, PCE works well for weakly nonlinear dynamical systems. However, for the strongly nonlinear dynamical systems, PCE requires large number of model evaluations with high degree polynomials which seems a similar limitation like MCS. Therefore, a few investigation has been made to address the issue of strong nonlinearity for the dynamical systems [7, 8, 9]. In this respect, recently a surrogate model was proposed by combining PCE with the Nonlinear AutoRegressive with eXogenous (NARX) input model [8, 3], this model is called PCE-NARX model. In this model, the problem of capturing the nonlinearity has been solved by the NARX model and the uncertain input parameters are propagated by the PCE model. However, the computation of the PCE model still requires high degree polynomials for the nonlinear dynamical systems [3] which is ultimately increasing the computational cost. On the other hand, the accuracy and the efficiency of the Kriging surrogate model [10, 11] has already been proved over the PCE model [12]. Therefore, to reduce the computational cost, the NARX model is formulated with the Kriging surrogate model in this paper.

The rest of the paper is organized as follows. A brief review of the Kriging surrogate model is described in the next section, and the computation of NARX model is illustrated briefly in section 3. Then, the proposed model is introduced in section 4 along with a suitable algorithm. Further, the applicability of the proposed model is illustrated through an example in section 5 and the conclusions drawn from this study are discussed in section 6.

2 REVIEW OF KRIGING

For an uncertain dynamical system, the d-dimensional random variables can be written in a vector form as $\hat{X} = \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_d\} \in \mathbb{R}^d$. The computation of the surrogate model is based on N samples of the random variables. The realizations of the random variables are denoted by the design of experiment (DoE) matrix $X = \{x_1, x_2, \ldots, x_d\} \in \mathbb{R}^{N \times d}$ (x_i is the i-th column of matrix X) and the corresponding responses are denoted by $Y = \{y(X_1), y(X_2), \ldots, y(X_N)\}^T$ (X_i is the i-th row of matrix X and is the i-th sample of the d random variables). Having the samples and the responses, the Kriging model performance function [10] is given by:

$$\mathcal{M}(X) = w^T \psi(X) + z(X) \tag{1}$$

In the above equation, $w^T \psi(X)$ represents the regression part of the polynomial, and $z(X)$ denotes the Gaussian process part. Various types of Kriging models are available in the literature according to the type of the polynomial $\psi(X)$. Out of all, an ordinary Kriging is used in this paper. The main parameters in the ordinary Kriging model are the coefficients in the regression part (w^T) and the Gaussian process part. The Gaussian process is modelled as zero mean with covariance:

$$\text{cov}\left[z(X_i, X_j)\right] = \sigma_z^2 r(X_i, X_j); \quad i, j = 1, 2, \ldots, N \tag{2}$$

where, σ_z^2 is the Gaussian process variance and $r(\bullet)$ is the auto-correlation function. Several auto-correlation functions are available in the literature [13, 10] and the Gaussian auto-correlation function is used for the present work.

The Kriging model predicts the response at the untried samples X_u by a best linear unbiased predictor (BLUB) which is given by:

$$\hat{\mathcal{M}}(X_u) = \hat{w}^T \psi(X_u) + \Re^T(X_u) \mathscr{R}^{-1} \left(\hat{w}^T \psi(X_u)\right) \tag{3}$$

where, $\Re(\bullet)$ is the correlation between the untried sample and the initial samples and $\mathscr{R}$ is the correlation matrix for the initial samples. Therefore, from the Kriging formulation [11, 12], the predicted coefficients are given by:

$$\hat{w} = \left(F^T \mathscr{R}^{-1} F\right)^{-1} F^T \mathscr{R}^{-1} Y \tag{4}$$

where F is the basis function matrix at the initial samples. The detail formulation procedure of the Kriging model parameters are given in [11]. The DACE toolbox [14] is utilized in this paper for the implementation of the Kriging model.

3 NARX MODEL

3.1 Full NARX model

The NARX model [15, 16] was proposed as the tool for nonlinear dynamic system identification. The basic phenomenon behind the formulation of the NARX model is that the dynamic response at a particular time-step is predicted by the responses of some previous time-steps and, the external force/excitation for some previous and present time-steps. Therefore, the response for a dynamic problem can be given by using the NARX model as:

$$y(t) = f\left[g(t)\right] + \varepsilon(t) \tag{5}$$

where, $g(t) = \{\xi(t_\tau), \xi(t_{\tau-1}), \xi(t_{\tau-2}), \ldots, \xi\left(t_{\tau-n_{\xi_m}}\right), y(t_{\tau-1}), y(t_{\tau-2}), \ldots, y\left(t_{\tau-n_{y_m}}\right)\}^T$ contains all the regressors for a dynamical system. ξ is the external force, y is the response and τ is the index of time-steps. n_{ξ_m} and n_{y_m} are the maximum time lags for the input and the response respectively. $f[\bullet]$ is the performance function for the NARX model and $\varepsilon(t)$ is the residual of the process in Equation 5. The main objective of the NARX model is to capture the nonlinearity of the dynamical systems. Therefore, the underlying performance function must have the nonlinear terms such that the nonlinearity can be captured easily. For the NARX model, several nonlinear performance functions have been proposed in the literature [17] which include polynomial, wavelet, sigmoid, radial basis function (RBF). Out of all, the simplest and widely used polynomial type performance function is chosen for the present study which is given by:

$$f\left[g\left(t\right)\right]=\sum_{i=1}^{M}\varpi_i\phi_i\left[g\left(t\right)\right] \tag{6}$$

where, ϖ_i are the coefficients corresponding to the polynomial bases $\phi_i\left[\bullet\right]$. Therefore, for the $\tau_{\max}$ number of time-steps, the polynomial basis matrix and the coefficient vector are given by:

$$\Phi_k\left(t_\tau, X_k\right)=\left[\phi_1\left[g_k\left(t_\tau, X_k\right)\right], \phi_2\left[g_k\left(t_\tau, X_k\right)\right], \ldots, \phi_M\left[g_k\left(t_\tau, X_k\right)\right]\right]^T \in \mathbb{R}^{M\times\tau_{\max}} \tag{7}$$

$$\varpi\left(X_k\right)=\left\{\varpi_1, \varpi_2, \ldots, \varpi_M\right\} \in \mathbb{R}^{1\times M} \tag{8}$$

In both the above equations, k in the subscript denotes the k-th sample point. Therefore, once the dynamic system is identified through Equation 5, the coefficients are available for the dynamic system which can be utilized to predict the response characteristics of the deterministic dynamical system. However, the coefficients of the NARX model (Equation 6) for a random dynamical system are random. For that reason, the NARX model is coupled with the Kriging model to propagate the uncertain characteristics of the system.

One of the important aspects of Equation 6 is the computation of the NARX coefficients. It is noticed from Equation 6 that the coefficients can be computed easily via the ordinary least square (OLS) method due to the form of the equation. However, it is evident from the previous studies [18, 8, 3] that all the terms (M terms) in the polynomial do not contribute for the system identification. Therefore, reducing the number of terms would make the system sparse in nature and would also enhance the computational efficiency. The important terms in the NARX model has been already identified in the literature by the Genetic algorithm [8] or by the least angle regression (LARS) [3]. In this paper, we have utilized the later one to identify the important terms in the NARX polynomial bases.

3.2 Formulation of the sparse NARX model

In a similar way to [3], a sparse NARX model is constructed in this paper. The main aim of this paper is to predict the stochastic response for the dynamical systems. The NARX model is constructed for the highly nonlinear samples as the representation of the stochastic system. For the selection of the highly nonlinear samples, the restoring force versus response curve is utilized in [8] and a threshold value approach is utilized in [3]. In this paper, a combination of these two approaches is adopted. Firstly, the restoring force versus response is plotted keeping the other values at their means until the nonlinearity is observed. Therefore, by locating the starting point of the nonlinearity, the threshold value for the dynamical system is decided. According to the threshold value, less number of samples would be retained as the nonlinear samples ($N_1 < N$). Therefore, only N_1 full NARX model are constructed in this step. Each of the full NARX model can be represented by Equation 5 as:

$$y\left(t_\tau, X_k\right)=\varpi\left(X_k\right)\Phi_k\left(X_k, t_\tau\right)+\varepsilon\left(t_\tau\right) \qquad k=1,2,\ldots,N_1 \tag{9}$$

In this step, the response series for the k-th sample point is known. Therefore, the polynomial basis matrix Φ_k is known beforehand. For the k-th sample point, the coefficient vector can be computed by minimizing the sum squared error which is given by:

$$\sum_{i=1}^{\tau_{\max}} \left[\varepsilon\left(t_i, X_k\right)\right]^2 = \sum_{i=1}^{\tau_{\max}} \left[y\left(t_i, X_k\right) - \varpi\left(X_k\right)\Phi_k\left(t_i, X_k\right)\right]^2 \tag{10}$$

The important terms for the full NARX model are captured at this step by the LARS [19] which leads the total terms in the polynomial to $M_1 < M$. Thereafter, the M_1 NARX coefficients are computed through the OLS using Equation 10. Similarly, for all the N_1 samples, the sparse NARX model is constructed. Therefore, at the end of this step, N_1 sparse NARX models are available.

Remark 1: It may happen that the same identical terms are captured in more than one sparse NARX model. For that reason, only the unique sparse NARX model are required to be selected at this step which may reduce the number of sparse NARX models to $N_2 \leq N_1$.

The N_2 number of unique sparse NARX models are utilized to reconstruct all the N initial response series separately and the relative error for each of the N samples is computed as follows:

$$\epsilon_k^p = \frac{\sum\limits_{i=1}^{\tau_{\max}} \left[y(t_i, X_k) - \hat{y}^p(t_i, X_k)\right]^2}{\sum\limits_{i=1}^{\tau_{\max}} \left[y(t_i, X_k) - \bar{y}(X_k)\right]^2} \quad k = 1, \cdots, N; \; p = 1, \cdots, N_2 \tag{11}$$

In Equation 11, $\hat{y}^p\left(\bullet\right)$ is the predicted response series at the k-th sample point using the p-th sparse NARX model and $\bar{y}\left(\bullet\right)$ is the mean of the actual time series which can be written as:

$$\bar{y}\left(X_k\right) = \frac{1}{\tau_{\max}} \sum_{i=1}^{\tau_{\max}} y\left(t_i, X_k\right) \tag{12}$$

After computing the relative error for all the sample points using a sparse NARX model, the mean relative error is computed as the mean of relative errors for all the samples. The mean relative error is given by:

$$\bar{\epsilon}^p = \frac{1}{N} \sum_{i=1}^{N} \epsilon_i^p \qquad p = 1, 2, \ldots, N_2 \tag{13}$$

Therefore, N_2 number of mean relative error is computed for the N_2 unique sparse NARX model. The final sparse model is selected as the one having the mean relative error less than a threshold value. For the current work, the threshold value is taken as 1×10^{-3}.

Remark 2: A situation may come that more than one unique sparse NARX model have the mean relative error less than the threshold value. In that case, the sparse NARX model having less number of terms is selected as the final sparse NARX model.

4 SPARSE KRIGING-NARX MODEL

The sparse NARX model has already been discussed in the previous section. Therefore, the sparse NARX model can be utilized to capture the strong nonlinearity of a dynamical system. However, the NARX model coefficients are random. Therefore, the independent surrogate model is constructed by combining the sparse NARX model with the Kriging model presented in this section. To formulate this, each of the coefficients of the final sparse NARX model are predicted by the Kriging model as:

$$\varpi_i\left(X\right) = w_i^T \psi\left(X\right) + z_i\left(X\right); \quad i = 1, 2, \ldots, M_1 \tag{14}$$

In the above equation, the coefficients of the sparse NARX model for the initial N samples are considered as the response quantity and X is the DoE matrix. Therefore, combining this equation with the sparse NARX model, the sparse Kriging-NARX (KNARX) model is given by:

$$y\left(t,X\right)=\sum_{i=1}^{M_1}\left(w_i^T\psi\left(X\right)+z_i\left(X\right)\right)\phi_i\left[g\left(t,X\right)\right] \tag{15}$$

The Kriging model in Equation 15 are constructed M_1 times to make the predicted model independent. The predictions at the untried samples are made by the BLUP estimator in accordance with the sparse NARX model which is given by:

$$\hat{y}\left(t,X_u\right)=\sum_{i=1}^{M_1}\left[\hat{w}_i^T\psi\left(X_u\right)+\Re^T\left(X_u\right)\mathscr{R}^{-1}\left(\hat{w}_i^T\psi\left(X_u\right)\right)\right]\phi_i\left[g\left(t,X_u\right)\right] \tag{16}$$

where $\hat{w}_i$ are the computed coefficients of the Kriging model.

The above-mentioned equation is utilized for the prediction of the stochastic dynamic responses in the time domain. Accordingly, the solution is made through two steps as discussed above. In the first step, the coefficients of the sparse NARX model are predicted by the Kriging surrogate model, and in the second step, the stochastic dynamic response is predicted by the sparse NARX model. A step by step flowchart for constructing the sparse KNARX model is given in Figure 1.

Remark 3: One of the important aspects in constructing the sparse KNARX model is the choice of the maximum time lags n_{ξ_m} and n_{y_m} for the input and the response respectively. In a similar way to [3], in the present paper these two maximum time lags are chosen as two times the number of degrees of freedom (DOF) of the dynamical system.

5 NUMERICAL APPLICATION TO A HALF OSCILLATOR

The sparse KNARX model (Figure 1) as developed in the previous section is applied to a simple nonlinear dynamical system for UQ. More specifically, a half oscillator [20], defined by Equation 17, is considered to check the applicability of the sparse KNARX model. Along with this, the result is also computed with the recently proposed sparse PCE-NARX model [3] and all the predicted results are compared with the full scale MCS results. The accuracy of the surrogate model predicted results are measured by the mean relative error (Equation 13) and the coefficient of correlation (R^2). For the construction of the surrogate models, the initial sample points are generated using the Latin hypercube sampling (LHS) scheme in the present work.

The governing differential equation of the half oscillator is given by:

$$\dot{y}\left(t\right)+\nu y\left(t\right)+\varepsilon y^3\left(t\right)=A\sin\left(\omega_\xi t\right) \tag{17}$$

where ν and ε are the system parameters, and A and ω_ξ are the parameters of the sinusoidal excitation. All the parameters of this problem are uncertain. The distribution type of all the uncertain parameters are presented in Table 1.

The main goal is to quantify the uncertain response parameter $y\left(t\right)$ of the half oscillator due to the uncertain input parameters as mentioned in Table 1. For the solution of the system, time integration has been performed for a total time of $T=30\,\mathrm{s}$ at a time-step $\Delta t=0.01\,\mathrm{s}$ using the MATLAB solver *ode45*. The initial condition for this problem is considered as $y\left(0\right)=0$. The problem has been solved by MCS, sparse PCE-NARX [3] and sparse KNARX model to predict the time dependent uncertain response quantity. MCS has been performed with 3×10^4 number

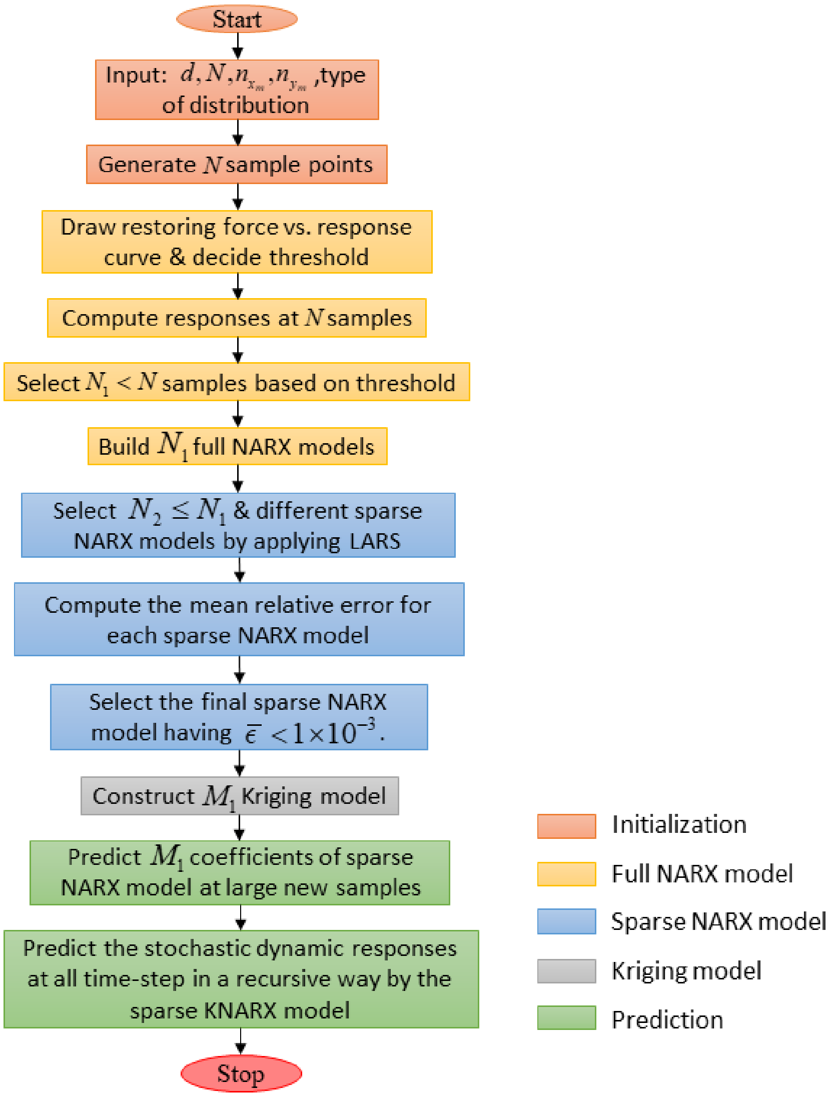

Start
Input: d, N, n_{x_m}, n_{y_m}, type of distribution
Generate N sample points
Draw restoring force vs. response curve & decide threshold
Compute responses at N samples
Select $N_1 < N$ samples based on threshold
Build N_1 full NARX models
Select $N_2 \leq N_1$ & different sparse NARX models by applying LARS
Compute the mean relative error for each sparse NARX model
Select the final sparse NARX model having $\bar{\epsilon} < 1 \times 10^{-3}$.
Construct M_1 Kriging model
Predict M_1 coefficients of sparse NARX model at large new samples
Predict the stochastic dynamic responses at all time-step in a recursive way by the sparse KNARX model
Stop
Initialization
Full NARX model
Sparse NARX model
Kriging model
Prediction

Figure 1: Flowchart to construct the sparse KNARX model

Table 1: Uncertain parameters for the half oscillator

Variables	Distribution type	Mean	Standard deviation	Unit
ν	Uniform	1	$\frac{0.15}{\sqrt{3}}$	–
ε	Uniform	1	$\frac{0.1}{\sqrt{3}}$	–
A	Normal	0.6	0.06	N
ω_ξ	Normal	1	0.1	$\mathrm{rad\,s^{-1}}$

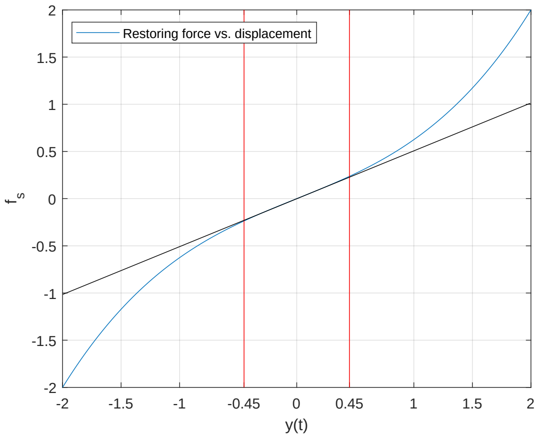


Figure 2: Restoring force versus displacement for the half oscillator

of sample points which is considered as the reference response characteristic and the error for the other methods is predicted with respect to the MCS result.

At the initial step, the restoring force $(f_s = \nu y(t) + \varepsilon y^3(t))$ versus the displacement $(y(t))$ is plotted in Figure 2 by fixing all other values at their means. It is seen from the figure that the displacement shows nonlinear behavior beyond the region $y(t) \in [-0.45\mathrm{m}, 0.45\mathrm{m}]$. Therefore, the threshold for the nonlinearity is decided as $\max |y(t)| > 0.45\mathrm{m}$. This threshold criterion is imposed on the oscillator to choose the highly nonlinear samples for constructing the surrogate models.

For the construction of the full NARX model, a suitable polynomial basis function is chosen which is given by:

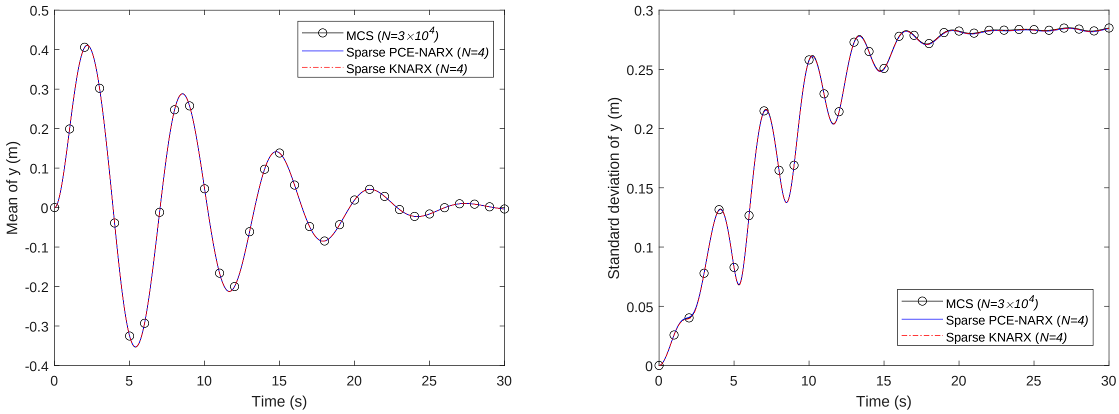
$$\phi_i [g(t)] = \xi^{l_i}_{\tau - n_{\xi_i}} y^{m_i}_{\tau - n_{y_i}} \tag{18}$$

In Equation 18, ξ and y are the excitation and the response of the half oscillator respectively. Right hand side of Equation 17 is the excitation part i.e. $\xi_\tau = A \sin(\omega_\xi t)$. n_ξ and n_y are the time lags for the excitation and the response respectively, whereas, l and m are the corresponding maximum degrees. It is important to note that for the simple formulation, the excitation and the response are expressed without the function t. For the half oscillator, $l = 1$ and $m = 3$ are chosen with a maximum degree of the polynomial i.e. $l + m \leq 3$ due to the cubic non-linearity of the problem. The maximum time lags are chosen as twice the number of DOF [8, 3] of the half oscillator i.e. 2 with $n_\xi = \{0, 1, 2\}$ and $n_y = \{1, 2\}$. As a result, 22 number of terms are found in the polynomial basis matrix for the full NARX model utilizing all the possible combinations including the constant term i.e. $i = 1, 2, \ldots, 22$ in Equation 18.

For the surrogate models, $N = 4$ number of initial sample points are generated using LHS which are listed in Table 2. Initially, the samples exhibiting high order non-linearity are selected based on the threshold value of displacement as decided previously. Therefore, $N_1 = 1$ sample is detected as the nonlinear sample (4-th sample in Table 2). Further, the only full NARX model is constructed and the most important terms for the NARX model are detected by the LARS algorithm. This procedure transforms the full NARX model in a sparse NARX model by reducing the total number of terms in the NARX polynomial basis. The final sparse NARX model is selected which predicts the mean error for all the N samples less than 1×10^{-3} and in this case, only 1 unique sparse NARX model was found. For that unique sparse NARX model, the coefficients for N samples are calculated by OLS method and the predicted mean

Table 2: Samples for all the random variables of the half oscillator generated using LHS

Sample number	ν	ε	A	ω_ξ
1	1.1175	1.0880	0.5840	0.9887
2	0.9772	0.9825	0.6237	1.1855
3	0.9174	1.0221	0.5390	1.0662
4	1.0376	0.9226	0.6885	0.8460



(a) Mean

(b) Standard deviation

Figure 3: Statistical response characteristics of the half oscillator

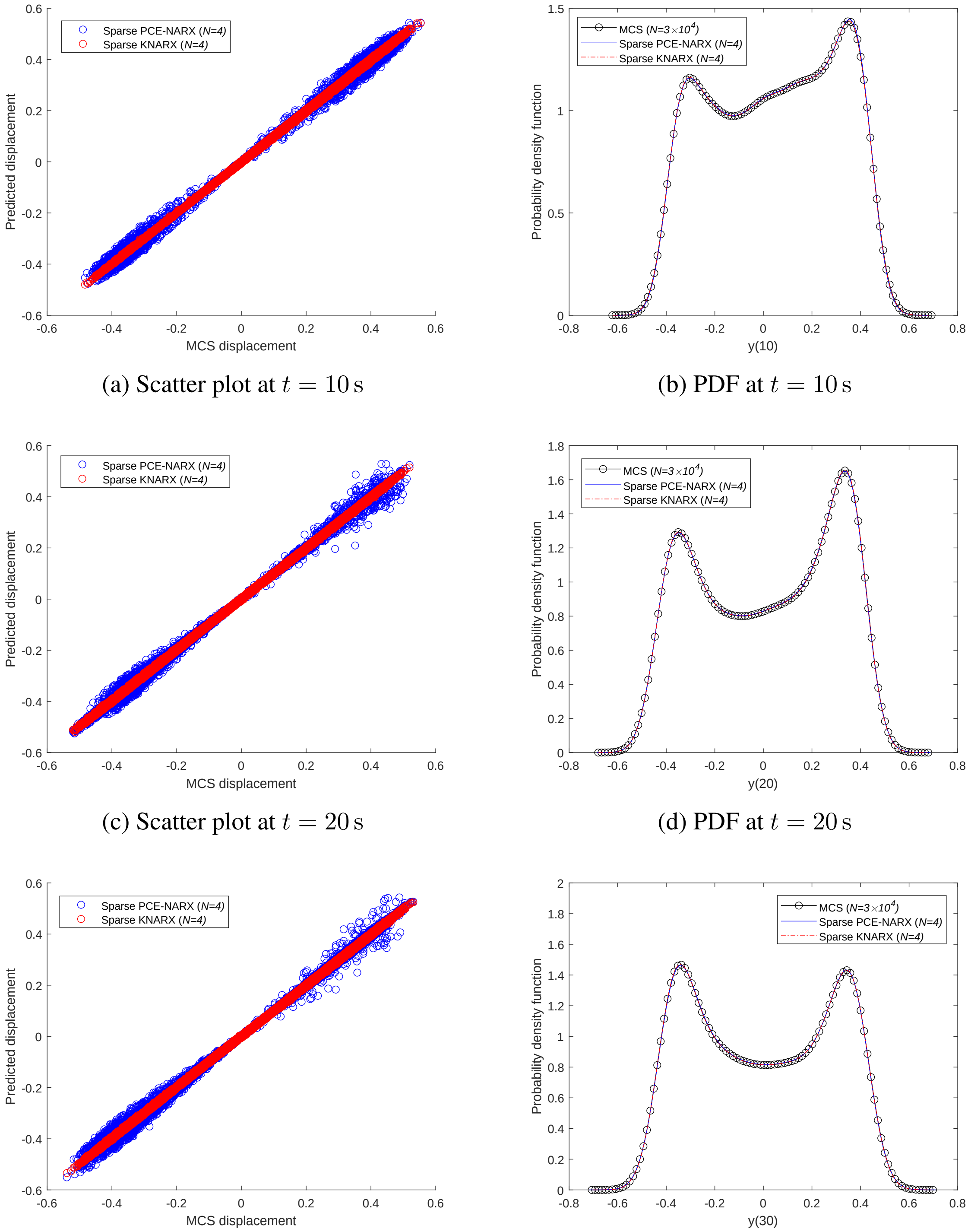
error is found as $\bar{\epsilon} = 9.16 \times 10^{-8}$ for $N = 4$ samples which satisfies the threshold criterion for selecting the sparse NARX model. It is observed that the selected final sparse NARX model has 5 terms in the polynomial basis matrix which are $\left\{y_{\tau-1}, y^3_{\tau-2}, \xi_{\tau-1}, \xi_{\tau-2}, \xi_{\tau-2}y^2_{\tau-2}\right\}$. Therefore, 5 surrogate models are required to be identified for the prediction of the stochastic response.

For the purpose of uncertainty quantification, the time dependent mean and standard deviation are plotted in Figure 3 by MCS $(N = 3 \times 10^4)$, sparse PCE-NARX $(N = 4)$ and by sparse KNARX model $(N = 4)$. It is clearly seen from Figure 3 that the time varying mean and standard deviation are predicted quite well by both sparse PCE-NARX and sparse KNARX model.

To compare the response characteristics at certain time instances, the scatter diagrams (the plot between the MCS and the predicted response) and the PDFs of the displacement are plotted at $10\,\mathrm{s}$, $20\,\mathrm{s}$ and $30\,\mathrm{s}$ in Figure 4. The R^2 value and the error (Equation 11) of the corresponding responses are listed in Table 3. Both surrogate models exhibit very promising results. However, the sparse KNARX outperforms the sparse PCE-NARX in predicting the R^2 value and the error $\epsilon_{y(t)}$ for all the time instances.

Further, the uncertain maximum absolute displacement $\max\left(|y\left(t\right)|\right)$ is also predicted by both the surrogate models which ultimately measures the safety of the system. The scatter diagram and the PDF of the $\max\left(|y\left(t\right)|\right)$ are plotted in Figure 5. It can be observed from the figure that the sparse KNARX outperforms the sparse PCE-NARX significantly in predicting $\max\left(|y\left(t\right)|\right)$.

For the overall assessment on the accuracy and the efficiency of the surrogate models, the overall mean error of the models is predicted by Equation 13 on 3×10^4 number of samples in Table 4. Along with this, the CPU times are also reported in Table 4. The error in predicting the



(a) Scatter plot at $t = 10\,\mathrm{s}$

(b) PDF at $t = 10\,\mathrm{s}$

(c) Scatter plot at $t = 20\,\mathrm{s}$

(d) PDF at $t = 20\,\mathrm{s}$

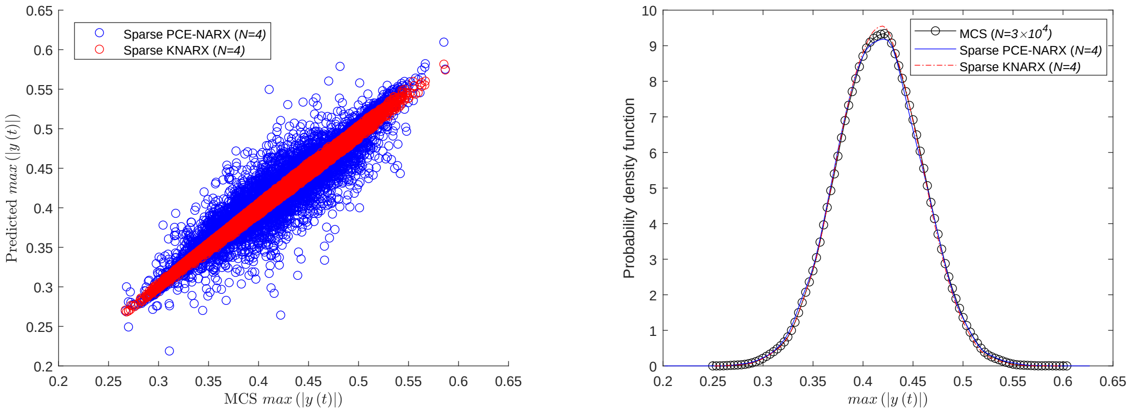
(e) Scatter plot at $t = 30\,\mathrm{s}$

(f) PDF at $t = 30\,\mathrm{s}$

Figure 4: Comparison of the instantaneous response characteristics at different time instances for the half oscillator

Table 3: Accuracy of the surrogate models in predicting the instantaneous response characteristics for the half oscillator

Method	Time instance	$\epsilon_{y(t)}$	R^2
Sparse PCE-NARX	$t = 10\,\mathrm{s}$	1.20×10^{-3}	0.9988
Sparse KNARX		6.62×10^{-5}	0.9999
Sparse PCE-NARX	$t = 20\,\mathrm{s}$	7.39×10^{-4}	0.9993
Sparse KNARX		5.05×10^{-5}	0.9999
Sparse PCE-NARX	$t = 30\,\mathrm{s}$	7.64×10^{-4}	0.9992
Sparse KNARX		4.95×10^{-5}	1.0000



(a) Scatter plot

(b) PDF

Figure 5: Prediction of the $\max\left(|y\left(t\right)|\right)$ for the half oscillator

Table 4: Prediction of the accuracy and the efficiency of the surrogate models for the half oscillator

Method	$\bar{\epsilon}$	$\epsilon_{\max(\lvert y(t)\rvert)}$	$R^2_{\max(\lvert y(t)\rvert)}$	CPU time
Sparse PCE-NARX	7.73×10^{-4}	7.01×10^{-2}	0.9299	$24.34\,\mathrm{s}$
Sparse KNARX	4.87×10^{-5}	4.30×10^{-3}	0.9957	$23.18\,\mathrm{s}$
MCS	–	–	–	$404.80\,\mathrm{s}$

$\max(|y(t)|)$ (using Equation 11) and R^2 value are reported in Table 4. All the results depict that both the sparse KNARX and the sparse PCE-NARX are very efficient and accurate, even if the former slightly outperforms the latter.

6 CONCLUSIONS

UQ of a dynamical system has been addressed in this paper. The main focus has been made towards developing an accurate and efficient surrogate model for solving the UQ problem of the nonlinear stochastic dynamical system in the time domain. In this line, a surrogate sparse KNARX model has been proposed in a similar way to [3]. The main concept of the proposed surrogate model is capturing the nonlinear behavior of a dynamical system through the NARX model and further, propagating the uncertainty by Kriging. The applicability of the proposed model has been shown through a simple nonlinear dynamic oscillator. Further, the implementation of this model on the multi degree of freedom system can be considered as the future scope of this study.

REFERENCES

[1] M. Grigoriu, Response of dynamic systems to poisson white noise. *Journal of Sound and Vibration*, **195** (3), 375–389, 1996.

[2] D. Lucor, C. H. Su, G. E. Karniadakis, Generalized polynomial chaos and random oscillators. *International Journal for Numerical Methods in Engineering*, **60** (3), 571–596, 2004.

[3] C. V. Mai, M. D. Spiridonakos, E. N. Chatzi, B. Sudret, Surrogate modeling for stochastic dynamical systems by combining nonlinear autoregressive with exogenous input models and polynomial chaos expansions. *International Journal for Uncertainty Quantification*, **6** (4), 313–339, 2016.

[4] X. Wan, G. E. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *Journal of Computational Physics*, **209** (2), 617–642, 2005.

[5] O. P. L. Maitre, L. Mathelin, O. Knio, M. Hussaini, Asynchronous Time Integration for Polynomial Chaos Expansion of Uncertain Periodic Dynamics. *Discrete Continuum Dynamic Systems - Series A*, **28** (1), 199–226, 2010.

[6] D. Xiu, G. E. Karniadakis, The Wiener-Askey polynomial chaos for stochastic differential equation. *SIAM Journal on Scientific Computing Scientific Computing*, **24** (2), 619–644, 2002.

[7] M. Gerritsma, J.-B. V. D. Steen, P. Vos, G. Karniadakis, Time-dependent generalized polynomial chaos. *Journal of Computational Physics* **229** (22), 8333–8363, 2010.

[8] M. Spiridonakos, E. Chatzi, Metamodeling of dynamic nonlinear structural systems through polynomial chaos NARX models. *Computers & Structures* **157**, 99–113, 2015.

[9] C. V. Mai, B. Sudret, Surrogate models for oscillatory systems using sparse polynomial chaos expansions and stochastic time warping. *SIAM/ASA Journal on Uncertainty Quantification*, **5** (1), 540–571, 2017.

[10] J. Sacks, W. J. Welch, T. J. Mitchell, H. P. Wynn, Design and Analysis of Computer Experiments. *Statistical Science*, **4** (4), 409–423, 1989.

[11] T. Santner, B. Williams, W. Notz, *The design and analysis of computer experiments, 1st Edition*. Springer, 2003.

[12] B. Bhattacharyya, A Critical Appraisal of Design of Experiments for Uncertainty Quantification. *Archives of Computational Methods in Engineering*, **25** (3), 727–751, 2018.

[13] I. Kaymaz, Application of kriging method to structural reliability problems. *Structural Safety*, **27** (2), 133–151, 2005.

[14] H. B. Nielsen and S. N. Lophaven and J. Søndergaard, *DACE - A Matlab Kriging Toolbox*. Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2002.

[15] S. Chen, S. A. Billings, Modelling and analysis of non-linear time series. *International Journal of Control*, **50** (6), 2151–2171, 1989.

[16] S. A. Billings, *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains, 1st Edition*. Wiley, 2013.

[17] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, A. Juditsky, Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, **31** (12), 1691–1724, 1995.

[18] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression. *Journal of Computational Physics*, **230** (6), 2345–2367, 2011.

[19] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression. *The Annals of Statistics*, **32** (2), 407–499, 2004.

[20] G. Muscolino, G. Ricciardi, P. Cacciola, Monte Carlo simulation in the stochastic analysis of non-linear systems under external stationary Poisson white noise input. *International Journal of Non-Linear Mechanics*, **38** (8), 1269–1283, 2003.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

VIBRATION BASED STRUCTURAL HEALTH MONITORING OF COMPOSITE CARBON FIBER STRUCTURAL SYSTEMS

Ilias Zacharakis[1], Alexandros Arailopoulos[1], Olga Markogiannaki, Dimitrios Giagopoulos*[1]

[1] Department of Mechanical Engineering, University of Western Macedonia
Kozani, Greece
izacharakis, aarailopoulos, omarkogiannaki, dgiagopoulos@uowm.gr

Abstract

A vibration based computational framework for damage identification of composite cylindrical parts, produced on a spinning axis by winded carbon fibers, cascaded on specified number of plies, in various angles and directions, was presented in this work. First, a discrete FE model of the examined structure is developed, by consecutive shell and solid elements, simulating each carbon fiber ply and resin matrix. Focusing on the updating methodology, coupled with robust, accurate and efficient finite element analysis software, the linear and non-linear behavior of the composite parts was examined under various load conditions followed by equivalent experimental trials, in order to classify the material properties (isotropic, orthotropic, anisotropic) and develop a high-fidelity FE model. This is achieved through combining modal residuals, that include the lowest identified modal frequencies and mode shapes, with response residuals, that include shape and amplitude correlation coefficients considering measured and analytical frequency response functions and time-histories of strains and accelerations. Single objective structural identification strategies without the need of sub-structuring methods, are used for estimating the parameters (material properties in each deformation plane) of the finite element model, based on minimizing the deviations between the experimental and analytical dynamic characteristics. A stochastic optimization evolution strategy is applied in parallel computing, to solve the single-objective optimization problem, arising from combining the above residuals. The effect of model error, finite element model parameterization, number of measured modes and number of mode shape components on the optimal models along with and their variability, are examined.

Keywords: Modal identification, Model updating, Large Scale Structures, Structural Dynamics

1 INTRODUCTION

Carbon fiber reinforced polymer (CFRP) composites have gained much attention in recent years through their industrialized implementation and use, as a structural material for static and dynamic load bearing as well as resistance to accidental excitations and actions. Due to its low-density, low thermal expansion and high strength, stiffness and corrosion resistance, applications from aerospace and automotive industry to building reinforcement and retrofit, as well as cryogenic fuel storage tanks are emerging rapidly [1-6]. CFRP composites are manufactured on a spinning axis of various radii, by compressing multiple cascaded plies of pre-tensed carbon fibers, which are winded in certain volume fractions and patterns of angles and directions, against a liquid resin polymer matrix. The final product is obtained after leaving the composite material in a furnace for specific duration in order to achieve full strength and hardening characteristics [7-9]. Being inherently sensitive to manufacturing treatment and due to its material variability, CFRPs strongly require certification results through numerical validation and hybrid (numerical - experimental) verification [10].

The most popular carbon fiber-reinforced composites, which have been extensively investigated by researchers, are the plain-woven CFRPs. Their popularity is attributed mainly to the low production cost combined to their effectiveness and efficiency under in-plane loading conditions. Presenting tension-compression asymmetric characteristics and orthotropic or even strong anisotropic mechanical behavior, due to varying fiber patterns, plain woven CFRPs are categorized to matrix-dominant presenting low strength and to fiber-dominant presenting high strength [11]. Thus, it is of high importance to fully understand and grow high confidence about the mechanical behavior and in-plane loading capacity of each CFRP made structure. Moreover, as most engineering applications require multi-axial loading strength, their behavior in such loading conditions need also to be examined. Combined experimental measurements, conducted in and out of laboratory, to numerical Finite Element (FE) model simulations are employed in order to investigate in the macroscopic mechanical characteristics and material properties of CFRP structures [10, 12, 13].

In this work, the material properties of a specific woven CFRP structure are classified and tuned reconciling experimental data to equivalent numerical (FE) model computations. This is achieved through combining modal residuals, that include the lowest identified modal frequencies and mode shapes, with response residuals, that include shape and amplitude correlation coefficients considering measured and analytical frequency response functions and time-histories of strains and accelerations [14-19]. Single objective structural identification strategies without the need of sub-structuring methods, are used for estimating the parameters of the finite element model. A state-of-the-art optimization algorithm, namely, covariance matrix adaptation evolution strategy (CMA-ES) [20-23], is applied in parallel computing, to solve the single-objective optimization problem, arising from combining the above residuals [24, 25]. The applicability and effectiveness of the methods applied, is explored by updating the finite element model of a lightweight small-scale CFRP pin-joined structure. Issues related to estimating unidentifiable solutions [26-29] arising in FE model updating formulations are also addressed. A systematic study is carried out to demonstrate the effect of model error, finite element model parameterization, number of measured modes and number of mode shape components on the optimal models and their variability.

The presentation in this work is organized as follows. The theoretical formulation of finite element model updating based on modal characteristics, frequency response functions is briefly presented in section 2. Section 3 presents the adopted residual in time domain. Section 4 presents the experimental application, the development of the FE model of small-scale cantilever CFRP beam, its modal identification along with the FE model updating parameterization and results for orthotropic material characterization. Finally, in section 5 the updated orthotropic material properties are verified on a small-scale pin-joined CFRP frame under dynamic excitation comparing experimental data and numerical results. Conclusions are summarized in section 6.

2 MODAL AND FREQUENCY RESPONSE RESIDUALS – LINEAR MODELS

Let a parameterized class of linear structural models used to model the dynamic behavior of the structure and let $\underline{\theta} \in R^{N_\theta}$ be the set of free structural model parameters to be identified using the measured modal data. The overall measure of fit of the linear model, between the measured and the model predicted characteristics is formed in the following expression, combining modal and frequency response residuals [30, 31]:

$$J(\underline{\theta};\underline{w}) = w_1 J_1(\underline{\theta}) + w_2 J_2(\underline{\theta}) + w_3 J_3(\underline{\theta}) + w_4 J_4(\underline{\theta}) \tag{1}$$

using equally weighting factors $w_i \geq 0$, $i = 1,2,3,4$, with $w_1 + w_2 + w_3 + w_4 = 1$.

For the first group, the measure of fit $J_1(\underline{\theta})$ is selected to represent the difference between the measured and the model predicted frequencies for all modes. For the second group, the measure of fit $J_2(\underline{\theta})$ is selected to represent the difference between the measured and the model predicted mode shape components for all modes, given by:

$$J_1(\underline{\theta}) = \sum_{r=1}^{m} \varepsilon^2_{\omega_r}(\underline{\theta}) \;\; and \;\; J_2(\underline{\theta}) = \sum_{r=1}^{m} \varepsilon^2_{\underline{\phi}_r}(\underline{\theta}) \tag{2}$$

where the modal data are used $\{\omega_r(\underline{\theta}),\ \underline{\phi}_r(\underline{\theta}) \in R^{N_0}, r = 1,\cdots,m\}$ to formulate the following residuals:

$$\varepsilon_{\omega_r}(\underline{\theta}) = \frac{\omega_r^2(\underline{\theta}) - \hat{\omega}_r^2}{\hat{\omega}_r^2} \quad and \quad \varepsilon_{\underline{\phi}_r}(\underline{\theta}) = \frac{\left\| \beta_r(\underline{\theta})\underline{\phi}_r(\underline{\theta}) - \hat{\underline{\phi}}_r \right\|}{\left\| \hat{\underline{\phi}}_r \right\|} \tag{3}$$

and for the second group the measure of fit $J_3(\theta)$ and $J_4(\theta)$ represent the frequency response measures of fit as follows:

$$J_3(\underline{\theta}) = \sum_{r=1}^{m} \left[1 - x_s(\hat{\omega}_r, \underline{\theta})^2\right] \text{ and } J_4(\underline{\theta}) = \sum_{r=1}^{m} \left[1 - x_a(\hat{\omega}_r, \underline{\theta})^2\right] \tag{4}$$

where

$$x_s(\omega_k) = \frac{\left|\{H_X(\omega_k)\}^H \{H_A(\omega_k)\}\right|^2}{\left(\{H_X(\omega_k)\}^H \{H_X(\omega_k)\}\right)\left(\{H_A(\omega_k)\}^H \{H_A(\omega_k)\}\right)} \tag{5}$$

and

$$x_a(\omega_k) = \frac{2\left|\{H_X(\omega_k)\}^H \{H_A(\omega_k)\}\right|}{\left(\{H_X(\omega_k)\}^H \{H_X(\omega_k)\}\right) + \left(\{H_A(\omega_k)\}^H \{H_A(\omega_k)\}\right)} \tag{6}$$

constitute the global and amplitude correlation coefficients [32], where $\{H_X(\omega_k)\}$ and $\{H_A(\omega_k)\}$ are the experimental (measured) and the numerical (predicted) response vectors at matching excitation - response locations, for any measured frequency point, ω_k .

3 TIME DOMAIN RESPONSE RESIDUALS – NONLINEAR MODELS

Additionally, parameter estimation of nonlinear model is based on response time history measurements such as acceleration and displacements. This formulation has the advantage of applicability over both linear and non-linear systems; it compares the measured raw data of the experimental arrangement to the equivalent predictions of the numerical model. In this way, all available information is preserved and systematic errors of the identification procedure are alleviated.

The measure of fit is given by:

$$J\left(\underline{\theta};M\right)=\frac{1}{m}\sum_{i=1}^{n}\frac{\sum_{j=1}^{m}\left(g_{ij}\left(\underline{\theta}_m \mid M\right)-\hat{y}_{ij}\right)^2}{\sum_{j=1}^{m}\left(\hat{y}_{ij}\right)^2} \qquad (7)$$

where $g_{ij}\left(\underline{\theta}_m \mid M\right)$ is the numerical time-history of the introduced FE model and $\hat{y}_{ij}$ is the respective experimental signal. Subscripts i correspond to the sensor (accelerometer) location and measurement direction, and j corresponds to the time-step instant. n is the total number of measured sensor locations and directions, whereas m is the total number of measured time-steps (number of observations).

4 EXPERIMENTAL APPLICATION

In order to examine the complexity and orthotropic material mechanical behavior of the used CFRP, two types of experimental arrangements were set. Firstly, a static tension-compression experimental test, as presented in **Figure 1**. Specifically, in this figure shown the tension-compression experimental device and its controller board, the strain gauge sensors placed on the CFRP tube and the equivalent FE model.

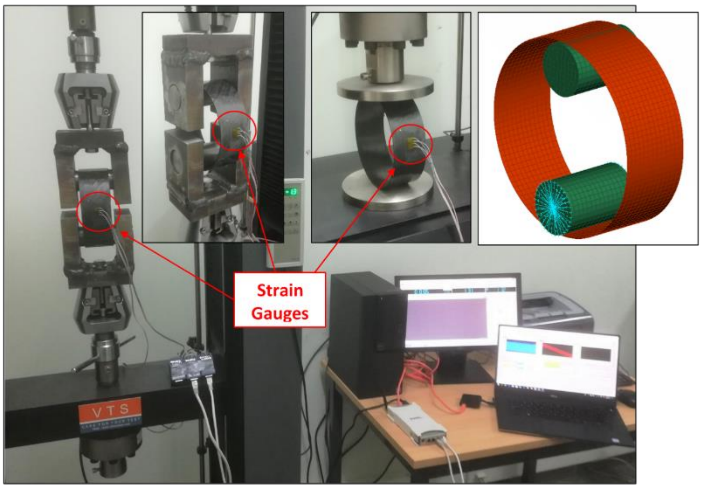


Figure 1 Experimental setup of composite cylindrical tube in tension-compression test device.

Moreover, dynamically induced excitation tests were conducted at a cantilever CFRP tubular beam as presented in **Figure 2**. Specifically, **Figure 2** presents the cantilever CFRP small-radius tube along with two (2) tri-axial accelerometers, a strain gauge sensor and a load cell at the free end of the cantilever beam, where an electromagnetic shaker device is mounted. Both arrangements were introduced in order to acquire knowledge of the mechanical behavior of the CFRP material and thus characterize its orthotropic behavior.

The CFRP is consisted of a stack of nine (9) plies with equal thickness and orientation angles apart from one ply. Specifically, plies 1 to 6 and 8 to 9 have a thickness of $t=0.175mm$ at q=55° and q=- 55° orientation angles consecutively. Ply 7 has a thickness of $t=0.16mm$ at q=86°. The nominal material parameters of the 2D orthotropic material used to model the CFRP was $E_1=146{,}45GPa$ and $E_2=7.73GPa$ for the modulus of elasticity in X and Y direction respectively, $v_{xy}=v_{yx}=0.12$ is the Poisson’s ration for in-plane bi-axial loading, and

$G_{12} = 3.54 GPa, G_{xz} = 3.95 GPa \, and \, G_{yz} = 2.80 GPa$ are the in-plane, transverse for shear in XZ plane and transverse for shear in YZ plane shear moduli and $r = 1600 \, kgr / m^3$ is the density.

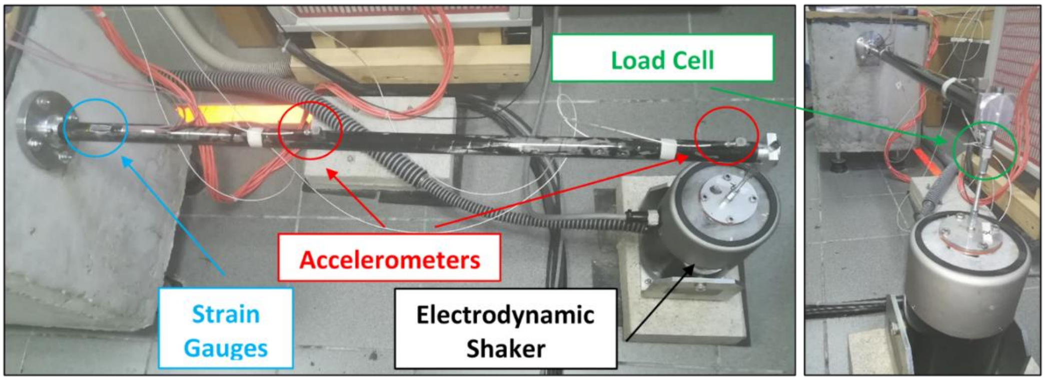


Figure 2 Experimental setup of cantilever CFRP tube under dynamic load excitation.

4.1 Development of the FE model and modal analysis of cantilever CFRP tube

The geometry of the cantilever CFRP beam is discretized with composite shell elements and tetrahedral solid elements for the aluminum ends using appropriate pre-processing commercial software [33]. The total number of DOFs was 1,500,000 [34]. The detailed FE model is presented in **Figure 3**. Indicative mode shapes of the predicted by the nominal FE model are presented in **Error! Reference source not found.** colored by spectrum colors of the normalized deformations.

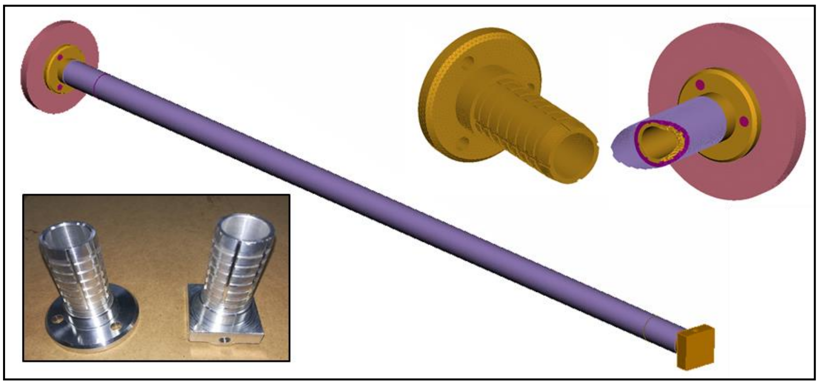


Figure 3 FE model of cantilever CFRP tube along with aluminum drop-outs.

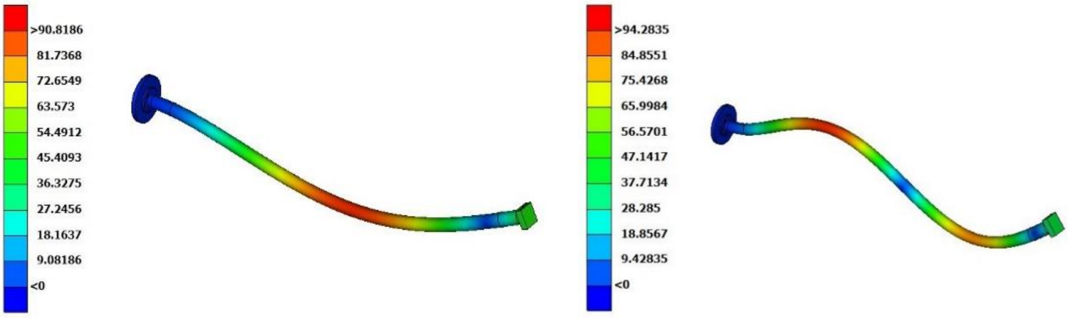


Figure 4 Typical eigenmodes predicted by the nominal FE model.

4.2 Experimental modal analysis

After developing the nominal finite element model, an experimental modal analysis procedure of the CFRP cantilever beam was performed in order to quantify the dynamic characteristics of the examined structure. First, all the necessary elements of the FRF matrix, required for determining the response of the structure were determined by imposing impulsive loading [14-17, 19]. The measured frequency range of the test was 0-600 Hz. An initial investigation indicated that the beam has seven (7) natural frequencies in this frequency range.

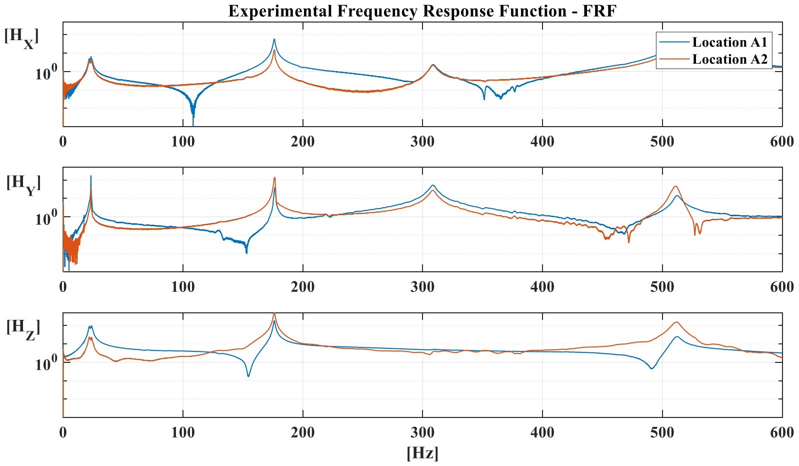


Figure 5 Typical FRFs for modal identification.

Figure 5 presents typical Frequency Response Functions (FRFs) at three components X, Y and Z for two specific measuring points under a specific impulse location and direction. Moreover, the top diagram of **Figure 6** presents a stabilization diagram of a detailed FRF for modal identification, whereas the lower diagram is the detailed view of the FRF.

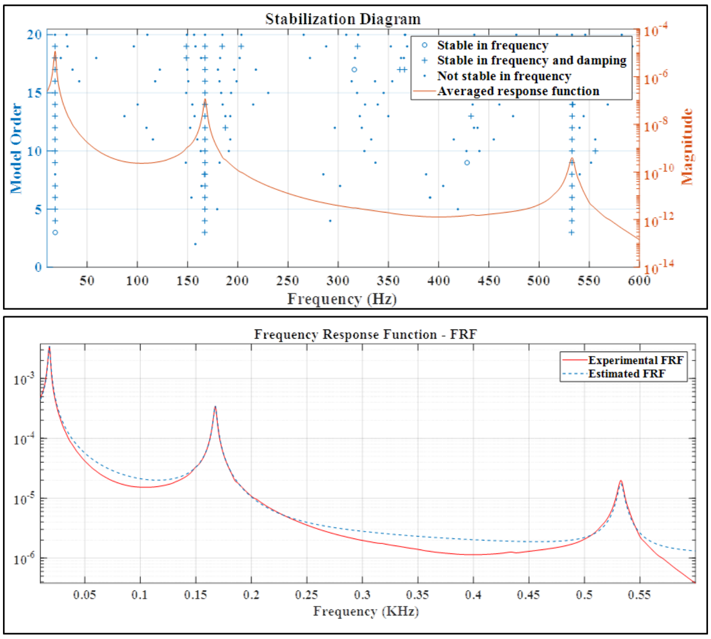


Figure 6 Detailed FRF and stabilization diagram for modal identification.

4.3 FE model parameterization and updating results

The parameterization of the finite element model is introduced in order to facilitate the applicability of the updating framework. The parameterized model is consisted of five (5) parts, as shown in **Figure 6**.

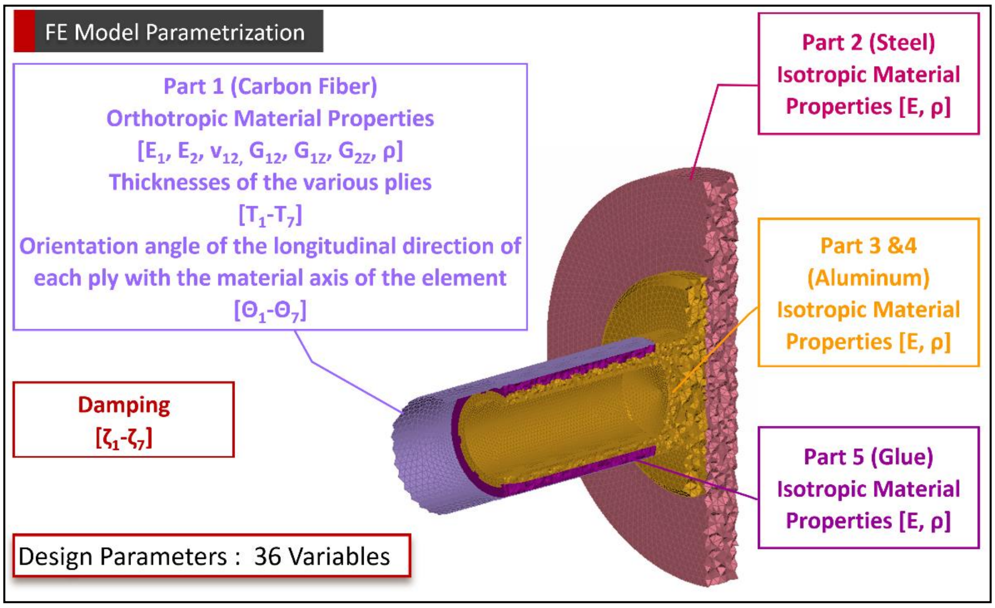

FE Model Parametrization
Part 1 (Carbon Fiber)
Orthotropic Material Properties
[E_1, E_2, v_{12}, G_{12}, G_{1Z}, G_{2Z}, ρ]
Thicknesses of the various plies
[T_1-T_7]
Orientation angle of the longitudinal direction of each ply with the material axis of the element
[$Θ_1$-$Θ_7$]
Part 2 (Steel)
Isotropic Material Properties [E, ρ]
Part 3 &4 (Aluminum)
Isotropic Material Properties [E, ρ]
Part 5 (Glue)
Isotropic Material Properties [E, ρ]
Damping
[$ζ_1$-$ζ_7$]
Design Parameters : 36 Variables

Figure 7 Parts of the parameterized FE model. Detail of CFRP tube and aluminum drop-out.

Part 1 is modeled with composite shell elements and orthotropic material properties while parts 2 to 5 are modeled with solid elements and isotropic material properties. Specifically, part 2 is a steel base, parts 3 and 4 represent the aluminum drop-outs of the beam and part 5 is the glue between the CFRP and the aluminum end. All orthotropic material properties along with the nine ply thicknesses t and orientation angles q were used as design variables of part 1. Additionally, Young's moduli and the material densities of isotropic material parts were also used as design variables. Apart from material properties parameters, the Rayleigh modal damping ratios are used as design variables. Specifically, modal damping ratios ζ_1 to z_7 pertaining to the first seven (7) eigenmodes are included in the design variables, so as to enhance fitting of compared time histories and FRFs, using nominal damping ratio of 3%, as the most common for a composite and steel structures. The total number of design variables for the FE model is thirty-six (36).

Table 1 Comparison between identified, nominal and updated FE predicted modal frequencies.

Mode	**Identified**		**Numerical (before updating)**		**Numerical (after updating)**	
	Frequency (Hz)	Damping (%)	Frequency (Hz)	Error (%)	Frequency (Hz)	Error (%)
1	18.16	0.85	15.54	16.86	18.48	1.73
2	18.18	0.63	16.62	11.79	19.12	2.82
3	149.45	0.82	137.23	8.90	147.53	1.30
4	167.51	0.25	144.22	16.15	168.8	0.76
5	414.32	0.71	408.12	1.52	415.34	0.25
6	436.14	0.91	428.12	1.87	435.54	0.14
7	531.36	1.2	472.35	12.49	524.87	1.24

The CMA-ES framework is applied at ±10% from the nominal values as design bounds, in order to update the developed FE model using the objective function of equation (1) in combination to equation (7), combining modal residuals that include the lowest identified modal frequencies with mode shapes and response residuals that include shape and amplitude correlation coefficients considering measured and numerical frequency response functions including components at all sensor locations, along with time domain acceleration time-histories. Finally, the results of the FE model-updating framework are presented in **Table 1**. A comparison between identified, nominal and updated FE predicted modal frequencies is also presented.

5 ANALYSIS OF A SMALL-SCALE PIN-JOINTED CFRP STRUCTURE

Finally, the experimental arrangement presented in **Figure 8** was set up in order to verify the updated material parameters of the CFRP. Four (4) tri-axial accelerometers were placed on the pin-joined CFRP frame structure, which was anchored on flat plate parallel to the ground, on a vertical concrete column. An electromagnetic shaker was mounted on a free end of the frame where a load cell sensor was placed to record imposed forces under dynamic excitation load.

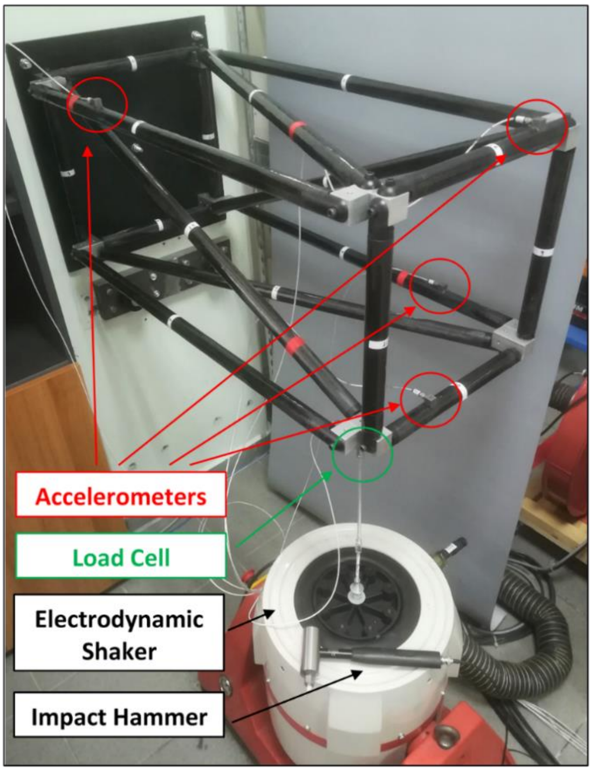


Figure 8 Experimantal setup of small-scale CFRP pin-joined structure.

Additionally, a detailed FE model of a small-scale CFRP pin-joined structure was also developed. The geometry of the structure is discretized with composite shell and solid elements as presented in **Figure 9**. The same figure also presents a detailed view of the FE model at a pin-joint and two indicative mode shapes of the FE model, using the updated orthotropic material parameters, colored by spectrum colors of the normalized deformations.

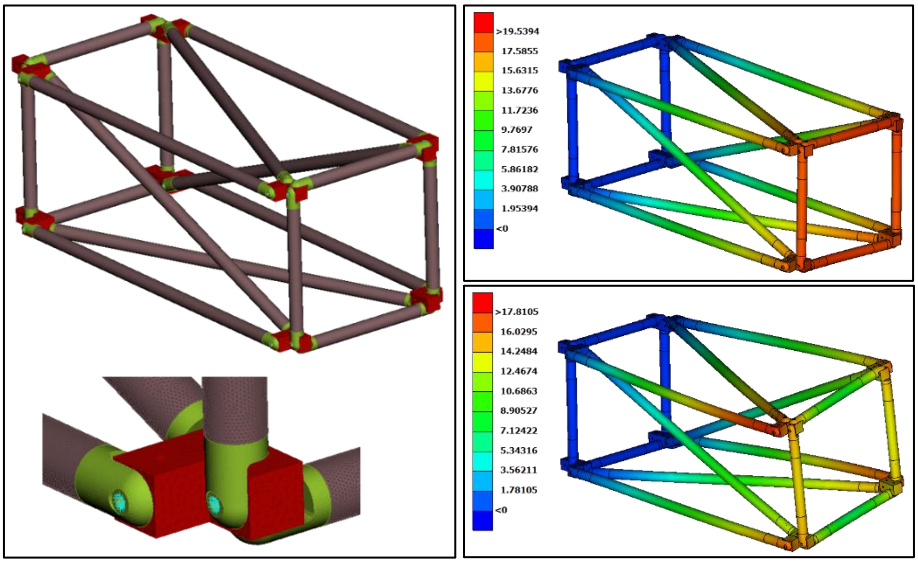


Figure 9 FE model and typical eigenmodes of CFRP pin-joined structure.

Finally, a comparison between experimental and numerical acceleration time histories at matching locations and excitation loading is presented in **Figure 10**. Specifically, time-histories of acceleration at the measured components X, Y and Z of the experimental arrangement under harmonic excitation is presented for two measured locations with black continuous line, whereas the numerically predicted equivalent response of the FE model using the updated parameters is presented with red continuous line. The experimentally obtained acceleration time histories, result very close to those numerically computed, concluding in a high fidelity FE model that could be used for damage identification of the composite cylindrical parts of the structure.

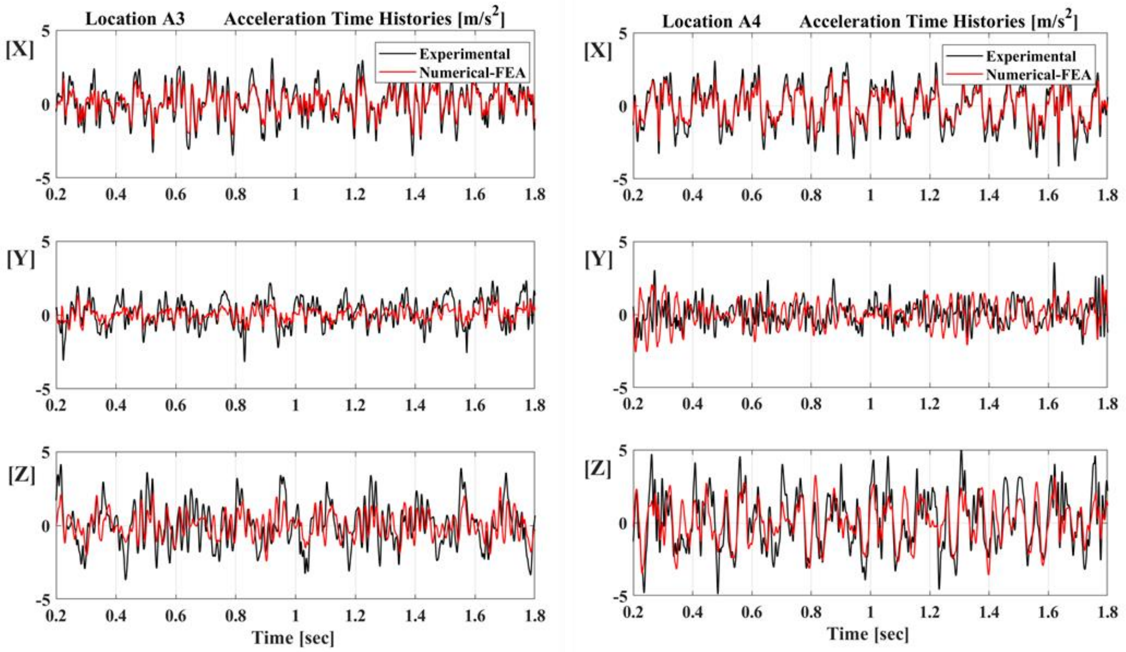


Figure 10 Comparison between experimental and numerical acceleration time histories in X, Y and Z local dirrections at a random force excitation.

6 CONCLUSIONS

In this work, a vibration based computational framework for developing a high fidelity FE model of a CFRP structure, characterizing its orthotropic material properties, that could be used for damage identification of its composite cylindrical parts, is presented. At first, a discrete FE model of a cantilever CFRP tubular beam is developed, by consecutive composite shell elements and solid elements, simulating each carbon fiber ply and resin matrix and its aluminum and steel dropouts respectively. A state of the art FE model updating framework, utilizing CMA-ES optimization algorithm coupled with robust, accurate and efficient finite element analysis software, was applied in order to reconcile modal residuals that include the lowest identified modal frequencies, mode shapes, response residuals that include shape and amplitude correlation coefficients and time-histories of accelerations of experimentally measured data and numerical FE model computation results, in order to classify the material properties (isotropic, orthotropic, anisotropic), update its parameters and develop a high-fidelity FE model. The updated orthotropic material properties are verified on a small-scale pin-joined CFRP frame under harmonic dynamic excitation comparing experimental data and numerical results.

Acknowledgment: *This research has been co ‐ financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH – CREATE – INNOVATE (project code: T1EDK-05393)*

REFERENCES

[1] Soutis, C., *Carbon fiber reinforced plastics in aircraft construction.* Materials Science and Engineering: A, 2005. **412**(1): p. 171-176.

[2] Suresh Kumar, M., M. Ambresha, K. Panbarasu, I. Kishore, and V.R. Ranganath, *A comparative study of failure features in aerospace grade unidirectional and bidirectional woven CFRP composite laminates under four-point bend fatigue loads.* Materialwissenschaft und Werkstofftechnik, 2015. **46**(6): p. 644-651.

[3] Tao, W., Z. Liu, P. Zhu, C. Zhu, and W. Chen, *Multi-scale design of three dimensional woven composite automobile fender using modified particle swarm optimization algorithm.* Composite Structures, 2017. **181**: p. 73-83.

[4] Yang, Y., X. Wu, and H. Hamada, *Application of fibre-reinforced composites beam as energy absorption member in vehicle.* International Journal of Crashworthiness, 2013. **18**(2): p. 103-109.

[5] Robinson, M., J. Stolzfus, and T. Owens, *Composite material compatibility with liquid and gaseous oxygen*, in *19th AIAA Applied Aerodynamics Conference*. 2001, American Institute of Aeronautics and Astronautics.

[6] Hongfei, Z., Z. Xuesen, Z. Jianbao, and S. Hongjie, *The Application of Carbon Fiber Composites in Cryotank*, in *Solidification*, A.E. Ares, Editor. 2018, IntechOpen.

[7] Park, S.Y., W.J. Choi, C.H. Choi, and H.S. Choi, *An experimental study into aging unidirectional carbon fiber epoxy composite under thermal cycling and moisture absorption.* Composite Structures, 2019. **207**: p. 81-92.

[8] Zhang, J., S. He, I. Walton, A. Kajla, and C. Wang, *Lightweight stiffened composite structure with superior bending strength and stiffness for automotive floor applications*, in *Sustainable Automotive Technologies 2012*. 2012, Springer. p. 75-80.

[9] Obradovic, J., S. Boria, and G. Belingardi, *Lightweight design and crash analysis of composite frontal impact energy absorbing structures.* Composite Structures, 2012. **94**(2): p. 423-430.

[10] Zhu, C., P. Zhu, and Z. Liu, *Uncertainty analysis of mechanical properties of plain woven carbon fiber reinforced composite via stochastic constitutive modeling.* Composite Structures, 2019. **207**: p. 684-700.

[11] Ryou, H., K. Chung, and W.-R. Yu, *Constitutive modeling of woven composites considering asymmetric/anisotropic, rate dependent, and nonlinear behavior.* Composites Part A: Applied Science and Manufacturing, 2007. **38**(12): p. 2500-2510.

[12] Karkkainen, R.L. and B.V. Sankar, *A direct micromechanics method for analysis of failure initiation of plain weave textile composites.* Composites Science and Technology, 2006. **66**(1): p. 137-150.

[13] Sun, C.T. and R.S. Vaidya, *Prediction of composite properties from a representative volume element.* Composites Science and Technology, 1996. **56**(2): p. 171-179.

[14] Ewins, D.J., *Modal Testing: Theory and Practice*. 1984, Somerset, England: Research Studies Press.

[15] Giagopoulos, D. and S. Natsiavas, *Hybrid (numerical-experimental) modeling of complex structures with linear and nonlinear components.* Nonlinear Dynamics, 2007. **47**(1): p. 193-217.

[16] Giagopoulos, D. and S. Natsiavas, *Dynamic Response and Identification of Critical Points in the Superstructure of a Vehicle Using a Combination of Numerical and Experimental Methods.* Experimental Mechanics, 2015. **55**(3): p. 529-542.

[17] Mohanty, P. and D.J. Rixen, *Identifying mode shapes and modal frequencies by operational modal analysis in the presence of harmonic excitation.* Experimental Mechanics, 2005. **45**(3): p. 213-220.

[18] Richardson, M.H. and D.L. Formenti, *Global curve fitting of frequency response measurements using the rational fraction polynomial method*, in *Third IMAC Conference*. 1985: Orlando, Florida.

[19] Spottswood, S.M. and R.J. Allemang, *On the Investigation of Some Parameter Identification and Experimental Modal Filtering Issues for Nonlinear Reduced Order Models.* Experimental Mechanics, 2007. **47**(4): p. 511-521.

[20] Hadjidoukas, P.E., P. Angelikopoulos, C. Papadimitriou, and P. Koumoutsakos, *Π4U: A high performance computing framework for Bayesian uncertainty quantification of complex models.* Journal of Computational Physics, 2015. **284**: p. 1-21.

[21] Hansen, N., *The CMA Evolution Strategy A Comparing Review.* Towards a New Evolutionary Computation, 2006. **192**(1): p. 75-102.

[22] Hansen, N., S.D. Müller, and P. Koumoutsakos, *Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES).* Evolutionary Computation, 2003. **11**(1): p. 1-18.

[23] Hansen, N. and A. Ostermeir, *Completely Derandomized Self-Adaptation in Evolution Strategies.* Evolutionary Computation, 2001. **9**(2): p. 159-195.

[24] Giagopoulos, D., A. Arailopoulos, V. Dertimanis, C. Papadimitriou, E. Chatzi, and K. Grompanopoulos, *Computational Framework for Online Estimation of Fatigue Damage using Vibration Measurements from a Limited Number of Sensors.* Procedia Engineering, 2017. **199**: p. 1906-1911.

[25] Giagopoulos, D., A. Arailopoulos, V. Dertimanis, C. Papadimitriou, E. Chatzi, and K. Grompanopoulos, *Structural health monitoring and fatigue damage estimation using vibration measurements and finite element model updating.* Structural Health Monitoring, 2018. **0**(0): p. 1475921718790188.

[26] Papadimitriou, C., E. Ntotsios, D. Giagopoulos, and S. Natsiavas, *Variability of updated finite element models and their predictions consistent with vibration measurements.* Structural Control and Health Monitoring, 2012. **19**(5): p. 630-654.

[27] Giagopoulos, D., D.-C. Papadioti, C. Papadimitriou, and S. Natsiavas, *Bayesian Uncertainty Quantification and Propagation in Nonlinear Structural Dynamics*, in *Topics in Model Validation and Uncertainty Quantification, Volume 5: Proceedings of the 31st IMAC, A Conference on Structural Dynamics, 2013*, T. Simmermacher, S. Cogan, B. Moaveni, and C. Papadimitriou, Editors. 2013, Springer New York: New York, NY. p. 33-41.

[28] Christodoulou, K., E. Ntotsios, C. Papadimitriou, and P. Panetsos, *Structural model updating and prediction variability using Pareto optimal models.* Computer Methods in Applied Mechanics and Engineering, 2008. **198**(1): p. 138-149.

[29] Ntotsios, E. and C. Papadimitriou. *Multi-objective optimization algorithms for finite element model updating*. in *23rd International Conference on Noise and Vibration Engineering 2008, ISMA 2008*. 2008.

[30] Arailopoulos, A., D. Giagopoulos, I. Zacharakis, and E. Pipili, *Integrated Reverse Engineering Strategy for Large-Scale Mechanical Systems: Application to a Steam Turbine Rotor.* Frontiers in Built Environment, 2018. **4**(55).

[31] Giagopoulos, D. and A. Arailopoulos, *Computational framework for model updating of large scale linear and nonlinear finite element models using state of the art evolution strategy.* Computers & Structures, 2017. **192**: p. 210-232.

[32] Grafe, H., *Model updating of large structural dynamics models using measured response function*, in *Department of Mechanical Engineering*. 1999, Imperial College: London.

[33] BETA CAE Systems, S.A., *ANSA & META-Post*. 2018, BETA CAE Systems, S.A.: Thessaloniki, Greece.

[34] Giagopoulos, D. and I. Chatziparasidis. *Optimum design, finite element model updating and dynamic analysis of a full laminated glass panoramic car elevator*. in *7th European Congress on Computational Methods in Applied Sciences and Engineering, ECCOMAS Congress 2016*. 2016. National Technical University of Athens.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

STOCHASTIC RESPONSE QUANTIFICATION OF FIXED-BASE AND BASE-ISOLATED RIGID-PLASTIC BLOCKS

Stavros Kasinos[1] **and Fai Ma**[2]

[1]Imperial College London
Department of Aeronautics, SW7 2AZ, London, UK
e-mail: s.kasinos@imperial.ac.uk

[2] University of California, Berkeley
Department of Mechanical Engineering, California 94720-1740, US
e-mail: fma@berkeley.edu

Keywords: Base Isolation, Earthquake, Rigid-plastic, Sliding, Statistical Linearisation, Stochastic Excitation.

Abstract. *The paper deals with the modelling, response quantification and vibration control of rigid-plastic blocks in presence of stochastic forcing with indicative application to seismic engineering. The full dynamic interaction between a rigid-plastic block and a linear base-isolation system is considered and efficient piecewise numerical solutions are derived for analysing the true nonlinear response, in comparison with the base-fixed counterpart. Stochastic forcing is modelled as stationary filtered white noise, characterised by a modified version of the Kanai-Tajimi power spectrum suggested by Clough and Penzien, commonly used in earthquake engineering applications. A statistical linearisation approach is adopted in view of approximating the strongly nonlinear systems during the sliding motion regime, which conveniently permits quantification of the steady-state, stationary response statistics. The accuracy of the linearisation approximation is investigated, and the effectiveness of the base isolation in suppressing the extreme forcing delivered to the block is assessed. The work delivers insights into the determination and understanding of the probabilistic characteristics of the response of dynamically driven base-fixed and base-isolated rigid-plastic systems, further encouraging investigations on other types of structures, isolation systems and hazard scenarios.*

1 INTRODUCTION

For the vast majority of the structural systems encountered in engineering, it is of paramount importance to understand the dynamics that underpin their response and reliability in the occurrence of extreme environmental loading conditions. Examples include motions of high speed crafts and ships in rough seas [1], vibration of buildings and offshore structures due to wave impacts [2], wind loads [3] and earthquakes [4]. It is widely common to cast idealised models for these structures or their subsystems, as a starting point in characterising their behaviour. The class of *block-type* models, for instance, can be considered representative of the *acceleration-sensitive rigid systems*, that is, a broad spectrum of the mechanical, electrical and electronic equipment of engineering interest (e.g. transformers, emergency generators, computer cabinets, compressors, medical and telecommunications equipment etc.) whose survivability and operational continuity during transportation and throughout their design life is critical. Inherent nonlinearities and uncertainties in their properties, the presence of randomness in the external excitation as well as the type of hazard, pose challenges that render the determination of their response statistics as a non-straightforward task.

Of interest is the case of the idealised sliding block, exhibiting rigid-plastic behaviour, a widely accepted model representing a broad range of structural and geotechnical systems, including buildings on moving foundation, equipment, retaining walls, slopes and masonry. Several studies have been devoted to the deterministic seismic analysis of such blocks, including those dealing with idealised ground acceleration pulses [5–7] and recorded earthquake ground motions [8]. The stochastic response of such systems has been examined in presence of white noise [9] and filtered white noise, characterised by the Kanai-Tajimi [10, 11] power spectrum [12–17], mostly for applications dealing with rigid structures resting on a frictional foundation. Modelling the excitation as white noise, however, implies infinite power of the resulting process, which is unphysical. Nonetheless, such idealisation can deliver useful insights in analysis, provided the results are carefully interpreted. The Kanai-Tajimi spectrum on the other hand, provides a more realistic model for earthquake engineering applications, however, it has been criticised due to the presence of low-frequency content [18].

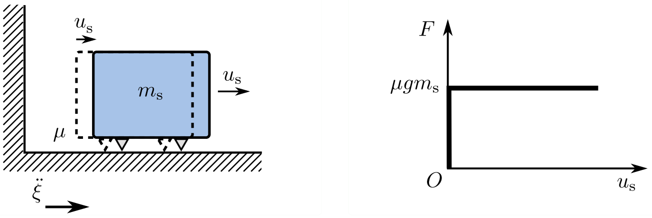
Among risk mitigation technologies, base isolation aims at limiting the vibration response of the system to be controlled via the use of supports that uncouple the structure from the ground. Theory and practice are covered in several books and papers; a comprehensive review of the subject is given by Kelly [19]. Previous endeavours in this context investigate the effectiveness of seismic isolation on the primary load-bearing structure [20], with limited efforts to examine such effects on the performance of components. The 'cascade' response of rigid-plastic systems, for instance, has been examined in base-isolated buildings subjected to broadband ground motions [21, 22]. Adequate characterisation of the nonlinear dynamics for the combined primary-secondary system assembly is in fact necessary, when the equipment vibrates close to, or is tuned with the primary structure. From a different viewpoint, isolation directly applied on the component can be a viable cost-effective strategy to protect sensitive equipment in critical facilities [19]. Nevertheless, to our knowledge, the only past publication dealing with isolation directly on the sliding component is the one by Roussis *et al.* [23], which tackles the problem on a conventional deterministic basis.

Recognising the importance of understanding the response probabilistic characteristics of such systems, this paper addresses the modelling, response quantification and vibration control of rigid-plastic blocks, in presence of stochastic forcing with indicative application to seismic engineering. The scope of the paper is fivefold: (1) to characterise the full dynamic interaction between a rigid-plastic block and a linear base-isolation system; (2) to derive efficient piecewise numerical solutions for quantifying the nonlinear response of fixed-base and base-isolated rigid-plastic blocks to a general-type excitation; (3) to quantify the statistics of the steady-state, stationary response of the associated equivalent linear systems during the sliding motion regime, in presence of excitation characterised by the Clough-Penzien spectrum; (4) to investigate the acuracy of the linearisation approximation; and (5) to assess the effectiveness of the isolation in suppressing the seismic forcing delivered to the block. The work will form the basis for extending our investigations to other types of systems and hazard scenarios.

2 VIBRATION OF FIXED-BASE AND BASE-ISOLATED RIGID-PLASTIC BLOCKS

2.1 Fixed-base rigid-plastic block

Let us consider first the case of a rigid-perfectly plastic single-degree-of-freedom (SDoF) block (S), as depicted in Figure 1(a). The block has a mass m_s and is subjected to the horizontal base acceleration $\ddot{\xi}(t)$, where the overdot denotes differentiation with respect to time and $u_\mathrm{s}(t)$ is the unidirectional displacement, relative to the ground.



(a) (b)

Figure 1: Free-standing sliding block (a) and force-displacement relationship (b).

The system exhibits infinite pre-yielding stiffness and infinite ductility, and the restoring force takes the form:

$$f_s = \begin{cases} \in \left[-\mu\, g\, m_\mathrm{s}, \mu\, g\, m_\mathrm{s}\right], & \dot{u}_\mathrm{s} = 0 \\ \mu\, g\, m_\mathrm{s}\, \mathrm{sgn}\left(\dot{u}_\mathrm{s}(t)\right), & \text{otherwise} \end{cases}, \tag{1}$$

in which $\mu = a_\mathrm{s}/g$ is the coefficient of sliding friction assuming horizontal contact surface, a_s being the system's specific strength (i.e. the level of the ground acceleration $\ddot{\xi}(t)$ required for S to yield), and g is the acceleration due to gravity; $\mathrm{sgn}(\bullet)$ denotes the signum function (i.e. $\mathrm{sgn}(x) = +1$ if $x > 0$, $\mathrm{sgn}(x) = -1$ if $x < 0$, and $\mathrm{sgn}(x) = 0$ if $x = 0$). Evidently, the formalism given by Eq. (1) contains information about two distinct motion regimes, namely, sticking (i.e. when $\dot{u}_\mathrm{s} = 0$), and slipping [24].

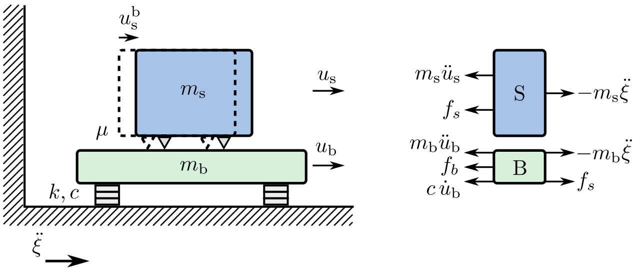
The equation of motion for S is:

$$\ddot{u}_\mathrm{s}(t) = \begin{cases} 0, & u_\mathrm{s}, \dot{u}_\mathrm{s} = 0 \\ -\mu\, g\, \mathrm{sgn}\left(\dot{u}_\mathrm{s}(t)\right) - \ddot{\xi}(t), & \text{otherwise} \end{cases}. \tag{2}$$

The initiation condition for the sliding regime is set to $|\ddot{\xi}(t)| = \mu\, g$ (Figure 1(b)). Following initiation, an instantaneous stop or a full stop can occur in the system once the velocity drops to zero ($\dot{u}_\mathrm{s} = 0$). In the former case, the motion will reverse or it will continue in the same direction, while in the latter case the system will remain at rest until the initiation condition is exceeded again.

2.2 Base-isolated rigid-plastic block

Consider now the case of a two-degree-of-freedom (TDoF) system, comprising of the block S being supported on a linear base isolation system (B) undergoing horizontal accelerated motion, as depicted in Figure 2(a), where $u_{\mathrm{b}}(t)$, $u_{\mathrm{s}}(t)$ are the unidirectional displacements of B and S, relative to the ground, and $u_{\mathrm{s}}^{\mathrm{b}}(t) = u_{\mathrm{s}}(t) - u_{\mathrm{b}}(t)$ is the motion of S relative to B.


$u_{\mathrm{s}}^{\mathrm{b}}$
m_{s}
u_{s}
μ
m_{b}
u_{b}
k, c
$\ddot{\xi}$
$m_{\mathrm{s}}\ddot{u}_{\mathrm{s}}$
f_s
S
$-m_{\mathrm{s}}\ddot{\xi}$
$m_{\mathrm{b}}\ddot{u}_{\mathrm{b}}$
f_b
$c\,\dot{u}_{\mathrm{b}}$
B
$-m_{\mathrm{b}}\ddot{\xi}$
f_s

(a) (b)

Figure 2: TDoF system: sliding block on a linear isolation system (a); free-body diagram (b).

Figure 2(b) shows the forces acting on B and S, where m_{b} and m_{s} are the associated masses. Further extending the formulation in [4], $f_b(t) = \omega_{\mathrm{b}}^2\, m_{\mathrm{b}}\, u_{\mathrm{b}}(t)$ represents the restoring force in B, where $\omega_{\mathrm{b}} = \sqrt{k/m_{\mathrm{t}}}$ is the associated natural circular frequency, k being the stiffness of a linear spring and $m_{\mathrm{t}} = m_{\mathrm{b}} + m_{\mathrm{s}}$ the total mass of the system. Furthermore, $c = 2\,\zeta\omega_{\mathrm{b}}\, m_{\mathrm{t}}$ is the viscous damping coefficient, where ζ is the equivalent viscous damping ratio. The rigid-perfectly plastic S system finally assumes a restoring force, f_s as in Eq. (1), where $\dot{u}_{\mathrm{s}}^{\mathrm{b}}$ is used in place of $\dot{u}_{\mathrm{s}}$.

Dynamic equilibrium of the mass m_{p} in the horizontal direction then gives:

$$\ddot{u}_{\mathrm{b}}(t) = -\gamma\, \ddot{u}_{\mathrm{s}}^{\mathrm{b}}(t) - 2\,\zeta\omega_{\mathrm{b}}\, \dot{u}_{\mathrm{b}}(t) - \omega_{\mathrm{b}}^2\, u_{\mathrm{b}}(t) - \ddot{\xi}(t)\,; \quad u_{\mathrm{b}}(0) = \dot{u}_{\mathrm{b}}(0) = 0\,, \tag{3}$$

where $\gamma = m_{\mathrm{s}}/m_{\mathrm{t}}$ is the ratio of the block's mass to the total mass of the system, controlling the relative significance of the feedback action on B.

Setting $u_{\mathrm{s}}^{\mathrm{b}}(t) = \dot{u}_{\mathrm{s}}^{\mathrm{b}}(t) = 0$ in the above for the sticking phase where no relative motion is exhibited for S, the resulting system can be interpreted as an equivalent oscillator with mass m_{t}.

Equilibrium of the forces (Figure 2(b)) gives the equation of motion for S:

$$\ddot{u}_{\mathrm{s}}^{\mathrm{b}}(t) = \begin{cases} 0, & u_{\mathrm{s}}^{\mathrm{b}}, \dot{u}_{\mathrm{s}}^{\mathrm{b}} = 0 \\ -\mu\, g\, \mathrm{sgn}\left(\dot{u}_{\mathrm{s}}^{\mathrm{b}}(t)\right) - \ddot{u}_{\mathrm{b}}(t) - \ddot{\xi}(t), & \text{otherwise} \end{cases} . \tag{4}$$

The initiation condition for sliding in the TDoF system is set to $|\ddot{u}_{\mathrm{b}}(t) + \ddot{\xi}(t)| = \mu\, g$, $\ddot{u}_{\mathrm{b}}(t)$ being a solution of Eq. (3).

Equations (3) and (4) are cast in a state space form (i.e. explicit expressions of the state variables) and are solved together. In this case, the state vector is:

$$\mathbf{y}\,(t) = \begin{cases} \left\{\, u_{\mathrm{b}}(t) \,\vdots\, \dot{u}_{\mathrm{b}}(t) \,\right\}^{\top}, & u_{\mathrm{s}}^{\mathrm{b}}, \dot{u}_{\mathrm{s}}^{\mathrm{b}} = 0 \\ \left\{\, u_{\mathrm{s}}^{\mathrm{b}}(t) \,\vdots\, \dot{u}_{\mathrm{s}}^{\mathrm{b}}(t) \,\vdots\, u_{\mathrm{b}}(t) \,\vdots\, \dot{u}_{\mathrm{b}}(t) \,\right\}^{\top}, & \text{otherwise} \end{cases} , \tag{5}$$

whose time derivative is:

$$\dot{\mathbf{y}}\,(t) = \begin{cases} \left\{ \begin{matrix} \dot{u}_{\mathrm{b}}(t) \\ -2\,\zeta\,\omega_{\mathrm{b}}\dot{u}_{\mathrm{b}}(t) - \omega_{\mathrm{b}}^2 u_{\mathrm{b}}(t) - \ddot{\xi}(t) \end{matrix} \right\}, & u_{\mathrm{s}}^{\mathrm{b}}, \dot{u}_{\mathrm{s}}^{\mathrm{b}} = 0 \\ \left\{ \begin{matrix} \dot{u}_{\mathrm{s}}^{\mathrm{b}}(t) \\ \frac{-\mu\,g\,\mathrm{sgn}\left(\dot{u}_{\mathrm{s}}^{\mathrm{b}}(t)\right)+2\,\zeta\omega_{\mathrm{b}}\dot{u}_{\mathrm{b}}(t)+\omega_{\mathrm{b}}^2 u_{\mathrm{b}}(t)}{1-\gamma} \\ \dot{u}_{\mathrm{b}}(t) \\ \frac{\gamma\,\mu\,g\,\mathrm{sgn}\left(\dot{u}_{\mathrm{s}}^{\mathrm{b}}(t)\right)-2\,\zeta\omega_{\mathrm{b}}\dot{u}_{\mathrm{b}}(t)-\omega_{\mathrm{b}}^2 u_{\mathrm{b}}(t)}{1-\gamma} - \ddot{\xi}(t) \end{matrix} \right\}, & \text{otherwise} \end{cases} . \tag{6}$$

During the sticking phase, integration is carried out solely for B based on the top part of Eq. (6), using the initial conditions from the last step. Following initiation integration proceeds thereafter using the bottom part of the equation.

It is worth mentioning that the formulation presented herein, is in agreement with an equivalent expression in [23] for the sliding motion regime, and delivers further insights during the sticking motion regime.

3 NUMERICAL PROCEDURE FOR PIECEWISE RESPONSE QUANTIFICATION

Owing to the piecewise linear form of the dynamical systems considered, a highly efficient numerical procedure is employed for quantifying the true nonlinear response due to a general-type of excitation. In what follows, each regime of motion is separately considered and the response time history is constructed by piecing together the individual segments.

3.1 Fixed-base rigid-plastic block

The response of the SDoF system in Eq. (2) is considered first during the sliding motion regime. Accordingly, a numerical scheme [4] is adopted for the response evaluation by interpolating the excitation over each time interval. The response vector is then readily determined through the recurrence formula:

$$\mathbf{y}(t_{i+1}) = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \cdot \mathbf{y}(t_i) - \begin{bmatrix} \frac{\Delta t^2}{3} \\ \frac{\Delta t}{2} \end{bmatrix} \cdot \eta(t_i) - \begin{bmatrix} \frac{\Delta t^2}{6} \\ \frac{\Delta t}{2} \end{bmatrix} \cdot \eta(t_{i+1})\,, \tag{7}$$

where $\mathbf{y}(t) = \{u_{\mathrm{s}}(t), \dot{u}_{\mathrm{s}}(t)\}^{\top}$ and $\eta\,(t) = \ddot{\xi}(t) + \mu\,g\,\mathrm{sgn}\,(\dot{u}_{\mathrm{s}}(t))$. Notably, the only restriction in Eq. (7) is that Δt is sufficiently low to closely approximate the excitation.

3.2 Base-isolated rigid-plastic block

The TDoF system in Eq. (6) is considered next. Similar to the fixed-base block, the response is separately derived for the sticking and sliding motion regimes. The response vector is then obtained from the recurrence formula:

$$\mathbf{y}(t_{i+1}) = \mathbf{\Theta}(\Delta t) \cdot \mathbf{y}(t_i) + \mathbf{\Gamma}_0(\Delta t) \cdot \mu\,g\,\mathrm{sgn}\left(\dot{u}_{\mathrm{s}}^{\mathrm{b}}(t)\right) + \mathbf{\Gamma}_1(\Delta t) \cdot \ddot{\xi}\,(t_i) + \mathbf{\Gamma}_2(\Delta t) \cdot \ddot{\xi}\,(t_{i+1})\ , \tag{8}$$

where $\mathbf{\Theta}(\Delta t)$ is the so-called transition matrix, and $\mathbf{\Gamma}_0(\Delta t)$, $\mathbf{\Gamma}_1(\Delta t)$ and $\mathbf{\Gamma}_2(\Delta t)$ are vectors depending on Δt, which is tacitly assumed sufficiently small so that the interpolation of the force is satisfactory.

During the sticking regime, $\mathbf{\Theta}(\Delta t)$ is given by:

$$\mathbf{\Theta}(\Delta t) = \begin{bmatrix} \mathcal{A}(\Delta t) & \mathcal{B}(\Delta t) \\ -\omega_{\mathrm{b}}^2 \mathcal{B}(\Delta t) & \mathcal{A}(\Delta t) - 2\,\zeta\omega_{\mathrm{b}}\mathcal{B}(\Delta t) \end{bmatrix}, \tag{9}$$

where $\mathcal{A}(\Delta t) = e^{-\zeta\omega_{\mathrm{b}}\Delta t}\cos(\omega_{\mathrm{d}}\Delta t) + \zeta\omega_{\mathrm{b}}\,\mathcal{B}(\Delta t)$, $\mathcal{B}(\Delta t) = \frac{e^{-\zeta\omega_{\mathrm{b}}\Delta t}}{\omega_{\mathrm{d}}}\sin(\omega_{\mathrm{d}}\Delta t)$, and $\omega_{\mathrm{d}} = \omega_{\mathrm{b}}\sqrt{1-\zeta^2}$ is the damped circular frequency.

Furthermore, $\mathbf{\Gamma}_0(\Delta t)$ is a zero vector and $\mathbf{\Gamma}_1(\Delta t)$, $\mathbf{\Gamma}_2(\Delta t)$ are given by:

$$\mathbf{\Gamma}_1(\Delta t) = \begin{bmatrix} \frac{\omega_{\mathrm{b}}\Delta t\mathcal{A}(\Delta t)+2\,\zeta(\mathcal{A}(\Delta t)-1)-\omega_{\mathrm{b}}\mathcal{B}(\Delta t)}{\omega_{\mathrm{b}}^3\Delta t} \\ \frac{1-\mathcal{A}(\Delta t)}{\omega_{\mathrm{b}}^2\Delta t} - \mathcal{B}(\Delta t) \end{bmatrix}; \tag{10}$$

$$\mathbf{\Gamma}_2(\Delta t) = \begin{bmatrix} \frac{-2\,\zeta\mathcal{A}(\Delta t)+\omega_{\mathrm{b}}\mathcal{B}(\Delta t)-\omega_{\mathrm{b}}\Delta t+2\,\zeta}{\omega_{\mathrm{b}}^3\Delta t} \\ \frac{\mathcal{A}(\Delta t)-1}{\omega_{\mathrm{b}}^2\Delta t} \end{bmatrix}. \tag{11}$$

For the sliding regime, the response depends on:

$$\mathbf{\Theta}(\Delta t) = \begin{bmatrix} 1 & \Delta t & 1-\zeta\omega_{\mathrm{b}}\,\mathbb{F}-\omega_{\mathrm{d1}}\mathbb{Q} & \Delta t+(\gamma-1)\mathbb{F} \\ 0 & 1 & \omega_{\mathrm{b}}^2\,\mathbb{F} & 1+\zeta\omega_{\mathrm{b}}\,\mathbb{F}-\omega_{\mathrm{d1}}\mathbb{Q} \\ 0 & 0 & \zeta\omega_{\mathrm{b}}\,\mathbb{F}+\omega_{\mathrm{d1}}\mathbb{Q} & (1-\gamma)\mathbb{F} \\ 0 & 0 & -\omega_{\mathrm{b}}^2\,\mathbb{F} & \omega_{\mathrm{d1}}\mathbb{Q}-\zeta\omega_{\mathrm{b}}\,\mathbb{F} \end{bmatrix}; \tag{12}$$

$$\mathbf{\Gamma}_0(\Delta t) = \begin{bmatrix} \frac{\gamma(\zeta\omega_{\mathrm{b}}\,\mathbb{F}+\omega_{\mathrm{d1}}\mathbb{Q}-1)}{\omega_{\mathrm{b}}^2} - \frac{\Delta t^2}{2} \\ -\gamma\,\mathbb{F}-\Delta t \\ -\frac{\gamma(\zeta\omega_{\mathrm{b}}\,\mathbb{F}+\omega_{\mathrm{d1}}\mathbb{Q}-1)}{\omega_{\mathrm{b}}^2} \\ \gamma\,\mathbb{F} \end{bmatrix}; \tag{13}$$

$$\mathbf{\Gamma}_1(\Delta t) = \begin{bmatrix} \frac{6(\gamma-1)\zeta^2\mathbb{F}\omega_{\mathrm{b}}+\omega_{\mathrm{b}}\left(\Delta t^3\left(-\omega_{\mathrm{b}}^2\right)+3(\gamma-1)^2\mathbb{F}+3(\gamma-1)\Delta t\mathbb{Q}\omega_{\mathrm{d1}}\right)+3(\gamma-1)\zeta\left(\Delta t\mathbb{F}\omega_{\mathrm{b}}^2+2\mathbb{Q}\omega_{\mathrm{d1}}-2\right)}{3\Delta t\omega_{\mathrm{b}}^3} \\ -\frac{\Delta t^2\omega_{\mathrm{b}}^2-2\mathbb{F}\omega_{\mathrm{b}}(\Delta t\omega_{\mathrm{b}}+\zeta)+2\gamma(\mathbb{F}\omega_{\mathrm{b}}(\Delta t\omega_{\mathrm{b}}+\zeta)+\mathbb{Q}\omega_{\mathrm{d1}}-1)-2\mathbb{Q}\omega_{\mathrm{d1}}+2}{2\Delta t\omega_{\mathrm{b}}^2} \\ -\frac{(\gamma-1)((\gamma-1)\mathbb{F}\omega_{\mathrm{b}}+\zeta(\mathbb{F}\omega_{\mathrm{b}}(\Delta t\omega_{\mathrm{b}}+2\zeta)-2)+\mathbb{Q}\omega_{\mathrm{d1}}(\Delta t\omega_{\mathrm{b}}+2\zeta))}{\Delta t\omega_{\mathrm{b}}^3} \\ \frac{(\gamma-1)(\mathbb{F}\omega_{\mathrm{b}}(\Delta t\omega_{\mathrm{b}}+\zeta)+\mathbb{Q}\omega_{\mathrm{d1}}-1)}{\Delta t\omega_{\mathrm{b}}^2} \end{bmatrix}; \tag{14}$$

$$\mathbf{\Gamma}_2(\Delta t) = \begin{bmatrix} -\frac{\Delta t^3\omega_{\mathrm{b}}^3+6(\gamma-1)\omega_{\mathrm{b}}(\Delta t+(\gamma-1)\mathbb{F})+12(\gamma-1)\zeta^2\mathbb{F}\omega_{\mathrm{b}}+12(\gamma-1)\zeta(\mathbb{Q}\omega_{\mathrm{d1}}-1)}{6\Delta t\omega_{\mathrm{b}}^3} \\ -\frac{\Delta t^2\omega_{\mathrm{b}}^2+2\zeta\mathbb{F}\omega_{\mathrm{b}}-2\gamma(\zeta\mathbb{F}\omega_{\mathrm{b}}+\mathbb{Q}\omega_{\mathrm{d1}}-1)+2\mathbb{Q}\omega_{\mathrm{d1}}-2}{2\Delta t\omega_{\mathrm{b}}^2} \\ \frac{(\gamma-1)(\omega_{\mathrm{b}}(\Delta t+(\gamma-1)\mathbb{F})+2\zeta(\zeta\mathbb{F}\omega_{\mathrm{b}}+\mathbb{Q}\omega_{\mathrm{d1}}-1))}{\Delta t\omega_{\mathrm{b}}^3} \\ -\frac{(\gamma-1)(\zeta\mathbb{F}\omega_{\mathrm{b}}+\mathbb{Q}\omega_{\mathrm{d1}}-1)}{\Delta t\omega_{\mathrm{b}}^2} \end{bmatrix}, \tag{15}$$

where the above depend on $\mathbb{F}(\Delta t) = \frac{1}{\omega_{\mathrm{d1}}}e^{\frac{\zeta\omega_{\mathrm{b}}\,\Delta t}{\gamma-1}}\sin(\frac{\omega_{\mathrm{d1}}\Delta t}{1-\gamma})$, $\mathbb{Q}(\Delta t) = \frac{1}{\omega_{\mathrm{d1}}}e^{\frac{\zeta\omega_{\mathrm{b}}\,\Delta t}{\gamma-1}}\cos(\frac{\omega_{\mathrm{d1}}\Delta t}{1-\gamma})$ and $\omega_{\mathrm{d1}} = \omega_{\mathrm{b}}\sqrt{1-\gamma-\zeta^2}$.

In using the piecewise linear solutions presented herein, the response is evaluated separately during the sticking and sliding motion regimes and once both components have been determined the overall response is constructed by piecing together the individual segments. Notably, an iterative scheme needs to be employed to identify the time of initiation for each regime of motion as well as subsequent changes in the regime when the velocity changes sign. Details on the numerical implementation procedure are provided in Appendix A.

4 STOCHASTIC MODEL OF SEISMIC FORCING

Let us now consider a ground acceleration $\ddot{\xi}(t)$, modelled as stationary filtered white noise process, characterised by a more realistic version of the Kanai-Tajimi power spectrum [10, 11], suggested by Clough and Penzien [25], commonly used in earthquake engineering applications.

The spectral density function takes the form:

$$S_{\ddot{\xi}}(\omega) = S_0 \cdot H_k(\omega) \cdot H_c(\omega)\,; \quad -\infty < \omega < \infty\,, \tag{16}$$

where S_0 represents a constant power spectral density level due to white noise, $H_k(\omega)$ and $H_c(\omega)$ represent the Kanai-Tajimi and Clough-Penzien filters, respectively, given by:

$$H_k(\omega) = \frac{1 + 4\,\zeta_g^2\,(\omega/\omega_g)^2}{\left(1-(\omega/\omega_g)^2\right)^2 + 4\,\zeta_g^2\,(\omega/\omega_g)^2}\,; \quad H_c(\omega) = \frac{(\omega/\omega_f)^4}{\left(1-(\omega/\omega_f)^2\right)^2 + 4\,\zeta_f^2\,(\omega/\omega_f)^2}\,, \tag{17}$$

where the parameters ω_g and ζ_g denote the frequency and damping ratio of the soil layer, respectively, and ω_f, ζ_f control the Clough-Penzien filter's characteristics.

In this model, the first filter $H_k(\omega)$ attenuates the frequency content for $\omega > \omega_g$ as $\omega \to \infty$, and amplifies the frequencies in the vicinity of $\omega = \omega_g$; the second filter $H_c(\omega)$ is then introduced to eliminate the low-frequency content, thus assuring finite power for the ground displacement.

Table 1 lists filter parameter values for producing reasonable spectral shapes for 'firm', 'medium' and 'soft' soils, as suggested in [26]. Figure 3 plots the corresponding curves for $S_0 = 1$, where the soft soil indicates a narrow-band process while the firm ground is broad-band with significant high frequency content.

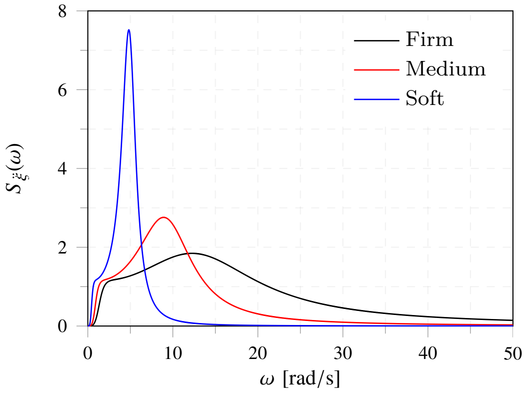


Figure 3: Clough-Penzien spectrum for different soil types [26].

Table 1: Filter parameters for different soil types [26].

Soil type	ω_g [rad/s]	ζ_g	ω_f [rad/s]	ζ_f
Firm	15.0	0.6	1.5	0.6
Medium	10.0	0.4	1.0	0.6
Soft	5.0	0.2	0.5	0.6

5 STEADY-STATE STATIONARY RESPONSE QUANTIFICATION

We next consider the case where the statistics of the steady-state, stationary response are of interest and the intensity of the base acceleration is sufficiently high such that the probability of sticking can be regarded negligible. In this case, the bottom part of Eq. (5) and (6) is valid for all time.

5.1 Fixed-base sliding block

Following the procedure delineated in [17] the nonlinear Eq. (2) is replaced with a linear one:

$$\ddot{u}_{\mathrm{s}}(t) = -\beta\, \dot{u}_{\mathrm{s}}(t) - \ddot{\xi}(t)\,, \tag{18}$$

where β represents a linear viscous damping term.

Minimising the mean square of the error $\varepsilon = \mu\, g\, \mathrm{sgn}\,(\dot{u}_{\mathrm{s}}(t)) - \beta\, \dot{u}_{\mathrm{s}}(t)$ with respect to β and after manipulation based on the standard assumption of zero mean Gaussian response, one obtains:

$$\beta = \left(\frac{2}{\pi}\right)^{\frac{1}{2}} \frac{\mu\, g}{\sigma_{\dot{u}_{\mathrm{s}}}}\,, \tag{19}$$

where $\sigma^2_{\dot{u}_{\mathrm{s}}} = \mathbb{E}\left\langle \dot{u}^2_{\mathrm{s}}(t)\right\rangle$ is the mean square of $\dot{u}_{\mathrm{s}}(t)$.

Further manipulation based on the specification of the spectrum, gives $\sigma_{u_{\mathrm{s}}}$ and $\sigma_{\dot{u}_{\mathrm{s}}}$ in terms of β:

$$\sigma^2_{u_{\mathrm{s}}} = \int_{-\infty}^{\infty} \left(\omega^2(\beta^2+\omega^2)\right)^{-1} S_{\ddot{\xi}}(\omega)\,\mathrm{d}\omega\,; \quad \sigma^2_{\dot{u}_{\mathrm{s}}} = \int_{-\infty}^{\infty} \left(\beta^2+\omega^2\right)^{-1} S_{\ddot{\xi}}(\omega)\,\mathrm{d}\omega\,. \tag{20}$$

Solution to Eq. (20) was presented in [17] for white noise excitation i.e. $S_{\ddot{\xi}}(\omega) = S_0$, in which case $\sigma^2_{\dot{u}_{\mathrm{s}}} = \pi S_0/\beta$ and $\beta = 2(\mu\, g)^2/\pi^2 S_0$. In the case where the excitation is characterised by the Kanai-Tajimi spectrum (i.e. setting $H_c(\omega) = 1$ in Eq. (16)), solution is reported in [16]. Notably, for both these cases, the first integral in Eq. (20) is infinite which implies that the mean-square of the displacement will indefinitely grow with time.

Further extending the existing contributions, we present here solutions for the Clough-Penzien spectrum in Eq. (16).

Analytical evaluation of Eq. (20) gives:

$$\sigma^2_{u_{\mathrm{s}}} = \frac{\pi S_0\, \omega_g^2(\mathcal{C}_1+\mathcal{C}_2)}{2\,\zeta_f\,\zeta_g\,\omega_f\,\mathcal{C}_3\,\mathcal{C}_4}\,; \quad \sigma^2_{\dot{u}_{\mathrm{s}}} = \frac{\pi S_0\, \omega_g^2(\mathcal{C}_5+\mathcal{C}_6)}{2\,\zeta_f\,\zeta_g\,\mathcal{C}_3\,\mathcal{C}_4}\,, \tag{21}$$

where the coefficients $\mathcal{C}_1$ - $\mathcal{C}_6$ are given by:

$$\mathcal{C}_1 = \beta\,(2\,\zeta_f\omega_f + 2\,\zeta_g\omega_g + \beta)\,\mathcal{C}_8\,; \tag{22a}$$

$$\mathcal{C}_2 = \omega_g\left(\omega_f^3\,\mathcal{C}_7 + \zeta_g\omega_g\left(4\omega_f^2\left(\zeta_f^2 + \zeta_g^2\right) + 4\zeta_f\zeta_g\omega_f\omega_g + \omega_g^2\right)\right)\;; \tag{22b}$$

$$\mathcal{C}_3 = \left(\beta^2 + 2\beta\zeta_f\omega_f + \omega_f^2\right)\left(\beta^2 + 2\beta\zeta_g\omega_g + \omega_g^2\right)\;; \tag{22c}$$

$$\mathcal{C}_4 = 2\,\omega_f^2\omega_g^2\left(2\zeta_f^2 + 2\zeta_g^2 - 1\right) + 4\zeta_f\zeta_g\omega_f^3\omega_g + 4\zeta_f\zeta_g\omega_f\omega_g^3 + \omega_f^4 + \omega_g^4\,; \tag{22d}$$

$$\mathcal{C}_5 = \omega_g(2\beta(\zeta_f\omega_g + \zeta_g\omega_f) + \omega_f\omega_g)\,\mathcal{C}_8\,; \tag{22e}$$

$$\mathcal{C}_6 = \beta^2\left(\omega_g^3\,\mathcal{C}_7 + 4\zeta_g^3\left(4\zeta_f^2\omega_f\omega_g^2 + \omega_f^3\right) + 16\zeta_f\zeta_g^4\omega_f^2\omega_g + \zeta_g\omega_f\omega_g^2\right)\;, \tag{22f}$$

in which $\mathcal{C}_7 = \zeta_f\left(4\,\zeta_g^2 + 1\right)$ and $\mathcal{C}_8 = 4\,\zeta_g^3\omega_f^2 + \zeta_g\omega_g^2 + \zeta_f\left(4\zeta_g^2 + 1\right)\omega_f\,\omega_g$.

On combining Eq. (19) with Eq. (21), the resulting algebraic equation can be solved numerically for β and therefore σ_{u_s} and $\sigma_{\dot{u}_\mathrm{s}}$ can be evaluated from Eq. (21).

5.2 Base-isolated sliding block

The TDoF base-isolated block is next considered. The system is of chain-like structure and statistical linearisation is admissible. Accordingly, the term $\operatorname{sgn}\left(\dot{u}_\mathrm{s}(t)\right)$ in Eq. (6), is replaced with the linear viscous damping term β, which assumes a similar form as the fixed-base block, except that $\sigma_{\dot{u}_\mathrm{s}^\mathrm{b}}$ (i.e. the standard deviation of $\dot{u}_\mathrm{s}^\mathrm{b}(t)$), is used in place of $\sigma_{\dot{u}_\mathrm{s}}$ in Eq. (19).

The equation of motion of the equivalent linear system then reads:

$$\ddot{u}_\mathrm{s}^\mathrm{b}(t) = \frac{-\beta\dot{u}_\mathrm{s}^\mathrm{b}(t) + 2\,\zeta\omega_\mathrm{b}\,\dot{u}_\mathrm{b}(t) + \omega_\mathrm{b}^2\,u_\mathrm{b}(t)}{1-\gamma}\;; \tag{23a}$$

$$\ddot{u}_\mathrm{b}(t) = \frac{\gamma\,\beta\dot{u}_\mathrm{s}^\mathrm{b}(t) - 2\,\zeta\omega_\mathrm{b}\,\dot{u}_\mathrm{b}(t) - \omega_\mathrm{b}^2\,u_\mathrm{b}(t)}{1-\gamma} - \ddot{\xi}(t)\,, \tag{23b}$$

where the spectral density matrix of the response process takes the form:

$$\mathbf{S}_u(\omega) = \mathbf{H}(\omega)\cdot\mathbf{S}_f(\omega)\cdot\mathbf{H}^{\top *}(\omega)\,, \tag{24}$$

in which $\mathbf{S}_f(\omega)$ denotes the spectral density matrix of the forcing, the symbols $\top$ and $*$ denote transposition and conjugation, respectively, and $\mathbf{H}(\omega)$ is the matrix of frequency response functions, given by:

$$\mathbf{H}(\omega) = \left[\begin{array}{c:c} \frac{\omega^2 - 2\,i\,\zeta\omega_\mathrm{b}\,\omega - \omega_\mathrm{b}^2}{\omega\,\mathbb{G}} & -\frac{\omega}{\mathbb{G}} \\ \hdashline -\frac{\gamma\,\omega}{\mathbb{G}} & \frac{\omega - \beta\,i}{\mathbb{G}} \end{array}\right]\,, \tag{25}$$

where $\mathbb{G}(\omega) = \omega^2((\gamma-1)\,\omega + i\,\beta) + 2\,\zeta\omega_\mathrm{b}(\beta + i\,\omega)\,\omega + \omega_\mathrm{b}^2(\omega - i\,\beta)$.

Further, the cross-variance of the response is evaluated through:

$$\mathbb{E}\left\langle u_i(t)u_j(t)\right\rangle = \int_{-\infty}^{\infty} S_{u_iu_j}(\omega)\,\mathrm{d}\omega\,;\quad \mathbb{E}\left\langle \dot{u}_i(t)\dot{u}_j(t)\right\rangle = \int_{-\infty}^{\infty} \omega^2 S_{u_iu_j}(\omega)\,\mathrm{d}\omega\,, \tag{26}$$

where $S_{u_iu_j}(\omega)$ is the (i,j)th element of $\mathbf{S}_u(\omega)$.

An alternative iterative procedure is employed for evaluating Eq. (26). Specifically, it is first assumed that $\beta = 0$ and the cross-variance terms in Eq. (26) are evaluated. These are used for determining a new estimate of β, which results in an update to Eq. (26). The procedure is repeated several times until accuracy is satisfactory.

6 NUMERICAL INVESTIGATIONS

The contributions presented in the preceding sections are next investigated by simulation techniques. Purpose is the quantification of the statistics of the steady-state stationary response of the systems under consideration due to filtered white noise excitation.

6.1 Piecewise linear solutions

The piecewise linear solutions presented in § 3 are first demonstrated on the nonlinear response quantification of fixed-base (FB) and base-isolated (IB) blocks with the purpose of assessing the validity of approximating the rigid-plastic behaviour with pure-sliding one (i.e. neglecting the rigid regime of motion, assuming that sliding is valid for all time). In the sequel, $u_{\mathrm{s}}(t)$ is used in place of $u_{\mathrm{s}}^{\mathrm{b}}(t)$, to represent the motion of the block relative to its base.

Figures 4(a) and 4(b) show two simulated realisations of the earthquake excitation, characterised by the Clough-Penzien power spectrum for a medium soil with $\omega_g = 10\,\mathrm{rad/s}$, $\zeta_g = 0.4$, $\omega_f = 1\,\mathrm{rad/s}$, $\zeta_f = 0.6$ and $S_0 = 0.0025\,\mathrm{m}^2/\mathrm{s}^3$. Details on the procedure used for generating the excitation time series are provided in Appendix B.

The relative displacement and relative velocity response time histories of the FB and IB systems have been quantified next using the proposed piecewise linear solutions. Each system has been successively modelled with idealised rigid-plastic (R) and sliding (S) behaviour, and the isolation parameters $\gamma = 0.04$, $\omega_{\mathrm{b}} = 1.5\,\mathrm{rad/s}$ and $\zeta = 0.05$ have been assumed.

Figures 4(c) and 4(e) show the response due to the first realisation of the excitation with $\mu = 0.02$, indicating excellent agreement between the rigid-plastic and sliding solutions. Plotting the response histories for the second realisation in Figures 4(d) and 4(f), with $\mu = 0.06$, shows pronounced variations between the rigid-plastic and sliding solutions for both the two systems under consideration.

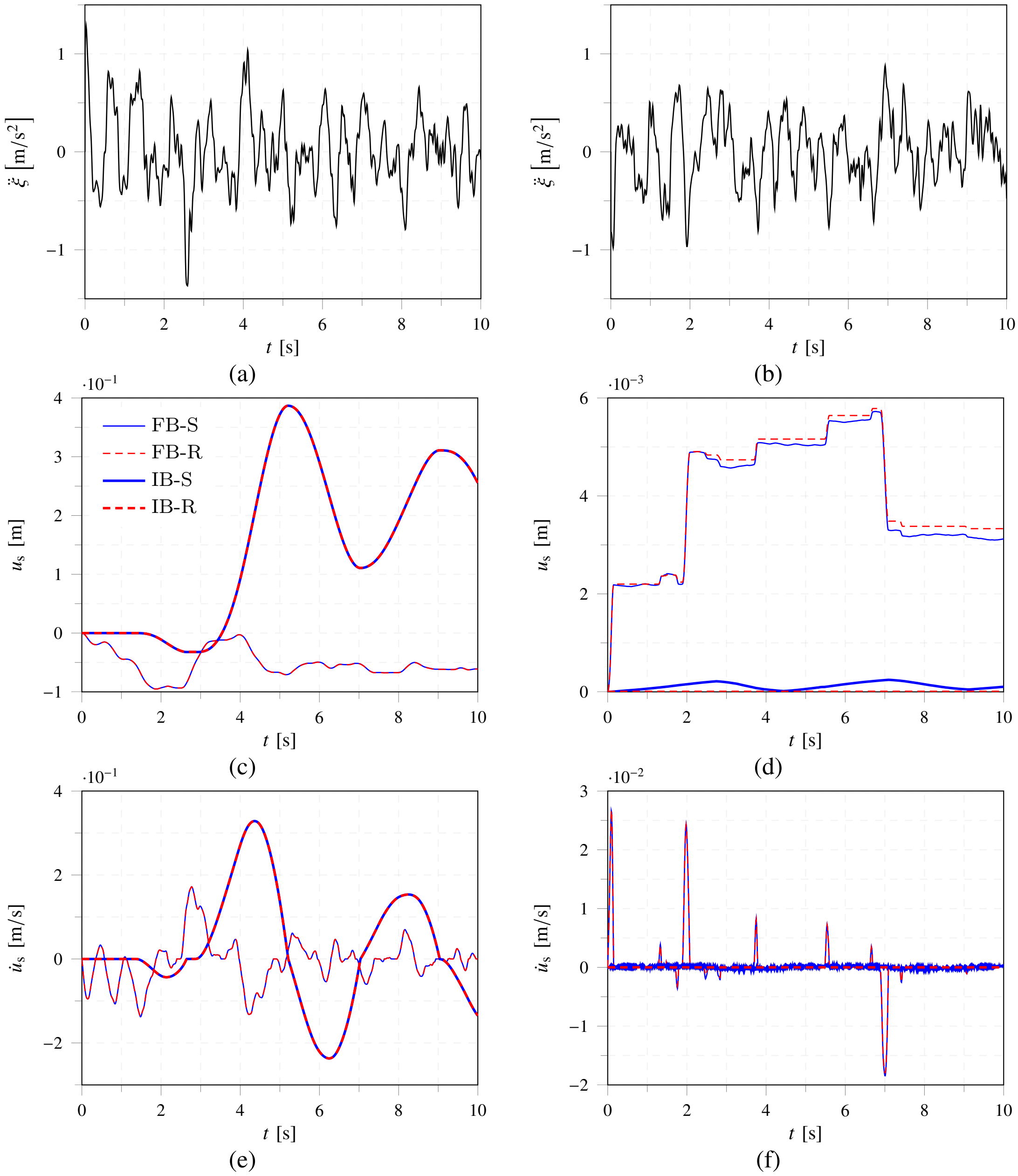
Overall, considering the probability of sticking negligible appears reasonable for low values of μ, or when the excitation is sufficiently high. Under these conditions, the approximation is admissible for use in the statistical linearisation procedure. In cases where these conditions are not met, such an approximation can be checked a priori. It is finally noted that demonstrating the validity of the piecewise linear solutions through comparisons with reference ones, falls outside the scope of this paper.

6.2 Statistical linearisation

The effectiveness of the statistical linearisation (SL) procedure described in § 5 is investigated next for the two systems under consideration.

Figure 5 compares the standard deviation of the relative velocity response determined using the SL procedure, with the nonstationary one numerically evaluated using the piecewise linear solutions via pertinent Monte Carlo (MC) simulation ($N = 200$ realisations), for various parameter combinations of ω_{b} and μ. The analysis has been carried out for a medium soil ($\omega_g = 10\,\mathrm{rad/s}$, $\zeta_g = 0.4$, $\omega_f = 1\,\mathrm{rad/s}$, $\zeta_f = 0.6$), and with parameters $S_0 = 0.003\,\mathrm{m}^2/\mathrm{s}^3$, $\gamma = 0.04$, and $\zeta = 0.05$.

As shown, for the fixed-base block, the standard deviation of the velocity response reaches stationarity in very short time. Further, for $\mu = 0.01$ and $\mu = 0.03$, there is good agreement between the MC and SL, confirming the validity of the expressions derived in § 5.1. Interestingly, the accuracy of the SL approximation deteriorates at higher values of μ, as evidenced by the large deviation for $\mu = 0.05$. This is in agreement with investigations carried out in [16] using the Kanai-Tajimi power spectrum.



(a) (b) (c) (d) (e) (f)

Figure 4: Response of fixed-base (FB) and base-isolated (IB) block, modelled with idealised rigid-plastic (R) and sliding (S) behaviour: realisations of base excitation (a, b) due to filtered white noise ($S_0 = 0.0025\,\mathrm{m}^2/\mathrm{s}^3$, $\omega_g = 10\,\mathrm{rad/s}$, $\zeta_g = 0.4$, $\omega_f = 1\,\mathrm{rad/s}$, $\zeta_f = 0.6$); corresponding relative displacement and relative velocity response time histories ($\gamma = 0.04$, $\omega_\mathrm{b} = 1.5\,\mathrm{rad/s}$ and $\zeta = 0.05$) for $\mu = 0.02$ (c, d) and $\mu = 0.06$ (e, f).

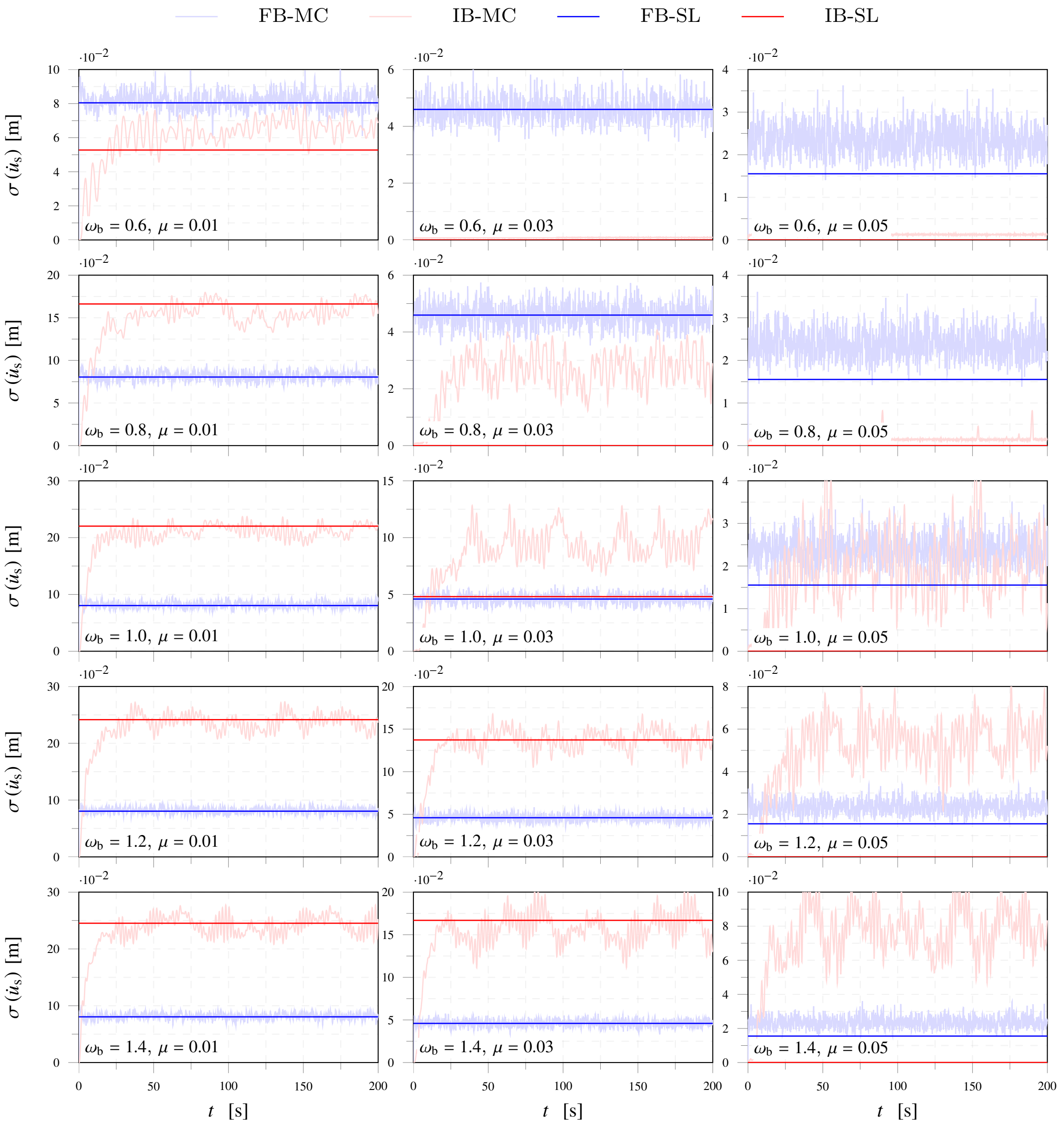


Figure 5: Standard deviation of relative velocity quantified for various parameter combinations of ω_{b} and μ: comparison of statistical linearisation (SL) and Monte Carlo (MC) simulation ($N = 200$ realisations), for the fixed-base (FB) and base-isolated (IB) block modelled with sliding behaviour. Reference parameters: $S_0 = 0.003\,\mathrm{m}^2/\mathrm{s}^3$; $\omega_g = 10\,\mathrm{rad/s}$, $\zeta_g = 0.4$, $\omega_f = 1\,\mathrm{rad/s}$, $\zeta_f = 0.6$ (medium soil); and $\gamma = 0.04$, $\zeta = 0.05$.

For the base-isolated block, the SL is found satisfactory for lower values of μ than those required for the fixed-base block, and for certain parameter combinations (e.g. $\omega_{\mathrm{b}} \geq 0.8$ and $\mu = 0.01$), while for other combinations (i.e. $\omega_{\mathrm{b}} = 0.8$ and $\mu = 0.03$) the iterative procedure employed for evaluating Eq. (26) does not converge and the solution breaks down. Further investigations are required to examine the influence of parameters γ and ζ on the effectiveness of the procedure.

6.3 Response spectra

A comparative study has been carried out with the purpose of assessing the effectiveness of the base isolation in suppressing the seismic forcing delivered to the block. Three soil types have been considered, namely, firm ($\omega_g = 15\,\mathrm{rad/s}$, $\zeta_g = 0.6$, $\omega_f = 1.5\,\mathrm{rad/s}$, $\zeta_f = 0.6$); medium ($\omega_g = 10\,\mathrm{rad/s}$, $\zeta_g = 0.4$, $\omega_f = 1\,\mathrm{rad/s}$, $\zeta_f = 0.6$) and soft ($\omega_g = 5\,\mathrm{rad/s}$, $\zeta_g = 0.2$, $\omega_f = 0.5\,\mathrm{rad/s}$, $\zeta_f = 0.6$). In all cases, a spectral density level $S_0 = 0.003\,\mathrm{m^2/s^3}$ has been considered, and the isolation parameters $\gamma = 0.04$ and $\zeta = 0.05$ have been assumed.

For each case, an ensemble of $N = 200$ synthetic ground motions has been generated using the procedure delineated in Appendix B, and Monte Carlo simulations have been used to quantify the stationary value of the standard deviation of the velocity response of each system.

Figure 6 plots the calculated standard deviation of the response, for several values of the parameters ω_b, and μ, where the standard deviation of the response of the base-isolated block ($\sigma_\mathrm{IB}(\dot{u}_\mathrm{s})$), has been normalised with respect to the corresponding value of the fixed-base ($\sigma_\mathrm{FB}(\dot{u}_\mathrm{s})$) model.

As shown, seismic isolation can attenuate the velocity response of the sliding block in all cases considered. Reducing the isolation frequency ω_b results in a reduction in the response standard deviation, and as $\omega_\mathrm{b} \to \infty$, the response of the isolated block approaches the response of the fixed-base block (i.e. $\sigma_\mathrm{IB}/\sigma_\mathrm{FB} \to 1$). Seismic isolation is effective for $\omega_\mathrm{b} < 1.25$, $\omega_\mathrm{b} < 1$ and $\omega_\mathrm{b} < 0.7$, for the firm, medium and soil, respectively, and higher values of ω_b are admissible as μ increases.

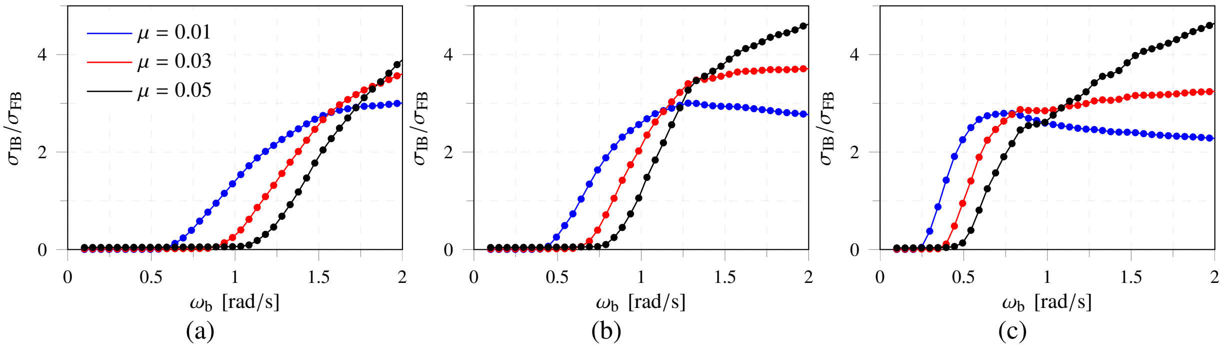


Figure 6: Isolated to non isolated standard deviation of relative velocity, quantified via $N = 200$ Monte Carlo realisations: (a) firm ($\omega_g = 15\,\mathrm{rad/s}$, $\zeta_g = 0.6$, $\omega_f = 1.5\,\mathrm{rad/s}$, $\zeta_f = 0.6$); (b) medium ($\omega_g = 10\,\mathrm{rad/s}$, $\zeta_g = 0.4$, $\omega_f = 1\,\mathrm{rad/s}$, $\zeta_f = 0.6$); and (c) soft ($\omega_g = 5\,\mathrm{rad/s}$, $\zeta_g = 0.2$, $\omega_f = 0.5\,\mathrm{rad/s}$, $\zeta_f = 0.6$) soil. Reference parameters: $S_0 = 0.003\,\mathrm{m^2/s^3}$, $\gamma = 0.04$ and $\zeta = 0.05$.

7 CONCLUSIONS

The modelling and response quantification of fixed-base and base-isolated rigid-plastic blocks were addressed in presence of stochastic forcing with indicative application to seismic engineering.

The dynamics of fixed-base rigid-plastic blocks were first overviewed, and equations governing their full dynamic interaction with a linear base-isolation system were presented. Highly-efficient piecewise numerical solutions were then derived for the two systems under consideration, which permit accurate quantification of the true nonlinear response due to a general-type excitation via pertinent Monte Carlo simulations.

A statistical linearisation approximation approach was adopted in view of approximating the strongly nonlinear systems during the sliding motion regime in presence of filtered white noise excitation, characterised by the Clough-Penzien stationary power spectrum, commonly used in earthquake engineering applications.

The accuracy of the linearisation approximation was examined and the effectiveness of the isolation system was assessed in attenuating the forcing delivered to the block.

The work delivers insights into the determination and understanding of the probabilistic characteristics of dynamically driven fixed-base and base-isolated rigid-plastic systems, motivating further investigations.

APPENDIX A. SOLVERS FOR THE SDOF AND TDOF NONLINEAR SYSTEMS

The piecewise linear solutions presented in § 3, govern the true nonlinear response of the systems considered and have been implemented in C++ resulting in standalone solver executable files. An iterative procedure based on the bisection method [27] has been adopted to identify state events (i.e. transition points such as the initiation and change in the regime of motion) and break down the solution in parts which have been later pieced together.

In order to confirm the validity of the solvers the solution has been compared to a MATLAB [28] implementation that has been prototyped using build-in Ordinary Differential Equation solvers. Specifically, ODE45 has been used, which is based on an explicit fourth- and fifth-order Runge-Kutta formulation. In this implementation, the continuous function $\tanh\left(\alpha\,\dot{u}_{\mathrm{s}}(t)\right)$ has been used in place of $\operatorname{sgn}\left(\dot{u}_{\mathrm{s}}(t)\right)$, where α is a large constant. Further, consistent initial conditions have been used, and MATLAB's odeset parameters have been set to AbsTol $=$ RelTol $=$ 10^{-8} and Refine $= 4$, which refer to relative and absolute solution tolerances and interpolation output, respectively. The option 'Events' has been invoked to identify state events.

APPENDIX B. SIMULATION OF STOCHASTIC FORCING

A stationary stochastic process representing the excitation time series ensemble, is generated through the summation of cosines with amplitudes and frequencies characterised by the power spectrum under consideration and random phases uniformly distributed over the interval $[0, 2\pi]$ [29]. In doing this, a frequency interval $[0, \tilde{\omega}]$ is considered, where $\tilde{\omega} = 100$ is an upper cut-off frequency, beyond which the spectral density is negligible. This interval is discretised using a frequency step $\Delta\omega = \tilde{\omega}/N_\omega$, where $N_\omega = \max\left\{N_0, \mathrm{ceil}\left(\frac{\tilde{\omega}T}{4\pi}\right)\right\}$ depends on $N_0 \approx 100$ (chosen such that the variance of the resulting process closely approximates the PSD) and on the temporal duration T of interest [30]. The time series is finally discretised using a time step $\Delta t \leq \frac{\pi}{4\tilde{\omega}}$.

References

[1] M. Riley, T. Coats, K. Haupt and D. Jacobson. Ride Severity Index - A new approach to quantifying the comparison of acceleration responses of high-speed craft. *11th International Conference on Fast Sea Transportation*, 2011.

[2] A. Malhotra and J. Penzien. Nondeterministic analysis of offshore structures. *Journal of the Engineering Mechanics Division*, **96**, 985–1003, 1970.

[3] S. Spence and M. Gioffre. Large scale reliability-based design optimization of wind excited tall buildings. *Probabilistic Engineering Mechanics*, **28**, 206–215, 2012.

[4] S. Kasinos. *Seismic response analysis of linear and nonlinear secondary structures*. PhD thesis, 2018.

[5] B. Westermo and F. Udwadia. Periodic response of a sliding oscillator system to harmonic excitation. *Earthquake Engineering and Structural Dynamics*, **11**, 135–146, 1983.

[6] N. Makris and M. Constantinou. Analysis of motion resisted by friction. I. Constant coulomb and linear/coulomb friction. *Mechanics of Structures and Machines*, **19**, 477–500, 1991.

[7] E. Voyagaki, G. Mylonakis and I. Psycharis. Rigid block sliding to idealized acceleration pulses. *Journal of Engineering Mechanics*, **138**, 1071–1083, 2012.

[8] E. Voyagaki, G. Mylonakis and I. Psycharis. A shift approach for the dynamic response of rigid-plastic systems. *Earthquake Engineering and Structural Dynamics*, **40**, 847–866, 2011.

[9] M. C. Constantinou, G. Gazetas and I. G. Tadjbakhsh. Stochastic seismic sliding of rigid mass supported through non-symmetric friction. *Earthquake Engineering and Structural Dynamics*, **12**, 777–793, 1984.

[10] K. Kanai. An empirical formula for the spectrum of strong earthquake motions. *Bulletin of the Earthquake Research Institute, University of Tokyo, Japan*, **39**, 1961.

[11] H. Tajimi. A statistical method of determining the maximum response of a building structure during an earthquake. *Proceedings of the 2nd World Conference in Earthquake Engineering*, 1960.

[12] S. Crandall, S. Lee and J. Williams. Accumulated slip of a friction-controlled mass excited by earthquake motions. *Journal of Applied Mechanics*, **41**, 1094–1098, 1974.

[13] S. Crandall and S. Lee. Biaxial slip of a mass on a foundation subject to earthquake motion. *Ingenieur-Archiv*, **45**, 361–370, 1976.

[14] G. Ahmadi. Stochastic Earthquake Response of Structures on Sliding Foundation. *International Journal of Engineering Science*, **21**, 93–102, 1983.

[15] T. Noguchi. The Response of a Building on Sliding Pads to Two Earthquake Models. *Journal of Sound and Vibration*, **103**, 437–442, 1985.

[16] M. C. Constantinou and I. G. Tadjbakhsh. Response of a sliding structure to filtered random excitation. *Journal of Structural Mechanics*, **12**, 401–418, 1984.

[17] J. Roberts and P. Spanos. *Random vibration and statistical linearisation*. Dover, 2003.

[18] I. Kougioumtzoglou and P. Spanos. Nonlinear MDOF system stochastic response determination via a dimension reduction approach. *Computers and Structures*, **126**, 135–148, 2013.

[19] J. Kelly. *Earthquake-resistant design with rubber*. Springer: London, 1997.

[20] B. Palazzo and L. Petti. Stochastic response comparison between base isolated and fixed-base structures. *Earthquake Spectra*, **13**, 77–96, 1997.

[21] F. Nikfar and D. Konstantinidis. Sliding response analysis of operational and functional components (OFC) in seismically isolated buildings. *3rd Specialty Conference on Disaster Prevention and Mitigation*, 2013.

[22] D. Konstantinidis and F. Nikfar. Seismic response of sliding equipment and contents in base-isolated buildings subjected to broadband ground motions. *Earthquake Engineering and Structural Dynamics*, **44**, 865–887, 2015.

[23] P. Roussis and S. Odysseos. Slide-rocking response of seismically-isolated rigid structures subjected to horizontal ground excitation. *2nd European Conference on Earthquake Engineering and Seismology*, 2014.

[24] H. Hong and S. Liu. Coulomb friction oscillator: modelling and responses to harmonic loads and base excitations. *Journal of Sound and Vibration*, **229**, 1171–1192, 2000.

[25] R. W. Clough and J. Penzien. *Dynamics of structures*. McGraw-Hill, 1975.

[26] A. Der Kiureghian and A. Neuenhofer. Response spectrum method for multi-support seismic excitations. *Earthquake Engineering and Structural Dynamics*, **21**, 713–740, 1992.

[27] K. Atkinson. *An introduction to numerical analysis*. John Wiley & Sons, 2008.

[28] The MathWorks Inc. MATLAB. *Release 8.2*, 2013.

[29] M. Shinozuka and G. Deodatis. Simulation of stochastic processes by spectral representation. *Applied Mechanics Reviews*, **44**, 191–204, 1991.

[30] G. Muscolino. *Dinamica delle strutture*. McGraw-Hill, 2001.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

RESPONSE SENSITIVITY OF STRUCTURAL SYSTEMS SUBJECTED TO FULLY NON-STATIONARY RANDOM PROCESSES

Tiziana Alderucci[1], Federica Genovese[1], Giuseppe Muscolino[1]

[1] Department of Engineering, University of Messina, Villaggio S. Agata, 98166 Messina, Italy;
e-mail: talderucci@unime.it; fedgenovese@unime.it; gmuscolino@unime.it

Abstract

A method for the evaluation of the statistics of response sensitivity of both classically and non-classically damped discrete linear structural systems under fully non-stationary stochastic seismic processes is presented. To do this the evolutionary frequency response function, also referred in literature as the time-frequency varying response function, plays a central role in the evaluation of the spectral characteristics of non-stationary response.

The proposed approach requires the following items: a) to write governing motion equations in state-variables, which are very suitable to evaluate the statistics of the response of both classically and non-classically damped discrete linear structural systems by an unified approach; b) to evaluate in explicit closed form solutions the derivatives of time-frequency response vector functions with respect to the parameters that define the modified structural model; c) to obtain the sensitivity of the structural response statistics by frequency domain integrals.

A numerical application shows that the proposed approach is suitable to cope with practical problems of engineering interest.

Keywords: Sensitivity analysis, Fully non-stationary processes, Non-geometric spectral moments, Evolutionary power spectral density function; Evolutionary frequency response function.

1 INTRODUCTION

During the analysis of structural systems, the reference structural parameters could be modified for design reasons. This is very frequent in optimization procedures, design of devices for vibrations control, etc. (see e.g., [1,2,3]). In this framework, the sensitivity analysis (i.e. the evaluation of partial derivatives of a performance measure with respect to system parameters) is a suitable vehicle to evaluate the response variation of structures under the influence of changes of parameter values.

Strong motion earthquakes are certainly the main critical actions for structures located in the seismically active regions of the earth. The analysis of recorded accelerograms in different sites shows that earthquake ground motion time-histories are non-stationary processes in both amplitude and frequency content. Then, the stationary models fail to reproduce the time-varying intensity, which is typical of real earthquakes ground-motion accelerograms. To take into account the time variability, the so-called *quasi-stationary* (or *uniformly modulated non-stationary*) random processes have been introduced [see e.g. 4,5]. These processes are constructed modulating the amplitude of a stationary zero-mean Gaussian random process through a deterministic function of time; for this reason they are also called *separable non-stationary stochastic processes*. However, these processes catch only the time-varying intensity of the accelerograms. To consider simultaneously both the amplitude and frequency changes, time-frequency varying deterministic modulating functions have been introduced in the characterization of the seismic process. The latter processes are referred as *fully or non-separable non-stationary stochastic processes* (see e.g., [6,7]).

Several papers have been devoted afterward to study the sensitivity of the response of structural systems subjected to stochastic excitations. As an example, Szopa [8] studied the stochastic sensitivity of the Van der Pol equation. Benfratello et al. [9] proposed a procedure, in the time domain, to evaluate the sensitivity of the statistical moments of the response of structural systems for stationary Gaussian and non-Gaussian white input processes. Proppe et al. [10] showed that the sensitivity analysis can be considered as an application of the Equivalent Linearization for design problem. Chaudhuri and Chakraborty [11] dealt with the response sensitivity evaluation in the frequency domain of structures subjected to non-stationary seismic processes. In Cacciola et al. [12] the sensitivities governing the evolution of spectral moments of the response are evaluated by solving set of differential equations once the Kronecker algebra is applied.

For linear structural systems subjected to non-stationary stochastic excitations, the *evolutionary frequency response function*, also referred in literature as the *time-frequency varying response function*, plays a central role in the evaluation of the statistics of the response [13]. In fact, by means of this function, it is possible to evaluate in explicit form the *evolutionary power spectral density* of the response and, consequently, *the non-geometric spectral moments*, which are required in the prediction of the safety of structural systems subjected to non-stationary random excitations (see e.g., [14-19]).

In recent studies [20,21], the senior authors, have evaluated in explicit form, for both classically and non-classically damped structural systems, the *time-frequency varying response function*.

In this study handy expressions for the sensitivities of *non-geometric spectral moments* of the structural response of linear classically or non-classically damped linear structural systems subjected to both separable and non-separable non-stationary excitations are evaluated. The proposed approach requires the following items: a) to determine sensitivities of *evolutionary frequency response functions* by means of explicit closed form solutions; b) to evaluate the sensitivity of the structural response statistics by frequency domain integrals.

A numerical application shows that the proposed approach is suitable to cope with practical problems of engineering interest.

2 DYNAMIC RESPONSE SENSITIVITIES FOR DETERMINISTIC LOADS

The sensitivity analysis consist in the evaluation of the change in the system response due to system parameter variations in the neighborhood of prefixed values, called “nominal parameter”. To this aim, preliminarily the set of significant parameters, for which the influence on the response has to be evaluated, are collected in the *r*-component vector $\boldsymbol{\alpha}$, where *r* being the number of the significant parameters taken into account. For a quiescent structural system at time $t = t_0$, the dependence of the damping and stiffness matrices of the structure, and of the response vector collecting the nodal displacements, on the actual value $\boldsymbol{\alpha}$ of the significant parameter vector, is expressed as:

$$\mathbf{M}\ddot{\mathbf{U}}(\boldsymbol{\alpha},t)+\mathbf{C}(\boldsymbol{\alpha})\dot{\mathbf{U}}(\boldsymbol{\alpha},t)+\mathbf{K}(\boldsymbol{\alpha})\mathbf{U}(\boldsymbol{\alpha},t)=-\mathbf{M}\boldsymbol{\tau}F(t);\quad \mathbf{U}(\boldsymbol{\alpha},t_0)=\mathbf{0} \tag{1}$$

where $\mathbf{M}$, $\mathbf{C}(\boldsymbol{\alpha})$, and $\mathbf{K}(\boldsymbol{\alpha})$ are the $n\times n$ mass, damping, and stiffness matrices of the structure, $\mathbf{U}(\boldsymbol{\alpha},t)$ is the *n*-dimensional vector of nodal displacements relative to the ground, $\boldsymbol{\tau}$ is the *n*-dimensional array listing the influence coefficients of the ground shaking, $F(t)$ is the time-dependent loading vector, and a dot over a variable denotes differentiation with respect to time.

Denoting with $\boldsymbol{\alpha}_0$ the vector of the significant parameters in correspondence of the nominal parameter~~s~~, any vector $\boldsymbol{\alpha}$ in the neighborhood of $\boldsymbol{\alpha}_0$ can be represented as:

$$\boldsymbol{\alpha}=\boldsymbol{\alpha}_0+\Delta\boldsymbol{\alpha}, \tag{2}$$

where $\Delta\boldsymbol{\alpha}$ is assumed to be a vector collecting small parameter variations with respect to the nominal parameter vector $\boldsymbol{\alpha}_0$. In order to evaluate the response sensitivity, the equation of motion (1) is written as:

$$\mathbf{M}\ddot{\mathbf{U}}(\boldsymbol{\alpha},t)+\left[\mathbf{C}(\boldsymbol{\alpha}_0)+\Delta\mathbf{C}(\boldsymbol{\alpha})\right]\dot{\mathbf{U}}(\boldsymbol{\alpha},t)+\left[\mathbf{K}(\boldsymbol{\alpha}_0)+\Delta\mathbf{K}(\boldsymbol{\alpha})\right]\mathbf{U}(\boldsymbol{\alpha},t)=-\mathbf{M}\boldsymbol{\tau}F(t);\quad \mathbf{U}(\boldsymbol{\alpha},t_0)=\mathbf{0} \tag{3}$$

in which $\mathbf{K}(\boldsymbol{\alpha}_0)$ and $\mathbf{C}(\boldsymbol{\alpha}_0)$ are the stiffness and damping matrices of the structure evaluated in correspondence of the nominal parameter vector $\boldsymbol{\alpha}_0$, while, $\Delta\mathbf{C}(\boldsymbol{\alpha})=\mathbf{C}(\boldsymbol{\alpha})-\mathbf{C}(\boldsymbol{\alpha}_0)$ and $\Delta\mathbf{K}(\boldsymbol{\alpha})=\mathbf{K}(\boldsymbol{\alpha})-\mathbf{K}(\boldsymbol{\alpha}_0)$. It follows that the structural system is non-classically damped. To solve Eq.(1) the equations of motion have to be rewritten in state variables:

$$\dot{\mathbf{Z}}(\boldsymbol{\alpha},t)=\mathbf{D}(\boldsymbol{\alpha})\mathbf{Z}(\boldsymbol{\alpha},t)+\mathbf{w}F(t);\quad \mathbf{Z}(\boldsymbol{\alpha},t_0)=\mathbf{0} \tag{4}$$

where $\mathbf{Z}(\boldsymbol{\alpha},t)$ is the *2n*-state vector variable while the $2n\times 2n$ matrix $\mathbf{D}(\boldsymbol{\alpha})$ and the *2n*-vector **w** are defined as:

$$\mathbf{Z}(\boldsymbol{\alpha},t)=\begin{bmatrix}\mathbf{U}(\boldsymbol{\alpha},t)\\ \dot{\mathbf{U}}(\boldsymbol{\alpha},t)\end{bmatrix};\quad \mathbf{D}(\boldsymbol{\alpha})=\begin{bmatrix}\mathbf{O}_{n,n} & \mathbf{I}_n\\ -\mathbf{M}^{-1}\mathbf{K}(\boldsymbol{\alpha}) & -\mathbf{M}^{-1}\mathbf{C}(\boldsymbol{\alpha})\end{bmatrix};\quad \mathbf{w}=\begin{bmatrix}\mathbf{0}_n\\ -\boldsymbol{\tau}\end{bmatrix}; \tag{5}$$

where $\mathbf{I}_n$ and $\mathbf{O}_{n,n}$ are respectively the identity and the zero matrices of $n\times n$ order while $\mathbf{0}_n$ stands for a *n*-dimensional vector. In order to evaluate the structural response the $2n\times 2n$

transition matrix $\boldsymbol{\Theta}(\boldsymbol{\alpha},t)$ has to be introduced [22,23], and for non-classically damped systems this matrix can be evaluated as:

$$\boldsymbol{\Theta}(\boldsymbol{\alpha},t)=\exp\left[\mathbf{D}(\boldsymbol{\alpha})t\right]=\boldsymbol{\Psi}(\boldsymbol{\alpha})\exp\left[\boldsymbol{\Lambda}(\boldsymbol{\alpha})t\right]\boldsymbol{\Psi}^{T}(\boldsymbol{\alpha})\mathbf{A}(\boldsymbol{\alpha})\equiv\boldsymbol{\Psi}^{*}(\boldsymbol{\alpha})\exp\left[\boldsymbol{\Lambda}^{*}(\boldsymbol{\alpha})t\right]\boldsymbol{\Psi}^{*T}(\boldsymbol{\alpha})\mathbf{A}(\boldsymbol{\alpha}) \tag{6}$$

in which $\mathbf{D}(\boldsymbol{\alpha})$ has been defined in Eq.(5), $\boldsymbol{\Lambda}(\boldsymbol{\alpha})$ and $\boldsymbol{\Psi}(\boldsymbol{\alpha})$ are the complex matrices collecting eigenvalues and eigenvectors respectively, depending of uncertain parameters $\boldsymbol{\alpha}$. Formally, these matrices can be evaluated by applying the *complex modal analysis*. According to this analysis the following coordinate transformation is introduced:

$$\mathbf{Z}(\boldsymbol{\alpha},t)=\boldsymbol{\Psi}(\boldsymbol{\alpha})\mathbf{X}(\boldsymbol{\alpha},t). \tag{7}$$

If *m* is the number of modes selected for the analysis, $\mathbf{X}(\boldsymbol{\alpha},t)$ is a complex vector of order 2*m* and the complex matrix $\boldsymbol{\Psi}(\boldsymbol{\alpha})$, of order $(2n\times 2m)$, collects the complex eigenvectors, solutions of the following eigenproblem:

$$\mathbf{D}^{-1}(\boldsymbol{\alpha})\boldsymbol{\Psi}(\boldsymbol{\alpha})=\boldsymbol{\Psi}(\boldsymbol{\alpha})\boldsymbol{\Lambda}^{-1}(\boldsymbol{\alpha});\quad \boldsymbol{\Psi}^{T}(\boldsymbol{\alpha})\mathbf{A}(\boldsymbol{\alpha})\boldsymbol{\Psi}(\boldsymbol{\alpha})=\mathbf{I}_{2m} \tag{8}$$

where the superscript T denotes the transpose operator, $\boldsymbol{\Lambda}$ is the diagonal matrix collecting the $2m$ complex eigenvalues and

$$\mathbf{A}(\boldsymbol{\alpha})=\begin{bmatrix}\mathbf{C}(\boldsymbol{\alpha}) & \mathbf{M}\\ \mathbf{M} & \mathbf{O}_{n,n}\end{bmatrix}. \tag{9}$$

In order to evaluate the first-order sensitivity, Eq.(4) must be differentiated with respect to $\boldsymbol{\alpha}$, setting $\boldsymbol{\alpha}=\boldsymbol{\alpha}_0$, leading to the following differential equation [12]:

$$\dot{\mathbf{s}}_{\mathbf{Z},i}(\boldsymbol{\alpha}_0,t)=\mathbf{D}(\boldsymbol{\alpha}_0)\mathbf{s}_{\mathbf{Z},i}(\boldsymbol{\alpha}_0,t)+\bar{\mathbf{F}}(\boldsymbol{\alpha}_0,t);\quad \mathbf{s}_{\mathbf{Z},i}(\boldsymbol{\alpha}_0,t_0)=\mathbf{0} \tag{10}$$

where the pseudo-force vector $\bar{\mathbf{F}}(\boldsymbol{\alpha}_0,t)$ is given by the equation

$$\bar{\mathbf{F}}(\boldsymbol{\alpha}_0,t)=\mathbf{D}'_i(\boldsymbol{\alpha}_0)\mathbf{Z}(\boldsymbol{\alpha}_0,t) \tag{11}$$

in which all the quantities are known. In Eq.(11) the matrix $\mathbf{D}'_i(\boldsymbol{\alpha}_0)$ can be readily determined deriving the matrix $\mathbf{D}(\boldsymbol{\alpha})$ with respect to *i*-th significant parameter α_i. That is,

$$\mathbf{s}_{\mathbf{Z},i}(\boldsymbol{\alpha}_0,t)=\left.\frac{\partial\mathbf{Z}(\boldsymbol{\alpha},t)}{\partial\alpha_i}\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0};\quad \mathbf{D}'_i(\boldsymbol{\alpha}_0)=\left.\frac{\partial}{\partial\alpha_i}\mathbf{D}(\boldsymbol{\alpha})\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0}=\begin{bmatrix}\mathbf{O}_{n,n} & \mathbf{O}_{n,n}\\ -\mathbf{M}^{-1}\mathbf{K}'_i(\boldsymbol{\alpha}_0) & -\mathbf{M}^{-1}\mathbf{C}'_i(\boldsymbol{\alpha}_0)\end{bmatrix} \tag{12}$$

where

$$\mathbf{K}'_i(\boldsymbol{\alpha}_0)=\left.\frac{\partial}{\partial\alpha_i}\mathbf{K}(\boldsymbol{\alpha})\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0};\quad \mathbf{C}'_i(\boldsymbol{\alpha}_0)=\left.\frac{\partial}{\partial\alpha_i}\mathbf{C}(\boldsymbol{\alpha})\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0}. \tag{13}$$

It is noted that the set of first-order ordinary differential in Eq.(10) is formally similar to Eq.(4), which represents the equation of motion of the structural system in the state variable space. This means that the derivatives of the response with respect to the *i*-th parameter can be calculated by means of the same procedures used for response evaluation, that is:

$$\mathbf{s}_{\mathbf{Z},i}\left(\boldsymbol{\alpha}_0,t\right)=\boldsymbol{\Psi}\left(\boldsymbol{\alpha}_0\right)\int_{t_0}^{t}\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0,t-\tau\right)\right]\boldsymbol{\Psi}^T\left(\boldsymbol{\alpha}_0\right)\mathbf{A}\left(\boldsymbol{\alpha}_0\right)\bar{\mathbf{F}}\left(\boldsymbol{\alpha}_0,\tau\right)\mathrm{d}\tau \tag{14}$$

It follows that the sensitivity vector of the response in state variables can be evaluated as:

$$\begin{aligned}\mathbf{s}_{\mathbf{Z},i}\left(\boldsymbol{\alpha}_0,t\right)&=\boldsymbol{\Psi}\left(\boldsymbol{\alpha}_0\right)\int_{t_0}^{t}\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0,t-\tau\right)\right]\boldsymbol{\Psi}^T\left(\boldsymbol{\alpha}_0\right)\mathbf{A}\left(\boldsymbol{\alpha}_0\right)\mathbf{D}_i'\left(\boldsymbol{\alpha}_0\right)\mathbf{Z}\left(\boldsymbol{\alpha}_0,\tau\right)\mathrm{d}\tau\\&=\boldsymbol{\Psi}\left(\boldsymbol{\alpha}_0\right)\int_{t_0}^{t}\left\{\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0,t-\tau\right)\right]\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\left[\int_{t_0}^{\tau}\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0,\tau-\rho\right)\right]F\left(\rho\right)\mathrm{d}\rho\right]\right\}\mathrm{d}\tau\,\mathbf{v}\left(\boldsymbol{\alpha}_0\right)\end{aligned} \tag{15}$$

where

$$\mathbf{Z}\left(\boldsymbol{\alpha}_0,\tau\right)=\boldsymbol{\Psi}\left(\boldsymbol{\alpha}_0\right)\left[\int_{t_0}^{\tau}\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0,\tau-\rho\right)\right]F\left(\rho\right)\mathrm{d}\rho\right]\mathbf{v}\left(\boldsymbol{\alpha}_0\right) \tag{16}$$

and

$$\mathbf{v}(\boldsymbol{\alpha})=\boldsymbol{\Psi}^T(\boldsymbol{\alpha})\mathbf{A}(\boldsymbol{\alpha})\mathbf{w};\quad \mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)=\boldsymbol{\Psi}^T\left(\boldsymbol{\alpha}_0\right)\mathbf{A}\left(\boldsymbol{\alpha}_0\right)\mathbf{D}_i'\left(\boldsymbol{\alpha}_0\right)\boldsymbol{\Psi}\left(\boldsymbol{\alpha}_0\right). \tag{17}$$

For deterministic excitation the sensitivity of the response can be evaluated by a step-by-step procedure [12,22,23].

3 DYNAMIC RESPONSE SENSITIVITY FOR FULLY NON-STATIONARY STOCHASTIC LOAD PROCESSES

3.1 Closed form solutions for the time-frequency varying response vector function

In the framework of non-stationary analysis of structures, the *spectral moments* can be evaluated in compact form by introducing the *pre-envelope covariance* (*PEC*) matrix. This matrix, in nodal space, is a $2n\times 2n$ Hermitian matrix, that, for non-classically damped systems, can be evaluated formally as [18,19]:

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\boldsymbol{\alpha},t)=\mathrm{E}\left\langle\mathbf{Z}(\boldsymbol{\alpha},t)\mathbf{Z}^{*T}(\boldsymbol{\alpha},t)\right\rangle&=\begin{bmatrix}\mathrm{E}\left\langle\mathbf{U}(\boldsymbol{\alpha},t)\mathbf{U}^{*T}(\boldsymbol{\alpha},t)\right\rangle & \mathrm{E}\left\langle\mathbf{U}(\boldsymbol{\alpha},t)\dot{\mathbf{U}}^{*T}(\boldsymbol{\alpha},t)\right\rangle\\ \mathrm{E}\left\langle\dot{\mathbf{U}}(\boldsymbol{\alpha},t)\mathbf{U}^{*T}(\boldsymbol{\alpha},t)\right\rangle & \mathrm{E}\left\langle\dot{\mathbf{U}}(\boldsymbol{\alpha},t)\dot{\mathbf{U}}^{*T}(\boldsymbol{\alpha},t)\right\rangle\end{bmatrix}\\&=\begin{bmatrix}\boldsymbol{\Lambda}_{0,\mathbf{UU}}(\boldsymbol{\alpha},t) & \mathrm{i}\boldsymbol{\Lambda}_{1,\mathbf{UU}}(\boldsymbol{\alpha},t)\\ -\mathrm{i}\boldsymbol{\Lambda}_{1,\mathbf{UU}}^{*T}(\boldsymbol{\alpha},t) & \boldsymbol{\Lambda}_{2,\mathbf{UU}}(\boldsymbol{\alpha},t)\end{bmatrix}\end{aligned} \tag{18}$$

where $\mathbf{Z}(\boldsymbol{\alpha},t)$ is the nodal state variable vector solution of Eq.(4), while the matrices $\boldsymbol{\Lambda}_{i,\mathbf{UU}}(\boldsymbol{\alpha},t)$ collect the *non-geometric spectral moments* (*NGSM*) [15-19]. After some algebra, the nodal *PEC* matrix, can be evaluated in time-domain, for quiescent structural systems (at time $t_0=0$), as follows:

$$\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\boldsymbol{\alpha},t)=\int_{t_0}^{t}\int_{t_0}^{t}\boldsymbol{\Theta}\left(\boldsymbol{\alpha},t-\tau_1\right)\mathbf{w}\mathbf{w}^T\boldsymbol{\Theta}^T\left(\boldsymbol{\alpha},t-\tau_2\right)R_{FF}\left(\tau_1,\tau_2\right)\mathrm{d}\tau_1\,\mathrm{d}\tau_2 \tag{19}$$

where $\mathbf{w}$ is the vector defined in Eq.(5), $R_{FF}(\tau_1,\tau_2)$ is the complex autocorrelation function and $\boldsymbol{\Theta}(\boldsymbol{\alpha},t)$ is the transition matrix defined in Eq.(6). By substituting the transition matrix (6) into Eq.(19), the nodal *PEC* matrix can be written also as [20,21]:

$$\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\boldsymbol{\alpha},t)=\boldsymbol{\Psi}^*(\boldsymbol{\alpha})\left\{\int_{t_0}^{t}\int_{t_0}^{t}\exp\left[\boldsymbol{\Lambda}^*\left(\boldsymbol{\alpha},t-\tau_1\right)\right]\mathbf{v}^*(\boldsymbol{\alpha})\mathbf{v}^T(\boldsymbol{\alpha})\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha},t-\tau_2\right)\right]R_{FF}\left(\tau_1,\tau_2\right)\mathrm{d}\tau_1\,\mathrm{d}\tau_2\right\}\boldsymbol{\Psi}^T(\boldsymbol{\alpha}) \tag{20}$$

where the vector $\mathbf{v}(\boldsymbol{\alpha})$ has been defined in Eq.(17). In this equation the autocorrelation function is defined as follows:

$$R_{FF}\left(t_1,t_2\right)=\int_0^\infty \exp\left[\mathrm{i}\omega\left(t_1-t_2\right)\right]a(\omega,t_1)\,a^*(\omega,t_2)G_0(\omega)\mathrm{d}\omega \tag{21}$$

where $a(\omega,t)\equiv a^*(-\omega,t)$ is the modulating function, that for *fully non-stationary processes* depends on both time and frequency. In Eq.(21) $G_0\left(\omega\right)$ is the one-sided *power spectral density* (*PSD*) function of the stationary counterpart of the fully not stationary input process having the one-sided *evolutionary PSD* (*EPSD*) defined as $G_{FF}(\omega,t)=\left|a(\omega,t)\right|^2 G_0(\omega)$. By substituting Eq.(21) into Eq.(20), it is possible to evaluate the nodal *PEC* matrix (18) as:

$$\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\boldsymbol{\alpha},t)=\int_0^\infty \mathbf{G}_{\mathbf{ZZ}}(\boldsymbol{\alpha},\omega,t)\,\mathrm{d}\,\omega=\boldsymbol{\Psi}^*\,(\boldsymbol{\alpha})\boldsymbol{\Sigma}_{\mathbf{XX}}(\boldsymbol{\alpha},t)\boldsymbol{\Psi}^T(\boldsymbol{\alpha}) \tag{22}$$

where $\boldsymbol{\Sigma}_{\mathbf{XX}}(\boldsymbol{\alpha},t)$ is the *PEC* matrix in the complex modal state subspace defined as:

$$\boldsymbol{\Sigma}_{\mathbf{XX}}(\boldsymbol{\alpha},t)=\int_0^\infty \mathbf{G}_{\mathbf{XX}}(\boldsymbol{\alpha},\omega,t)\,\mathrm{d}\,\omega \tag{23}$$

where $\mathbf{G}_{\mathbf{XX}}(\boldsymbol{\alpha},\omega,t)$ is the one-sided *EPSD* function matrix of the modal complex response, that is:

$$\mathbf{G}_{\mathbf{XX}}(\boldsymbol{\alpha},\omega,t)=G_0(\omega)\,\mathbf{X}^*(\boldsymbol{\alpha},\omega,t)\,\mathbf{X}^T(\boldsymbol{\alpha},\omega,t). \tag{24}$$

Notice that, in evaluating the nodal *PEC* matrix of Eq.(22), the following coordinate transformation has been introduced:

$$\mathbf{Z}(\boldsymbol{\alpha},\omega,t)=\boldsymbol{\Psi}(\boldsymbol{\alpha})\,\mathbf{X}(\boldsymbol{\alpha},\omega,t) \tag{25}$$

where $\mathbf{Z}(\boldsymbol{\alpha},\omega,t)$ is the *time-frequency varying response* (*TFR*) vector function of the nodal response, while $\mathbf{X}(\boldsymbol{\alpha},\omega,t)$ is the *TFR* vector function of the modal response, defined as:

$$\mathbf{X}(\boldsymbol{\alpha},\omega,t)=\int_{t_0}^{t}\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha},t-\tau\right)\right]\exp\left(\mathrm{i}\omega\tau\right)a(\omega,\tau)\mathrm{d}\tau\,\mathbf{v}(\boldsymbol{\alpha}). \tag{26}$$

In order to evaluate in explicit form the *TFR* vector function of modal response, the vector $\mathbf{X}(\boldsymbol{\alpha},\omega,t)$ can be evaluated as the solution of a set of $2m$ first order uncoupled differential equations, since the following relationship holds [20]:

$$\dot{\mathbf{X}}(\boldsymbol{\alpha},\omega,t)=\boldsymbol{\Lambda}(\boldsymbol{\alpha})\,\mathbf{X}(\boldsymbol{\alpha},\omega,t)+\mathbf{v}(\boldsymbol{\alpha})\exp\left(\mathrm{i}\omega t\right)a(\omega,t)\boldsymbol{\mathcal{U}}\left(t-t_0\right);\quad \mathbf{X}(\boldsymbol{\alpha},\omega,t_0)=\mathbf{X}_0(\boldsymbol{\alpha},\omega) \tag{27}$$

where $\mathbf{X}(\boldsymbol{\alpha},\omega,t_0)\equiv\mathbf{X}_0(\boldsymbol{\alpha},\omega)$ is the vector of the initial condition at time $t=t_0$ and $\mathcal{U}(t)$ is the *unit step function.*

If the particular solution of Eq.(27), $\mathbf{X}_\mathrm{p}(\boldsymbol{\alpha},\omega,t)$, can be determined in explicit form, the *TFR* vector function, according to the dynamics of non-classically damped systems, can be written as [21]:

$$\mathbf{X}(\boldsymbol{\alpha},\omega,t)=\left\{\mathbf{X}_\mathrm{p}(\boldsymbol{\alpha},\omega,t)+\exp\left[\boldsymbol{\Lambda}(\boldsymbol{\alpha})t\right]\left[\mathbf{X}_0(\boldsymbol{\alpha},\omega)-\mathbf{X}_\mathrm{p}(\boldsymbol{\alpha},\omega,t_0)\right]\right\}\mathcal{U}(t-t_0). \tag{28}$$

The analytical expression of the particular solution vector $\mathbf{X}_\mathrm{p}(\boldsymbol{\alpha},\omega,t)$, can be easily obtained in closed form for the most common models of modulating function $a(\omega,t)$ proposed in literature [4-7]. It has been recently shown that the most useful time-frequency functions to model the fully non-stationary seismic excitation can be written as [6]:

$$a(\omega,t)=\varepsilon(\omega)\,(t-t_0)\,\exp\left[-\alpha_a(\omega)(t-t_0)\right]\mathcal{U}(t-t_0); \tag{29}$$

where $\varepsilon(\omega)$ and $\alpha_a(\omega)$ could be complex functions which have to be chosen to satisfy the condition: $a(\omega,t)\equiv a^*(-\omega,t)$.

It has been demonstrated that for quiescent structural systems at time $t_0=0$, $\mathbf{X}_0(\boldsymbol{\alpha},\omega)=\mathbf{0}$, and for the modulating function, defined in Eq.(29), the vector $\mathbf{X}(\boldsymbol{\alpha},\omega,t)$, defined in Eq.(28), can be evaluated in explicit form as [20,21]:

$$\mathbf{X}(\boldsymbol{\alpha},\omega,t)=-\,\varepsilon(\omega)\left\{\exp\left[-\beta(\omega)\,t\right]\left[\boldsymbol{\Gamma}^2(\omega)+t\,\boldsymbol{\Gamma}(\omega)\right]-\exp\left[\boldsymbol{\Lambda}(\boldsymbol{\alpha})t\right]\boldsymbol{\Gamma}^2(\omega)\right\}\mathbf{v}(\boldsymbol{\alpha})\,\mathcal{U}(t) \tag{30}$$

where $\beta(\omega)=\alpha_a(\omega)-\mathrm{i}\omega$ and $\boldsymbol{\Gamma}(\omega)$ is the diagonal matrix, function of the $\boldsymbol{\alpha}$ vector too that for simplicity's sake is omitted, defined as:

$$\boldsymbol{\Gamma}(\omega)\equiv\boldsymbol{\Gamma}(\boldsymbol{\alpha},\omega)=\left[\boldsymbol{\Lambda}(\boldsymbol{\alpha})+\beta(\omega)\mathbf{I}_{2m}\right]^{-1}. \tag{31}$$

Then, it is possible to evaluate, in explicit form, the *EPSD* function matrix of the modal response by substituting Eq.(30) into Eq.(24) which can be written as:

$$\mathbf{G}_{\mathbf{ZZ}}(\boldsymbol{\alpha},\omega,t)=G_0(\omega)\,\boldsymbol{\Psi}^*(\boldsymbol{\alpha})\,\mathbf{X}^*(\boldsymbol{\alpha},\omega,t)\,\mathbf{X}^T(\boldsymbol{\alpha},\omega,t)\,\boldsymbol{\Psi}^T(\boldsymbol{\alpha}) \tag{32}$$

3.2 Closed form solutions for the sensitivity of time-frequency varying response vector function

By differentiating the *PEC* matrix, defined in Eq.(18), it is possible to evaluate its sensitivity with respect to the *i*-th parameter, as follows:

$$\boldsymbol{\Sigma}_{\mathbf{s}_{\mathbf{Z},i}\mathbf{s}_{\mathbf{Z},i}}(\boldsymbol{\alpha}_0,t)=\left.\frac{\partial\boldsymbol{\Sigma}_{\mathbf{ZZ}}(\boldsymbol{\alpha},t)}{\partial\alpha_i}\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0}=\mathrm{E}\left\langle\mathbf{Z}^*(\boldsymbol{\alpha}_0,t)\mathbf{s}_{\mathbf{Z},i}^T(\boldsymbol{\alpha}_0,t)\right\rangle+\mathrm{E}\left\langle\mathbf{Z}^*(\boldsymbol{\alpha}_0,t)\mathbf{s}_{\mathbf{Z},i}^T(\boldsymbol{\alpha}_0,t)\right\rangle^{*T} \tag{33}$$

where the vector $\mathbf{s}_{\mathbf{Z},i}(\boldsymbol{\alpha}_0,t)$ has been defined in Eq.(12). It follows that, analogously to Eq.(22), the following relationship holds:

$$\mathrm{E}\left\langle\mathbf{Z}^*(\boldsymbol{\alpha}_0,t)\mathbf{s}_{\mathbf{z},i}^T(\boldsymbol{\alpha}_0,t)\right\rangle=\boldsymbol{\Psi}^*(\boldsymbol{\alpha}_0)\left\{\int_0^\infty\mathbf{X}^*(\boldsymbol{\alpha}_0,\omega,t)\mathbf{Y}_i^T(\boldsymbol{\alpha}_0,\omega,t)\,G_0(\omega)\mathrm{d}\omega\right\}\boldsymbol{\Psi}^T(\boldsymbol{\alpha}_0) \tag{34}$$

where the vector $\mathbf{Y}_i(\boldsymbol{\alpha}_0,\omega,t)$ is the *sensitivity* of *TFR* vector function with respect to the parameter α_i:

$$\mathbf{Y}_i(\boldsymbol{\alpha}_0,\omega,t)=\int_0^t \exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0,t-\tau\right)\right]\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\mathbf{X}(\boldsymbol{\alpha}_0,\omega,\tau)\,\mathrm{d}\tau. \tag{35}$$

Alternatively the sensitivity of *PEC* matrix can be defined as:

$$\boldsymbol{\Sigma}_{\mathrm{s}_{\mathrm{Z},i}\mathrm{s}_{\mathrm{Z},j}}\left(\boldsymbol{\alpha}_0,t\right)=\begin{bmatrix} \left.\dfrac{\partial}{\partial\alpha_i}\boldsymbol{\Lambda}_{0,\mathbf{UU}}(\boldsymbol{\alpha},t)\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0} & \left.\mathrm{i}\dfrac{\partial}{\partial\alpha_i}\boldsymbol{\Lambda}_{1,\mathbf{UU}}(\boldsymbol{\alpha},t)\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0} \\ \left.-\mathrm{i}\dfrac{\partial}{\partial\alpha_i}\boldsymbol{\Lambda}^{*T}_{1,\mathbf{UU}}(\boldsymbol{\alpha},t)\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0} & \left.\dfrac{\partial}{\partial\alpha_i}\boldsymbol{\Lambda}_{2,\mathbf{UU}}(\boldsymbol{\alpha},t)\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0} \end{bmatrix}=\int_0^\infty \mathbf{G}_{\mathrm{s}_{\mathrm{Z},i}\mathrm{s}_{\mathrm{Z},j}}\left(\boldsymbol{\alpha}_0,t\right)\mathrm{d}\omega \tag{36}$$

whose elements are the sensitivity of first three spectral moments with respect to the parameter α_i. In this equation $\mathbf{G}_{\mathrm{s}_{\mathrm{Z},i}\mathrm{s}_{\mathrm{Z},j}}\left(\boldsymbol{\alpha}_0,t\right)$ is the sensitivity of the one-sided *EPSD* function of nodal response, that is:

$$\begin{aligned}\mathbf{G}_{\mathrm{s}_{\mathrm{Z},i}\mathrm{s}_{\mathrm{Z},j}}\left(\boldsymbol{\alpha}_0,t\right)&=\left.\frac{\partial\mathbf{G}_{\mathbf{ZZ}}(\boldsymbol{\alpha},\omega,t)}{\partial\alpha_i}\right|_{\boldsymbol{\alpha}=\boldsymbol{\alpha}_0}\\ &=G_0(\omega)\,\boldsymbol{\Psi}^*(\boldsymbol{\alpha})\left[\mathbf{X}^*(\boldsymbol{\alpha}_0,\omega,t)\,\mathbf{Y}_i^T(\boldsymbol{\alpha}_0,\omega,t)+\mathbf{Y}_i^*(\boldsymbol{\alpha}_0,\omega,t)\,\mathbf{X}^T(\boldsymbol{\alpha}_0,\omega,t)\right]\boldsymbol{\Psi}^T(\boldsymbol{\alpha}).\end{aligned} \tag{37}$$

The main problem is now to evaluate the vector $\mathbf{Y}_i(\boldsymbol{\alpha}_0,\omega,t)$, defined in Eq.(35), taking into account Eq.(30). This vector function can be evaluated as solution of the following differential equation with zero start conditions at time $t_0=0$:

$$\dot{\mathbf{Y}}_i(\boldsymbol{\alpha}_0,\omega,t)=\boldsymbol{\Lambda}(\boldsymbol{\alpha}_0)\,\mathbf{Y}_i(\boldsymbol{\alpha}_0,\omega,t)+\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\mathbf{X}(\boldsymbol{\alpha}_0,\omega,\tau)\,\mathcal{U}\left(t-t_0\right);\quad \mathbf{Y}_i(\boldsymbol{\alpha}_0,\omega,0)=\mathbf{0}. \tag{38}$$

To perform the solution of this set of differential equations the vector defined in Eq.(30) is rewritten as:

$$\mathbf{X}\left(\boldsymbol{\alpha}_0,\omega,t\right)=\mathbf{X}_1\left(\boldsymbol{\alpha}_0,\omega,t\right)+\mathbf{X}_2\left(\boldsymbol{\alpha}_0,\omega,t\right) \tag{39}$$

where

$$\begin{aligned}&\mathbf{X}_1\left(\boldsymbol{\alpha}_0,\omega,t\right)=-\varepsilon(\omega)\exp\left[-\beta(\omega)\,t\right]\left[\boldsymbol{\Gamma}^2(\omega)+t\,\boldsymbol{\Gamma}(\omega)\right]\mathbf{v}(\boldsymbol{\alpha}_0)\,\mathcal{U}\left(t\right);\\ &\mathbf{X}_2\left(\boldsymbol{\alpha}_0,\omega,t\right)=\varepsilon(\omega)\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0\right)t\right]\,\boldsymbol{\Gamma}^2(\omega)\mathbf{v}(\boldsymbol{\alpha}_0)\,\mathcal{U}\left(t\right).\end{aligned} \tag{40}$$

It follows that it is possible to split the vector solution of Eq.(38) as the sum of two vectors, solutions of the following two sets of differential equations, with zero start initial conditions at time $t_0=0$:

$$\begin{aligned}&\dot{\mathbf{Y}}_{i,1}(\boldsymbol{\alpha}_0,\omega,t)=\boldsymbol{\Lambda}(\boldsymbol{\alpha}_0)\,\mathbf{Y}_{i,1}(\boldsymbol{\alpha}_0,\omega,t)+\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\mathbf{X}_1(\boldsymbol{\alpha}_0,\omega,t);\quad \mathbf{Y}_{i,1}(\boldsymbol{\alpha}_0,\omega,0)=\mathbf{0}\\ &\dot{\mathbf{Y}}_{i,2}(\boldsymbol{\alpha}_0,\omega,t)=\boldsymbol{\Lambda}(\boldsymbol{\alpha}_0)\,\mathbf{Y}_{i,2}(\boldsymbol{\alpha}_0,\omega,t)+\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\mathbf{X}_2(\boldsymbol{\alpha}_0,\omega,t);\quad \mathbf{Y}_{i,2}(\boldsymbol{\alpha}_0,\omega,0)=\mathbf{0}\end{aligned} \tag{41}$$

It follows that the *sensitivity TFR*'s vector function can be evaluate in closed form solution as:

$$\begin{aligned}\mathbf{Y}_i\left(\boldsymbol{\alpha}_0,\omega,t\right)=\mathbf{Y}_{i,1}\left(\boldsymbol{\alpha}_0,\omega,t\right)+\mathbf{Y}_{i,2}\left(\boldsymbol{\alpha}_0,\omega,t\right)=&\left\{\mathbf{Y}_{i,1,\mathrm{p}}\left(\boldsymbol{\alpha}_0,\omega,t\right)+\mathbf{Y}_{i,2,\mathrm{p}}\left(\boldsymbol{\alpha}_0,\omega,t\right)\right.\\ &\left.-\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0\right)t\right]\left[\mathbf{Y}_{i,1,\mathrm{p}}\left(\boldsymbol{\alpha}_0,\omega,0\right)+\mathbf{Y}_{i,2,\mathrm{p}}\left(\boldsymbol{\alpha}_0,\omega,0\right)\right]\right\}\mathcal{U}\left(t\right)\end{aligned} \tag{42}$$

where the particular solution vectors of Eqs.(41), can be evaluated, after some algebra, as follows:

$$\begin{aligned}&\mathbf{Y}_{i,1,\mathrm{p}}\left(\boldsymbol{\alpha}_0,\omega,t\right)=\varepsilon(\omega)\exp\left[-\beta(\omega)\,t\right]\boldsymbol{\Gamma}(\omega)\left[\boldsymbol{\Gamma}(\omega)\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)+\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\boldsymbol{\Gamma}(\omega)+t\,\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)\right]\boldsymbol{\Gamma}(\omega)\mathbf{v}(\boldsymbol{\alpha}_0);\\&\mathbf{Y}_{i,2,\mathrm{p}}(\boldsymbol{\alpha}_0,\omega,t)=\varepsilon(\omega)\mathbf{P}_i(\boldsymbol{\alpha}_0,t)\exp\left[\boldsymbol{\Lambda}\left(\boldsymbol{\alpha}_0\right)t\right]\boldsymbol{\Gamma}^2(\omega)\mathbf{v}(\boldsymbol{\alpha}_0);\end{aligned}\tag{43}$$

where $\mathbf{P}_i(\boldsymbol{\alpha}_0,t)$ is a matrix of order $(2m\times 2m)$ whose elements, $P_{i,jk}(\boldsymbol{\alpha}_0,t)$, are defined as follows:

$$P_{i,jj}(\boldsymbol{\alpha}_0,t)=t\;B_{i,jj}(\boldsymbol{\alpha}_0);\quad P_{i,jk}(\boldsymbol{\alpha}_0,t)=\frac{B_{i,jk}(\boldsymbol{\alpha}_0)}{\lambda_k-\lambda_j},\,j\neq k\tag{44}$$

with $B_{i,jk}(\boldsymbol{\alpha}_0)$ elements of the matrix $\mathbf{B}_i\left(\boldsymbol{\alpha}_0\right)$. Finally, the sensitivity of *PEC* matrix with respect to the parameter α_i, defined in Eq.(36), can be evaluated by substituting Eqs.(39) and (42) into Eq.(37), and then the result (the explicit closed form of the nodal *EPSD* function matrix) into Eq.(36).

4 NUMERICAL APPLICATIONS

In this section, the accuracy of the proposed procedure has been verified, through the comparison of the results of a numerical application with the *Monte Carlo Simulation* (*MCS*) method (1000 samples). The analysed system is composed by two interconnected three-story selected structures, having the same floor elevation, as depicted in Figure 1. The two neighbouring floors are connected by a damper device. Each fluid damper device is modelled as a combination of a linear spring, having stiffness $k_{\mathrm{d},i}=\left(1+\alpha\right)\times 10^5\,\mathrm{N/m}$, and a linear dashpot, having damping coefficient $c_{\mathrm{d},i}=\left(1+\alpha\right)\times 10^6\,\mathrm{N\ s/m}$, with $\alpha>0$ a dimensionless parameter. It follows that the vector $\boldsymbol{\alpha}$ becomes a scalar quantity and the nominal structural matrices are evaluated setting $\alpha=0$.

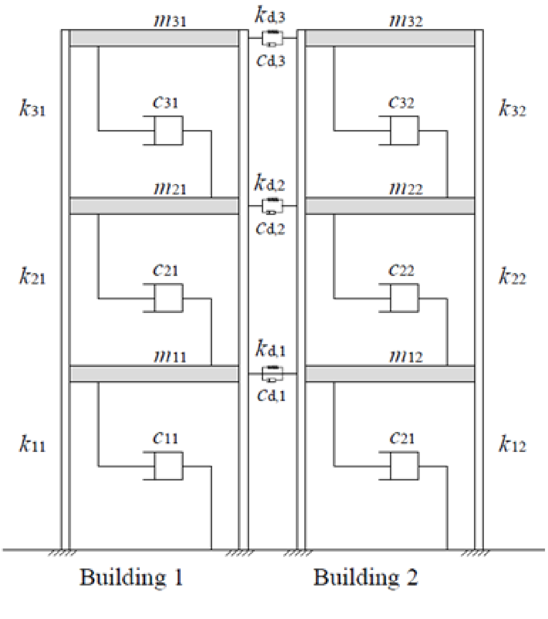


Figure 1: Geometric configuration of the analyzed structure.

The characteristics of each floor (mass m_i, stiffness k_i and damping coefficient c_i) for the two buildings are summarized in Table 1. In Table 2 the modal characteristics of the two unlinked buildings (circular frequency ω_i, period T_i and modal participating mass ratio ε_i), together with the global system are reported.

	Building 1	Building 2
$k_i\left[\mathrm{N/m}\right]$	2×10^{11}	2×10^{9}
$m_i\left[\mathrm{kg}\right]$	1.29×10^{6}	1.29×10^{6}
$c_i\left[\mathrm{Ns/m}\right]$	1×10^{5}	1×10^{5}

Table 1: Characteristics of the analysed buildings.

Building 1			Building 2			Global system		
$\omega_i\left[\mathrm{rad/s}\right]$	$T_i\left[\mathrm{s}\right]$	$\varepsilon_i\left[\%\right]$	$\omega_i\left[\mathrm{rad/s}\right]$	$T_i\left[\mathrm{s}\right]$	$\varepsilon_i\left[\%\right]$	$\omega_i\left[\mathrm{rad/s}\right]$	$T_i\left[\mathrm{s}\right]$	$\varepsilon_i\left[\%\right]$
175.235	0.036	91.408	17.523	0.359	91.408	17.527	0.359	47.704
490.998	0.013	7.488	49.099	0.128	7.488	49.100	0.128	3.744
709.512	0.009	1.104	70.951	0.089	1.104	70.952	0.089	0.552

Table 2: Modal information of the analysed buildings.

The selected structures are subjected to a fully non-stationary seismic input whose *EPSD* function can be expressed as:

$$G_{FF}(\omega,t)=a^2(\omega,t)\,G_0(\omega). \tag{45}$$

In the previous equation the parameters of the modulating function, defined in Eq.(29), have been set as: $\alpha_a\left(\omega\right)=\frac{1}{2}\left(0.15+\frac{\omega^2}{25\pi^2}\right)$, $\varepsilon\left(\omega\right)=\frac{\sqrt{2}}{5\pi}\omega$ and $t_0=0$; the Tajimi-Kanai *PSD* function is used to model the *PSD* function of the stationary counterpart of the input stochastic process:

$$G_0(\omega)=G_\mathrm{W}\,\frac{4\zeta_\mathrm{K}^2\,\omega_\mathrm{K}^2\,\omega^2+\omega_\mathrm{K}^4}{\left(\omega_\mathrm{K}^2-\omega^2\right)^2+4\zeta_\mathrm{K}^2\,\omega_\mathrm{K}^2\,\omega^2} \tag{46}$$

where $G_\mathrm{W}=0.05\ \mathrm{m}^2/\mathrm{s}^3$, $\omega_\mathrm{K}=4\,\pi\,\mathrm{rad/s}$ is the filter frequency that determines the dominant input frequency and $\zeta_\mathrm{K}=0.6$ is the filter damping coefficient that indicates the sharpness of the *PSD* function.

In order to show the accuracy of the proposed method, the *sensitivity* $S_{\lambda_{\ell,u_r}}\left(t\right)$ (indicated in Eq.(47)) with respect to the parameter α of *NGSM*s of the generic *r*-th floor displacement $u_r(t)$ are compared with *MCS* results and are depicted in Figures 2-4.

$$S_{\lambda_{\ell,u_r}}\left(t\right)=\left.\frac{\partial\lambda_{\ell,u_r}\left(\alpha,t\right)}{\partial\alpha}\right|_{\alpha=0} \tag{47}$$

In Eq.(47), $\lambda_{\ell,u_r}\left(\alpha,t\right)$ are the *r*-diagonal elements of the $\mathbf{\Lambda}_{\ell,\mathbf{UU}}(\boldsymbol{\alpha},t)$ matrix defined in Eq.(36) while the subscript ℓ indicates the order of the *NGSM*s.

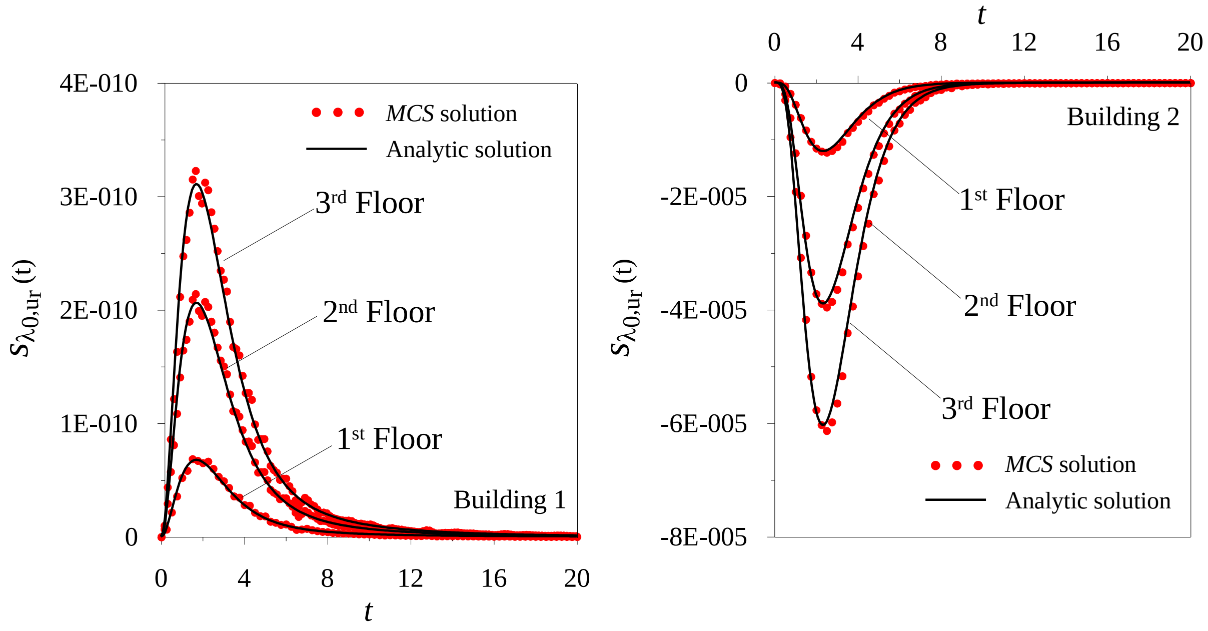


Figure 2: Time histories of the sensitivity of the *NGSM* $\lambda_{0,u_r}(t)$, for the six relative to ground floor displacements of the buildings (black line) and comparison with the *MCS* (red dots).

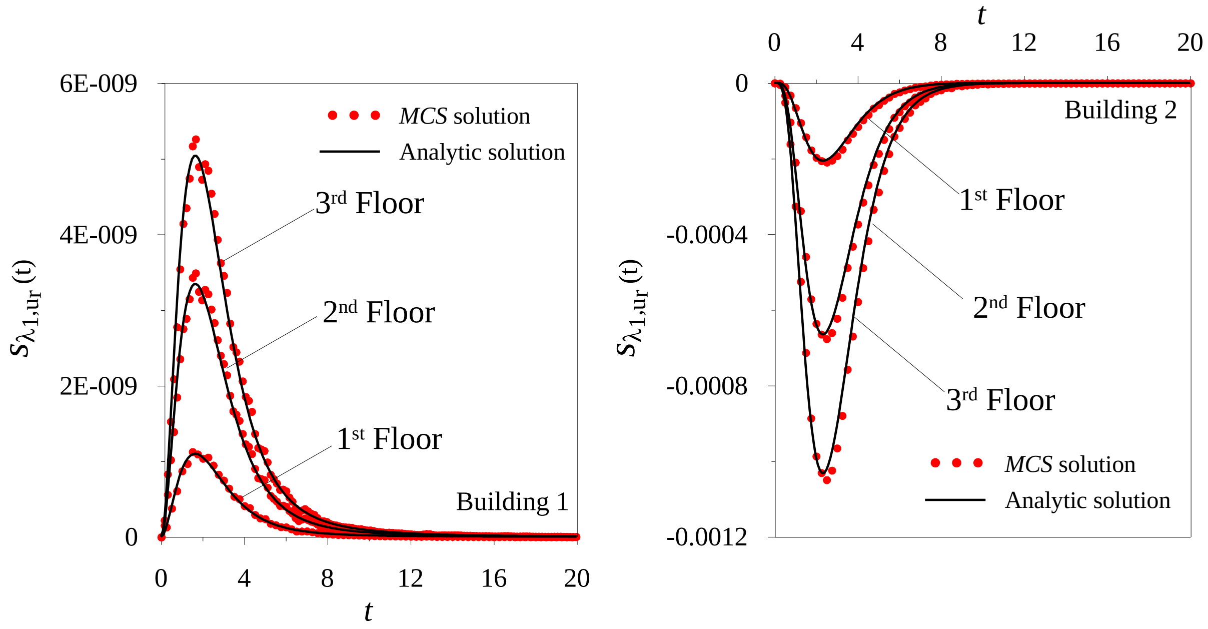


Figure 3: Time histories of the sensitivity of real part of the *NGSM* $\lambda_{1,u_r}(t)$, for the six relative to ground floor displacements of the buildings (black line) and comparison with the *MCS* (red dots).

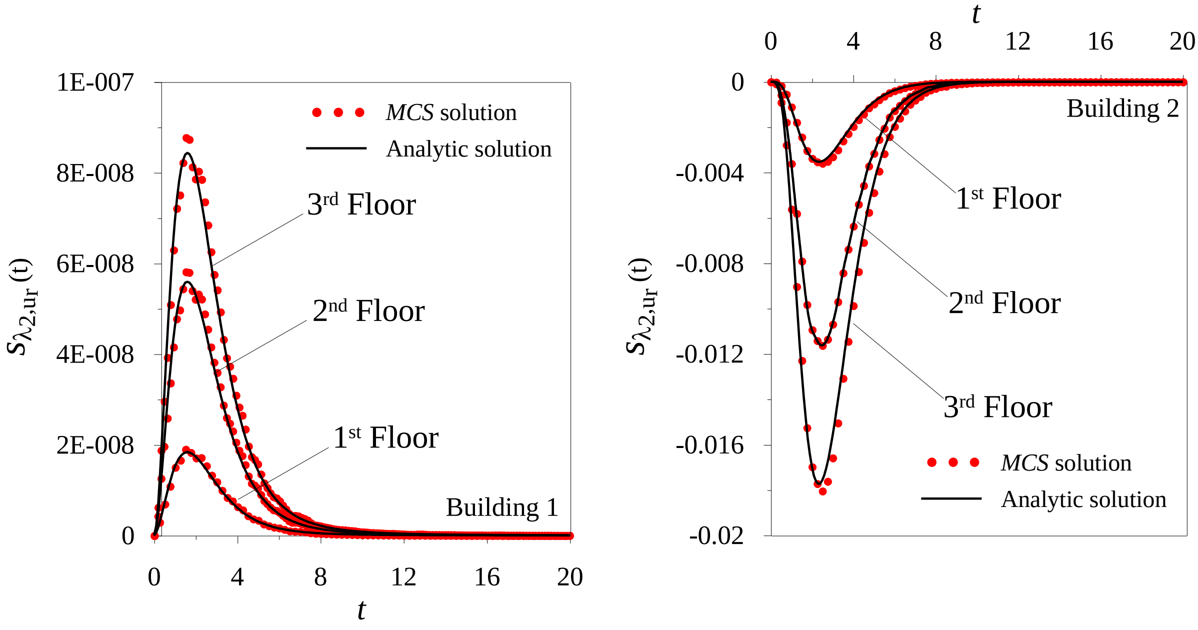


Figure 4: Time histories of the sensitivity of the *NGSM* $\lambda_{2,u_r}(t)$, for the six relative to ground floor displacements of the buildings (black line) and comparison with the *MCS* (red dots).

The figures 2-4 evidence a perfect superposition between the proposed analytical solution and the *MCS* method, demonstrating the accuracy of the proposed procedure. Obviously a positive sensitivity indicates an increment of the corresponding *NGSM*, when the parameter α changes, while a negative sensitivity means that the *NGSM* decreases when the parameter changes.

Finally, in Figure 5 the *sensitivity* of the first *NGSM* of the response of third floors, for both buildings, and for five different ratio of the stiffness, have been depicted. In the first three cases the stiffness of building 1 is assumed: $k_i = 2\times10^{11}\,\mathrm{N/m}$. In the latest two cases the stiffness of building 2 is: $k_i = 2\times10^{11}\,\mathrm{N/m}$. These figures show that the sensitivity of the first *NGSM* is positive for the more rigid building, while it is negative for the more deformable building. Namely, for the presence of devices, changing the parameter α the response of more rigid structures increases, while the response of lighter structures decreases. In the third case, when the two structures have the same stiffness, the sensitivity for building 1 is negative for the former time instants and positive for the following ones. The opposite result is obtained for building 2. This means that in the third case the sign of the sensitivity changes in the time. Zero sensitivity means that a change of parameter α does not modify the response of two buildings with respect to the nominal case.

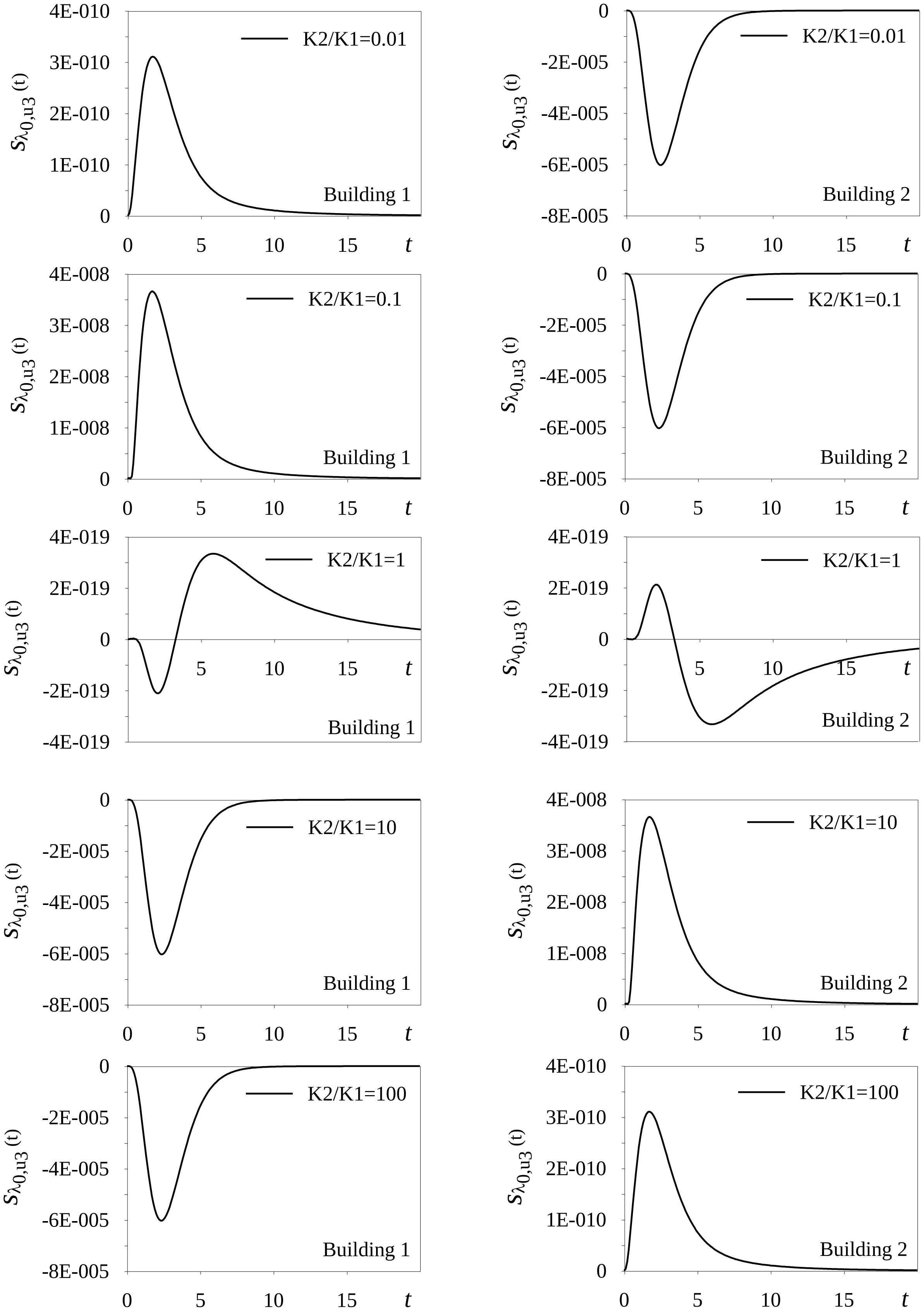


Figure 5: Time histories of the sensitivity of the *NGSM* $\lambda_{0,u_3}(t)$, for the third relative to ground floor displacements of the buildings for different ratio of the stiffness.

5 CONCLUSIONS

In the framework of optimization procedures, especially during the design of vibration control devices, the sensitivity analysis is a very powerful tool to evaluate how the structural response is modified with reference structural parameters changes.

In this paper a novel method for the evaluation of the sensitivities of *non-geometric spectral moments* of the structural response of linear classically or non-classically damped linear structural systems subjected to both separable and non-separable non-stationary excitations is proposed.

The proposed procedure is based on two fundamental steps: first, it is necessary to determine sensitivities of *evolutionary frequency response functions*, and it is possible thanks to the herein obtained explicit closed form solutions; then, by simple frequency domain integrals, it is possible to evaluate the sensitivity of the structural response statistics.

The presented method has a unified approach for both classically and non-classically damped discrete linear structural systems, thanks to use of the state-variables.

The numerical application on a plane-frame demonstrated the effectiveness of the proposed method, since a validation with *MCS* method has been done.

REFERENCES

[1] P.M. Frank, *Introduction to System Sensitivity Theory*. Academic Press, NY. 1978.

[2] E.J. Haug, V. Komkov, K.K. Choi, *Design sensitivity analysis of structural system*, Academic Press; Orlando. 1985.

[3] T.D. Hien, M. Kleiber, Stochastic design sensitivity in structural dynamics, *International Journal for Numerical Methods in Engineering*, **32**, 1247-1265, 1991.

[4] P.C. Jennings, G.W. Housner, C. Tsai, Simulated earthquake motions for design purpose, *Proceeding 4th International Conference on Earthquake Engineering*, Santiago; A-**1**, 145-160, 1969.

[5] T.I. Hsu, M.C. Bernard, A random process for earthquake simulation, *Earthquake Engineering and Structural Dynamics*, **6**, 347–362, 1978.

[6] P. Spanos, G.P. Solomos. Markov approximation to transient vibration, *Journal of Engineering Mechanics (ASCE)*, **109**, 1134-1150, 1983.

[7] J.P. Conte, B-.F. Peng, Fully nonstationary analytical earthquake ground-motion model, *Journal of Engineering Mechanics (ASCE)*,;**123,** 15–24, 1997.

[8] J. Szopa, The stochastic sensitivity of the Van der Pol equation, *Journal of Sound and Vibration*, **100**, 135-140, 1985.

[9] S. Benfratello, S. Caddemi, G. Muscolino, Gaussian and non-Gaussian stochastic sensitivity analysis of discrete structural system, *Computer and Structures*, **78**, 425-434, 2000.

[10] C. Proppe, H.J. Pradlwarter, G.I. Schueller, Equivalent linearization and Monte Carlo simulation in stochastic dynamics, *Probabilistic Engineering Mechanics*, **18**, 1-15, 2003.

[11] A. Chaudhuri, S.Chakraborty, Sensitivity evaluation in seismic reliability analysis of structures, *Computer Methods in Applied Mechanics and Engineering*, **193**, 59-68, 2004.

[12] P. Cacciola, P. Colajanni, G. Muscolino, A modal approach for the evaluation of the response sensitivity of structural systems subjected to non-stationary random processes, *Computer Methods in Applied Mechanics and Engineering,* **194,** 4344–4361, 2005.

[13] J. Li, J.B. Chen, *Stochastic Dynamics of Structures*, Singapore: John Wiley & Sons; 2009.

[14] M. Di Paola, Transient Spectral Moments of Linear Systems, *SM Archives,* **10** 225-243, 1985.

[15] G. Muscolino, Nonstationary Envelope in Random Vibration Theory, *Journal of Engineering Mechanics* (*ASCE*), **114**, 1396-1413, 1988.

[16] G. Michaelov, S. Sarkani, L.D. Lutes, Spectral Characteristics of Nonstationary Random Processes – A Critical Review, *Structural Safety,* **21**, 223-244,1999.

[17] G. Michaelov, S. Sarkani, L.D. Lutes, Spectral Characteristics of Nonstationary Random Processes – Response of a simple oscillator, *Structural Safety,* **21**, 245-244, 1999.

[18] M. Di Paola, G. Petrucci, Spectral Moments and Pre-Envelope Covariances of Nonseparable Processes, *Journal of Applied Mechanics (ASME),* **57**, 218-224, 1990.

[19] G. Muscolino, Nonstationary Pre-Envelope Covariances of Nonclassically Damped Systems, *Journal of Sound and Vibration,* **149**, 107-123, 1991.

[20] G. Muscolino, T. Alderucci, Closed-form solutions for the evolutionary frequency response function of linear systems subjected to separable or non-separable non-stationary stochastic excitations, *Probabilistic Engineering Mechanics,* **40**, 75–89, 2015.

[21] T. Alderucci, G. Muscolino, Time–frequency varying response functions of non-classically damped linear structures under fully non-stationary stochastic excitations, *Probabilistic Engineering Mechanics,* **54**, 95–109, 2018.

[22] G. Borino, G. Muscolino, Mode-superposition methods in dynamic analysis of classically and non-classically damped linear systems, *Earthquake Engineering and Structural Dynamics,* **14**, 705-717, 1986.

[23] G. Muscolino, Dynamically Modified Linear Structures: Deterministic and Stochastic Response, *Journal of Engineering Mechanics (ASCE),* **122**, 1044-1051, 1996.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

TRACKING THE MODAL PARAMETERS OF BAIXO SABOR CONCRETE ARCH DAM WITH UNCERTAINTY QUANTIFICATION

Sérgio Pereira[1], Edwin Reynders[2], Filipe Magalhães[1], Álvaro Cunha[1] and Jorge Gomes[3]

[1] Construct-ViBest, Faculty of Engineering (FEUP), University of Porto
Rua Dr.Roberto Frias, 4200-465 Porto, Portugal
{pereira.sergio, filipema, acunha}in@fe.up.pt

[2] University of Leuven (KU Leuven), Department of Civil Engineering
Kasteelpark Arenberg 40, B-3001 Leuven, Belgium
edwin.reynders@kuleuven.be

[3] National Laboratory for Civil Engineering (LNEC)
Av. do Brasil 101, 1700-066 Lisboa, Portugal
jgomes@lnec.pt

Abstract

During the last decade, many vibration-based structural health monitoring systems have been successfully implemented in different structures such as bridges, towers, stadia roofs and wind turbines, with the aim of studying the structures dynamics and its evolution over time and eventually detecting the occurrence of novel structural behaviour that may indicate the presence of damage.

Such vibration-based monitoring systems generally rely on the identification of modal properties, which are then used as monitoring features. Therefore, from operational modal analysis to the tracking of those features, many processing steps occur that depend on the accuracy of the identified modal properties in order to produce good results. Thus, the calculation of the uncertainties associated with the identified modal properties increase the robustness of this process.

In this context, data obtained from the continuous dynamic monitoring of a concrete arch dam has been used to test the effect of taking the uncertainties of identified modal properties into consideration when performing operational modal analysis and modal tracking.

Keywords: Uncertainties in Modal Properties, Operational Modal Analysis, Continuous Dynamic Monitoring, Concrete Arch Dam.

1 INTRODUCTION

Integrated monitoring systems considering real time data directly obtained from structures are very important to the long-term management of large civil infrastructures, such as dams. Though health monitoring systems are historically associated with static data, vibration-based systems have already been successfully implemented in different structures such as bridges [1], wind turbines [2], stadia roofs [3] or bell-towers [4].

Such vibration-based health monitoring systems rely on operational modal analysis to continuously identify the structure's modal properties, which can be used as monitoring features to evaluate the structures health condition evolution over time. From operational modal analysis to the tracking of these features, many processing steps occur that depend on the accuracy of the identified modal properties in order to produce good results. Therefore, the calculation of the uncertainties associated with the identified modal properties may act as an important tool in this process, helping to quantify confidence levels and to eliminate misidentifications of modal properties, thus creating more robust and reliable monitoring databases.

In order to test the influence of the consideration of the uncertainties associated with modal properties in automatic operational modal analysis, the data obtained during one year of monitoring of Baixo Sabor arch dam is used.

After a brief description of the dynamic monitoring system installed in Baixo Sabor arch dam, the results obtained between 01/12/2016 and 01/12/2017 and the methodology used to achieve such results are presented. This paper refers as well the method used to quantify the uncertainty associated with the obtained modal properties and it presents the effect of the consideration of such uncertainties in the tracking of the dam's first four vibration modes.

2 CONTINUOUS DYNAMIC MONITORING OF BAIXO SABOR ARCH DAM

2.1 Instrumented dam and monitoring system

The Baixo Sabor hydroelectric development is located in Sabor river, a tributary of Douro river in the northeast of Portugal, and has been operating since 2016. This concrete double-curvature arch dam is 123 meters high and its crest is 505 meters long. The arch is composed by 32 concrete blocks, separated by vertical contraction joints, and includes six horizontal visit galleries. The left part of Figure 1 shows an aerial picture of the dam and the reservoir, dated May 2016, after the monitoring had started.

A vibration-based health monitoring system was installed in the dam in December 2015, right after it started operating, in order to identify the dam's dynamic characteristics and their evolution over time, taking into account the variation of ambient and operational conditions, as well as the possible evolution of the materials mechanical properties.

The continuous dynamic monitoring system consists of 20 uniaxial accelerometers that have been radially disposed in the dam's three upper visit galleries, whose synchronization is achieved using GPS antennas. The right part of Figure 1 shows the position of the accelerometers installed in the dam, marked with red dots in a picture of the structure. The dynamic monitoring system is configured to continuously record acceleration time series with a sampling rate of 50 Hz and a duration of 30 minutes at all instrumented points, thus producing 48 groups of time series per day [5].

The continuously collected data is processed with a monitoring software developed at ViBest/FEUP called DynaMo [6]. Besides backing up the original data samples, this monitoring software performs the pre-processing of the acceleration time series, through trend elimination, filtering and re-sampling, it characterizes vibration levels and it performs the identification of the dam modal properties through automatic operational modal analysis. The continuous and automatic identification of modal parameters is achieved by combining the Covariance Driven Stochastic Subspace Identification method (SSI-Cov) with a routine based on cluster analysis that automatizes its application. A brief description of this approach is presented in the next section.

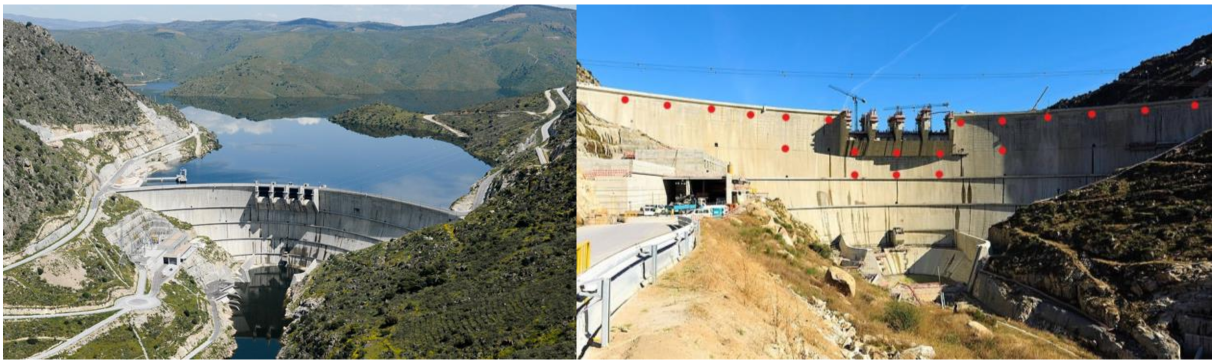


Figure 1 – Baixo Sabor arch dam: a) aerial view (on the left) [7]; b) position of accelerometers marked with red dots (on the right)

2.2 Automated operational modal analysis and modal tracking

In the context of continuous dynamic monitoring, it is crucial to automate modal analysis, in order to process the enormous amount of produced data more easily and to obtain results in real time. As it was mentioned before, in this application the automation of operational modal analysis is achieved through the combination of the SSI-Cov method with a routine based on cluster analysis.

After the application of the SSI-Cov method to each time series of accelerations and after the construction of stabilization diagrams, the methodology based on a hierarchical clustering algorithm proposed in [8] is used to group poles with similar modal properties, thus producing groups with high internal (within-cluster) homogeneity and high external (between-cluster) heterogeneity. Similarity between poles is measured using a metric that depends on the relative differences between natural frequencies and consistency of mode shapes.

A predefined number of different clusters are obtained from the application of cluster analyses, corresponding to both physical and numerical modes. The final modal estimates associated with the dataset under analysis are defined as the mean values of the poles included in each cluster. To separate physical modes from numerical ones, the modal estimates identified in each setup are compared with a set of reference values, which were obtained from selected datasets with very clear stabilization diagrams. Each new set of modal properties is only accepted as a physical mode if the MAC (Modal Assurance Criterion) [9] between the estimated mode shape and the reference mode shape is higher than 0.55 and the variation between their frequency values is lower than 4.5%.

This methodology was applied to one year of data that was continuously collected by the Baixo Sabor arch dynamic monitoring system between 01/12/2016 and 01/12/2017 and the structures first four vibration modes were tracked. In this sense, the evolution of the natural frequencies of the dam’s first four modes was represented over time (Figure 2), being each mode represented by a different color, whereas Figure 3 presents the evolution of the four modes damping ratios during the same period, using the same color system.

Since 48 daily datasets were considered during 366 days, resulting in a total 17568 modal estimates considered per mode, the tracked modal properties show daily and seasonal variability. On the one hand, the seasonal variation provoked by operational and environmental conditions (essentially water level and structure temperature) is clear on natural frequencies, and on the other, a higher daily scatter is observed with the damping estimates.

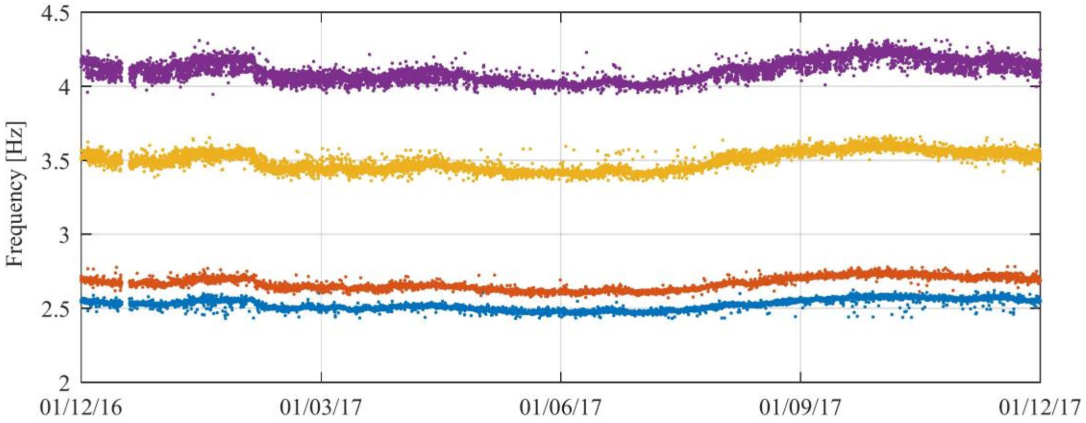


Figure 2 – Evolution of the first four modes natural frequencies between 01/12/2016 and 01/12/2017

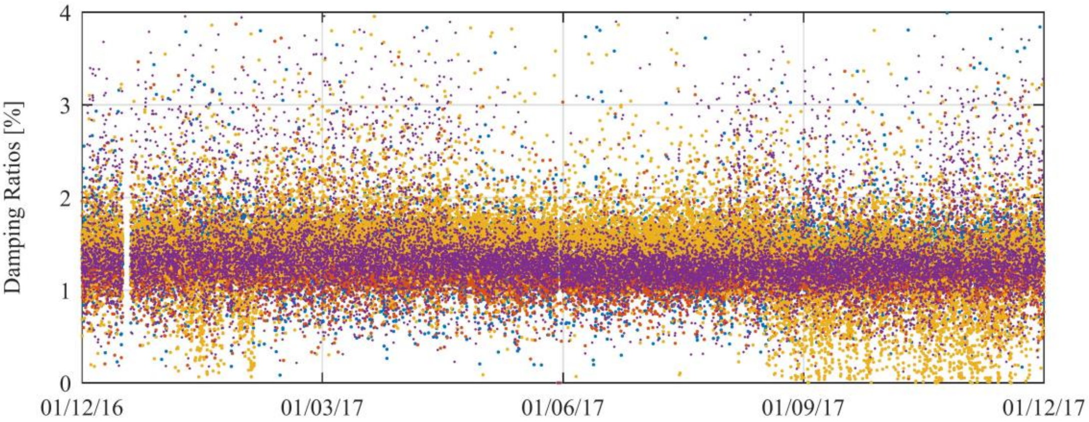


Figure 3 – Evolution of the first four modes damping ratios between 01/12/2016 and 01/12/2017

3 UNCERTAINTY QUANTIFICATION OF MODAL PARAMENTERS ESTIMATES

3.1 Introduction

Modal parameters of a structure, estimated from ambient vibration measurements using state of art identification methods, are always subject to bias and variance errors. Since identified modal characteristics are quite often used for calibration and validation of dynamic structural models, for structural control or for structural health monitoring it becomes important to analyze the accuracy associated with estimated parameters. Errors introduced in the identification process may be due to several reasons [10]. For instance: the use of a finite number of data samples; the inputs may not be a white noise; nonlinear distortions may be present in the data because of material or geometrical nonlinearities; non-stationary nature of structures, activated by external factors as temperature or wind; analog and/or digital filters introducing spurious poles; human induced errors.

Depending both on the identification method used and on the structure characteristics, each of the previous error causes may gain relevance over the others, and even different causes that were not mentioned may arise. Both the physical part of the problem, more strongly associated with the data collection, and the processing part, strictly related with the identification procedure, must be conducted with extreme care, in order to reduce the number of error sources and to mitigate the effects of those that cannot be completely suppressed.

To mitigate bias errors, stabilization diagrams are of good use. However, unlike bias errors, which can be mitigated, variance errors can only be estimated, they cannot be not removed. In [10] a detailed sensitivity analysis of the reference-based covariance-driven stochastic subspace identification method (SSI-Cov) yielded a novel expression for the covariance of the system matrices that are identified using this method.. Additionally, subsequent sensitivity analysis yielded expressions for the covariances of the modal parameters estimates, allowing the estimation of uncertainty bounds. A computationally faster version of this algorithm [11] was later developed along with a multi-order extended implementation that allows the calculation of uncertainty bounds for all elements of a stabilization diagram. This second version was integrated on the available routines for automated operational modal analysis to obtain standard deviation values associated with all the modal estimates. This method was validated with two application examples that were presented in [12].

3.2 Application example

Using the methodology of section 2 for automatic operational modal analysis, a group of clusters is obtained after the application of SSI-Cov to each 30 minute sets of recorded accelerations. Moreover, each cluster is composed by a group of poles that resulted from the multitude of orders considered during the application of SSI-Cov. Besides the three regular quantities obtained from the common application of SSI-Cov, two more are obtained with this new version, thus five different quantities are associated with each pole:

- Natural frequency (f [Hz]);
- Damping ratio (d [%]);
- Mode Shape;
- Standard deviation of natural frequency (f_{std} [Hz]);
- Standard deviation of damping ratio (d_{std} [%]).

With this implementation, it would be possible to obtain as well the standard deviations associated with each modal ordinate, which would indicate the uncertainty related to the final mode shape. However, this was not considered in this work. Additionally, relative values of standard deviations may be computed, dividing each standard deviation value by its natural frequency ($f_{std\text{-}relative}$) or damping ratio ($d_{std\text{-}relative}$). In the end, the estimates of modal properties and respective standard deviation values are calculated as the mean value of all the poles integrating the cluster, eventually disregarding estimates with high uncertainty.

An example using real data from the dynamic monitoring system of Baixo Sabor is presented in **Table 1**, comprehending five poles belonging to the same cluster, thus resulting from the same set of 30 minutes acceleration time series. Most of the five poles present natural frequencies close to 3.52 Hz, and damping ratios around 1.5 %. It is worth noticing that the frequency value of pole 3, the only one in the group closer to 3.51 than to 3.52 Hz, shows the highest standard deviation, indicating that higher uncertainty is associated with this pole. The same is observed with the standard deviation of its damping ratio, which is so high that it leads to a relative standard deviation of 101.7 %. In this sense, the estimates of modal properties for this cluster were calculated considering both the mean of the five poles and the mean of four poles, excluding pole 3. Even if small differences were obtained, it is worth pointing out that the uncertainty associated with the modal properties was minimized in the second case, which gives the analyst more confidence in the results obtained and prospects better chances of successfully building damage detection models in the future.

Pole	f [Hz]	f_{std} [Hz]	$f_{std\text{-}relative}$ [%]	d [%]	d_{std} [%]	$d_{std\text{-}relative}$ [%]
1	3.522	0.0041	0.117	1.488	0.110	7.4
2	3.521	0.0053	0.152	1.503	0.229	15.2
3	3.513	0.0419	1.194	1.540	1.566	101.7
4	3.524	0.0086	0.245	1.624	0.335	20.6
5	3.520	0.0113	0.321	1.585	0.500	31.6
Mean $_{all}$	3.520	0.0143	0.406	1.548	0.548	35.3
Mean $_{[1,2,4,5]}$	3.522	0.0073	0.209	1.550	0.294	18.7

Table 1 – Pole quantities example

4 EFFECT OF UNCERTAINTIES ON MODAL TRACKING

To test the effect of considering uncertainties in the algorithm used for modal tracking, the results presented in Figure 2 and Figure 3 will be used as baseline for comparison. These results were obtained applying the methodology presented in section 2, in which the uncertainty of modal estimates was quantified but was not included in any part of the tracking algorithm. For reference in this work this processing will be named Processing A.

In this sense, each mode natural frequencies and damping ratios obtained with Processing A were represented independently in Figure 4 and Figure 5, to provide a closer evaluation of their evolution over time. However, in this case, the color of each modal estimate was represented as a linear function of its standard deviation. Therefore, natural frequencies were represented in blue if their standard deviation values were close to 0 Hz, and they were represented in yellow if they were close to 0.05 Hz, or higher than this value. For the modal damping ratios estimates, relative standard deviations were used to choose the color of each estimate. Thus, damping ratios were represented in blue if their relative standard deviation values were close to 0 %, and they were represented in yellow if they were close to 50 %, or higher than this value.

The value of 0.05 Hz was picked taking into account the authors’ experience on modal analysis, and it represents a substantial uncertainty that for the structure under consideration should not be accepted. Furthermore, the 50 % limit for t-he relative standard deviation results from the consideration that values higher than 50 % would mean negative values could be admitted for damping ratios (assuming a normal distribution and a 95% confidence interval), which is not physically possible.

Figure 4 shows that the frequency estimates of the first two modes are generally associated with lower standard deviation values than the third and fourth modes, which present a higher scatter during the period under analysis. However, all the four modes show many yellow estimates that clearly diverge from the main tracking line. Additionally, in the figure that corresponds to the evolution of mode 3, a thin blue horizontal alignment seems to be defined between 3.55 and 3.60 Hz, indicating a specific frequency that always presents very low standard deviations.

The damping ratios of all four modes present as well many estimates represented in yellow, indicating a considerable number of estimates associated with relative standard deviations higher than 50 %. Furthermore, besides the normal variability around the mean, mode 3 presents a significant number of estimates with low damping, between 0 and 1 %, and mode 4 presents a significant number of estimates with high damping, between 2 and 4 %. The majority of the estimates with seemingly abnormal damping values present high relative standard deviations, as indicated by its yellow color.

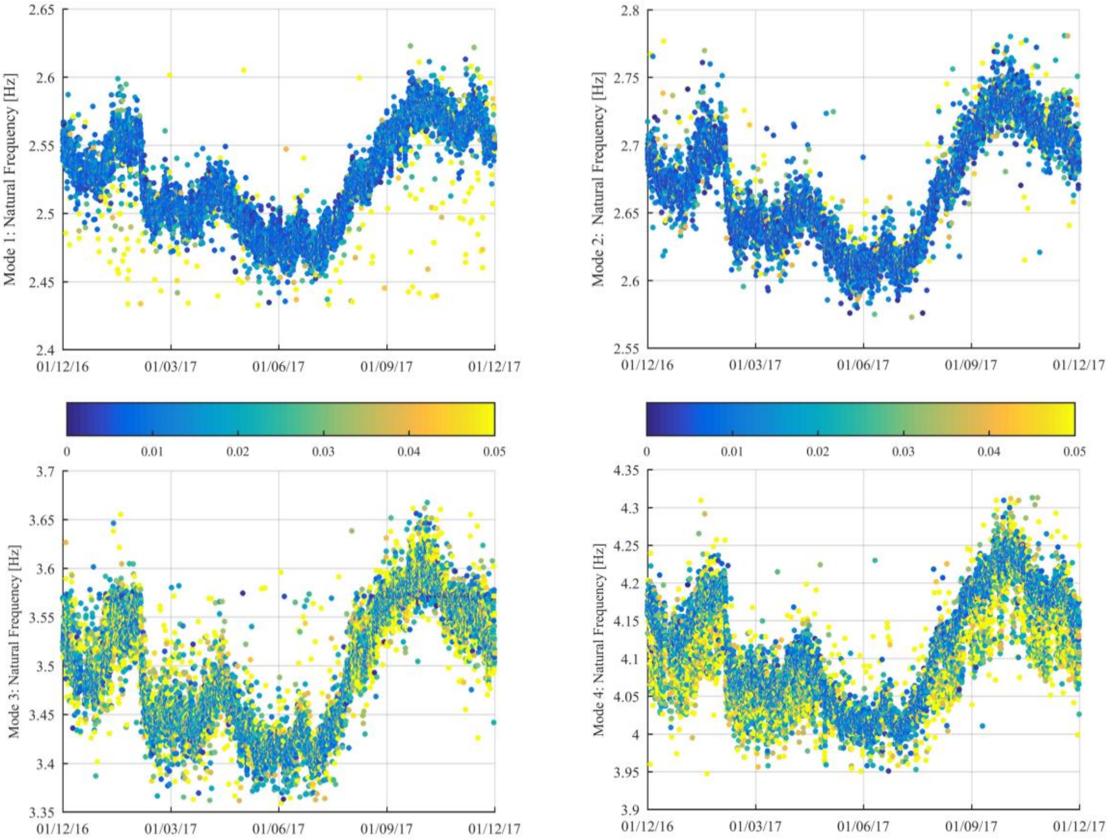


Figure 4 – Natural frequencies with color as function of standard deviation (Processing A)

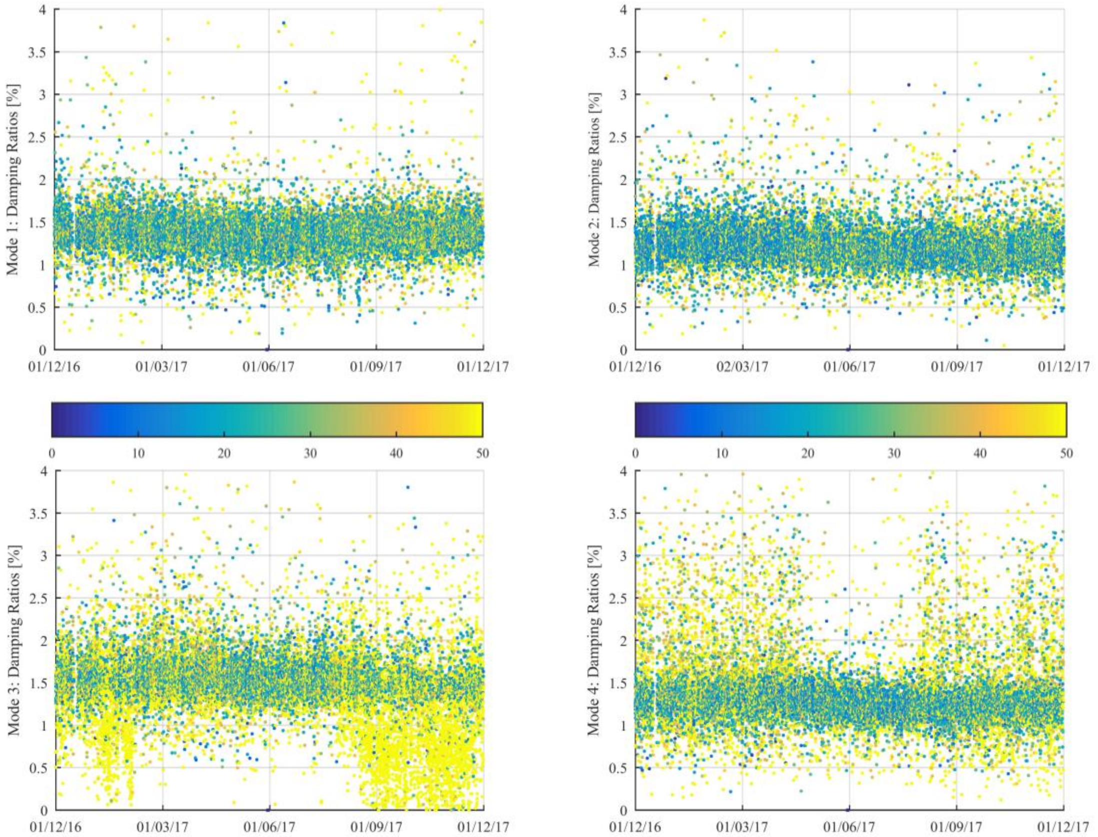


Figure 5 – Damping ratios with color as function of relative standard deviation (Processing A)

After the analysis of the results provided by processing A, a new tracking strategy was put through, yet this time, before the comparison between each new set of modal properties estimates and the previously defined references, all the clusters obtained from the application of SSI-Cov to each 30 minutes time series of accelerations were analysed considering the uncertainties of each pole estimate. This analysis, which will be referenced as Processing B, consisted on the evaluation of standard deviations and, consequently, all the poles whose frequency standard deviation was equal or higher than 0.05 Hz and all the poles whose relative damping standard deviation was equal or higher than 50 % were eliminated. In conclusion, most clusters became more homogeneous, and many clusters that were composed mainly by poles associated with high standard deviations disappeared. However, the number of setups for which one or mode modes were not tracked increased, diminishing the number of successful identifications.

In the perspective of the first two modes, this strategy turned out to be profitable, and the good results obtained will be presented hereafter. However, in a first stage, in the case of the third mode, it had negative consequences, and in the case of the fourth mode it was not very efficient. Figure 6 presents the distribution of frequencies and damping ratios identified for the third mode with Processing B. On the one hand there is an abnormal number of identifications between 3.55 and 3.60 Hz, and on the other there is an abnormal number of identifications with damping close to 0 %. This indicates that the elimination of poles with higher uncertainty was favorable to a systematic identification of the turbine rotation frequency (3.57 Hz) as the frequency of mode 3, increasing the number of misidentifications.

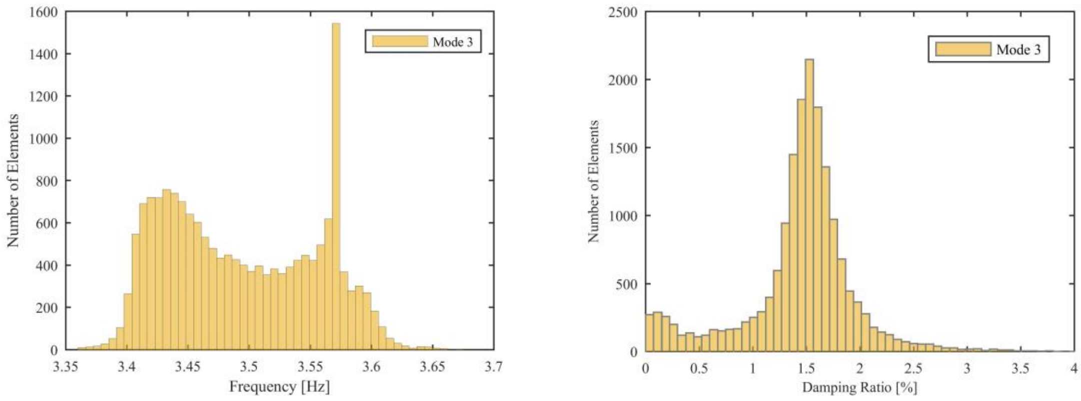


Figure 6 – Histograms of frequency and damping of mode 3 (Processing B)

In order to minimize the number of times the turbine rotation frequency is identified as mode 3 natural frequency, the characteristics of the estimates associated with this parasite frequency were studied. Thus, besides presenting modal damping ratios close to 0, these estimates systematically present very low standard deviation values for both frequency and damping. In this sense, in the left part of Figure 7, all the frequency estimates identified for mode 3 which presented frequency standard deviations lower than 0.005 Hz were represented in yellow, while all the other frequency estimates were represented in black. The vast majority of yellow points correspond to frequencies very close to 3.57 Hz, therefore associated with the turbine frequency.

In the case of the fourth mode, Processing B did not introduce any kind of bias, on the contrary, though many misidentifications were eliminated, it is still possible to improve the modal tracking of this mode. In this sense, frequency estimates with relative standard deviations higher than 0.3 % were represented in yellow, in the right side of Figure 7, while all the other frequency estimates were represented in black. This leads to a figure with a high number of yellow points dispersed all over the frequency range, but mostly associated with estimates located far from the main frequency track.

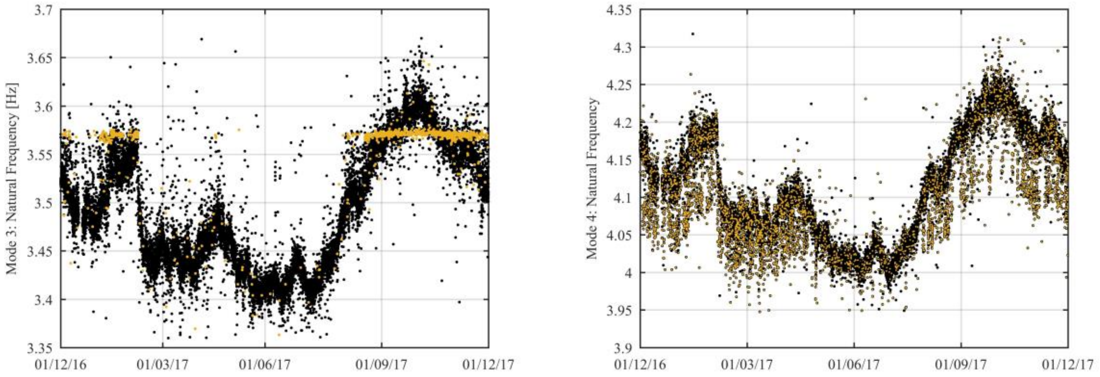


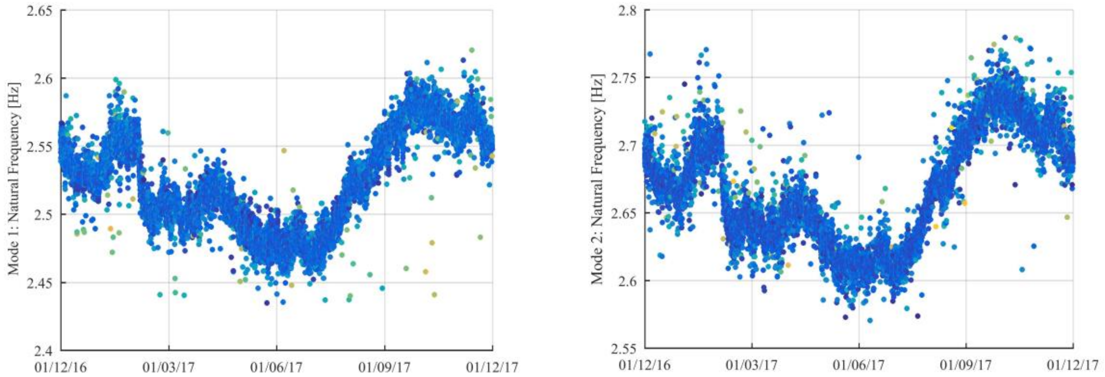
Figure 7 – Misidentifications of the third and fourth modes after Processing B

Taking into account previous considerations, Processing C was put through, combining the conditions used in Processing B with the elimination of poles with frequency standard deviations lower than 0.005 Hz, in the range between 3.566 Hz and 3.576 Hz. Additionally, poles with relative frequency standard deviations higher than 0.3 % in the range between 3.90 and 4.35 Hz were also eliminated.

This time, good results were achieved for the four modes. The color of natural frequency (Figure 8) and damping (Figure 9) estimates was once again represented as function of their respective standard deviation values, using the same limits used in Figure 4 and Figure 5.

In the case of natural frequency, there are still some outliers in the four modes, but its number reduced considerably. Moreover, the four figures generally present darker colors, indicating a significant reduction in the value of standard deviations, thus increasing the confidence level of individual estimates. Though mode 3 still presents a few estimates with high uncertainty, the thin blue horizontal alignment associated with the turbine frequency is not distinguishable anymore.

The general level of accuracy of damping ratios increased as well, especially with the fourth mode, which presented many estimates with high damping values and high relative damping standard deviations that were eliminated. The third mode is again the one which presents the higher number of estimates with high uncertainty, though many have already been eliminated. Most of these estimates present damping values between 0.5 and 1 %, indicating that in some cases the SSI-Cov algorithm was not capable to separate the third mode estimate from the turbine harmonic.



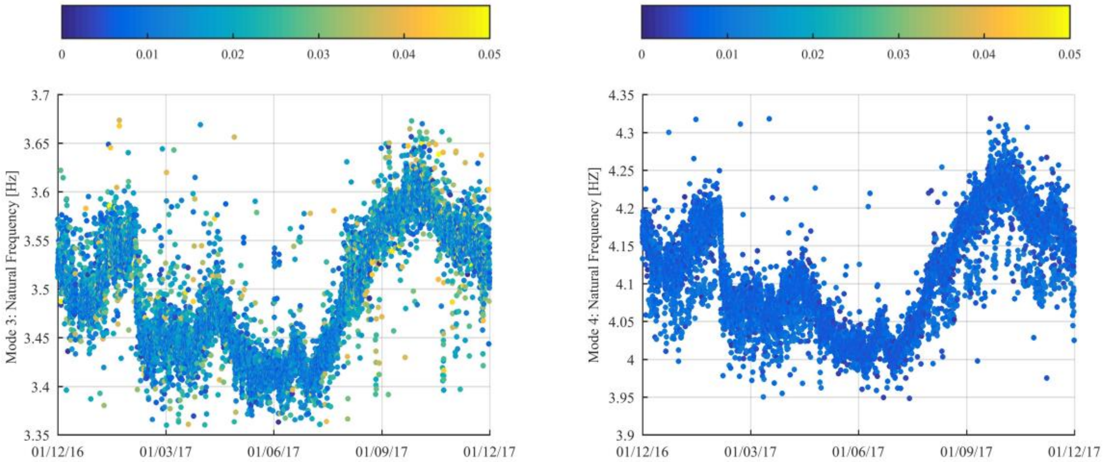


Figure 8 – Natural frequencies with color as function of standard deviation (Processing C)

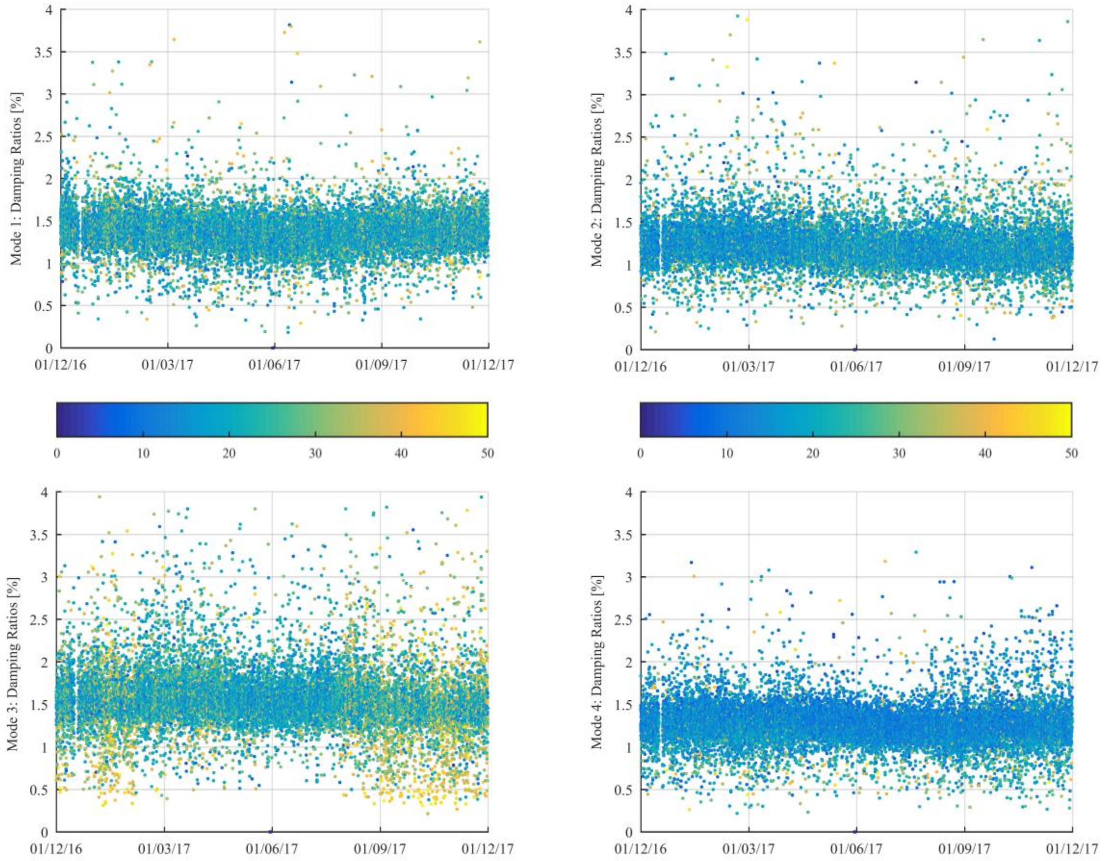


Figure 9 – Damping ratios with color as function of relative standard deviation (Processing C)

Finally, the results obtained with Processings A and C, the first and last ones, were summarized in Table 2 for comparison. For each vibration mode, means and standard deviations of the estimates obtained for the entire analyzed period were calculated for four parameters: natural frequencies (mean (f) and std (f)), damping ratios (mean (d) and std (d)), frequency standard deviations (mean (f_{std}) and std (f_{std})) and relative damping standard deviations (mean ($d_{std\text{-}relative}$) and std ($d_{std\text{-}relative}$)).

In the case of natural frequencies, the mean and standard deviation for the whole studied period did not present considerable changes from Processing A to C, which was expected, since the effect of operational and environmental conditions on natural frequencies is predominant when compared to the effect of random errors. The elimination of estimates associated with the turbine frequency, however, led to a decrease of the third mode natural frequency mean. On the other hand, the mean and standard deviation of frequency standard deviations decreased substantially for the four modes, indicating a more accurate set of estimates.

Considerable variations were observed in the means of damping ratios of third and fourth modes. The third mode damping ratio mean for the studied period increased in agreement with the elimination of turbine rotation generated poles, and the fourth mode damping ratio mean decreased in agreement with the elimination of outliers with high uncertainty, which presented high damping values as well. Estimates that are more accurate were achieved also in the case of damping ratios, as shown by the clear reduction of the relative standard deviation values of damping.

Processing	mean (f)	std (f)	mean (d)	std (d)	mean (f_{std})	std (f_{std})	mean ($d_{std-relative}$)	std ($d_{std-relative}$)
Mode 1 – A	2.516	0.143	1.38	0.379	0.012	0.014	35.4	52.7
Mode 1 – C	2.516	0.144	1.37	0.301	0.008	0.004	20.9	7.5
Mode 2 – A	2.660	0.146	1.22	0.284	0.011	0.012	34.5	44.0
Mode 2 – C	2.659	0.146	1.23	0.276	0.006	0.004	18.1	7.0
Mode 3 – A	3.484	0.194	1.45	0.485	0.027	0.026	82.0	107.5
Mode 3 – C	3.479	0.199	1.57	0.412	0.014	0.007	23.3	8.8
Mode 4 – A	4.095	0.227	1.39	0.472	0.025	0.028	45.9	69.5
Mode 4 – C	4.100	0.230	1.26	0.260	0.007	0.002	15.0	6.0

Table 2 – Results comparison between Processing A and C

5 CONCLUSIONS

Automated operational modal analysis was performed to one year of data recorded by the continuous dynamic monitoring of Baixo Sabor arch dam, which was used to track the modal properties of its first four vibration modes. The tracking was based on the application of SSI-Cov method, an algorithm using cluster analysis and on the comparison between modal estimates and reference values.

A version of SSI-Cov was used that allows the quantification of the uncertainty associated with the modal parameters estimates. It was verified that several frequency and damping estimates presented high standard deviation values, which should not be accepted, and the majority of them correspond to outliers. In this context, the effect of the consideration of uncertainties within the tracking algorithm was tested, in order to minimize the number of outliers and increase the confidence level of modal estimates. The intended purpose was accomplished and good results were achieved with this processing, that is, the standard deviation of both natural frequencies and damping ratios estimates was considerably decreased for the four studied modes, and many outliers corresponding to misidentifications were eliminated, especially with the fourth mode.

Additionally, the consideration of the uncertainty of modal estimates in the tracking algorithm revealed to be adequate to remove as well the presence of parasite frequencies from tracked data, namely the influence of the rotation frequency of the turbines in the hydroelectric development, which was achieved through the elimination of poles with very low standard deviations.

In short, the quantification of uncertainties demonstrated to be a useful tool for the accurate tracking of modal parameters estimates. In the future, the authors expect to test the influence of the inclusion of said uncertainties in methods related to mitigation of environmental and operational effects and damage detection.

ACKNOWLEGMENTS

This work was financially supported by: UID/ECI/04708/2019- CONSTRUCT - Instituto de I&D em Estruturas e Construções funded by national funds through the FCT/MCTES (PIDDAC); PTDC/ECM-EST/0805/2014|16761 – DAM_AGE - Advanced Online Dynamic Structural Health Monitoring of Concrete Dams, funded by FEDER funds through COMPETE2020 - Programa Operacional Competitividade e Internacionalização (POCI) – and by national funds through FCT - Fundação para a Ciência e a Tecnologia; FCT PhD Scholarship SFRH/BD/100587/2014 provided to the first author. The authors would like also to acknowledge all the collaboration and support provided by EDP Produção.

REFERENCES

1. Magalhães, F., A. Cunha, and E. Caetano, *Vibration based structural health monitoring of an arch bridge: From automated OMA to damage detection.* Mechanical Systems and Signal Processing, 2012. **28**: p. 212-228.
2. Oliveira, G., F. Magalhães, Á. Cunha, and E. Caetano, *Development and implementation of a continuous dynamic monitoring system in a wind turbine.* Journal of Civil Structural Health Monitoring, 2016. **6**(3): p. 343-353.
3. Martins, N., E. Caetano, S. Diord, F. Magalhães, and T. Cunha, *Dynamic monitoring of a stadium suspension roof: Wind and temperature influence on modal parameters and structural response.* Engineering Structures, 2014. **59**: p. 80-94.
4. Ubertini, F., G. Comanducci, and N. Cavalagli, *Vibration-based structural health monitoring of a historic bell-tower using output-only measurements and multivariate statistical analysis.* Structural Health Monitoring, 2016. **15**(4): p. 438-457.
5. Pereira, S., F. Magalhães, J.P. Gomes, Á. Cunha, and J.V. Lemos, *Dynamic monitoring of a concrete arch dam during the first filling of the reservoir.* Engineering Structures, 2018.
6. Magalhães, F., S. Amador, Á. Cunha, and E. Caetano. *DynaMo - Software for vibration based structural health monitoring*. in *Bridge Maintenance, Safety, Management, Resilience and Sustainability - Proceedings of the Sixth International Conference on Bridge Maintenance, Safety and Management*. 2012.
7. EDP. *Energias de Portugal*. [cited 2018 21/09/2018]; Available from: http://www.a-nossa-energia.edp.pt/centros_produtores/.
8. Magalhães, F., A. Cunha, and E. Caetano, *Online automatic identification of the modal parameters of a long span arch bridge.* Mechanical Systems and Signal Processing, 2009. **23**(2): p. 316-329.
9. Allemang, R.J., *The modal assurance criterion–twenty years of use and abuse.* Sound and vibration, 2003. **37**(8): p. 14-23.

10. Reynders, E., R. Pintelon, and G. De Roeck, *Uncertainty bounds on modal parameters obtained from stochastic subspace identification.* Mechanical Systems and Signal Processing, 2008. **22**(4): p. 948-969.

11. Döhler, M. and L. Mevel, *Efficient multi-order uncertainty computation for stochastic subspace identification.* Mechanical Systems and Signal Processing, 2013. **38**(2): p. 346-366.

12. Reynders, E., K. Maes, G. Lombaert, and G. De Roeck, *Uncertainty quantification in operational modal analysis with stochastic subspace identification: Validation and applications.* Mechanical Systems and Signal Processing, 2016. **66-67**: p. 13-30.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

DATA FEATURES-BASED LIKELIHOOD-INFORMED BAYESIAN FINITE ELEMENT MODEL UPDATING

Xinyu Jia[1], Costas Papadimitriou[1*]

[1] Department of Mechanical Engineering, University of Thessaly, Volos 38334, Greece
e-mail: jia@uth.gr, costasp@uth.gr

Abstract

A new formulation for likelihood-informed Bayesian inference is proposed in this work based on probability models introduced for the features between the measurements and model predictions. The formulation applies to both linear and nonlinear dynamic models of structures. A relation between likelihood-informed and likelihood-free approximate Bayesian computation (ABC) is also established in this study, demonstrating that both formulations yield reasonable and consistent uncertainties for the model parameters. In particular, the uncertainties obtained with the new formulation account better for the fact that different sampling rates used in recording response time history measurements often yield measurements that contain the same information and so the sampling rate should not affect the uncertainty in the model parameters. The effectiveness of the proposed approach is demonstrated using an example from model updating of a linear model of a dynamical spring-mass chain system.

Keywords: Uncertainty quantification, Bayesian learning, model updating, structural dynamics, Likelihood-informed Bayesian computation, data features

1 INTRODUCTION

Bayesian model updating has gained more interest because of its effectiveness in practical engineering problems [1-3]. In Bayesian updating, the prior probability density function (PDF) of model parameters is updated to the posterior PDF by accounting for the information obtained from the measurements. Using probability models for the prediction errors, often formulated as the discrepancy between model predictions and the measurements, the likelihood function is developed. Asymptotic methods and sampling techniques have been developed to solve the parameter inference problem. In particular, sampling methods include versions of Markov Chain Monte Carlo (MCMC) (e.g. [4]), adaptive MCMC [5] as well as Transitional MCMC (TMCMC) [6, 7]. For likelihood-free parameter inference, the approximate Bayesian computation (ABC) has been developed. Among the algorithms proposed to solve the ABC, the subset simulation [8, 9] is shown to be computational effective alternative.

Bayesian model updating in structural dynamics using response time histories measurements such as accelerations, displacements or strains is often formulated by introducing point-to-point probabilistic descriptions of the discrepancy between the measurements and model predictions [10]. Spatially and temporally uncorrelated prediction error models used to quantify these discrepancies, result in very peaked posterior probability distributions for the model parameters due to the large number of data points available from high sampling rates. Spatially and temporally correlated prediction error models are more reasonable for quantifying uncertainties [11, 12]. However, the uncertainty depends on the correlation structure assumed which is often unknown and needs to be selected from a family of user-introduced correlation structures that might not be representative for the application. In general, the uncertainty quantified by the posterior probability distribution depends highly on the prediction error models and the correlation structure introduced between time instances as well as between measurements at different locations.

Herein we address the problem of Bayesian learning given response time history measurements. It is expected that for sufficiently small sampling rate, the information contained in the response time histories is independent of the sampling rate used to represent the time histories. Conventional techniques fail to quantify such independence and also give unrealistically small uncertainties due to the large number of data points used to represent the time histories. To properly quantify uncertainties, we propose a new formulation for likelihood-informed Bayesian inference based on probability models introduced for the features between the measured data and model predictions. Specifically, a probability model is assigned to the square of the discrepancy of the response time history between the measurement and the model prediction. Different probability models are investigated, such as a truncated Gaussian model and an exponential distribution model. It is demonstrated that reasonable uncertainties are obtained for the model parameters that are independent of the sampling rate used to represent the response time histories. A relation between likelihood-informed and likelihood-free Bayesian computations is also established, demonstrating that both formulations yield reasonable and consistent uncertainties for the model parameters. A spring-mass chain model with simulated, noise contaminated, measured acceleration time histories is used to demonstrate the effectiveness of the proposed approach.

The rest of this paper is organized as follows. In Section 2, the new likelihood-informed formulation for Bayesian model updating is proposed and compared with ABC formulation. The effectiveness of the proposed method is demonstrated using a spring-mass chain system in Section 3. Section 4 reports the conclusions of this study.

2 PROPOSED BAYESIAN FORMULATIONS

In Bayesian framework, the probabilities of unknown parameter sets $\underline{\theta}$ in the model class M can be first estimated from the prior probability density functions (PDF), and then it can be updated based on the following Bayesian formula when some measurements D are available:

$$p(\underline{\theta} \mid D, M) = \mathrm{c}\, p(D| \, \underline{\theta}, M)\, p(\underline{\theta} \mid M) \tag{1}$$

where $p(\underline{\theta} \mid D, M)$ is the posterior PDF of the model parameters given the measurements D and the model class M ; $p(\underline{\theta} \mid M)$ is the prior PDF; c is the constant which is selected so that the posterior PDF integrates to one; $p(D| \, \underline{\theta}, M)$ is the likelihood function of observing the data from the model class.

2.1 Model parameter estimation

Consider a parameterized class of structures models $\underline{g}\left(\underline{\theta}; M\right)$, where *M* is the model, $\underline{\theta}$ is the set of model parameters which can be identified using the measurements D . Let $D = \left\{ \hat{y}_j(k\Delta t) \in R^{N_0}, j = 1, 2, \cdots, N_0; k = 1, 2, \cdots, N_D \right\}$ be the measured response time histories data from the structure, where N_0 is the number of degrees of freedom (DOF) of the models, N_D is the number of the sampled data using a sampling rate Δt, *j* and *k* denote the *j*-th modes and time index at time $k\Delta t$, respectively.

Conventional methods for parameter estimation in structural dynamics using direct response time history measurements are based on prediction error equations formulated at time $t = k\Delta t$ as follows:

$$\hat{y}_j\left(k\right) = g_j\left(k; \underline{\theta}, M\right) + \varepsilon_j(k; \underline{\theta}) \tag{2}$$

Using a zero-mean Gaussian model for the prediction errors $\varepsilon_j(k; \underline{\theta})$, $k = 1, 2, \cdots, N_D$, and assuming of the prediction errors between the different sensor DOF $j = 1, 2, \cdots, N_0$, one can readily built the likelihood in the form given in [10, 13].

Herein, a new formulation for the likelihood is presented based on introducing probabilistic models for the features between the data and the model predictions. Specifically, it is assumed that the average of the square of the discrepancy between the measurements $\hat{y}_j\left(k\right)$ and the model prediction $g_j\left(k; \underline{\theta} \mid M\right)$, $k = 1, 2, \cdots, N_D$, satisfy the following equation:

$$\frac{1}{N_D} \sum_{k=1}^{N_D} \left[\hat{y}_j\left(k\right) - g_j\left(k; \underline{\theta}; M\right) \right]^2 = e_j \tag{3}$$

Due to the fact that the square error is always larger than zero, the uncertainty in e_j can be quantified with the following two kinds of distributions: 1. the truncated normal distribution; 2. the exponential distribution.

Regarding the case 1, the PDF of each variable e_j can be written as [14]:

$$p(e_j) = \frac{\sqrt{2}}{\sqrt{\pi\sigma}} \exp(-\frac{e_j^2}{2\sigma^2}) \tag{4}$$

where σ is the prediction error parameter of the truncated Gaussian probability models. The likelihood-informed based on the data features can be derived by the following formula:

$$p(\underline{e}|\ \underline{\theta},M)=\prod_{j=1}^{N_0} p(e_j|\underline{\theta},M) \tag{5}$$

The proposed likelihood is then given by:

$$p(\underline{e}|\ \underline{\theta},M)=\left(\frac{\sqrt{2}}{\sqrt{\pi}}\right)^{N_0}\cdot\left(\frac{1}{\sigma}\right)^{N_0}\cdot\exp\left(-\frac{1}{2}J\left(\underline{\theta};\sigma;M\right)\right) \tag{6}$$

where $J\left(\underline{\theta};\sigma;M\right)=\frac{1}{\sigma^2}\sum_{j=1}^{N_0}J_j\left(\underline{\theta};M\right)$, and $J_j\left(\underline{\theta};M\right)=\frac{1}{N_D}\sum_{k=1}^{N_D}\left[\hat{y}_j\left(k\right)-g_j\left(k;\underline{\theta};M\right)\right]^2$. Consequently, the logarithmic of the likelihood is:

$$L\left(\underline{\theta}\right)=\ln p(\underline{e}|\ \underline{\theta},M)=\mathrm{c}_0-N_0\ln\sigma-\frac{1}{2\sigma^2N_D}\sum_{j=1}^{N_0}\sum_{k=1}^{N_D}\left[\hat{y}_j\left(k\right)-g_j\left(k;\underline{\theta};M\right)\right]^2 \tag{7}$$

where $c_0=N_0\ln\sqrt{\frac{2}{\pi}}$.

In the case 2, the PDF of each variable e_j, assuming that it follows an exponential distribution, is given by:

$$p(e_j)=\begin{cases}\lambda\exp(-\lambda e_j) & e_j\geq 0\\ 0 & e_j<0\end{cases} \tag{8}$$

where the parameter λ is reparameterized by $\lambda=\frac{1}{2\sigma^2}$, which can make the exponent term equal to that of the truncated normal distribution. Similarly, the logarithmic likelihood function is calculated as:

$$L\left(\underline{\theta}\right)=\ln p(\underline{e}|\ \underline{\theta},M)=\mathrm{c}_1-2N_0\ln\sigma-\frac{1}{2\sigma^2N_D}\sum_{j=1}^{N_0}\sum_{k=1}^{N_D}\left[\hat{y}_j\left(k\right)-g_j\left(k;\underline{\theta};M\right)\right]^2 \tag{9}$$

where $c_1=-N_0\ln 2$.

When the prior PDF and likelihood function are determined, the posterior PDF of the model parameters $\underline{\theta}$ is further solved according to the Eq. (1). It should be noted that the new method extends a recent likelihood-informed formulation developed for the case where the modal frequencies and mode shape components are available as the measured data [15].

Several methods have been introduced to estimate the model parameters and their uncertainties. Specifically, Monte Carlo Markov Chain (MCMC) [4], adaptive MCMC [5] as well as Transitional MCMC (TMCMC) [6], etc, can be used for populating with samples the support of the posterior distribution. Herein, the TMCMC algorithm is applied.

2.2 Relationship between likelihood-informed formulation and ABC

Based on the Eq. (9), the most probable value (MPV) $\hat{\sigma}^2$ of the posterior PDF can be obtained. Equivalently, it can be solved by maximizing the logarithmic likelihood function L:

$$\left.\frac{\partial L_1}{\partial \sigma^2}\right|_{\sigma^2=\hat{\sigma}^2}=0 \tag{10}$$

where $L_1=-L$. The best estimate is then given by:

$$\hat{\sigma}^2=\frac{1}{2N_0}\varepsilon \tag{11}$$

where ε is defined as a prediction error, which is given by:

$$\varepsilon=\sum_{j=1}^{N_0}\frac{1}{N_D}\sum_{k=1}^{N_D}\left[\hat{y}_j(k)-g_j(k;\underline{\theta};M)\right]^2 \tag{12}$$

Equivalently, Eq. (11) can be rewritten as follows:

$$\varepsilon=2N_0\hat{\sigma}^2 \tag{13}$$

In ABC algorithms, a summary statistics η and a tolerance parameter δ are first introduced [16]:

$$\rho\left(\eta(X),\eta(D)\right)\leq\delta \tag{14}$$

where $X\in D$ denotes a simulated dataset from $p(D|\underline{\theta},M)$, and $\rho(\cdot,\cdot)$ is a distance measure on the model output space. In general, the measure $\rho(\cdot,\cdot)$ is chosen to be the least square measure of the distance between the measurements and the model prediction from a parameterized class of structures models. Specifically for the model with predictions $\underline{g}(\underline{\theta};M)$, it is written as:

$$\rho=\sum_{j=1}^{N_0}\frac{1}{N_D}\sum_{k=1}^{N_D}\left[\hat{y}_j(k)-g_j(k;\underline{\theta};M)\right]^2 \tag{15}$$

It can be readily found that the right side term in Eq. (15) is exactly the same as that in Eq. (12), thus the tolerance value δ can be then selected based on the best estimate $\hat{\sigma}^2$:

$$\delta=2N_0\hat{\sigma}^2 \tag{16}$$

The effectiveness of choosing the tolerance value is also demonstrated using examples in the next section.

3 NUMERICAL EXAMPLE

3.1 Description of a 10-DOF Spring-Mass Chain model

Consider a 10-DOF spring-mass chain system excited at the base. The equation of motion with base excitation $\ddot{y}_g(t)$:

$$M\underline{\ddot{y}}(t)+C\underline{\dot{y}}(t)+K(\underline{\theta})\underline{y}(t)=-M\underline{1}\ddot{y}_g(t) \tag{17}$$

where $\underline{1}=[1,1,\cdots,1]^T$ is a 10×1 vector. The system is created based on the following assumptions:

a) The mass matrix M is diagonal having elements equal to 1kg.

b) The springs are assumed to have the same stiffness equal to 1000N/m, and the spring matrix K is given by the following stiffness matrix when the parameter $\underline{\theta}=\underline{1}$:

$$K=\begin{bmatrix} k_1+k_2 & -k_2 & 0 & \cdots & 0 \\ -k_2 & k_2+k_3 & -k_3 & 0 & 0 \\ 0 & -k_3 & \ddots & \ddots & 0 \\ 0 & 0 & \ddots & k_9+k_{10} & -k_{10} \\ 0 & 0 & 0 & -k_{10} & k_{10} \end{bmatrix} \tag{18}$$

c) Rayleigh damping is assumed with the damping matrix written as

$$C=\alpha M+\beta K \tag{19}$$

where the coefficient α and β are taken to be 0.2265 and $6.7515\,e^{-4}$, respectively, corresponding to given damping ratios $\zeta_1=\zeta_5=0.02$ for the first and fifth modes of the system. .

d) Given the above system properties, the natural frequencies $\omega_1,\omega_2,\omega_3$ of the first three modes are estimated to be 1.0Hz, 3.0Hz and 4.9Hz.

e) The base excitation $\ddot{y}_g$ is obtained from an earthquake excitation, as shown in Fig. 1.

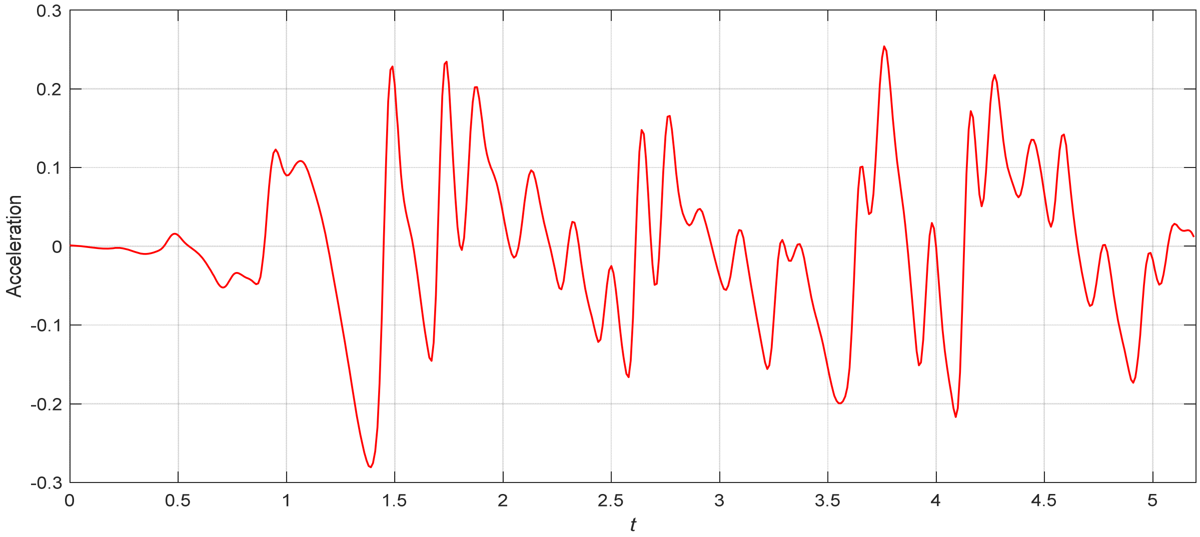


Fig. 1 Earthquake excitation

Eq. (17) can be also expressed with respect to the modal coordinates using the transformation $\underline{y}(t)=\Phi\underline{\xi}(t)$, as follows:

$$\ddot{\underline{\xi}}(t)+C^*\dot{\underline{\xi}}(t)+\Omega\underline{\xi}(t)=-\Phi^T M\underline{1}\ddot{y}_g(t) \tag{20}$$

where C^* and Ω are two diagonal matrices with elements $2\zeta\omega_i$ and ω_i^2, respectively. The state-space form is next constructed:

$$\dot{\underline{x}}(t)=A_c\underline{x}(t)+B_c p(t) \tag{21}$$

where $\underline{x}$ is state vector, $\dot{\underline{x}}$ is first derivative of the state, $\mathbf{A}_c$ is system state matrix and $\mathbf{B}_c$ is the input to state matrix given as:

$$x(t)=\begin{bmatrix}\xi(t)\\ \dot{\xi}(t)\end{bmatrix},\ A_c=\begin{bmatrix}\underline{0} & I\\ -\Omega & -C^*\end{bmatrix},\ B_c=\begin{bmatrix}\underline{0}\\ -\Phi M\underline{1}\end{bmatrix},\ p(\mathrm{t})=\ddot{y}_g(t) \tag{22}$$

The observation equation can also be written in the form:

$$\underline{d}(t)=G_c\underline{x}(t)+J_c p(t) \tag{23}$$

For absolute acceleration measurements $\underline{d}(t)$, the matrixes G_c and J_c are given by:

$$G_c=\left[-S_a\Phi\Omega\ -S_a\Phi C^*\right],\ J_c=\left[S_a(\underline{1}-\Phi\Phi^T M\underline{1})\right] \tag{24}$$

where S_a is the selection matrix. Thus, the system of equations (23) and (24) can be applied to predict the acceleration measurements.

3.2 Results

The proposed method is now applied to the system mentioned above. Two cases are investigated in this section. The first one studies the problem of parameter estimation using the data features to formulate the likelihood, with truncated normal (TN) distribution assumed for the square of the discrepancy between the measured and model predicted response time histories (Case 1). The other one formulates the likelihood in a similar way but assumes an exponential (EXP) distribution for the square error, instead of a truncated Gaussian distribution. The exponential distribution is also used to explore the relationship between likelihood-informed algorithm and the likelihood-free ABC algorithm. All methods are compared with the conventional Bayesian formulation assuming normal (NORM) distribution for the prediction errors at each time instant to construct the likelihood.

Results are presented for simulated measurements that are generated for a nominal spring-mass chain model. To simulate the effect of model error, 5% Gaussian noise is added to the acceleration response time histories generated from the nominal model. The acceleration measurements from all ten DOF of the system are considered. For demonstration purposes, a single stiffness parameter is considered as the model parameter to be updated. This parameter included the stiffness of the first three springs in the spring-mass chain system.

Parameter estimation results along with their uncertainties (5 and 95% quantiles) are presented in Figs. 2 and 3 for different sampling rates Δt ranging from $0.1\Delta t$ to $10\Delta t$ of the same time history. The number of the samples N_D are decreased accordingly from $10N_D$ to $0.1N_D$. Specifically, results from the proposed truncated Gaussian distribution (TN) are compared with results obtained from the conventional Bayesian method. It should be noted that the different sampling rates chosen do not affect the information contained in the data. Both methods give almost the same MAP estimates for the structural model parameter (Fig. 2). However, uncertainty bounds are substantially different for the two methods. Specifically, from the results in Fig. 2, it becomes evident that the conventional Bayesian method gives very small uncertainties that decrease as the number of sampling points increase. The proposed method based on the data features provides much higher uncertainties that are independent on the number of data points used. This is consistent with intuition since the information contained in the acceleration time history is almost independent of the sampling rate used in this example.

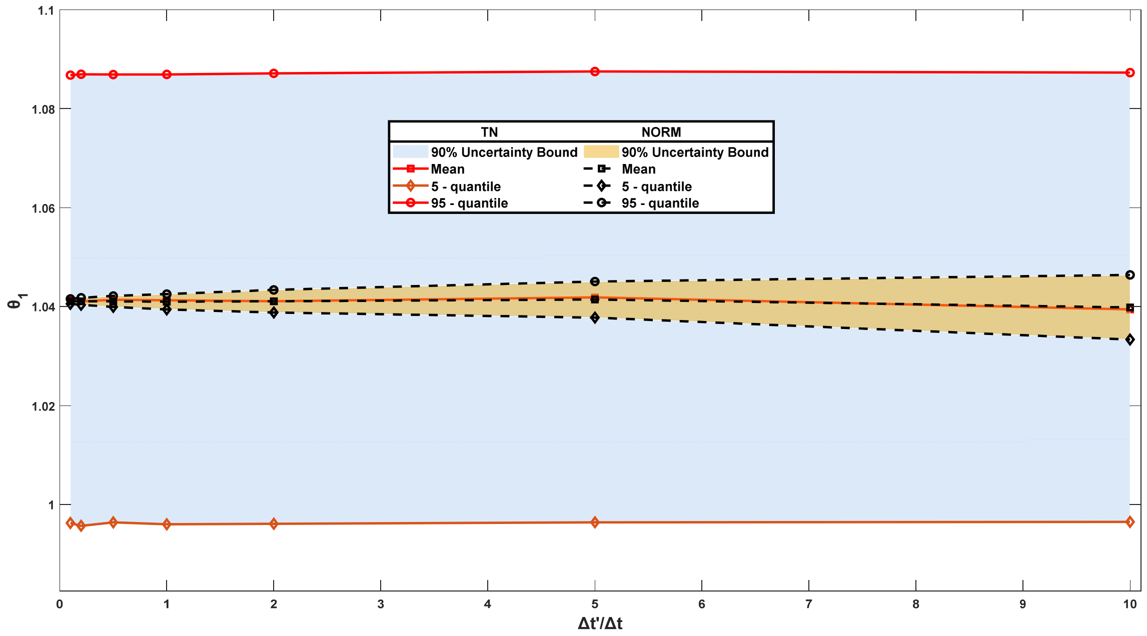


Fig. 2 Parameter estimation of θ_1 using TN (Case 1) and NORM

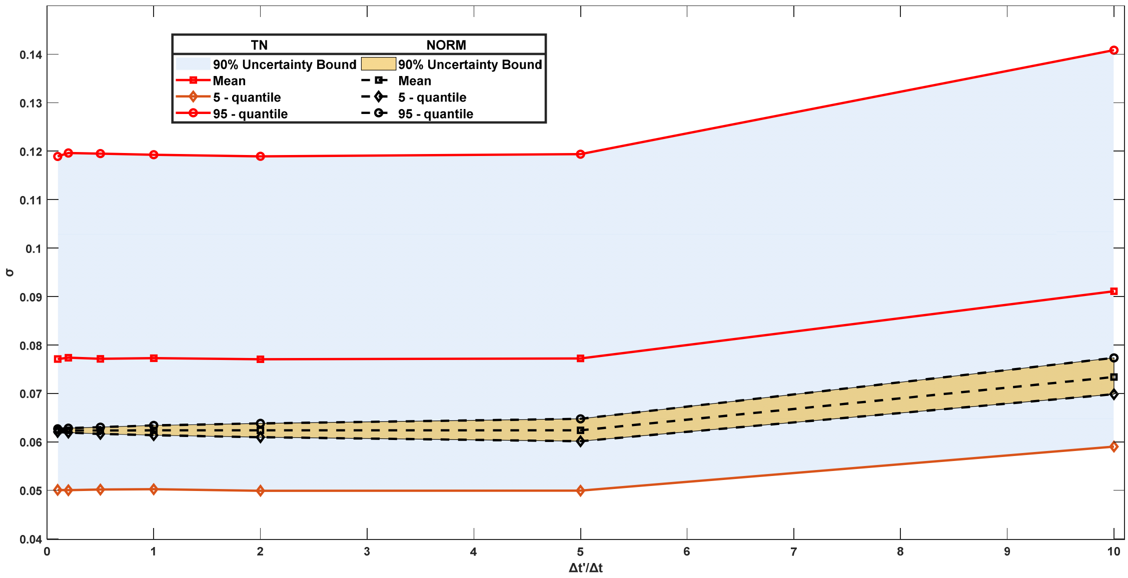


Fig. 3 Parameter estimation of σ using TN (Case 1) and NORM

Next, parameter estimation results along with their uncertainties (5 and 95% quantiles) are compared in Fig. 4 for the TN case (Case 2), the conventional Bayesian method (NORM) and the ABC method. Again the sampling rates Δt range from $0.1\Delta t$ to $10\Delta t$ of the same time history. The best estimate $\hat{\sigma}$ of the standard deviation is specified as the value σ_{90} (90-quantile) obtained from EXP. Then the tolerance value δ in ABC algorithm can be calculated based on the formulation in Eq. (16). Although all three methods predict the same MAP estimate, the uncertainty bounds computed from the conventional Bayesian methods are again substantially smaller than the bounds computed from the other two methods. Also, the uncertainty in the model parameter decreases as the sampling rate increases which is contrary to intuition, since there is not extra information contained in the time history with higher sampling rate. The uncertainty predicted by the proposed likelihood-informed method is similar to the uncertainty estimated by the ABC method. Both methods (TN and ABC) provide uncertainty bounds that are almost independent on the sampling rate. The small discrepancies in the uncertainty bounds are due to the choice of the tolerance value in ABC. A slightly different tolerance can zero the discrepancy between the two methods.

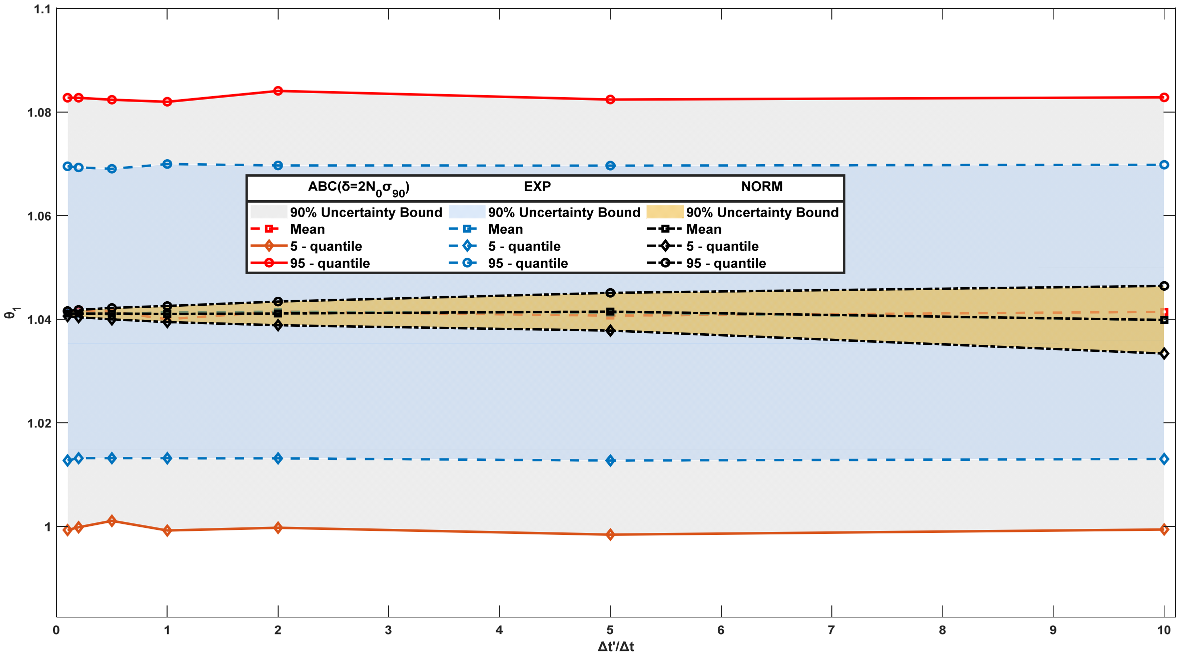


Fig. 4 Parameter estimation of θ_1 using EXP (Case 2), NORM and ABC

4 CONCLUSIONS

A new formulation based on the data features for likelihood-informed Bayesian inference has been presented and discussed in this paper. The effectiveness of the proposed formulation has been demonstrated by a spring-mass chain model. The main conclusions of this work are:

- The proposed data-features likelihood-based Bayesian methodology correctly accounts for the uncertainty in the model parameters, making such uncertainty independent of sampling rate of the measured response time histories. In contrast, the uncertainty in the model parameters obtained from conventional Bayesian inference formulation depends on the sampling rate of the response time histories, despite the fact that the information contained in the response time history data is independent of the sampling rate.
- The proposed likelihood-informed Bayesian formulation provides results that are consistent with the ones obtained from likelihood-free ABC formulations.
- The proposed method applied herein to linear structural systems can also be extended to non-linear structural systems given response time history measurements.

Acknowledgment: This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 764547.

REFERENCES

[1] M. Muto, J.L. Beck, Bayesian updating and model class selection for hysteretic structural models using stochastic simulation, *Journal of Vibration and Control*, **14**, 7-34, 2008.

[2] F. DiazDelaO, A. Garbuno-Inigo, S. Au, I. Yoshida, Bayesian updating and model class selection with Subset Simulation, *Computer Methods in Applied Mechanics and Engineering*, **317**, 1102-1121, 2017.

[3] O. Sedehi, C. Papadimitriou, L.S. Katafygiotis, Probabilistic hierarchical Bayesian framework for time-domain model updating and robust predictions, *Mechanical Systems and Signal Processing*, **123**, 648-673, 2019.

[4] S. Au, J.L. Beck, A new adaptive importance sampling scheme for reliability calculations, *Structural safety*, **21**, 135-158, 1999.

[5] J.L. Beck, S.-K. Au, Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation, *Journal of Engineering Mechanics*, **128**, 380-391, 2002.

[6] J. Ching, Y.-C. Chen, Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging, *Journal of Engineering Mechanics*, **133**, 816-832, 2007.

[7] S. Wu, P. Angelikopoulos, C. Papadimitriou, P. Koumoutsakos, Bayesian annealed sequential importance sampling: an unbiased version of transitional Markov chain Monte Carlo, *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, **4**, 011008, 2018.

[8] M. Chiachio, J.L. Beck, J. Chiachio, G. Rus, Approximate Bayesian computation by subset simulation, *SIAM Journal on Scientific Computing*, **36**, A1339-A1358, 2014.

[9] M.K. Vakilzadeh, Y. Huang, J.L. Beck, T. Abrahamsson, Approximate Bayesian Computation by subset simulation using hierarchical state-space models, *Mechanical Systems and Signal Processing*, **84**, 2-20, 2017.

[10] J.L. Beck, L.S. Katafygiotis, Updating models and their uncertainties. I: Bayesian statistical framework, *Journal of Engineering Mechanics*, **124**, 455-461, 1998.

[11] E. Simoen, C. Papadimitriou, G. Lombaert, On prediction error correlation in Bayesian model updating, *Journal of Sound and Vibration*, **332**, 4136-4152, 2013.

[12] C. Papadimitriou, G. Lombaert, The effect of prediction error correlation on optimal sensor placement in structural dynamics, *Mechanical Systems and Signal Processing*, **28**, 105-127, 2012.

[13] C. Papadimitriou, Bayesian uncertainty quantification and propagation (UQ+ P): state-of-the-art tools for linear and nonlinear structural dynamics models, in *Identification Methods for Structural Health Monitoring*, E. Chatzi and C. Papadimitriou (Eds), Springer, pp. 137-170, 2016.

[14] S. Wilhelm, B. Manjunath, tmvtnorm: A package for the truncated multivariate normal distribution, *Sigma*, **2**, 2010.

[15] C. Argyris, *Bayesian Uncertainty Quantification and Optimal Experimental Design in Data-Driven Simulations of Engineering Systems*, PhD Thesis, University of Thessaly, Department of Mechanical Engineering, Greece, 2017.

[16] R.D. Wilkinson, Approximate Bayesian computation (ABC) gives exact results under the assumption of model error, *Statistical Applications in Genetics and Molecular Biology*, **12**, 129-141, 2013.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

TOWARDS A GENERAL THEORY FOR DATA-BASED POSSIBILISTIC PARAMETER INFERENCE

Dominik Hose and Michael Hanss

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
e-mail: {dominik.hose, michael.hanss}@itm.uni-stuttgart.de

Keywords: Possibility Theory, Imprecise Probabilities, Distribution Estimation, Parameter Identification, Uncertainty Propagation, (Inverse) Fuzzy Arithmetic.

Abstract. *This paper unifies several recent results from possibilistic uncertainty analysis in order to contribute to a general theory of possibilistic parameter estimation by providing an exemplary procedure to estimating possibilistic distributions of model parameters from samples of an aggregated output quantity.*

This task is accomplished by dividing the problem in two subproblems. In the first step, the output samples are represented in a structured manner by a possibility distribution. The second step deals with the backpropagation of the output distribution through a model, thus arriving at a distribution of the input quantity to be estimated.

The theoretical basis for this two-step scheme lies in the theory of imprecise probabilities, giving the computed distributions an immediate and meaningful interpretation. It is intended to provoke the development of a novel theory complementary to classical statistics.

1 Introduction

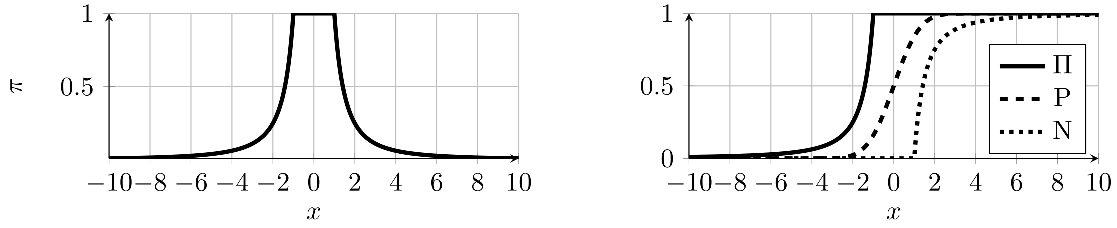
In some cases, e.g. in mechanical engineering [1], it is necessary to infer an input (a-priori) distribution of an unknown quantity, e.g. masses, stiffnesses or damping from the a-posteriori distribution of a measurable output quantity, e.g. energy, displacements, or eigenfrequencies. In the general probabilistic case, this typically requires the solution of a system of equations involving stochastic variables and possessing infinitely many solutions. Thus, it cannot be solved without additional assumptions, such as maximum entropy of the input distribution or its shape (Gaussian, uniform, etc.). This problem can be resolved if one is willing to revert back to the coarser framework of possibility theory. A solution scheme is proposed here, which is divided into two steps.

First of all, the available output data are represented in a structured manner by means of a possibility distribution, which is motivated by results about consistent probability-possibility transformations. Few scholars have investigated this topic, and literature is sparse or uninspiring. Many authors simply assume a reasonable range and a nominal value of some parameter and then construct a triangular fuzzy number from these values. This approach may be justifiable in some cases [2], but often it falls short of the capabilities that possibility theory has to offer for representing uncertainty. Apart from the methods gathered in this contribution, namely percentage sets and possibilistic moment matching, Dubois and Prade provide some basic approaches to this problem in [3], and Masson and Denœux show how the empirical probabilities (relative frequencies) of multinomial probability distributions can be used to construct possibility distributions by means of confidence intervals [4]. For practical methods for the construction of possibility distributions from probability distributions or families thereof interested readers may refer to [5]. Secondly, relying on results about inverse fuzzy arithmetic, the output distribution is propagated backwards through the given model, yielding the desired possibility distribution of the parameter to be inferred. The solution of such fuzzy equations [6], where the parameters are of possibilistic rather than probabilistic nature, has also received little attention by scholars in the past and therefore requires further analysis. The most rigorous pursuer of this line of research is certainly Tanaka who solves fuzzy linear equations with very restrictive assumptions about the shape of the involved fuzzy parameters in the context of fuzzy linear programming, but is able to provide strong results, e.g. in [7, 8, 9]. Furthermore, Hukuhara introduces the Hukuhara difference for set-valued functions in [10], which Bede and Stefanini employ on an α-cut basis in [11] to propose an inverse to the addition and multiplication of fuzzy-valued functions. In a recent review, Lodwick and Dubois argue that the solution of interval linear systems is a first step to solving systems of fuzzy linear equations [12]. They identify four cases – depending on the kind of uncertainty that is encoded in the interval – how interval linear systems should be categorized and solved. Consequently, they recommend applying techniques for the solution of interval linear systems, such as contractor programming [13] or the identification of robust solution spaces [14], on an α-cut basis for the solution of fuzzy linear systems. Here, a generalization of the above mentioned results is sought.

The remainder of this contribution is organized as follows. In Section 2, a brief overview of possibility theory is given. Section 3 is concerned with the structured representation of data by means of a possibility distribution, and in Section 4, inverse fuzzy arithmetic is introduced to infer a-priori possibility distributions. In order to demonstrate the usefulness of the suggested solution scheme, a well-known application example, the GARTEUR SM-AG-19 testbed, is employed in Section 5, where the two basic steps of the proposed procedure are performed. Some concluding remarks are given in Section 6.

2 Possibilistic Uncertainty Descriptions

Possibility theory provides a unified framework for a robust treatment of polymorphic uncertainties. Therein, the possibility measure $\Pi : \Omega \to [0,1]$, the possibilistic counterpart to a probability measure P, assigns varying levels of confidence between $\Pi(\emptyset) = 0$ and $\Pi(\Omega) = 1$ to all subsets of the universe of discourse Ω. In contrast to the probability measure fulfilling the additivity axiom $\mathrm{P}(A \cup B) = \mathrm{P}(A) + \mathrm{P}(B)$ for disjunctive events $A, B \in 2^{\Omega}$, the possibility measure fulfills the maxitivity axiom $\Pi(A \cup B) = \max(\Pi(A), \Pi(B))$. This allows for a very general description of aleatory and epistemic uncertainties [15]. Arguing from the point of view that in a state of perfect knowledge only aleatory uncertainties remain and epistemic uncertainties arise do to a lack of knowledge about the true underlying probability distribution, an interval is the least specific representation of uncertainty and a probability distribution is its most specific representation [16] as it describes the uncertain outcome of an experiment perfectly. For instance, a fair coin will show heads fifty percent of the time. This aleatory uncertainty is irreducible; there is not a more specific way to describe it. Possibility theory can be employed to represent an additional epistemic lack of knowledge and to encode several probability distributions in just one possibility distribution. Encoding e.g. the confidence levels of such a probability distribution in a possibility distribution [5] loses some of the information, but still allows for a conservative assessment of upper and lower probabilities, i.e. the description of uncertainty is coarsened. In this context, the concept of *consistency* is of fundamental importance. Here, the definition by Dubois and Prade [17], viewing a possibility measure as an upper probability measure, is employed. More precisely, a probability measure P and a possibility measure Π are called consistent if the probability of an event $U \in 2^{\Omega}$ is bounded from above by its possibility, i.e. $\mathrm{P}(U) \leq \Pi(U)$, and consequently from below by the dual necessity $\mathrm{N}(U) = 1 - \Pi(\Omega \setminus U) \leq \mathrm{P}(U)$. Therefore, any possibility measure induces a credal set of consistent probability measures $\mathcal{P}_{\Pi} = \left\{ \mathrm{P} : \mathrm{P}(U) \leq \Pi(U) \ \forall \, U \in 2^{\Omega} \right\}$. Illustrative results about which probability distributions are actually contained in $\mathcal{P}_{\Pi}$ may be found in [16]. A possibility distribution consistent with all probability distributions with zero mean and unit variance is visualized in Figure 1. The corresponding possibility density function is shown in Figure 1a. The cumulative possibility and necessity distribution and the cumulative probability distribution of the normalized Gaussian distribution, all evaluating the event $U(x) = (-\infty, x]$, are depicted in Figure 1b. In the continuous case, an $\mathbb{R}^{n_x}$-valued uncertain (random or fuzzy) variable $\boldsymbol{X}$ may possess a probability distribution $\mathrm{P}_{\boldsymbol{X}}$ or a possibility distribution $\Pi_{\boldsymbol{X}}$ on the Borel σ-algebra $\mathcal{B}^{n_x}$. The possibility density function (fuzzy set membership function) $\pi_{\boldsymbol{X}}$ and the probability density function $p_{\boldsymbol{X}}$ have to satisfy $\Pi_{\boldsymbol{X}}(U) = \sup \pi_{\boldsymbol{X}}(U)$ and $\mathrm{P}_{\boldsymbol{X}}(U) = \int_U p_{\boldsymbol{X}} \, \mathrm{d}\lambda$, where λ is the Lebesgue measure, for all events $U \in \mathcal{B}^{n_x}$.



(a) Possibility density function.

(b) Cumulative distribution functions.

Figure 1: Possibility distribution consistent with all probability distributions with zero mean and unit variance.

3 Possibility Distribution Estimation

The construction of meaningful possibility distributions has received relatively little attention in the past. Below, two methods for the construction of possibility distributions from samples are presented.

3.1 Percentage Sets

The most intuitive method for constructing a possibility distribution from samples are percentage sets. The basic idea in one dimension, percentage intervals, is presented in [18] without mentioning the powerful theoretical results that can be shown about them. For the one-dimensional case, a well-known result from possibility theory, which can be found e.g. in [5], states that any cumulative probability distribution function can act as a possibility density function, e.g. $\pi_X = C_X(x) = \mathrm{P}_X(\{\xi \in \mathbb{R} : \xi \geq x\})$ for $x \in \mathbb{R}$, inducing a possibility distribution Π_X which is consistent with P_X since

$$\mathrm{P}_X(U) \overset{\hat{x}=\inf U}{\leq} C_X(\hat{x}) = \pi_X(\hat{x}) \overset{(\dagger)}{=} \sup \pi_X(U) = \Pi_X(U) \qquad \forall\, U \in \mathcal{B}\,, \tag{1}$$

where $(\dagger)$ follows from the fact that π_X is monotonously decreasing.

The $\{0,1\}$-valued uncertain variable $Y_x = \mathbb{1}_{\{X \geq x\}}$ assuming one if $X \geq x$ and zero otherwise is, thus, Bernoulli distributed with probability $C_X(x)$. Given n_r realizations $x_1, \dots, x_{n_r}$ of the independent and identically distributed (iid) random variables $X_1, \dots, X_{n_r} \sim \mathrm{P}_X$, the relative frequency ν_x of realizations with $x_i \geq x$ is an unbiased estimator of $C_X(x)$, and for $n_r \to \infty$ this yields $\nu_x \to C_X(x)$ according to Borels strong law of large numbers for the Bernoulli distribution of Y_x. This motivates choosing $\hat{\pi}_{X^{\mathrm{PS}}}(x) = \frac{1}{n_r}\sum_{x_i \geq x} 1$ to approximate a possibility distribution that is consistent with P_X.

An extension to the M-dimensional case is also feasible by approximating

$$\pi_{\boldsymbol{X}}(\boldsymbol{x}) = \mathrm{P}_{\boldsymbol{X}}(\{\boldsymbol{\xi} \in \mathbb{R}^{n_x} : ||\boldsymbol{\xi} - \boldsymbol{c}|| \geq ||\boldsymbol{x} - \boldsymbol{c}||\}) \qquad \forall\, \boldsymbol{x} \in \mathbb{R}^M \tag{2}$$

via $\hat{\pi}_{\boldsymbol{X}^{\mathrm{PS}}}(\boldsymbol{x}) = \frac{1}{n_r}\sum_{||\boldsymbol{x}_i - \boldsymbol{c}|| \geq ||\boldsymbol{x} - \boldsymbol{c}||} 1$ for any $\boldsymbol{c} \in (\mathbb{R} \cup \pm\infty)^{n_x}$. Depending upon the choice of the norm this yields different geometrical shapes, such as **Percentage Boxes** for the ∞-norm $||\cdot||_\infty$, **Percentage Spheres** for the 2-norm $||\cdot||_2$, **Percentage Hyperellipsoids** for the weighted 2-norm $||\boldsymbol{A}\cdot||_2$ where $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$ is a non-singular matrix, etc. An outer approximation of the **Percentage Intervals** in [18] can be obtained for $M = 1$ and $c = \frac{1}{n_r}\sum_{i=1}^{n_r} x_i$.

3.2 Possibilistic Moment Matching

Moment matching is a standard technique in statistics for the parameters of probability distributions. Following a similar line of reasoning as in the previous section, Mauris argues in [19] that the possibilistic representation of families of probabilities with certain characteristics, i.e. moments μ_i, is fundamentally linked to the solution of the possibilistic moment matching problem

$$\begin{aligned} \pi_X(x) = \qquad & \max_{\mathrm{P}_X} && \mathrm{P}_X(|X - c| \geq |x - c|) \\ & \text{subject to} && p_X(\xi) \geq 0, \qquad\qquad \forall\, \xi \in \mathbb{R} \\ &&& \int_{\mathbb{R}} p_X(\xi)\,\mathrm{d}\xi = 1, \\ &&& \int_{\mathbb{R}} p_X(\xi) h_i(\xi)\,\mathrm{d}\xi = \mu_i, \quad i = 1, \dots, n_h\,, \end{aligned} \tag{3}$$

where c is usually the suspected mode of the probabilities and h_i are the moment functions. This allows for a more general procedure for the construction of possibility densities. In principle, all moments can be estimated from the samples without bias by computing $\hat{\mu}_i = \frac{1}{n_r}\sum_{i=1}^{n_r} h_i(x_i)$. However, the variance of these estimates grows quickly such that matching the moments of higher order is generally not recommendable unless the number of samples is large. Therefore, only the sample moments $\hat{\mu}_i$ of lower order are matched and the resulting family of probabilities is captured via the estimated possibility density $\hat{\pi}_{X^{\mathrm{MM}}}$.

Notice, that the possibility density in Figure 1a may be obtained by matching the first-order raw moment, i.e. the mean by $h_1(\xi) = \xi$ with $\mu_1 = 0$ and the second-order central moment, i.e. the variance by $h_2(\xi) = \xi^2$ with $\mu_2 = 1$.

4 Inverse Fuzzy Arithmetic

The forward propagation of possibilistic variables through models of the form $\boldsymbol{Y} = \phi(\boldsymbol{X})$ is performed by the application of Zadeh's extension principle [20] providing the possibility distribution of $\boldsymbol{Y}$ according to

$$\pi_{\boldsymbol{Y}}(\boldsymbol{y}) = \sup_{\boldsymbol{x}:\boldsymbol{y}=\phi(\boldsymbol{x})} \pi_{\boldsymbol{X}}(\boldsymbol{x}) \tag{4}$$

where the supremum of the empty set is defined to be zero. For the inverse propagation, $\boldsymbol{Y}$ is an $\mathbb{R}^{n_y}$-valued fuzzy variable with known membership function $\pi_{\boldsymbol{Y}}$ and the membership function $\pi_{\boldsymbol{X}}$ of the $\mathbb{R}^{n_x}$-valued fuzzy variable $\boldsymbol{X}$ is sought.

Hose and Hanss provide an approach to inverse fuzzy arithmetic in [21] which can serve as a basis for a more general solution to fuzzy equations. This approach is further investigated here. It is a variation of the approaches provided in [9], [11] and [12]. The functional dependency is generalized to the nonlinear case, yet the dependency on additionally known fuzzy parameters is not considered. This overall reduced involvement allows for showing some desirable properties within the theory of imprecise probabilities [22].

Specifically, in [21] it is argued that the minimum specific inverse possibility distribution given by

$$\pi^*_{\phi^{-1}(\boldsymbol{Y})}(\boldsymbol{\xi}) = \pi_{\boldsymbol{Y}}(\phi(\boldsymbol{\xi})) \qquad \forall \boldsymbol{\xi} \in \mathbb{R}^{n_x} \tag{5}$$

is a sensible choice, purely approaching the problem in the framework of fuzzy set theory as an inverse to Zadeh's extension principle. This proposition may also be investigated in the framework of imprecise probabilities, yielding the following powerful results:

Suppose $\boldsymbol{X}$ is an $\mathbb{R}^{n_x}$-valued uncertain variable, $\phi : \mathbb{R}^{n_x} \to \mathbb{R}^{n_y}$ a Borel measurable and surjective function, and $\boldsymbol{Y} = \phi(\boldsymbol{X})$ an $\mathbb{R}^{n_y}$-valued uncertain variable. If $\boldsymbol{Y}$ possesses a probability distribution $\mathrm{P}_{\boldsymbol{Y}}$, then in general there exists an infinite number of probability distributions $\mathrm{P}_{\boldsymbol{X}}$ yielding this pushforward measure under ϕ. These extensions may be gathered in the set $\mathcal{I}^{\phi}_{\mathrm{P}_{\boldsymbol{Y}}} = \{\mathrm{P}_{\boldsymbol{X}} : \mathrm{P}_{\boldsymbol{Y}}(V) = \mathrm{P}_{\boldsymbol{X}}(\phi^{-1}(V))\ \forall V \in \mathcal{B}^{n_y}\}$ which is generally hard to compute. Theorem 2 in [23] states that an outer approximation of this credal set may be found by transforming $\mathrm{P}_{\boldsymbol{X}}$ into any consistent possibility distribution $\Pi_{\boldsymbol{X}}$ (refer e.g. to [2]) and then computing its minimum specific inverse possibility distribution $\Pi^*_{\phi^{-1}(\boldsymbol{Y})}$, whose induced credal set satisfies $\mathcal{I}^{\phi}_{\mathrm{P}_{\boldsymbol{Y}}} \subseteq \mathcal{P}_{\Pi^*_{\phi^{-1}(\boldsymbol{Y})}}$.

The univariate densities $\pi^*_{X_1}, \ldots, \pi^*_{X_{n_x}}$ can then be obtained by the marginalization

$$\pi^*_{X_i}(x_i) = \sup_{x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_{n_x}} \pi^*_{\phi^{-1}(\boldsymbol{Y})}(x_1,\ldots,x_{i-1},x_i,x_{i+1},\ldots,x_{n_x}), \qquad i = 1,\ldots,n_x\,. \tag{6}$$

The framework of imprecise probabilities facilitates the computation of upper and lower bounds of the expected values of the X_i [24] by evaluating the Choquet integrals [25]

$$\overline{\mathbb{E}}[X_i] = \int X_i \, \mathrm{d}\Pi^*_{X_i} = \int\limits_0^\infty \Pi^*_{X_i}\left(X_i \geq \xi\right) \, \mathrm{d}\xi + \int\limits_{-\infty}^0 \left[\Pi^*_{X_i}\left(X_i \geq \xi\right) - 1\right] \, \mathrm{d}\xi \tag{7}$$

and

$$\underline{\mathbb{E}}[X_i] = \int X_i \, \mathrm{d}\mathrm{N}^*_{X_i} = \int\limits_0^\infty \mathrm{N}^*_{X_i}\left(X_i \geq \xi\right) \, \mathrm{d}\xi + \int\limits_{-\infty}^0 \left[\mathrm{N}^*_{X_i}\left(X_i \geq \xi\right) - 1\right] \, \mathrm{d}\xi \tag{8}$$

which may serve as an interval-valued estimator $\hat{X}_i = \left[\underline{\mathbb{E}}[X_i], \overline{\mathbb{E}}[X_i]\right]$.

5 Application

In order to illustrate the general solution scheme for possibilistic parameter inference, the two steps provided in Sections 3 and 4 shall now be applied to the GARTEUR SM-AG-19 testbed presented in [26], which was originally designed to provide a benchmark problem for the various techniques available for stationary oscillation testing. A finite-element model of the GARTEUR structure is exhibited in Figure 2.

Several model updating techniques have been applied using experimental data [27, 28, 29, 30, 31]. However, these conflicting measurement results are a perfect example of epistemic uncertainties since the deviations do not stem from stochastic disturbances in the measurement process, but rather from the application of different measuring techniques. Simply fitting a model with a Gaussian noise term would fail to address this form of uncertainty in a reasonable manner.

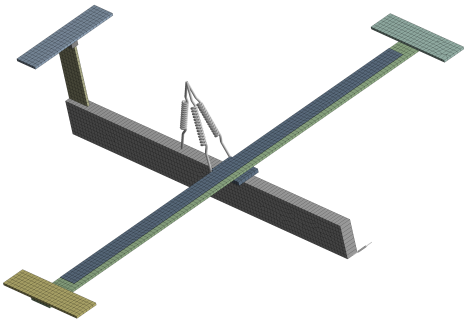
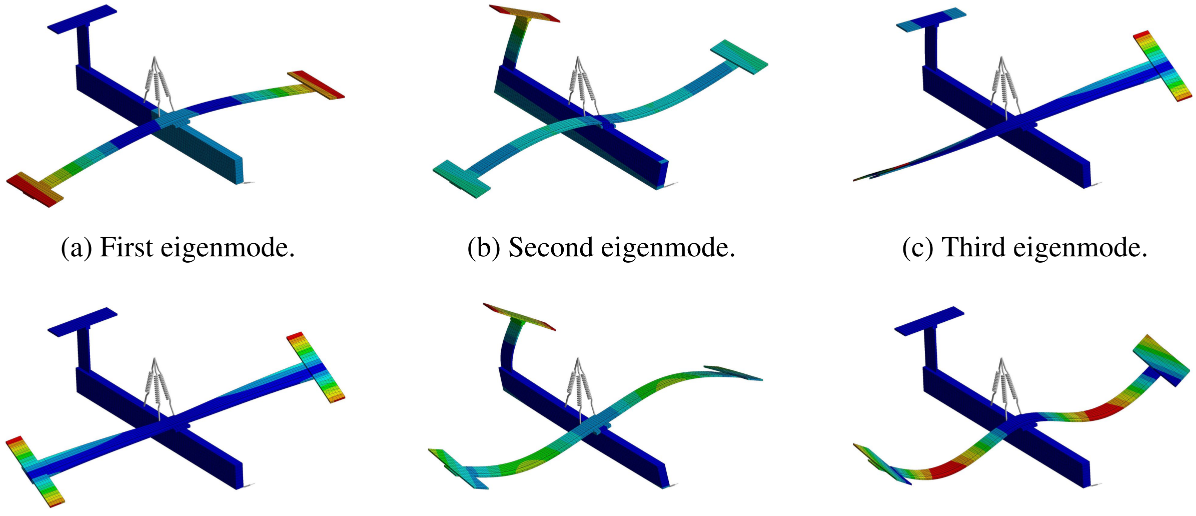


Figure 2: Finite-element model of the GARTEUR SM-AG-19 testbed.

5.1 STEP 1: Possibilisty Distribution Estimation

To ellucidate the methods of percentage sets and possibilistic moment matching, these methods will be applied to the experimental results of the groups who collaborated in the GARTEUR project. These data are gathered in Table 1, consisting of the six lowest identified eigenfrequencies with the associated modes shown in Figure 3.



(a) First eigenmode.

(b) Second eigenmode.

(c) Third eigenmode.

(d) Fourth eigenmode.

(e) Fifth eigenmode.

(f) Sixth eigenmode.

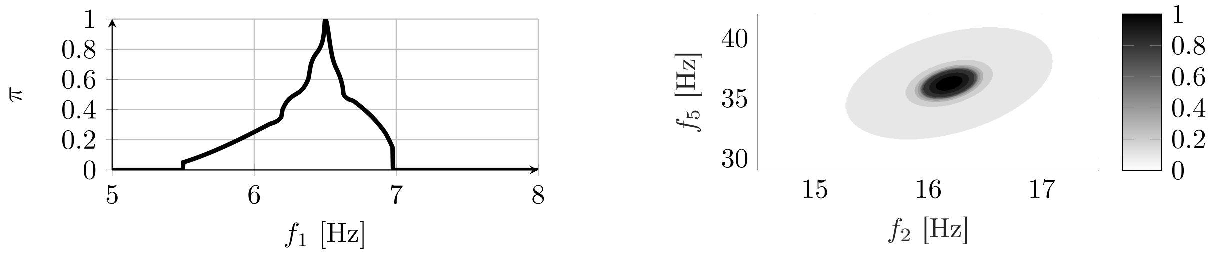
Figure 3: Mode shapes of the GARTEUR SM-AG-19 testbed.

Table 1: Eigenfrequencies of the GARTEUR SM-AG-19 testbed determined by different modal analysis techniques.

Source	Laboratory	f_1 [Hz]	f_2 [Hz]	f_3 [Hz]	f_4 [Hz]	f_5 [Hz]	f_6 [Hz]
[32]	-	5.50	14.91	31.96	32.33	34.38	43.89
[33]	Intespace a	6.1	16.3	33.7	36.0	40.7	49.6
[33]	Conservatoire National des Arts et Métiers	6.19	16.16	32.45	32.96	35.63	49.08
[33]	Intespace b	6.2	16.3	33.3	35.8	41.4	49.4
[34]	Deutsches Zentrum für Luft- und Raumfahrt	6.38	16.10	33.13	33.53	35.65	48.38
[33]	Aerospatiale a	6.39	15.98	31.84	32.33	35.12	48.47
[33]	Aerospatiale b	6.4	16.01	31.92	34.66	35.13	48.49
[33]	Saab	6.48	16.33	33.31	33.75	35.73	48.85
[33]	Defense Research Agency a	6.49	16.41	33.42	33.87	36.26	49.55
[33]	Defense Research Agency b	6.50	16.45	33.49	33.97	36.34	49.85
[33]	Defense Research Agency c	6.50	16.47	33.47	33.97	36.38	49.84
[35]	Imperial College a	6.54	16.55	34.86	35.30	36.53	49.81
[35]	University of Wales	6.55	16.61	34.88	35.36	36.71	50.09
[33]	Imperial College b	6.623	16.210	35.420	37.177	37.464	48.421
[33]	Office National D'Etudes et de Recherches Aerospatiales	6.63	16.25	33.16	33.57	35.36	48.62
[33]	University of Manchester	6.71	16.40	33.46	33.94	36.12	49.65
[33]	Aerospatiale c	6.92	16.09	32.96	33.48	35.33	48.41
[33]	Aerospatiale d	6.949	15.996	32.867	33.375	34.726	48.07
[33]	Imperial College c	6.974	16.079	33.862	33.938	34.915	46.080
[33]	Sopemea	6.974	16.079	33.682	33.938	34.915	46.080

a) Percentage Sets

Figure 4a shows the possibility density $\hat{\pi}_{F_1^{\mathrm{PS}}}$ obtained from the percentage intervals of the first eigenfrequency, and Figure 4b shows the percentage hyperellipsoid possibility density $\hat{\pi}_{\boldsymbol{F}_{2,5}^{\mathrm{PS}}}$ obtained from the empirical cumulative probability distribution function of the eigenfrequencies f_2 and f_5 after a student normalization, i.e. by centering about the sample means and scaling with the Cholesky decomposed sample covariance matrix.



(a) Possibility density $\hat{\pi}_{F_1^{\mathrm{PS}}}$ of the first eigenfrequency.

(b) Joint possibility density $\hat{\pi}_{\boldsymbol{F}_{2,5}^{\mathrm{PS}}}$ of the eigenfrequencies f_2 and f_5.

Figure 4: Percentage sets of the data given in Table 1.

b) Moment Matching

The raw ith moments $\hat{\mu}_{i,j}$ of the jth eigenfrequencies in Table 1 are given in Table 2. The possibility density $\hat{\pi}_{F_1^{\mathrm{MM}}}$ in Figure 5 are obtained by possibilistic moment matching, i.e. by solving (3) for the first three raw sample moments of the first eigenfrequency.

Table 2: Raw sample moments of the data in Table 1.

Moment function	f_1	f_2	f_3	f_4	f_5	f_6
$h_1(\xi) = \xi$	6.5	16.2	33.4	34.2	36.2	48.5
$h_2(\xi) = \xi^2$	42.4	262	$1.11 \cdot 10^3$	$1.17 \cdot 10^3$	$1.32 \cdot 10^3$	$2.36 \cdot 10^3$
$h_3(\xi) = \xi^3$	277	$4.24 \cdot 10^3$	$3.72 \cdot 10^4$	$4 \cdot 10^4$	$4.79 \cdot 10^4$	$1.15 \cdot 10^5$

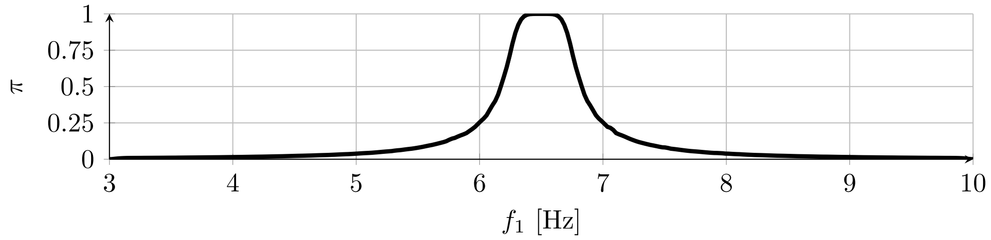


Figure 5: Possibility density $\hat{\pi}_{F_1^{\mathrm{MM}}}$

5.2 STEP 2: Inverse Fuzzy Arithmetic

Using the possibility distributions identified above, inverse fuzzy arithmetic now enables the identification of e.g. a possibility distribution of the Young's modulus of the material or of the Rayleigh damping parameters.

a) Young's Modulus

The GARTEUR testbed may be described by the second-order differential equation

$$\boldsymbol{M}\,\ddot{\boldsymbol{y}} + \boldsymbol{K}(E)\,\boldsymbol{y} = \boldsymbol{0}\,, \tag{9}$$

resulting from the finite-element formulation. Its dependency on the Young's modulus E can formally be expressed as a function ϕ_1 allowing to estimate a possibility distribution for E.

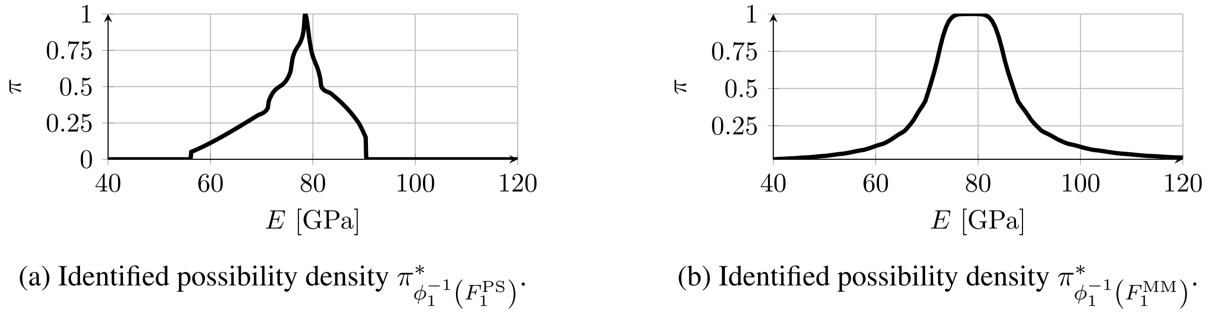
Using the percentage intervals of the first eigenfrequency F_1^{PS} in Figure 4a as the output distribution, one obtains the minimum specific inverse solution $\Pi^*_{\phi_1^{-1}(F_1^{\mathrm{PS}})}$ shown in Figure 6a, yielding the expected value bounds

$$\underline{\mathbb{E}}[E^{\mathrm{PS}}] = 71.24\,\mathrm{GPa} \qquad \text{and} \qquad \overline{\mathbb{E}}[E^{\mathrm{PS}}] = 83.93\,\mathrm{GPa}\,. \tag{10}$$

Using the possibility distribution $\hat{\Pi}_{F_1^{\mathrm{MM}}}$ in Figure 5 as the output distribution, one obtains the minimum specific inverse solution $\Pi^*_{\phi_1^{-1}(F_1^{\mathrm{MM}})}$ shown in Figure 6b and

$$\underline{\mathbb{E}}[E^{\mathrm{MM}}] = 67.91\,\mathrm{GPa} \qquad \text{and} \qquad \overline{\mathbb{E}}[E^{\mathrm{MM}}] = 90.01\,\mathrm{GPa} \tag{11}$$

as the bounds for the expected value. Since the latter bounds are more conservative, one may deduce that the information given by the sample moments is less specific than the information contained in the percentage sets.



(a) Identified possibility density $\pi^*_{\phi_1^{-1}(F_1^{\mathrm{PS}})}$.

(b) Identified possibility density $\pi^*_{\phi_1^{-1}(F_1^{\mathrm{MM}})}$.

Figure 6: Estimation of the possibility density of the Young's modulus by inverse fuzzy arithmetic.

b) Rayleigh Damping

The proposed procedure can also be employed to identify Rayleigh damping, i.e. to estimate the possibility distributions of the parameters α and β which constituting the damping matrix $\boldsymbol{C}(\alpha, \beta) = \alpha\boldsymbol{M} + \beta\boldsymbol{K}$. In this case, a function $\psi_{2,5}$ formally describes the dependency of the eigenfrequencies f_2 and f_5 of the system equations

$$\boldsymbol{M}\,\ddot{\boldsymbol{y}} + \boldsymbol{C}(\alpha,\beta)\,\dot{\boldsymbol{y}} + \boldsymbol{K}\,\boldsymbol{y} = \boldsymbol{0} \tag{12}$$

on the Rayleigh parameters. Applying inverse fuzzy arithmetic to the percentage hyperellipsoids of $\boldsymbol{F}_{2,5}^{\mathrm{PS}}$, the density of the joint possibility distribution $\Pi^*_{\psi_{2,5}^{-1}\left(\boldsymbol{F}_{2,5}^{\mathrm{PS}}\right)}$ of the Rayleigh parameters shown in Figure 7a is obtained. A marginalization yields the univariate densities

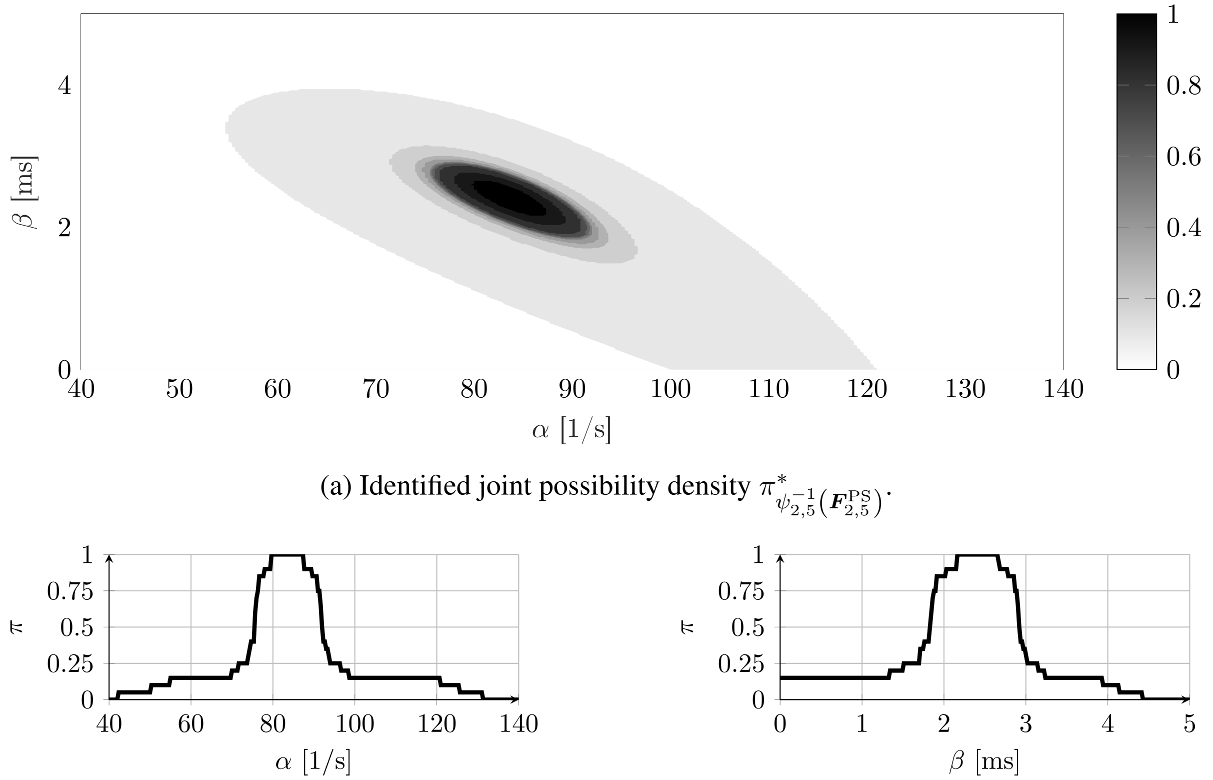
$$\pi^*_\alpha(a) = \sup_{b \geq 0} \pi^*_{\psi_{2,5}^{-1}\left(\boldsymbol{F}_{2,5}^{\mathrm{PS}}\right)}(a,b) \qquad \text{and} \qquad \pi^*_\beta(b) = \sup_{a \geq 0} \pi^*_{\psi_{2,5}^{-1}\left(\boldsymbol{F}_{2,5}^{\mathrm{PS}}\right)}(a,b) \tag{13}$$

shown in Figures 7b and 7c. The bounds on the expected values are given by

$$\underline{\mathbb{E}}[\alpha^{\mathrm{PS}}] = 72\,\frac{1}{\mathrm{s}} \qquad \text{and} \qquad \overline{\mathbb{E}}[\alpha^{\mathrm{PS}}] = 97\,\frac{1}{\mathrm{s}} \tag{14}$$

and

$$\underline{\mathbb{E}}[\beta^{\mathrm{PS}}] = 1.6\,\mathrm{ms} \qquad \text{and} \qquad \overline{\mathbb{E}}[\beta^{\mathrm{PS}}] = 3.1\,\mathrm{ms}\,. \tag{15}$$



(a) Identified joint possibility density $\pi^*_{\psi_{2,5}^{-1}\left(\boldsymbol{F}_{2,5}^{\mathrm{PS}}\right)}$.

(b) Identified marginal possibility density π^*_α.

(c) Identified marginal possibility density π^*_β.

Figure 7: Estimation of the possibility densities of the Rayleigh damping parameters by inverse fuzzy arithmetic.

6 Conclusion

Since the proposed procedure is based on theoretical results about probability-possibility transformations and inverse possibility propagation, it provides a meaningful interpretation of the involved uncertainty descriptions within the framework of imprecise probabilities. It is, furthermore, conceptually simple and straightforward to compute as the application to the

GARTEUR testbed shows, and thus it promises to be a suitable method for a broad range of applications.

The next sensible step is to investigate whether similar results can be proven for non-deterministic process models. In order to do so, a more general inverse fuzzy arithmetic has to be developed, capable of solving fuzzy equations of the form

$$\boldsymbol{Y} = f\left(\boldsymbol{P}, \boldsymbol{X}\right) , \tag{16}$$

where $\boldsymbol{Y}$ and $\boldsymbol{P}$ are considered to be known fuzzy quantities and the distribution of $\boldsymbol{X}$ is sought. This would prove useful when e.g. measurement noise has to be taken into account. Amongst other things, this requires a careful examination of the results presented in [12] within the framework of imprecise probabilities.

Acknowledgements

The authors gratefully acknowledge the support provided by the German Research Foundation (DFG) in the framework of the research project no. 319924547. They would also like to express their sincere appreciation of the work contributed by Andreas Baumann in the framework of his student thesis under supervision of the authors.

REFERENCES

[1] E. T. Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957.

[2] D. Dubois, L. Foulloy, G. Mauris, and H. Prade. Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. *Reliable computing*, 10(4):273–297, 2004.

[3] D. Dubois and H. Prade. *Possibility theory: an approach to computerized processing of uncertainty*. Springer Science & Business Media, 2012.

[4] M.-H. Masson and T. Denœux. Inferring a possibility distribution from empirical data. *Fuzzy Sets and Systems*, 157(3):319–340, 2006.

[5] D. Dubois and H. Prade. Practical methods for constructing possibility distributions. *International Journal of Intelligent Systems*, 31(3):215–239, 2016.

[6] D. Dubois, E. Kerre, R. Mesiar, and H. Prade. Fuzzy interval analysis. In *Fundamentals of fuzzy sets*, pages 483–581. Springer, 2000.

[7] H. Tanaka, H. Ichihashi, and K. Asai. Fuzzy decision in linear programming problems with trapezoid fuzzy parameters. *Management Decision Support Systems Using Fuzzy Sets and Possibility Theory, J. Kacprzyk and R. Yager (Eds), Verlag TÜV Rheinland*, pages 146–155, 1985.

[8] H. Tanaka. Fuzzy data analysis by possibilistic linear models. *Fuzzy Sets and Systems*, 24(3):363–375, 1987.

[9] H. Tanaka, I. Hayashi, and J. Watada. Possibilistic linear regression analysis for fuzzy data. *European Journal of Operational Research*, 40(3):389–396, 1989.

[10] M. Hukuhara. Integration des applications mesurables dont la valeur est un compact convexe. *Funkcial. Ekvac*, 10:205–223, 1967.

[11] L. Stefanini. A generalization of hukuhara difference and division for interval and fuzzy arithmetic. *Fuzzy Sets and Systems*, 161(11):1564–1584, 2010.

[12] W. A. Lodwick and D. Dubois. Interval linear systems as a necessary step in fuzzy linear systems. *Fuzzy Sets and Systems*, 281:227–251, 2015.

[13] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter. *Applied interval analysis: with examples in parameter and state estimation, robust control and robotics*, volume 1. Springer Science & Business Media, 2001.

[14] M. Zimmermann and J. Edler von Hoessle. Computing solution spaces for robust design. *International Journal for Numerical Methods in Engineering*, 94(3):290–307, 2013.

[15] S. Destercke and D. Dubois. A unified view of some representations of imprecise probabilities. In *Soft Methods for Integrated Uncertainty Modelling*, pages 249–257. Springer, 2006.

[16] C. Baudrit and D. Dubois. Practical representations of incomplete probabilistic knowledge. *Computational statistics & data analysis*, 51(1):86–108, 2006.

[17] D. Dubois and H. Prade. When upper probabilities are possibility measures. *Fuzzy Sets and Systems*, 49(1):65–74, 1992.

[18] A. Hanselowski, S. Ihrle, and M. Hanss. A fuzzy model updating technique motivated by bayesian inference. In *Proceedings of the UNCECOMP 2015, 1st ECOMMAS Thematic Conference on Uncertainty Quantification in Computational Sciences and Engineering*, pages 548–559, 2015.

[19] G. Mauris. A review of relationships between possibility and probability representations of uncertainty in measurement. *IEEE Trans. Instrumentation and Measurement*, 62(3):622–632, 2013.

[20] L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning - i. *Information sciences*, 8(3):199–249, 1975.

[21] D. Hose and M. Hanss. On inverse fuzzy arithmetical problems in uncertainty analysis. In *Proceedings of the 7th International Conference on Uncertainties in Structural Dynamics USD 2018*, 2018.

[22] P. Walley. *Statistical reasoning with imprecise probabilities*. Chapman and Hall, 1991.

[23] D. Hose and M. Hanss. Consistent inverse probability and possibility propagation. In *Proceedings of the EUSFLAT 2019 conference (submitted), Prague*, 2019.

[24] D. Dubois. Possibility theory and statistical reasoning. *Computational statistics & data analysis*, 51(1):47–69, 2006.

[25] G. Choquet. Theory of capacities. In *Annales de l'institut Fourier*, volume 5, pages 131–295, 1954.

[26] A. J. Persoon and E. Balmes. A ground vibration test on the garteur testbed sm ag-19. Technical report, Nationaal Lucht-en Ruimtevaartlaboratorium, 1997.

[27] C. Thonon and J.-C. Golinval. Finite element model updating of the garteur sm-ag19 structure. In *International Conference on Structural System Identification*, September 2001.

[28] Y. Govers and M. Link. Model updating using uncertain experimental modal data. In *IFASD 2011 - 15th International Forum on Aeroelasticity and Structural Dynamics*, 2011.

[29] C. Mares, J. E. Mottershead, and M. I. Friswell. Selection and updating of parameters for the garteur sm-ag19 testbed. In *Proceedings of the international seminar on modal analysis*, volume 2, pages 635–640. KU Leuven; 1998, 2001.

[30] I. Boulkaibet, L. Mthembu, T. Marwala, M. I. Friswell, and S. Adhikari. Finite element model updating using the separable shadow hybrid monte carlo technique. In Randall Allemang, editor, *Topics in Modal Analysis II, Volume 8*, pages 267–275, Cham, 2014. Springer International Publishing.

[31] H. Haddad Khodaparast, Y. Govers, I. Dayyani, S. Adhikari, M. Link, M. I. Friswell, J. E. Mottershead, and J. Sienz. Fuzzy finite element model updating of the dlr airmod test structure. *Applied Mathematical Modelling*, 52:512 – 526, 2017.

[32] Y. Govers, H. Haddad Khodaparast, M. Link, and J. E. Mottershead. A comparison of two stochastic model updating methods using the dlr airmod test structure. *Mechanical Systems and Signal Processing*, 52-53:105 – 114, 2015.

[33] A. Gravelle. Ground vibration test techniques. Technical report, GARTEUR (Group for Aeronautical Research and Technology in Europe), April 1999.

[34] M. Degener and M. Hermes. Ground vibration test and finite element analysis of the garteur sm-ag19 testbed. Technical report, 1996.

[35] M. Link and M. Friswell. Working group 1: Generation of validated structural dynamic models - results of a benchmark study utilising the garteur sm-ag19 test-bed. *Mechanical Systems and Signal Processing*, 17(1):9 – 20, 2003.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

ON PROBABILITY-POSSIBILITY CONSISTENCY IN HIGH-DIMENSIONAL PROPAGATION PROBLEMS

Dominik Hose, Markus Mäck and Michael Hanss

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
e-mail: {dominik.hose, markus.maeck, michael.hanss}@itm.uni-stuttgart.de

Keywords: Possibility Theory, Imprecise Probabilities, High Dimensionality, Uncertainty Propagation, Probability-Possibility Transformations, Triangular Fuzzy Numbers.

Abstract. *The concept of consistency between multivariate probability and possibility distributions is of essential importance for solving possibilistic inference problems in large-scale applications where potentially many uncertain variables are involved.*

The transformation of probability distributions to possibility distributions with minimal loss of information has been treated before. For instance, it has been shown that the (maximum specific) possibility distribution of a univariate uniform probability distribution is a triangular fuzzy number. Unfortunately, this result does not directly translate to higher dimensions, a problem which has received little attention among scholars. Yet, the construction of joint possibility distributions in the multivariate case and, consequently, their propagation are mandatory for the interpretation of the possibilistic results in the context of imprecise probabilities.

In this contribution, the consistent aggregation of marginal possibility distributions in a joint distribution is investigated, with the aim of enabling the consistent propagation of uncertainty of a high dimension using possibility theory.

In particular, rather than deriving the joint possibility distribution from consistent marginal distributions, which may result in an inconsistent joint distribution, the consistency of the joint distribution has to be ensured first, and the marginal distributions can be deduced therefrom.

This approach is motivated by the transformation of multivariate uniform probability distributions. It highlights that the often-used triangular shape of the possibility density function, which is optimal in the univariate case, yields an inconsistent distribution in the multivariate case for independent variables. The conclusions are then generalized for arbitrary possibility distributions, resulting in a mathematical program for the construction of multivariate joint possibility distributions.

The main result is that, instead of modeling uncertainty by means of triangular fuzzy numbers, the dimension of the uncertainty space has to be accounted for and the provided alternative which is highly meaningful in the context of imprecise probabilities should be chosen. Additionally, it is shown that for an increasing number of uncertain variables, the resulting possibility densities become less and less specific, and possibilistic analysis degenerates to interval calculus.

1 Introduction

This contribution initially intendeds to provide a concise summary of possibility theory within the framework of imprecise probabilities for engineers in Sections 2 – 6. Supplementary literature is referenced where it is needed. An emphasis is put on the problem of high-dimensional uncertainty propagation of possibility distributions, and some difficulties are pointed out in Section 7, building upon the results of Baudrit and Dubois who are the main pursuers of this line of research with many useful results, e.g. in [1] and [2]. However, they specialize in the problem of the joint propagation of probability and possibility distributions. Here, some novel theoretical approaches regarding the construction of joint possibility distributions from marginal possibility distributions are illustrated from the point of view of imprecise probabilities. In a first step, an optimal transformation of multivariate uniform probability distributions is derived manually in Section 8. Generalizing these results, a mathematical program for obtaining the joint possibility distribution is given in Section 9 and the core of this contribution, an argument for the replacement of triangular fuzzy numbers, is presented. Some final remarks in Section 10 conclude the discussion.

2 Probability and Possibility Spaces

The definitions of a probability space $(\mathcal{X}, \mathcal{S}, \mathrm{P})$, consisting of a sample space $\mathcal{X}$, the σ-algebra $\mathcal{S}$ on $\mathcal{X}$, and a probability measure P, are well-known [3]. The set of all probability distributions on the measurable space $(\mathcal{X}, \mathcal{S})$ is denoted by $\mathbb{P}\left(\mathcal{X}, \mathcal{S}\right)$.

Many scholars (e.g. Destercke, Dubois and Chojnacki in [4]) have argued that in certain scenarios a more general description of uncertainty is necessary in order to avoid having to agree on just one probability distribution.

One alternative description of uncertain bodies of evidence is the theory of possibility which Zadeh proposes [5]. It is based on the possibility measure, a sub-additive Choquet capacity [6], which is a set function $\Pi : 2^{\mathcal{X}} \to [0, 1]$ that satisfies $\Pi\left(\emptyset\right) = 0$ and $\Pi\left(\mathcal{X}\right) = 1$, and for two disjoint sets $U_1, U_2 \subseteq \mathcal{X}$ the possibility of their union is given by $\Pi\left(U_1 \cup U_2\right) = \max\left(\Pi(U_1), \Pi(U_2)\right)$. The introduction of a σ-algebra is not required for possibility measures but it is certainly possible to consider their restrictions $\Pi|_{\mathcal{S}}$ in order to compare them with the respective probability measures.

Finally, analogously to a probability measure inducing a probability density, a possibility measure Π induces a possibility density π and vice versa, satisfying the identities

$$\Pi\left(U\right) = \sup_{x \in U} \pi(x) \quad \forall\, U \in \mathcal{S} \qquad \text{and} \qquad \pi\left(x\right) = \Pi(\{x\}) \quad \forall\, x \in \mathcal{X}\,. \tag{1}$$

3 Possibility as an Imprecise Probability

As stated in [7], the Dempster-Shafer Theory of Evidence [8] can serve as a general framework for descriptions of uncertainty by imprecise probabilities [9]. Therein, the belief mass plays a central role in assigning belief values to subsets of the sample space. If certain restrictions are imposed on this belief mass function, it degenerates to a probability distribution or a possibility distribution demonstrating how closely they are interconnected. A nice review of the different representations of uncertainty and their relation to each other is given in [4].

Therefore, one can attempt to measure their consistency in order to determine how closely a specific possibility and probability distribution are actually connected. Several concepts of a measure of consistency have been proposed, see [10] for a review. Here, the definition by Dubois and Prade [11], viewing a possibility measure as an upper probability measure, is employed. Namely, a probability measure P and a possibility measure Π are called consistent if the probability of all events $U \in \mathcal{S}$ is dominated by its possibility, i.e.

$$\mathrm{P}\,(U) \leq \Pi\,(U)\ . \tag{2}$$

From the upper bound provided by the possibility measure, it follows immediately that the probability measure is also bounded from below by the necessity measure

$$\mathrm{N}\,(U) = 1 - \Pi\,(\mathcal{X} \setminus U) \leq 1 - \mathrm{P}\,(\mathcal{X} \setminus U) = \mathrm{P}\,(U) \qquad \forall\, U \in \mathcal{S}\,. \tag{3}$$

Therefore, necessity and possibility measures may be viewed as upper and lower probabilities [12].

4 Credal Sets

A possibility distribution induces a credal set of consistent probability distributions [13]

$$\mathcal{P}_\Pi = \{\mathrm{P} \in \mathbb{P}\,(\mathcal{X},\mathcal{S}) : \mathrm{P}\,(U) \leq \Pi\,(U)\ \forall\, U \in \mathcal{S}\}\ . \tag{4}$$

Generally, a possibility measure Π' is called more specific than Π'', denoted by $\Pi' \preceq \Pi''$, if for all events $U \in \mathcal{S}$ it holds that $\Pi'(U) \leq \Pi''(U)$. It is easy to see that this implies $\mathcal{P}_{\Pi'} \subseteq \mathcal{P}_{\Pi''}$. More illustrative results about which probability distributions are actually contained in $\mathcal{P}_\Pi$ may be found in [14].

A necessary and sufficient criterion to check consistency between a probability measure P and a possibility measure Π, i.e. $\mathrm{P} \in \mathcal{P}_\Pi$, is to check if Equation (2) is fulfilled only for the sub-level sets

$$S_\Pi^\alpha = \{x \in \mathcal{X} : \pi(x) \leq \alpha\} \tag{5}$$

for all $\alpha \in [0,1]$ instead of all $U \in \mathcal{S}$, see [15], making this verification computationally tractable. This, or course, requires π to be $\mathcal{S}$-measurable, which is usually the case in engineering applications.

5 Probability-Possibility Transformations

Since possibility theory is able to provide insight about the bounds of the true probabilities of an event, it is a broader framework that at the same time handles coarser knowledge and can never be as precise as probability theory. However, it can be favorable to replace a probability distribution on some evidence by a possibility distribution in certain situations, e.g. in order to facilitate further calculations or for the solution of inverse problems [16]. Of course, this possibility distribution should be consistent with the original probability distribution in order to account for it. In [14], Dubois and Prade argue that given a probability measure P with probability density p, an optimal transform is given by the possibility density

$$\pi(x) = 1 - \mathrm{P}\,(\{\zeta \in \mathcal{X} : p(\zeta) \geq p(x)\}) \qquad \forall\, x \in \mathcal{X}\,. \tag{6}$$

Notice that optimal transforms are not unique and other possibilities exist. For instance, the cumulative distribution function $F(x) = \mathrm{P}\,(\{\zeta \in \mathcal{X} : \zeta \leq x\})$ for $x \in \mathcal{X}$ may also serve as a suitable optimal transform depending on the application scenario. For alternative methods for the construction of possibility distributions refer e.g. to [15].

In Figure 1, several optimal transforms of a standard Gaussian distribution are depicted. Therein, π_1 is obtained from Equation (6), π_2 represents the cumulative distribution and π_3 represents the complementary cumulative distribution.

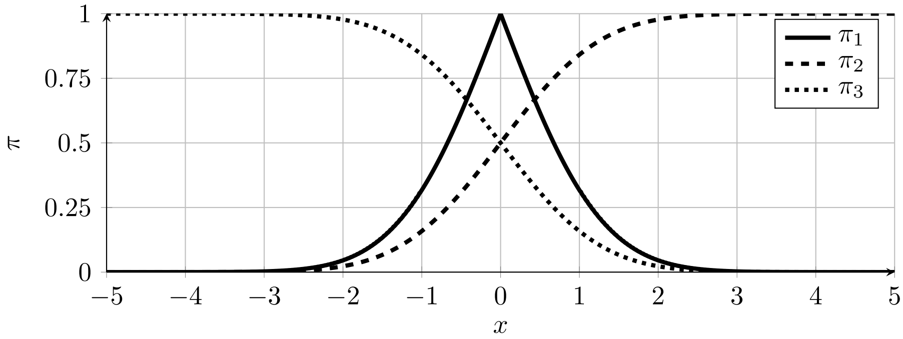


Figure 1: Possibility densities of the optimal transforms of the standard Gaussian distribution.

6 Uncertainty Propagation

The propagation of uncertainties through models represented by a (measurable) function $\varphi : \mathcal{X} \to Y$ mapping from the uncertain input space $\mathcal{X}$ to the uncertain output space $\mathcal{Y}$ is often the main concern in uncertainty quantification. Formally, the pushforward of a possibility or probability measure μ is defined by

$$\mu^{\varphi}(V) = \mu\left(\varphi^{-1}(V)\right) \qquad \forall V \in \mathcal{Y}, \tag{7}$$

where $\mathcal{Y}$ is a σ-algebra on Y and $\varphi^{-1}(V) = \{x \in \mathcal{X} : \varphi(x) \in V\}$. This definition can then by extended to credal sets by defining $\mathcal{P}^{\varphi} = \{\mathrm{P}^{\varphi} : \mathrm{P} \in \mathcal{P}\}$. A well-known result (e.g. Proposition 1 in [17]) states that $\mathcal{P}_{\Pi}^{\varphi} = \mathcal{P}_{\Pi^{\varphi}}$. The forward propagation of consistent measures yields consistent pushforward measures.

Next, the case of inverse uncertainty propagation is investigated. Given a pushforward measure μ^{φ}, only a restriction of the original measure $\mu|_{\mathfrak{X}}$ on the algebra $\mathfrak{X} = \{\varphi^{-1}(V) : V \in \mathcal{Y}\} \subseteq \mathcal{S}$ is known and there exists a (possibly infinite) set of extensions on $\mathcal{S}$

$$\mathcal{I}_{\mu^{\varphi}} = \left\{\mu' \text{ on } \mathcal{S} : (\mu')^{\varphi} = \mu^{\varphi}\right\} \tag{8}$$

yielding this pushforward measure. In the case of possibility measures, the outer extension induced by the possibility density

$$\pi^* : x \in \mathcal{X} \mapsto \pi^{\varphi}\left(\varphi(x)\right) \tag{9}$$

satisfies $\Pi' \preceq \Pi^*$ for all $\Pi' \in \mathcal{I}_{\Pi^{\varphi}}$, i.e. it is the least specific possibility distribution $\Pi^* \in \mathcal{I}_{\Pi^{\varphi}}$. Moreover, for all $\mathrm{P}^{\varphi} \in \mathcal{P}_{\Pi^{\varphi}}$ it follows that $\mathcal{I}_{\mathrm{P}^{\varphi}} \subseteq \mathcal{P}_{\Pi^*}$, as shown e.g. in Theorem 2 in [18].

7 High-Dimensional Consistency

Suppose that two independent $\mathbb{R}$-valued uncertain (stochastic) variables ξ_1 and ξ_2 with given possibility densities π_{ξ_1} and π_{ξ_2} are to be propagated through a model $\varphi : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ in order to compute the possibility distribution of the output variable $\eta = \varphi\left(\xi_1, \xi_2\right)$. Equation (7) then translates to computing

$$\pi_{\eta}(y) = \max_{\xi_1, \xi_2 \in \mathbb{R} : y = \varphi(x_1, x_2)} \pi_{\xi_1, \xi_2}\left(x_1, x_2\right). \tag{10}$$

A reader, unaware of the extensive discussion on possibilistic independence summarized in [19], now may assume that the propagation of possibilistic uncertainties, traditionally known as fuzzy arithmetic [20], could serve as a conservative alternative to probabilistic calculus, i.e. that the joint probability density

$$p_{\xi_1,\xi_2}\left(x_1,x_2\right) = p_{\xi_1}\left(x_1\right) \cdot p_{\xi_2}\left(x_2\right) \qquad \forall\, x_1, x_2 \in \mathbb{R} \tag{11}$$

ought to be consistent with the the joint possibility density

$$\pi^{\text{Zadeh}}_{\xi_1,\xi_2}\left(x_1,x_2\right) = \min\left\{\pi_{\xi_1}\left(x_1\right), \pi_{\xi_2}\left(x_2\right)\right\} \qquad \forall\, x_1, x_2 \in \mathbb{R}\,, \tag{12}$$

from the original formulation of Zadeh's extension principle [21], if $\mathrm{P}_{\xi_1} \in \mathcal{P}_{\Pi_{\xi_1}}$ and $\mathrm{P}_{\xi_2} \in \mathcal{P}_{\Pi_{\xi_2}}$. However, this is not the case as Baudrit et al. demonstrate in Section III.B of [1]. Instead, if one wishes to maintain meaningful results within the framework of imprecise probabilities, the aim is to construct a (in some sense optimal) joint possibility distribution Π_{ξ_1,ξ_2} from the marginal possibilities Π_{ξ_1} and Π_{ξ_2} such that the joint probability distribution P_{ξ_1,ξ_2} of all marginal probabilities $\mathrm{P}_{\xi_1} \in \mathcal{P}_{\Pi_{\xi_1}}$ and $\mathrm{P}_{\xi_2} \in \mathcal{P}_{\Pi_{\xi_2}}$ is contained in $\mathcal{P}_{\Pi_{\xi_1,\xi_2}}$.

This would also enable the joint propagation of any number of marginal possibility distributions by recursively applying this procedure. A sensible sub-problem to be considered first is that of transforming high-dimensional probability distributions into possibility distributions as done in the following section.

8 High-Dimensional Probability-Possibility Transformations

The most commonly used possibility density is of triangular shape with zero density outside its support $[\xi_\mathrm{l}, \xi_\mathrm{u}]$ and a linear increase towards its nominal value $\bar{\xi}$ with density one, refer to Figure 2. A triangular possibility density function represents the optimal transform of a uniform probability density function and bounds all density functions transformed from symmetric probability densities with the same (bounded) support as an upper envelope [22]. However, the extension from the one dimensional case to the multivariate case of N uniformly and independently distributed probability densities cannot be achieved by assuming independent triangular shaped possibilistic densities.

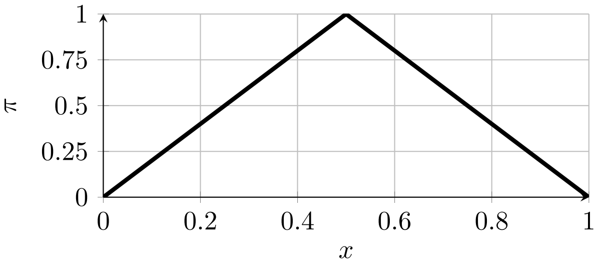


Figure 2: Symmetric triangular possibility density on the support $[0, 1]$.

In the following, the optimal transform of N-dimensional, uniform probability densities is derived. Let $\mathrm{P}_{\xi_1,\ldots,\xi_N}$ be an N-dimensional uniform probability distribution. As explained in Section 4, a consistent possibility measure $\Pi_{\xi_1,\ldots,\xi_N}$ has to satisfy

$$\mathrm{P}_{\xi_1,\ldots,\xi_N}\left(S^{\alpha}_{\Pi_{\xi_1,\ldots,\xi_N}}\right) \leq \Pi_{\xi_1,\ldots,\xi_N}\left(S^{\alpha}_{\Pi_{\xi_1,\ldots,\xi_N}}\right) \leq \alpha \qquad \forall\, \alpha \in [0,1]\,. \tag{13}$$

Defining C^α to be the α-cut given through $C^\alpha = \mathbb{R}^N \backslash S^{\alpha}_{\Pi_{\xi_1,\ldots,\xi_N}}$, its probability can be expressed by

$$\mathrm{P}\left(C^\alpha\right) = 1 - \mathrm{P}_{\xi_1,\ldots,\xi_N}\left(S^{\alpha}_{\Pi_{\xi_1,\ldots,\xi_N}}\right) \overset{\text{Eq. (13)}}{\geq} 1 - \alpha \tag{14}$$

and by

$$\mathrm{P}\left(C^{\alpha}\right)=\int_{C^{\alpha}}\underbrace{p_{\xi_1,\ldots,\xi_N}(\boldsymbol{x})}_{\varrho=\text{const.}}\mathrm{d}\boldsymbol{x}=\varrho\underbrace{\int_{C^{\alpha}}\mathrm{d}\boldsymbol{x}}=\varrho V\left(C^{\alpha}\right)\,. \tag{15}$$

Consequently, the volume of the α-cut is bounded by $V(C^{\alpha})\geq\frac{1-\alpha}{\varrho}$, where the constant factor ϱ is determined from the normalization condition $\int_{\mathbb{R}^N}p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}=1$. For independent and symmetric possibility density functions, it holds that

$$C^{\alpha}=\left\{\boldsymbol{x}\in\mathbb{R}^N:||\bar{\boldsymbol{\xi}}-\boldsymbol{x}||_{\infty}\leq R(\alpha)\right\} \tag{16}$$

where $\bar{\boldsymbol{\xi}}$ is the center of C^{α}, i.e. the N-dimensional nominal vector, and $R:[0,1]\rightarrow\mathbb{R}_0^+$ is a monotonously decreasing positive function. Hence, the volume of C^{α} may also be expressed as $V\left(C^{\alpha}\right)=\left(2R(\alpha)\right)^N$ and consequently

$$R(\alpha)\geq\frac{1}{2}\left(\frac{1-\alpha}{\varrho}\right)^{\frac{1}{N}}\,. \tag{17}$$

Choosing the most specific and consistent marginal possibility density function corresponds to replacing the inequality by an equality, and therefore

$$\pi_{\xi_i}(x_i)=1-\varrho\left(2|\bar{\xi}_i-x_i|\right)^N\qquad\forall\,i=1,\ldots,N\text{ and }x_i\in\mathbb{R}\,. \tag{18}$$

This means that the actual shape of the marginal densities depends on the dimension and is given by the expression in Equation (18). Triangular fuzzy numbers are just a special case of this result for $N=1$.

In the two dimensional case for $\xi_i\in[0,1]$, the optimal, consistent and symmetric marginal possibility densities are given by $\pi_{\xi_i}(x_i)=1-4\left(x_i-\bar{\xi}_i\right)^2$, $i=1,2$, and are shown, together with the resulting joint density function, in Figure 3.

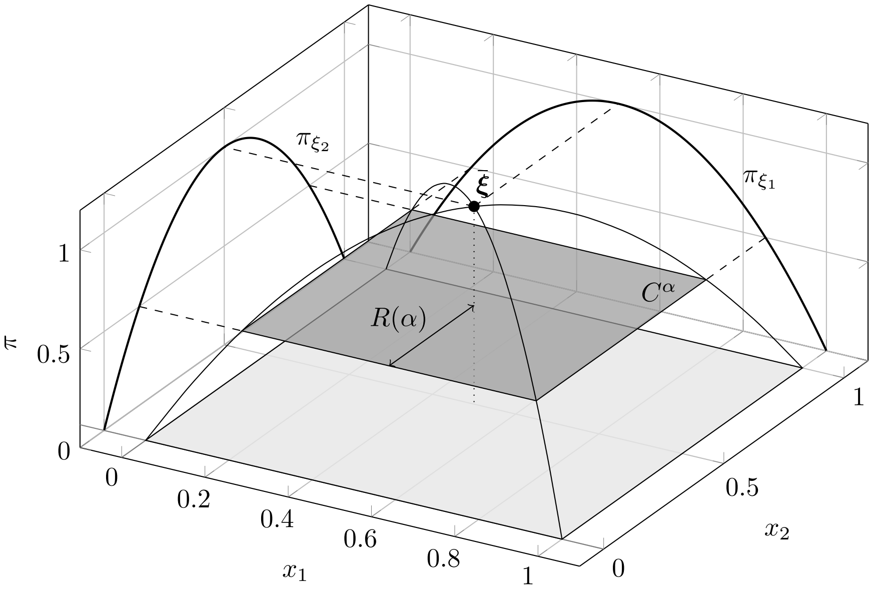


Figure 3: Maximum specific possibility density of a two dimensional uniform probability density.

9 Computation of Joint Possibility Distributions

The considerations in the previous section may be generalized in the following way: Given N uncertain $\mathbb{R}$-valued variables $\xi_1, \dots, \xi_N$ with marginal possibility distributions $\Pi_{\xi_1}, \dots, \Pi_{\xi_N}$, a maximally specific joint possibility distribution $\Pi_{\xi_1,\dots,\xi_N}$ gathering all joint probability distributions $\mathrm{P}_{\xi_1,\dots,\xi_N}$ of the independent marginal probability distributions $\mathrm{P}_{\xi_1} \in \mathcal{P}_{\Pi_{\xi_1}}, \dots, \mathrm{P}_{\xi_N} \in \mathcal{P}_{\Pi_{\xi_N}}$ is the solution of the mathematical program

$$\begin{aligned} \pi_{\xi_1,\dots,\xi_N}(\boldsymbol{x}) = \quad \max \quad & \mathrm{P}_{\xi_1,\dots,\xi_N}\left(\{\boldsymbol{\zeta} \in \mathbb{R}^N : ||\boldsymbol{\zeta} - \boldsymbol{c}||_\infty \geq ||\boldsymbol{x} - \boldsymbol{c}||_\infty\}\right) \\ \text{s.t.} \quad & \mathrm{P}_{\xi_i} \in \mathcal{P}_{\Pi_{\xi_i}} \qquad \text{for } i = 1, \dots, N \end{aligned} \tag{19}$$

for all $\boldsymbol{x} \in \mathbb{R}^N$. The center point $\boldsymbol{c} \in \mathbb{R}^N$ will be the nominal vector of the resulting joint possibility density and can be chosen arbitrarily. However, it is recommendable to choose the nominal values of the marginal possibility distributions such that $\boldsymbol{c} = \bar{\boldsymbol{\xi}}$.

Once again, consider the case of two uncertain variables ξ_1 and ξ_2 with triangular possibility densities $\pi_\xi(x) = \pi_{\xi_1}(x) = \pi_{\xi_2}(x) = 1 - 2\left|x - \frac{1}{2}\right|$ for $x \in [0, 1]$, as shown in Figure 2. For $\boldsymbol{x} \in [0, 1]^2$, denote $r = ||\boldsymbol{x} - \boldsymbol{c}||_\infty$ and define $U_r = \{\boldsymbol{\zeta} \in \mathbb{R}^2 : ||\boldsymbol{\zeta} - \boldsymbol{c}||_\infty \geq r\}$, where $\boldsymbol{c} = \left(\frac{1}{2}, \frac{1}{2}\right)^{\mathrm{T}}$. Then, Equation (19) simplifies to

$$\begin{aligned} \pi_{\xi_1,\xi_2}(\boldsymbol{x}) = \quad \max \quad & 1 - \mathrm{P}_{\xi_1,\xi_2}(U_r) \\ \text{s.t.} \quad & \mathrm{P}_{\xi_i} \in \mathcal{P}_{\Pi_{\xi_i}} \qquad i = 1, 2\,. \end{aligned} \tag{20}$$

Notice, that U_r may be decomposed into $U_r = I_r^2$ with $I_r = [\frac{1}{2} - r, \frac{1}{2} + r]$. Since P_{ξ_1} and P_{ξ_2} are considered independent, one can then bound $\mathrm{P}_{\xi_1,\xi_2}(U_r)$ from below via

$$\mathrm{P}_{\xi_1,\xi_2}(U_r) = \mathrm{P}_{\xi_1}(I_r) \cdot \mathrm{P}_{\xi_2}(I_r) \geq \mathrm{N}_{\xi_1}(I_r) \cdot \mathrm{N}_{\xi_2}(I_r) = (1 - \underbrace{\max_{x \notin I_r} \pi_\xi(x)}_{1-2r})^2 = 4r^2 \tag{21}$$

and thus $\pi_{\xi_1,\xi_2}(\boldsymbol{x}) = 1 - 4r^2$ which is the same result as above and shown in Figure 3. Naturally, this derivation extends to higher dimensions and the implications are manifold.

Most importantly, since the univariate triangular possibility density bounds all density functions transformed from symmetric probability densities with the same support, the joint possibility distribution obtained here bounds all their independent combinations.

The main conclusion is, thus, that if a possibilistic uncertainty analysis is performed with more than one uncertain parameter, instead of assuming triangular possibility densities, they should be chosen according to Equation (18).

A second implication is that with an increasing dimension N the optimal possibility densities become less and less specific and converge to unit function over the support. Thus, for many uncertain parameters, this becomes an argument to switching from a possibilistic uncertainty assessment to interval calculus.

Of course, Equation (19) may be employed to construct an even larger variety of joint possibility distributions from marginal ones. Propagating the resulting joint possibility density through any model will yield a possibility distribution that accounts for all possible combinations of probability distributions that could have arisen given the marginal possibility distributions. This may, of course, also be accomplished by propagating the marginal densities obtained by the marginalization

$$\tilde{\pi}_{\xi_i}(x_i) = \max_{x_1,\dots,x_{i-1},x_{i+1},\dots,x_N \in \mathbb{R}} \pi_{\xi_1,\dots,\xi_N}(x_1, \dots, x_N) \qquad \forall\, x_i \in \mathbb{R} \tag{22}$$

according to the original formulation of Zadeh's extension principle. Some examples of these marginal densities for varying values of N are shown in Figure 4.

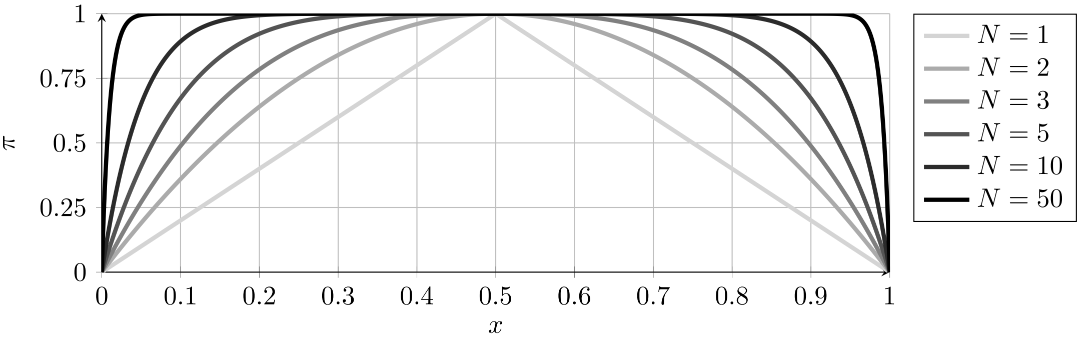


Figure 4: Marginal densities $\tilde{\pi}_{\xi_i}$ of the joint distribution of N triangular and symmetric possibility densities on the support $[0, 1]$.

10 Conclusions

Possibility theory can provide useful solutions for the problem of considering polymorphic uncertainties in many types of models. However, when the results ought to be meaningful within the framework of imprecise probabilities, special care must be taken in order to compute with the correct distributions. In this contribution, the authors hope to have highlighted some possible pitfalls and ways to avoid them.

The main takeaway message is that when computing in uncertainty spaces of higher order, triangular fuzzy numbers ought to be replaced by the expression in Equation (18).

In its spatially discretized form, Equation (19) is an optimization problem with linear equality and inequality constraints and a bilinear objective function, and there should exist adequate methods for exploiting its specific structure. The authors are planning further investigations regarding computationally efficient solution algorithms and potentially better-suited basis representations of the probability distributions.

Acknowledgements

The authors gratefully acknowledge the support provided by the German Research Foundation (DFG) in the framework of the research project no. 319924547 and in the framework of the priority programme 1886 “Polymorphic Uncertainty Modelling for the Numerical Design of Structures”.

REFERENCES

[1] Cédric Baudrit, Didier Dubois, and Dominique Guyonnet. Joint propagation and exploitation of probabilistic and possibilistic information in risk assessment. *IEEE transactions on fuzzy systems*, 14(5):593–608, 2006.

[2] Cédric Baudrit, Inés Couso, Didier Dubois, et al. Joint propagation of probability and possibility in risk analysis: Towards a formal framework. *International Journal of Approximate Reasoning*, 45(1):82–105, 2007.

[3] Willliam Feller. *An introduction to probability theory and its applications*, volume 2. John Wiley & Sons, 2008.

[4] Sébastien Destercke, Didier Dubois, and Eric Chojnacki. Unifying practical uncertainty representations–i: Generalized p-boxes. *International Journal of Approximate Reasoning*, 49(3):649 – 663, 2008.

[5] Lotfi A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 100:9–34, 1999.

[6] Gustave Choquet. Theory of capacities. In *Annales de l'institut Fourier*, volume 5, pages 131–295, 1954.

[7] George Klir and Mark Wierman. *Uncertainty-based information: elements of generalized information theory*, volume 15. Springer Science & Business Media, 1999.

[8] Arthur P. Dempster. Upper and lower probabilities induced by a multivalued mapping. In *Classic Works of the Dempster-Shafer Theory of Belief Functions*, pages 57–72. Springer, 2008.

[9] Peter Walley. *Statistical reasoning with imprecise probabilities*. Chapman and Hall, 1991.

[10] Miguel Delgado and Serafın Moral. On the concept of possibility-probability consistency. *Fuzzy Sets and Systems*, 21(3):311–318, 1987.

[11] Didier Dubois, Henri Prade, and Sandra Sandri. On possibility/probability transformations. In *Fuzzy logic*, pages 103–112. Springer, 1993.

[12] Didier Dubois and Henri Prade. When upper probabilities are possibility measures. *Fuzzy Sets and Systems*, 49(1):65–74, 1992.

[13] Didier Dubois. *Fuzzy information and decision processes*, chapter On Several Presentations of uncertain body of evidence, pages 167–189. North-Holland, Amsterdam, 1982.

[14] Cédric Baudrit and Didier Dubois. Practical representations of incomplete probabilistic knowledge. *Computational statistics & data analysis*, 51(1):86–108, 2006.

[15] Didier Dubois and Henri Prade. Practical methods for constructing possibility distributions. *International Journal of Intelligent Systems*, 31(3):215–239, 2016.

[16] Dominik Hose and Michael Hanss. Towards a general theory for data-based possibilistic parameter inference. In *Proceedings of the 3rd International Conference on Uncertainty Quantification in Computational Sciences and Engineering, Crete (Greece)*, 2019.

[17] Andrey Bronevich and George J. Klir. Measures of uncertainty for imprecise probabilities: An axiomatic approach. *International Journal of Approximate Reasoning*, 51(4):365 – 390, 2010.

[18] Dominik Hose and Michael Hanss. Consistent inverse probability and possibility propagation. In *Proceedings of the EUSFLAT 2019 conference (submitted), Prague (Czech Republic)*, 2019.

[19] Gert De Cooman. Possibility theory ii: Conditional possibility. *International Journal Of General System*, 25(4):325–351, 1997.

[20] Michael Hanss. *Applied fuzzy arithmetic - an introduction with engineering applications*. Springer, Berlin, 2010.

[21] Lotfi A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning-i. 8(3):199–249.

[22] Didier Dubois, Laurent Foulloy, Gilles Mauris, and Henri Prade. Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. *Reliable computing*, 10(4):273–297, 2004.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

IMPRECISE RANDOM FIELD ANALYSIS FOR TRANSIENT DYNAMICS

Matthias Faes [1] and David Moens [2]

[1]KU Leuven - Department of Mechanical Engineering
Jan De Nayerlaan 5 - 2860 St.-Katelijne-Waver
e-mail: matthias.faes [at] kuleuven.be

Keywords: Imprecise probability, random field, p-box

Abstract. *The objective modeling of (spatio-)temporal randomness following a random field approach usually requires data with a high stochastic and (spatio-)temporal resolution. The framework of imprecise probabilities has been shown to alleviate this burden of data by explicitly acknowledging epistemic uncertainty that originates e.g., from such lack of data, in the analysis. However, work on imprecise random fields is only being performed very recently, and up until now, only epistemic uncertainty in the definition of the first two statistical moments of the random field is considered. This paper presents an approach to account for imprecision in the first two statistical moments, but also in the covariance structure of the random field. An efficient approach for application in linear transient dynamics applications is presented and applied to the study of the dynamics of a car suspension subjected to an imprecisely known road profile. It is shown that the presented approach indeed is capable of provinding an analyst with the bounds on some comfort indicators of the car model, at greatly reduced computational cost when compared to brute-force approaches.*

1 INTRODUCTION

In the context of including non-determinism into numerical models, usually either a probabilistic or a possibilistic (interval/fuzzy) approach is followed, and recent work has been dedicated to the comparison of both philosophies in a forward [9] and inverse setting [6]. Some specific considerations have to be made in case multivariate (spatial) uncertain parameters is considered. The interval framework, while highly objective under scarce data, is less suited for the description of such multivariate non-deterministic quantities, as intervals are by definition independent. Therefore, application of a classical interval framework will yield in this context over-conservative results. Methods to cope with dependence in a multivariate interval or fuzzy context where only introduced very recently [16, 11, 12, 7, 8]. On the other hand, when sufficient data are available, the probabilistic framework is highly suited for the description of multivariate uncertain non-deterministic quantities, e.g. following a random fields approach [14]. However, data with high spatial and stochastic resolution are usually necessary to construct an objective random field description [3].

As a remediation for the strict requirements on the data that are necessary to accurately represent quantities in the probabilistic framework, the concept of imprecise probabilities is gaining more and more traction [1]. Following an imprecise probabilistic framework, the analyst acknowledges existing epistemic uncertainty in key attributes of the probabilistic quantities under consideration, rather than assuming a certain crisp value. In practice, this is usually obtained by assigning intervals to the statistical moments of a family of distributions belonging to a predefined credal set [17]. In the context of imprecise random field analysis, Verhaeghe et al. [15] where the first to study the effect of computing with interval-valued correlation lengths in a random field with exponential covariance kernel. Similarly, Dannert et al. [5] recently introduced a p-box framework for the propagation of imprecise random fields with interval-valued correlation length where they select samples from the correlation length interval a priori. Gao et al. [10] also recently proposed an efficient sampling approach to cope with impreciseness in the first two central moments of a random field analysis to determine bounds on the reliability of structural components under mixtures of stochastic and non-stochastic system inputs. This paper is concerned with the analysis of imprecise random fields, where imprecision is present in both the central moments of the random field, as well as in the definition of the underlying covariance of the field.

2 RANDOM FIELD ANALYSIS

In a probabilistic context, model parameters $\boldsymbol{x}(\mathbf{r})$ that are subjected to spatial variability are modelled as a random field $x(\mathbf{r}, \theta)$. Such a random field $x(\mathbf{r}, \theta)$ describes a set of correlated random variables $x(\theta)$, assigned to each location $\mathbf{r} \in \Omega$ in the continuous model domain $\Omega \subset \mathbb{R}^d$ with dimension $d \in \mathbb{N}$. Each such a random variable $x(\theta)$ provides a mapping $x : (\Theta, \sigma, P) \mapsto \mathbb{R}$ with $\theta \in \Theta$ a coordinate in sample space Θ and σ the sigma-algebra. For a given event θ_i, $x(\mathbf{r}, \theta_i)$ is a realisation of the random field. A random field is considered Gaussian if the distribution of $(x(\mathbf{r}_1, \theta), x(\mathbf{r}_2, \theta), \ldots, x(\mathbf{r}_n, \theta))$ is jointly Gaussian $\forall \mathbf{r} \in \Omega$. In this case, $x(\mathbf{r}, \theta)$ is completely described by its mean function $\mu_x(\boldsymbol{r}) : \Omega \mapsto \mathbb{R}$ and its auto-covariance function $\boldsymbol{\Gamma}_x(\mathbf{r}, \mathbf{r'}) : \Omega \times \Omega \mapsto \mathbb{R}$. Commonly, (squared) exponential or Matérn covariance functions are applied [4].

In an engineering context, the application of random fields for the modelling of spatial non-deterministic material quantities requires a discretisation of $\boldsymbol{x}(\mathbf{r})$ over Ω. Specifically, this means that the continuous random field $x(\mathbf{r}, \theta)$ is represented by a finite set of $M \in \mathbb{N}^+$

correlated random variables $\zeta_i, i = 1, \ldots, M$, as well as a set of deterministic functions that describe the spatial behaviour of the field. Usually, such discretisation is obtained following a Karhunen-Loève (KL) series expansion [13]. At the core of the method lies a spectral decomposition of a continuous, bounded, symmetric and positive definite auto-covariance function $\mathbf{\Gamma}_x(\mathbf{r}, \mathbf{r'}) : \Omega \times \Omega \mapsto \mathbb{R}$ following Mercer's theorem:

$$\mathbf{\Gamma}_x(\mathbf{r}, \mathbf{r'}) = \sum_{i=1}^{\infty} \lambda_i \boldsymbol{\psi}_i(\mathbf{r}) \boldsymbol{\psi}_i(\mathbf{r}') \tag{1}$$

where $\lambda_i \in [0, \infty)$ and $\boldsymbol{\psi}_i(\mathbf{r}) : \Omega \mapsto \mathbb{R}$ are respectively the eigenvalues and eigenfunctions of $\mathbf{\Gamma}_x(\mathbf{r}, \mathbf{r'})$. These are in practice obtained by solving the homogeneous Fredholm integral equation of the second kind:

$$\int\limits_{\Omega} \mathbf{\Gamma}_x(\mathbf{r}, \mathbf{r}') \boldsymbol{\psi}_i(\mathbf{r}') d\mathbf{r}' = \lambda_i \boldsymbol{\psi}_i(\mathbf{r}) \tag{2}$$

Since $\mathbf{\Gamma}_x(\mathbf{r}, \mathbf{r}')$ is bounded, symmetric and positive definite, these eigenvalues λ_i are non-negative and the eigenfunctions $\boldsymbol{\psi}_i(\mathbf{r})$ satisfy an orthogonality condition such that they form a complete orthogonal basis on $\mathcal{L}_2$. In this case, the series expansion in eq. (1) is convergent [13] and the random field can be expressed as a series expansion:

$$x(\mathbf{r}, \theta) = \mu_x(\mathbf{r}) + \sigma_x \sum_{i=1}^{\infty} \sqrt{\lambda_i} \boldsymbol{\psi}_i(\mathbf{r}) \xi_i(\theta) \tag{3}$$

with σ_x the variance of the random field (in case $\mathbf{\Gamma}_x(\mathbf{r}, \mathbf{r'}) : \Omega \times \Omega \mapsto [0, 1]$) and $\xi_i(\theta), i = 1, \ldots, \infty$ standard uncorrelated random variables, which can be shown to be independent standard normally distributed in the case of a Gaussian random field. In case the field is non-Gaussian, the joint distribution of $\xi_i(\theta)$ is very hard to obtain.

To limit the computational cost, the series expansion in eq. (3) is usually truncated by retaining only the $m \in \mathbb{N}$ largest eigenvalues and corresponding eigenfunctions of $\mathbf{\Gamma}_x(\mathbf{r}_i, \mathbf{r}_j)$ [2]. A closed form solution for the Fredholm integral equation presented in eq. (2) exists only for very simple domains and Gaussian random fields. Therefore, it is usually approximated via numerical methods such as numerical integration via Nystrom's method or Galerkin projection to find a finite dimensional representation of the continuous basis functions. For a recent overview on such numerical procedures, the reader is referred to [2].

3 Imprecise random field analysis

In case a Gaussian random field $x(\mathbf{r}, \theta)$ with a auto-covariance function $\mathbf{\Gamma}_{\boldsymbol{x}}(L)$ with $L \in \mathbb{R}^+$ the correlation length, is considered over the domain Ω, it is fully described by the triplet $(\boldsymbol{\mu}_x, \sigma, L)$. However, in engineering practice, it is often difficult or even intractable to objectively provide a crisp estimate for these quantities, leading to often subjective estimates to obtain a random field description of the phenomenon under consideration. Especially given the importance of the correlation length on both the numerical and statistical aspects of the random field simulation, such approach is not desirable. In the context of a random field $x(\mathbf{r}, \theta)$, given epistemic uncertainty on (some of) its hyper-parameters (mean, variance and correlation length), the field becomes an imprecise random field $[x](\mathbf{r}, \theta)$. The KL expansion of an imprecise random field in this case becomes:

$$[x](\mathbf{r}, \theta) = \mu_x^I(\mathbf{r}) + \sigma_x^I \sum_{i=1}^{\infty} \sqrt{\lambda_i^I} \boldsymbol{\psi}_i^I(\mathbf{r}) \xi_i(\theta) \tag{4}$$

with $\lambda_i^I \in \mathbb{IR}$ interval-valued eigenvalues and $\boldsymbol{\psi}_i^I(\mathbf{r}) : \Omega \mapsto \mathbb{IR}$ interval fields representing the bounds on the corresponding eigenfunctions. It can therefore be understood that an imprecise random field describes a set of correlated P-boxes $[x](\theta)$ for each location in the model domain. Similarly, for a given θ_i, also realizations $[x](\mathbf{r}, \theta_i)$ are generated, which are interval field valued:

$$[x](\mathbf{r}, \theta_i) = \mu_x^I(\mathbf{r}) + \sigma_x^I \sum_{i=1}^{\infty} \sqrt{\lambda_i^I} \boldsymbol{\psi}_i^I(\mathbf{r}) \xi_i(\theta_i) \tag{5}$$

It should be noted that, in case $\mathcal{F}$ extends towards more than Gaussian random fields, the same considerations concerning the correlation and dependence in ξ_i as made for regular random fields have to be made.

4 PROPAGATION OF IMPRECISE RANDOM FIELDS IN TRANSIENT DYNAMICS

We consider the case of a transient dynamic problem, which is governed by the dynamic equation:

$$M\ddot{X}(t) + C\dot{X}(t) + KX(t) = F(t) \tag{6}$$

with $\ddot{\bullet}$ and $\dot{\bullet}$ representing respectively the second and first time derivative of $\bullet$ and $M \in \mathbb{R}^{n_{dof} \times n_{dof}}, C \in \mathbb{R}^{n_{dof} \times n_{dof}}$ and $K \in \mathbb{R}^{n_{dof} \times n_{dof}}$ respectively the mass, damping and stiffness matrices of the system under consideration. $X \in \mathbb{R}^{n_{dof}}$ is the solution of this ODE and represents a vector of displacements. In case the system is discretized by a finite element model, the terms in X represent the nodal displacements.

Let $H(t)$ denote the impulse response function of the system at a certain time instant τ. When the force excitation $F(t)$ is discretized into n_t time steps Δt, the response $x(t_j)$ at a time instant $t_j,\ j = 1, \ldots, n_t$ is given by:

$$X(t_j) = \sum_{i=1}^{j} F(t_i) H(t_j - t_i) \Delta t \tag{7}$$

In the limit case where $\lim_{\Delta t \to 0}$, the problem reduces to the solution of the following convolution integral:

$$X(t) = \int_0^t F(\tau) H(t - \tau) d\tau \tag{8}$$

Hence, in case $H(t-\tau)$ is a monotonic function of t, $X(t)$ is a monotonic function as well with respect to $F(t)$. In this case it is sufficient to propagate only those values in $\mathcal{H} = \{\mu, \sigma, L\}$ that bound the eigenfunctions $\sqrt{\lambda_i}\psi_i(\mathbf{r}$ of the imprecise random field.

Let $G(\Omega, L) : \Omega \times L \mapsto \{\boldsymbol{\lambda}, \boldsymbol{\psi}(\mathbf{r})\}$ denote the process of solving eq. (2) for m eigenpairs of $\boldsymbol{\Gamma}_x$ given a crisp value for L (e.g., following Galerkin or Nyström procedures), The main idea is to apply a global optimization scheme to determine those values for L that yield extreme values in $\sqrt{\lambda_i}\boldsymbol{\psi}_i(\mathbf{r})$:

$$\overline{L}_i^* = \arg\max_{G(L)} ||\sqrt{\lambda_i}\psi_i(\mathbf{r})||_2, \qquad \text{s.t. } L \in L^I \tag{9a}$$

$$\underline{L}_i^* = \arg\min_{G(L)} ||\sqrt{\lambda_i}\psi_i(\mathbf{r})||_2, \qquad \text{s.t. } L \in L^I \tag{9b}$$

with $i = 1, \ldots, m$. The underlying idea to look for those L that correspond to extrema in the $\mathcal{L}_2$ norm of the basis function in each mode of the random field is that as such, a complete bounding set is obtained. Furthermore, due to the differentiability of the $\mathcal{L}_2$ norm, this is a smooth, convex, non-linear optimization problem in limited dimension. Therefore, any sequential quadratic programming approach can be followed to obtain the bounds without excessive computational overhead. Note that the problem is not necessarily convex. As such, it is advised to try different randomized initial estimates.

The maximally $2m$ solutions are then concatenated in a single vertex set $\mathcal{L}$:

$$\mathcal{L} = \left\{ \underline{L}_1^*, \overline{L}_1^*, \underline{L}_2^*, \overline{L}_2^*, \dots \underline{L}_M^*, \overline{L}_M^*, \right\} \tag{10}$$

and the eigen pairs $\boldsymbol{\lambda}$, $\boldsymbol{\psi}(\mathbf{r})$ are computed using (2) for each of these $L \in \mathcal{L}$. In this way, a set of complete orthogonal bases with corresponding scale factors is obtained that bound the possible variation in the imprecise random field basis, given the interval uncertainty on the correlation length. It should be noted that due to the smoothness of the decay of the eigenvalues of $\boldsymbol{\Gamma}$, the cardinality $\mathfrak{C}(\mathcal{L})$ of $\mathcal{L}$ will be considerably smaller than $2m$ in practice.

5 CASE STUDY: VEHICLE SUSPENSION COMFORT ESTIMATION

The second case study is concerned with assessing the bounds on the comfort of a vehicle suspension, given an imprecise random field description of the road profile. Hereto, a quarter-car model is applied to model the car dynamics. Also this system can be regarded as a linear transient dynamic system of the form shown in eq. (6). For this specific case, a state-space model is employed:

$$\frac{d}{dt}\begin{bmatrix} x_{us}-x_0 \\ \dot{x}_{us} \\ x_s - x_{us} \\ \dot{x}_s \end{bmatrix} = A \begin{bmatrix} x_{us}-x_0 \\ \dot{x}_{us} \\ x_s - x_{us} \\ \dot{x}_s \end{bmatrix} + \begin{bmatrix} -1 \\ \frac{4c_t}{m_{us}} \\ 0 \\ 0 \end{bmatrix} \dot{x}_0 \tag{11}$$

with x_{us} the displacement of the unsprung mass, x_s the displacement of the sprung mass, $\dot{\bullet}$ the time derivative of $\bullet$, m_{us} and m_s the unsprung and sprung mass of a quarter of the car, c_s and c_t respectively the damping coefficients of the suspension and tire, k_s and k_t respectively the stiffness coefficients of the suspension and tire and the matrix A equal to:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-4k_t}{m_{us}} & \frac{-4(c_s+c_t)}{m_{us}} & \frac{4k_s}{m_{us}} & \frac{4c_s}{m_{us}} \\ 0 & -1 & 0 & 1 \\ 0 & \frac{4c_s}{m_s} & \frac{-4k_s}{m_s} & \frac{-4c_s}{m_s} \end{bmatrix} \tag{12}$$

The system is excited at the basis, with x_0 modelling the vertical displacement of the tire. The complete road profile is denoted $x_0(t)$. Four state variables are considered, being respectively the tire deflection; the unsprung mass velocity; the suspension stroke, and sprung mass velocity. Typically, in the context of assessing the comfort of a car, two parameters are of interest: the suspension stroke (i.e., the relative displacement of the car body with respect to the unsprung mass) and the acceleration of the sprung mass (car body).

In this example, the suspension of the car is tuned for performance. The parameters of the state-space model are listed in table 1. The dynamics of the car are simulated over a distance of $100\ m$, when the car is travelling at a speed of $10\ m/s$. The one dimensional spatial domain is discretized into 200 equidistant points and the time domain is discretized into time intervals of $0.005\ s$

Table 1: Parameters of the quarter car state-space model

Parameter	Value
m_s	325 kg
m_{us}	65 kg
$c_s + c_t$	1898 N.s/m
k_t	2325 N/m
k_s	505 N/m

The uncertain road profile is modelled as a zero-mean imprecise Gaussian random field with exponential covariance kernel. Imprecision is present in the variance of the field, as well as in the correlation length of the covariance kernel. The former corresponds in this case to to the height of road roughness values, whereas the latter corresponds to their spatial frequency. Specifically, the intervals are set as $\sigma^I = [0.0015; 0.003]\ m$ and $L^I = [2; 15]\ m$.

A solution to the optimization problem introduced in eq. (9) indicates that a set $\mathcal{L}$ with cardinality of 16 is necessary to capture all spatial variation. This is a direct result from the comparably large interval on the correlation length. As such, 32 vertex combinations are needed to propagate the epistemic uncertainty in the imprecise random field. The bounds on the first four basis functions are shown in figure 1.

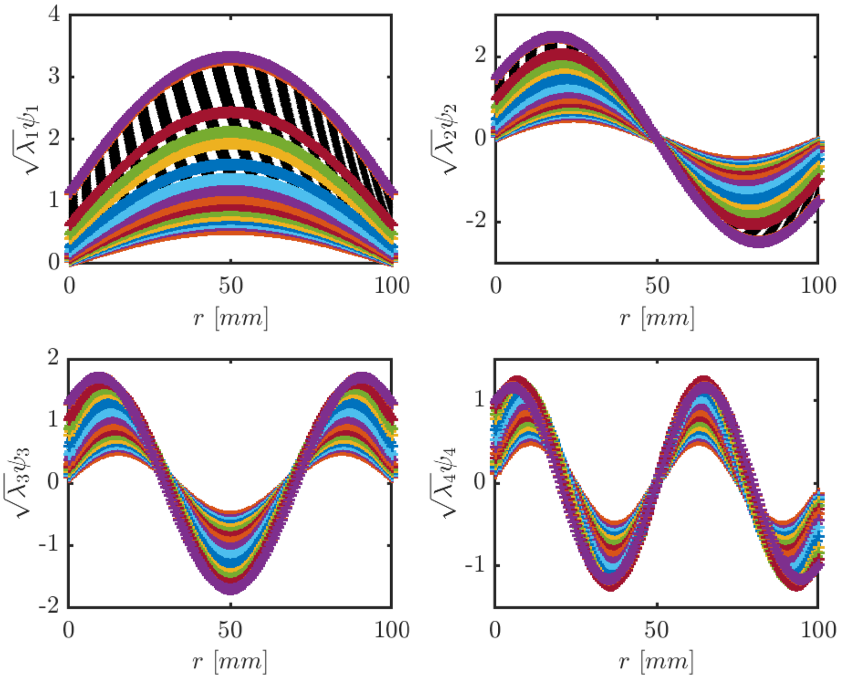


Figure 1: Bounds on the basis functions of the imprecise Gaussian random field with exponential covariance kernel

The stochastic propagation is performed by means of Monte Carlo simulation with 1000 samples. The results of this propagation are compared to a simulation where the epistemic uncertainty is propagated using a Sobol set consisting of 500 samples in between the intervals on σ^I and L^I. Figure 2 illustrates those realisations of the sobol set in the hyper-parameters that yield an extremum in the profile of the relative displacement between the sprung and unsprung mass during the time interval $[0,2]$ s, as well as the extreme bounds $[\underline{\Delta x}(t);\overline{\Delta x}(t)]$ that are predicted by taking:

$$\underline{\Delta x}(t) = \min \Delta x(t_i \mid \mathcal{V}) \qquad \forall t_i \in [0,2]\ s \tag{13a}$$
$$\overline{\Delta x}(t) = \max \Delta x(t_i \mid \mathcal{V}) \qquad \forall t_i \in [0,2]\ s \tag{13b}$$

with $\mathcal{V}$ denoting the vertex set of hyper-parameters that bound the basis functions of the imprecise random field. As can be noted from this figure, the bounds on the possible displacement profiles are captured perfectly by the bounding of the basis functions at strongly reduced computational cost.

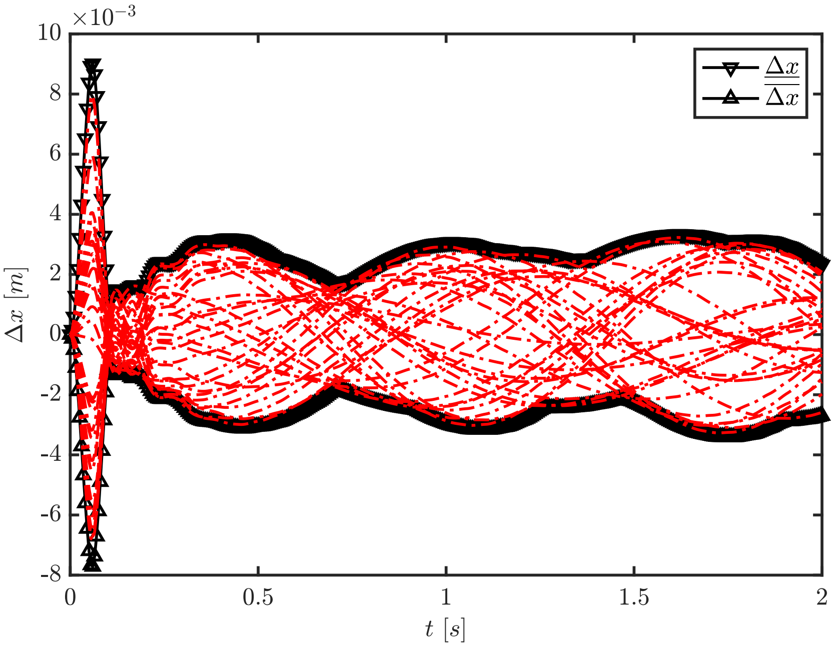


Figure 2: Bounds on the acceleration of the sprung mass, obtained by propagating the hyper-parameter combinations that yield the bounds on the basis functions (in black), as well as extremum-yielding realisations of the Sobol-set simulation (in red).

The interval on the maximal stroke during the simulated time period (i.e., the relative displacement between the sprung and unsprung mass of the car) is $[4.453e-05; 0.00901]$ m in case only those parameters in $\mathcal{H}$ that yield the bounds on the basis functions are propagated, and $[4.91e-05; 0.00794]$ m when a large-space filling design between the intervals on those parameters is propagated. As such, even when very large bounds are imposed on the uncertain road profile, the proposed methodology is able to give an exact estimate of the bounds on selected quantities of interest of the car dynamics. Furthermore, this estimate is obtained at greatly reduced cost.

6 Conclusions

The definition of covariance kernels and their parameters is often performed based on limited data or the engineering judgement of the analyst. To overcome this possible bias, this paper presents an approach to model and simulate random fields using the Karhunen-loève expansion with imprecise covariance kernels in the context of transient dynamic problems. The problem is approached from an interval standpoint, and an iterative procedure is proposed to generate a set of complete $\mathcal{L}_2$ bases that effectively bound the realisations of the imprecise random field. A discussion on when such approach is applicable is included, and specifically focussed on transient dynamic problems. A case study concerning the dynamics of a car suspension while driving over a road that is modelled as an imprecise random field is studied. It is shown that the method is indeed capable of efficiently and effectively computing the bounds on stochastic quantities of interest, such as e.g., the probability of failure or cumulative density function of the response, given the imprecision in the random field input.

Acknowledgement

Matthias Faes is a post-doctoral researcher of the Research Foundation Flanders (FWO) under grant number 12P3519N.

REFERENCES

[1] Beer, M., Ferson, S., and Kreinovich, V. Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing 37*, 1-2 (2013), 4–29.

[2] Betz, W., Papaioannou, I., and Straub, D. Numerical methods for the discretization of random fields by means of the Karhunen-Loeve expansion. *Computer Methods in Applied Mechanics and Engineering 271* (2014), 109–129.

[3] Charmpis, D., Schuëller, G., and Pellissetti, M. The need for linking micromechanics of materials with stochastic finite elements: A challenge for materials science. *Computational Materials Science 41*, 1 (2007), 27 – 37.

[4] Ching, J., and Phoon, K.-K. Impact of Autocorrelation Function Model on the Probability of Failure. *Journal of Engineering Mechanics 145*, 1 (2019), 04018123.

[5] Dannert, M. M., Fau, A., Fleury, R. M., Broggi, M., Nackenhorst, U., and Beer, M. A probability-box approach on uncertain correlation lengths by stochastic finite element method. *PAMM 18*, 1 (2018), e201800114.

[6] Faes, M., Broggi, M., Patelli, E., Govers, Y., Mottershead, J., Beer, M., and Moens, D. A multivariate interval approach for inverse uncertainty quantification with limited experimental data. *Mechanical Systems and Signal Processing 118* (mar 2019), 534–548.

[7] Faes, M., and Moens, D. Identification and quantification of spatial interval uncertainty in numerical models. *Computers and Structures 192* (2017), 16–33.

[8] Faes, M., and Moens, D. Multivariate dependent interval finite element analysis via convex hull pair constructions and the extended transformation method. *Computer Methods in Applied Mechanics and Engineering 347* (2019), 85–102.

[9] Faes, M., and Moens, D. Recent trends in the modeling and quantification of non-probabilistic uncertainty. *Archives of Computational Methods in Engineering* (Feb 2019).

[10] Gao, W., Wu, D., Gao, K., Chen, X., and Tin-loi, F. Structural reliability analysis with imprecise random and interval fields. 49–67.

[11] SOFI, A. Structural response variability under spatially dependent uncertainty: Stochastic versus interval model. *Probabilistic Engineering Mechanics 42* (2015), 78–86.

[12] SOFI, A., MUSCOLINO, G., AND ELISHAKOFF, I. Static response bounds of Timoshenko beams with spatially varying interval uncertainties. *Acta Mechanica 226*, 11 (2015), 3737–3748.

[13] SPANOS, P., AND GHANEM, R. Stochastic finite element expansion for random media. *Journal of engineering mechanics 115*, 5 (1989), 1035–1053.

[14] VANMARCKE, E. H., AND GRIGORIU, M. Stochastic Finite Element Analysis of Simple Beams. *Journal of Engineering Mechanics 109*, 5 (1983), 1203–1214.

[15] VERHAEGHE, W., DESMET, W., VANDEPITTE, D., AND MOENS, D. Random field expansion with interval correlation length using interval fields. In *Eurodyn 2011* (Leuven, apr 2011), pp. 2662–2667.

[16] VERHAEGHE, W., DESMET, W., VANDEPITTE, D., AND MOENS, D. Interval fields to represent uncertainty on the output side of a static FE analysis. *Computer Methods in Applied Mechanics and Engineering 260*, 0 (2013), 50–62.

[17] WEI, P., SONG, J., BI, S., BROGGI, M., BEER, M., LU, Z., AND YUE, Z. Non-intrusive stochastic analysis with parameterized imprecise probability models: I. performance estimation. *Mechanical Systems and Signal Processing 124* (2019), 349 – 368.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

IDENTIFICATION OF VISCO-PLASTIC MATERIAL MODEL PARAMETERS USING INTERVAL FIELDS

C. van Mierlo[1], M. Faes[1], D. Moens[1]

[1]KU Leuven, Department of Mechanical Engineering
Technology campus De Nayer, Jan De Nayerlaan 5, St.-Katelijne-Waver, Belgium
e-mail: koen.vanmierlo[at]kuleuven.be

Keywords: Visco-Plastic material model, Interval Fields, Uncertainty Quantification, Multiscale Analysis.

Abstract. *This paper concerns the application of interval fields as a way to determine the uncertainty of material parameters based on limited test data. More specifically, we focus on the identification of a state-of-the-art non-linear visco-plastic material model based on stress-strain data obtained on a limited set of experiments at different strain rates. This is especially challenging as the corresponding parameters highly depend on the model, and in general, little to no reference values are available from literature. In practice, most of these parameters are determined using specialized curve-fitting software. Typically, a global optimization algorithm is used, minimizing the error between prediction and test data. The problem arising is that the model is fitted on all the sample data in a least-square error sense. This means that the model averages out the (possibly large) scatter among the tested samples. In this manner the intra-variability of the test samples and corresponding model non-determinism is neglected.*

Therefore, this paper presents a novel methodology to include uncertainty in the material model, while preserving the ability to be used in a non-intrusive way for transient numerical analysis. This is achieved by using interval fields to represent the non-determinism in the scarce amount of testing data. The application of the Inverse Distance Weighting interval field definition is studied in this context. It is shown how the choice of the control points and base functions affect the quality of the interval field. Based on this interval field, a virtual set of complete models can be generated, enabling the use of the normal workflow for propagation through the transient numerical model. To demonstrate the methodology, a case study is performed starting from a limited amount of actual test data, obtained from tensile tests conducted on additively manufactured polymer samples. This data is used to construct an interval field with three interval scalars, from where the visco-plastic material model is fitted to determine the uncertainty on a transient numerical model.

1 INTRODUCTION

Today, advanced numerical solvers are being used with complex material models to be able to numerically predict real life complex cases involving high strains, high strain rates, creep, relaxation, and so forth. These material models can capture the complex nature of materials, and as such, play a key role in the overall performance of the numerical simulation. However, such complex material models often need a large amount of parameters, ranging from 2 for a Neo-Hookean (NH) model up to 17 for a Three Network (TN) model [1]. Typically, these parameters need to be determined using a large number of tests (e.g. uni-axial tension, -compression, bi-axial, split Hopkinson bar, . . .). A rule of thumb to make the model as comprehensive as possible is to have at least as many tests as there are invariants in the model [2]. Conducting these tests is usually not straightforward, as special measurement equipment and a conditioned environment are necessary to minimize external influences and the measurement error. This makes it difficult to have all necessary data for a complete model characterisation in an early design stage, especially if there are new materials and production process being considered, like additive manufacturing (AM). In addition to the latter, conventional parameters (e.g. Young's modulus, shear modulus, bulk modulus, . . .) that are typically obtained from material data sheets provided by the supplier, are not very informative, as the required material models for the visco-plastic behaviour often depend on non-conventional parameters specific for the model at hand. For the determination of these parameters, specialized curve fitting tools are used to calibrate the model parameters to the test data. Specifically, this is done by solving an optimization problem where a least-squares error between model prediction and experimental test data is minimized.

The problem arising is that when tests show a large amount of variability, it becomes very hard to use one deterministic model to truthfully represent the stress strain relationship. Obtaining a good model that incorporates this -possibly high- scatter however is complicated by the fact that these non-conventional parameters are highly influenced by the error metric used in the minimization during curve-fitting, introducing variability in the identification procedure itself. Furthermore, there usually exists only limited reference literature on these parameters. This in contrast to conventional parameters which can be determined quite confidently by standard tests, and reference values are omnipresent in literature for frequently used materials. Combined with the fact that the information at hand generally stems from a limited amount of tests conducted, rendering the amount of data rather scarce, it is clear that the uncertainty quantification (UQ) of these non-conventional model parameters is a challenging problem.

Interval methods have been proven to provide an analyst with an objective estimate of the uncertainty under scarce data, as e.g., compared to Bayesian approaches [3]. One way to approach this problem could be by applying a multi-variate convex-hull based identification procedure. However, there exists a physical coupling between the model parameters as they represent the stress-strain relationship of the material. When independent intervals are defined on these model parameters, it can no longer be guaranteed that for each realisation, the model represents a physically feasible material model. So a different approach is necessary, starting from feasible stress-strain curves rather than individual model parameters.

This paper applies the recently introduced framework of interval fields in this context. These enable to account for non-deterministic material parameters that are variable and non-homogeneous over the model domain. This can be regarded as a possibilistic counterpart to random fields [4, 5, 6]. The big advantage of these techniques is that the input fields can be defined in an intuitive sense, while remaining non-intrusive, i.e., enabling that the propagation of the uncertainty consists of multiple deterministic evaluations.

The paper is organized in the following manner: section 2 introduces the interval field concept. Next, the considered visco-plastic material model is discussed in section 3. Section 4 then describes how non-homogeneous interval fields can be used to represent the available test data, obtained by performing experiments at the material level. Section 5 discusses how the obtained interval fields can be used in a numerical simulation for a specific case study. Finally, section 6 discusses the conclusions and future work.

2 INTERVAL FIELD ANALYSIS

This section gives a general description of interval fields as recently introduced in [6]. By definition interval parameters are indicated using apex I: x^I. Vectors are indicated as lower-case boldface characters $\mathbf{x}$, whereas matrices are expressed as upper-case boldface characters $\mathbf{X}$. For the remainder of the text, interval parameters are either represented using the bounds of the interval $x^I = [\underline{x}; \overline{x}]$ where $\underline{x}$ stands for the lower bound and $\overline{x}$ stands for the upper bound, or by their centre point $\hat{x} = \frac{\underline{x}+\overline{x}}{2}$ and the radius $\Delta x = \frac{\overline{x}-\underline{x}}{2}$. An interval is *closed* when both the upper and lower bound are a member of the interval. The domain of a real-valued closed interval is denoted as $\mathbb{IR}$. $x^I \in \mathbb{IR}$ is specifically defined as:

$$x^I = [\underline{x}; \overline{x}] = \{x \in \mathbb{R} \mid \underline{x} < x < \overline{x}\} \tag{1}$$

2.1 Explicit interval fields

The definition of an explicit interval field as introduced in [6] is given in equation (2), where the field consists of a superposition of $n_b \in \mathbb{N}$ base functions ψ_i defined over the domain Ω of the FE model. Each of these base functions is scaled with an independent interval scalar $\alpha_i^I \in \mathbb{IR}$. When considering a discretised FE domain evaluated at k locations (e.g., one per element) used to model the field variability, these base functions provide a mapping from the full *k* dimensional input space to a reduced n_b dimensional input space:

$$x^I(\mathbf{r}) = \mu_x^I + \sum_{i=1}^{n_b} \psi_i(\mathbf{r})\alpha_i^I \tag{2}$$

with $n_b \ll k$. It is clear that through this definition, the intervals control the amount of uncertainty in a strongly reduced dimensionality, while the base functions provide the coupling between the local intervals at the element level.

2.2 Inverse distance weighting interval fields

Different base functions ψ_i can be defined to model the spatial complexity of the field. In this paper, the Inverse Distance Weighting interpolation (IDW) is used to create the base functions. For a comprehensive description of alternative base function definitions, the reader is referred to [7]. When base functions are defined by IDW as in equation (3), the spatial complexity of the field is based on a number of selected control points $\mathbf{r}_i \in \mathbb{R}$ inside the model domain Ω. The assumption is that an independent interval is available to describe the level of uncertainty at each of these locations. The base functions are then constructed such that for each other point in the domain, the field value is a weighted sum of the known intervals at the selected locations, the uncertainty of which is weighted by the Eucledian distance in physical space to the control points. As such, the base functions are given as:

$$\psi_i(\mathbf{r}) = \frac{w_i(\mathbf{r})}{\sum_{j=1}^{n_b} w_j(\mathbf{r})} \tag{3}$$

with $w_i(\mathbf{r}) \in \Omega$ and $i = 1, \ldots, n_b$:

$$w_i(\mathbf{r}) = [D(\mathbf{r}_i, \mathbf{r})]^{-p} \tag{4}$$

with $D()$ a distance measure in Ω, and $p \in \mathbb{R}$ a tunable scaling parameter. This technique allows for an intuitive modelling of non-homogeneous uncertain fields based on local uncertainty, making it well suited for representing uncertainty in material testing data, as envisaged in this work.

2.3 Propagation of interval fields

The propagation of an interval field typically yields a d-dimensional non-convex set $\tilde{y}$ of uncertain model responses given by the numerical model $\mathcal{M}() : \mathbb{R}^k \mapsto \mathbb{R}^d$. Since a general solution to this problem is not feasible, the exact solution set $\tilde{y}$ is approximated by an uncertain realisations set $\tilde{y}_s$ consisting of q deterministic propagations of the interval field $x_j(\mathbf{r})$. The advantage of the interval fields as defined above, is that the propagation reduces to adequately sampling the resulting n_b-dimensional hypercubic input space spanned by the interval scalars. For the application of a transient numerical model $\mathcal{M}(t)$ this typically results in a time dependent output set $\tilde{y}(r, t)$, containing the output of d-degrees of freedom, or corresponding strains or stresses for each time step. This set is explicitly defined as:

$$\tilde{y}_{s,q} = \left\{ y_{s,j}(\mathbf{r}, t_i) | y_{s,j}(\mathbf{r}, t_i) = \mathcal{M}(x_j(\mathbf{r}), t_i); \alpha_{i,j} \in \alpha_i^I; i = 1, \ldots n_b; j = 1, \ldots, q \right\} \tag{5}$$

where in addition to the general formulation this set is defined as time dependent. This stems from the implementation of the time-dependent analysis envisaged in this work, where an implicit or explicit solver will use a number of time steps to reach the final result. Typically, the solver gives an output for every time step $t_i \in \mathbb{R}$, part of the solution set at this time step.

3 VISCO-PLASTIC MATERIAL MODELLING

The material model used throughout this study is the Three-Network model (TN), which is found to be able to represent the material behaviour in the visco-plastic regime [8], specifically developed for thermoplastic materials, and capable of predicting large strain deformation mechanical behaviour in cyclic multiaxial stress states. The model takes a total number of 17 parameters, 11 of which are determined by means of curve fitting when there is no temperature dependence to be modelled, and there is only tensile test data available.

3.1 The Three-Network visco-platicity model

Here, the material model under consideration is shortly described. For a comprehensive description of the model, the reader is referred to [1], where the Three Network model is described in detail. The model is similar to the hybrid model as described in [9], but it is numerically more efficient. As the name of the model suggests, it is composed of three molecular networks acting in parallel. The total deformation gradient $\mathbf{F}^{appl}$ consists of a terminal expansion part $\mathbf{F}^{th}$ and mechanical deformation part $\mathbf{F}$:

$$\mathbf{F}^{appl} = \mathbf{F}\mathbf{F}^{th} \tag{6}$$

where on network A and B the deformation gradient is decomposed into a visco-plastic and visco-elastic component:

$$\mathbf{F} = \mathbf{F}^{e}_{A,B}\mathbf{F}^{p}_{A,B} \tag{7}$$

This can be interpreted as crystalline and amorphous zones in the material. The Cauchy stress acting on these networks is given by an eight-chain representation [10]. For network C this is given by a model with first order I_2 dependence, where I_2 stands for the second invariant. The total Cauchy stress in the system is now given by the sum of these individual stresses $\sigma = \sigma_A + \sigma_B + \sigma_C$. The velocity gradient of network A and B is decomposed in the same way into an elastic and viscous component. In summary the velocity gradient of the viscoelastic flow $\dot{\mathbf{F}}^v$ for network A and B can be written as follows:

$$\dot{\mathbf{F}}^v_{A,B} = \dot{\gamma}_{A,B}\mathbf{F}^{e-1}_{A,B}\frac{\mathrm{dev}[\boldsymbol{\sigma}_a]}{\tau_{A,B}}\mathbf{F}^p_{A,B} \tag{8}$$

where $\dot{\gamma}$ gives the effective flow rate which is considered to follow a power law form, and the driving deviatoric stress on the relaxed configuration convected to the current configuration is given by $\mathrm{dev}[\boldsymbol{\sigma}_A]$. By defining an effective stress by the Frobenius norm τ, the direction of the driving deviatoric stress is given by:

$$\mathrm{dev}[\boldsymbol{\sigma}_A]/\tau \tag{9}$$

3.2 Uni-axial test data

The material under consideration here is the Durable Resin V1 printed on a Formlabs Form 2 stereolithography printer. To characterise the material parameters of the model, uniaxial tension tests have been preformed in accordance with the NIST report on additive manufacturing [11]. These tests are summarized in figure 1 where two tests have been performed at different strain rates, respectively at 50 mm^{-1} and 5 mm^{-1}, further referred to as high speed (HS) and low speed (LS). This difference is indicated by a dashed line for the HS tests and a full line for the LS test results. From these tests, it is clear that the material behaves in accordance with most polymers, where after an initial deformation the macromolecules start to orientate and cause a stiffening of the material before failure occurs. The available data is very limited but one can clearly state that there is a large variability on the data.

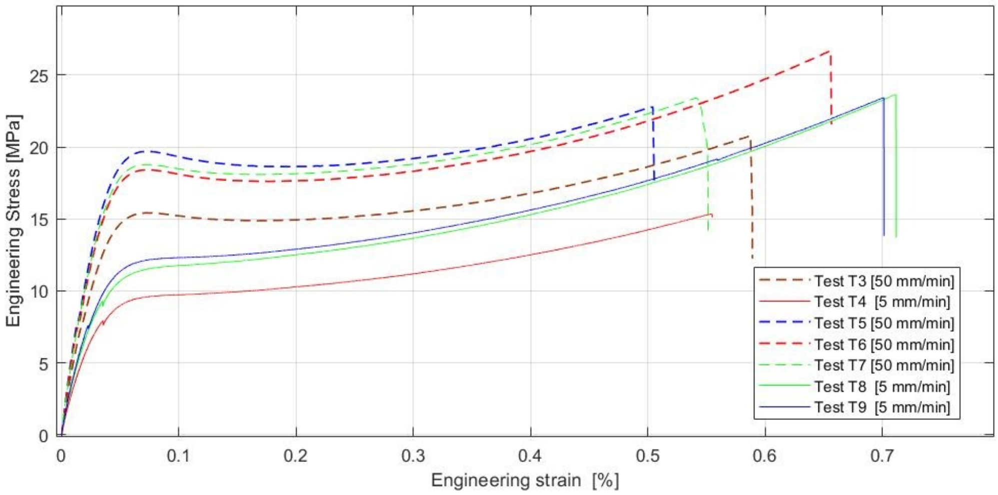


Figure 1: Test results of uni-axial tensile tests (ASTM D638)

4 NON-HOMOGENEOUS INTERVAL FIELDS

When considering the test data as described in the previous section, a first (naive) interval approach could be to calculate independent intervals on all model parameters by calibrating material models to each individual test sample. All underlying dependencies of the material parameters are as such disregarded, and hence, the possibility of non-physical realisations arises. This is shown in figure 2 where the material model prediction envelopes (black) are plotted based on the vertex combinations of the resulting 11-dimensional hypercubic material parameter space. The minimum and maximum value of the tests are indicated in blue and red for the tests at high speed (full line) and low speed (dashed line). The realisations obtained by this approach are not realistic, as the prediction of the upper bounds of these envelopes starts yielding at a stress higher than the ultimate strength of the material observed during testing. Also, it is clear that the amount of non-conservatism is very high in the obtained result, clearly indicating that unfeasible predictions will be included if dependencies between material parameters are not taken into account. In addition to this, a multivariate vertex propagation on these naive intervals requires a total number of $2^{11} = 2048$ runs. This is computationally intractable for the complex numerical simulations under consideration, as they require several hours of wall-clock time on high-performance computers to complete.

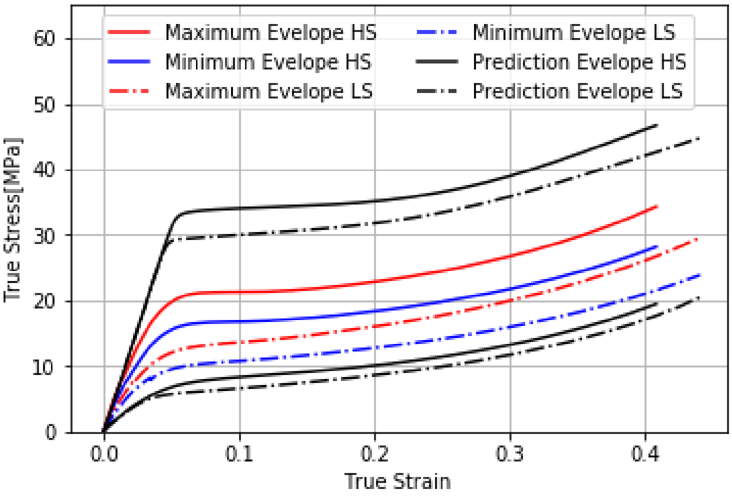


Figure 2: Data of the tests (red, blue) and the naive model prediction (black)

Therefore, this work focuses on the use of an alternative approach where the measurement data is first pre-processed as an interval field that describes the bounds on stress-strain relationship. Then, realisations of this interval field will be used to calibrate the model parameters. This enables a reduction of the amount of evaluations depending on the number n_b of base functions used. Indeed, if this would be successful with for example $n_b = 3$, a vertex approach takes only a total of $2^{n_b} = 8$ evaluations. Furthermore, every realisation of the interval field yields a physically admissible stress-strain relationship and provides the physical coupling between parameters as needed. Still, at the control points the interval field stays perfectly decoupled enabling the use of different sampling schemes. These will be examined in this section.

4.1 Interval field approach

In order to represent the non-determinism in the obtained experimental data using the interval field concept, an envelope is considered that contains the lower and upper stress-strain curves from the test data as shown in figure 1. This is done separately for the LS and HS data. By considering the constructed envelopes $\mathcal{A}_{LS}$ and $\mathcal{A}_{HS}$ as bounds on the feasible stress-strain curves at the respective reference speeds, we implicitly assume that the tests cover the complete range of possible material behaviour. The challenge is now in defining base vectors and corresponding interval scalars such that the dimensionality of the problem remains tractable, while at the same time, a good correspondence with the test data is achieved.

For the construction of the interval fields, the Inverse Distance Weighting approach is applied. As described earlier, this approach relies on the exact quantification of field parameter intervals at specific locations in the domain. Once these are chosen, the base functions over the domain follow directly from equations 3 and 4. In this case, exact stress intervals σ_i^I have to be determined at selected strain locations ϵ_i. This can be done using the midpoint and radius of the envelopes $\mathcal{A}_{LS}$:

$$\hat{\sigma}_{i,LS} = \frac{min(\mathcal{A}_{LS}(\epsilon_i)) + max(\mathcal{A}_{LS}(\epsilon_i))}{2} \tag{10a}$$

$$\Delta\sigma_{i,LS} = \frac{max(\mathcal{A}_{LS}(\epsilon_i)) - min(\mathcal{A}_{LS}(\epsilon_i))}{2} \tag{10b}$$

and $\mathcal{A}_{HS}$:

$$\hat{\sigma}_{j,HS} = \frac{min(\mathcal{A}_{HS}(\epsilon_j)) + max(\mathcal{A}_{HS}(\epsilon_j))}{2} \tag{11a}$$

$$\Delta\sigma_{j,HS} = \frac{max(\mathcal{A}_{HS}(\epsilon_j)) - min(\mathcal{A}_{HS}(\epsilon_j))}{2} \tag{11b}$$

This means that the complete specification of the interval fields depends only on the chosen reference locations, which consequently, have to be chosen with care. This is especially true as the number of control points directly influences the number of evaluations and therefore, the computational effort in the propagation step. These considerations are described in the following subsection.

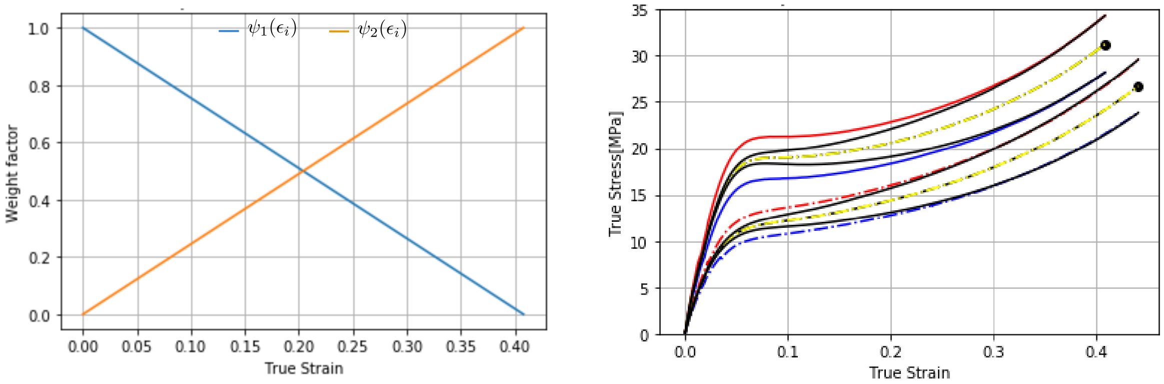
4.2 Controlling spatial complexity

When an interval field is discretised over the model domain the number of control points influences the spatial complexity. Even when the interval scalars are homogeneous, some realisations of the field exhibit a large gradient between two control points, this is referred to as the spatial complexity of the field. This is controlled by choosing the correct location of these control points. This is also of crucial importance to avoid non-physical material behaviour. For example, when the material becomes ‘softer’ after yielding, this could result in an unstable material model [12]

Hence this section describes the considerations of the number of control points and discusses several interpolation approaches to construct the interval field, respectively for the case of two and four control points.

4.2.1 Two control point implementation

For the first case, the minimum number of two control points is applied, taken respectively at the start and end of each envelope. The first interval scalar is placed at the origin of the curve, ensuring all realisations to start at zero stress-strain. This also implies that the interval scalar for the first base vector reduces to a crisp zero, lowering at the same time the interval space dimension by one. The other interval scalar corresponding to the end-point of the envelope as such acts as a scaling parameter. Physically, this means that the material characteristics in all realisations of this interval field are coupled (e.g., a stiffer material will have a higher yield stress and ultimate stress). This is obviously an assumption, as this information is not present in the data.



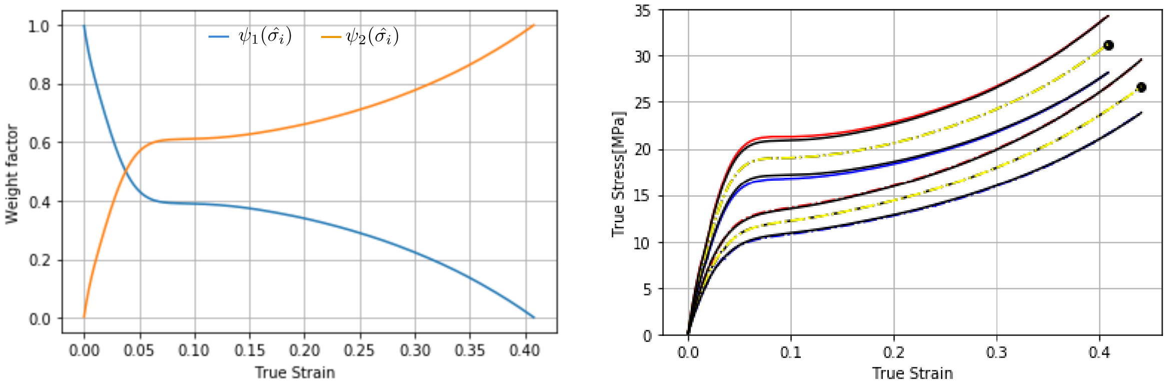
(a) Weight factors measured over strain axis

(b) Interval field realisations for two control points

Figure 3: Field realisation and weight functions over the strain axis, for two control points.

Figure 3a shows the linear base functions resulting from the one-dimensional distance-weighting. Figure 3b shows the outer field realisations of the resulting interval field envelopes (black) on top of the envelopes of the test data (blue/red). In this figure, the centre point is given by the yellow line and the black dots describes the location of the control points. From this figure, it becomes clear that the field can't capture the whole envelope. This is due to the fact that there is a constant increase of uncertainty along the strain axis. As a result, the uncertainty is underestimated for a large part of the stress-strain curve.

The problem at hand is inherent to the use of IDW to model the interval field, where the weight factor increases linear (when $p = 1$) with the distance along the strain axis. An intuitive solution to this problem would be to change the distance measure $D()$ of the weight functions such that the distance is measured over the continuous midpoint curve of the stress-strain envelope (yellow line in figure 3b). This approach results in the interval field as depicted in figure 4, where now the full experimental envelope is captured using again only two control points. Here the influence of the first interval scalar decays rapidly over the strain-axis, while the influence of the second weight function increases quickly up to a strain level of 0.05, while slowly increasing further on. This is due to the fact that the distance is now measured along the centre line, having a large effect on what is happening at the beginning of the curve where there is a fast increase of stress.

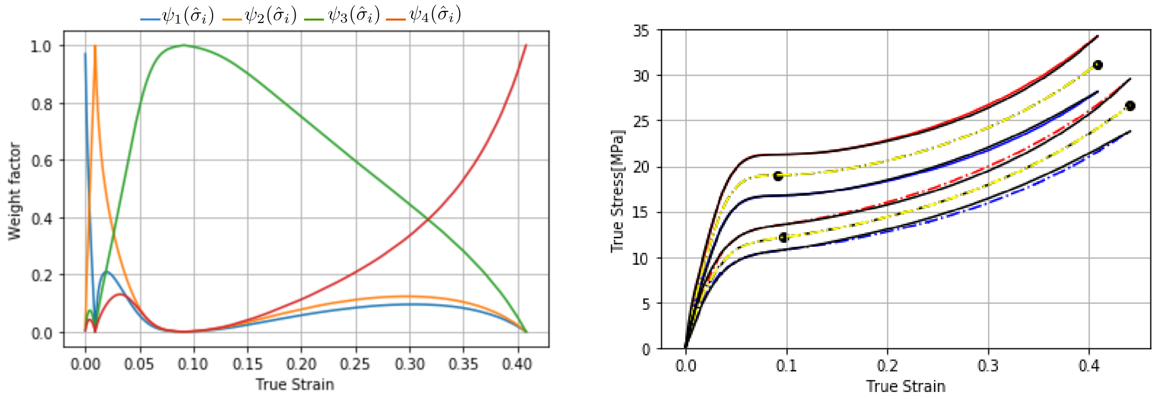


(a) Weight factors measured over midpoint curve (b) Interval field realisations for two control points

Figure 4: Field realisation and weight functions over the midpoint, for two control points.

4.2.2 Four control point implementation

The limitation of the two control point implementation as discussed in the previous section is that, by having only one interval scalar, the uncertainty analysis comes down to a homogeneous scaling of a reference curve over the strain domain. This however does not cover for all variations observed in the experiments. In order to increase the achievable complexity, the field is now constructed with four control points located (1) at the start, (2) at one tenth of the flexural point, (3) at the flexural point and (4) at the end point of the envelope. In this manner, the material can differ in initial stiffness, yield at a different stress and fail at different ultimate strengths, all in a decoupled sense. Also, through this definition, the yielding stress is always lower than the ultimate strength, even for the combination of vertex extrema on the interval scalars at these four locations. This guarantees physically feasible realisations within the hypercubic interval space. The resulting interval field is shown in figure 5.

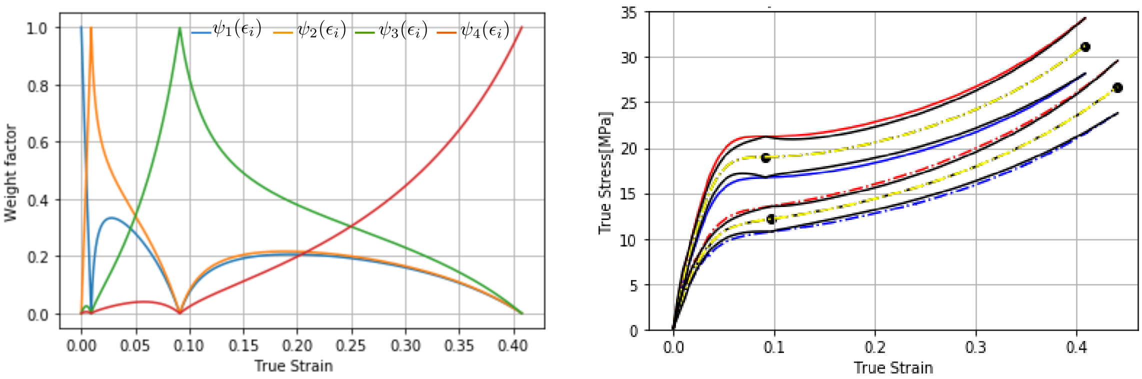


(a) Weight factors measured over midpoint curve (b) Interval field realisations for four control points

Figure 5: Field realisation and weight functions over the midpoint, for four control points.

As the first interval again condenses to a crisp zero, the number of independent interval scalars is now equal to 3. The field again clearly captures the experimental envelope, but now from a physical perspective, the stiffness, yield strength and ultimate strength of the material are decoupled. This makes this approach better suited to capture the non-determinism in the model and explore more of the parameter domain. This however comes at the cost of larger amount of runs needed to propagate the uncertainty.

Also, in this implementation, the distance measure is taken over the mean curve rather than directly in the strain domain. It could be stated that, through the addition of the intermediate control points, this is no longer necessary, as these additional points provide exactly the required flexibility to cover the full experimental envelope. However, figure 6 clearly shows that if this approach is followed, the fit is less accurate. This can be explained by looking at the base functions plotted in 6a, which show the weight of every scalar at a specified strain. The problem arising from using IDW to construct a non-homogeneous interval field is that the weight factors are non-zero even beyond the ‘next’ control point. This makes that the first two small intervals at the start manifest themselves in the end by lowering the average between the control points. This effect can be seen in figure 6a where the second relatively small scalar (orange) still plays a role at a strain of 0.25, even though it was constructed at 0.009 strain. When using the centre curve of the envelope, the distance between the scalars is influenced, making them less sensitive to the small intervals at the start of the curve, as the distance between them is larger. This is clear when comparing figure 5a and 6a, where a substantial decrease of the influence of the first two scalars can be seen in figure 5a.



(a) Weight factors measured over strain axis

(b) Interval field realisations for four control points

Figure 6: Field realisation and weight functions over the strain axis, for four control points.

It can be concluded that the proposed IDW-based approach is able to capture the non-determinism in non-homogeneous testing data without overestimating the uncertainty, ensuring physical realisations, and this at a controllable low dimensionality.

5 CASE STUDY

To test the proposed approach, a case study is performed where a component, as illustrated in figure 7 is produced by means of additive manufacturing. One of the problems withholding the true potential of additive manufacturing for many industrial applications, is that material properties and their variability are difficult to characterise, and the production process has a large influence herein [13]. An example of this can be found in the NIST report on additive manufacturing [11] where guidelines are given for testing of additively manufactured samples.

5.1 Description

The material that is used for this case is Durable Resin V1 printed on a Formlabs Form 2 stereolithography printer, the test results of which are summarized in figure 1. The material is assumed to be isotropic and the build direction is not taken into account during the simulation of its mechanical response. The goal of this case study is to study the stress at the corner node (nr. 1122) of the component, considering the uncertainty that is present in the material parameters. A load is applied by setting a velocity of 50 mm^{-1} at the top surface of the component at node (nr. 3498), for a duration of one second. For this work, only the output at the end time of the simulation is considered.

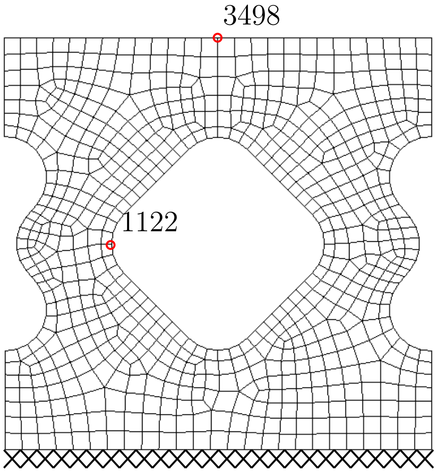


Figure 7: Numerical model of the biaxial structure (the elements displayed are for illustration purposes, a finer discretization is used during simulation)

5.2 Results

The interval field at the input side is assembled using the four control point approach as discussed in section 4.2.2. The field is created as described in section 2, resulting in the vertex realisations as show in figure 8. The output of the simulation at the 8 vertices is given in table 1. These results show that the stress of the simulation lies within the interval of $[14.76 \;\; 18.51]$ MPa for node (nr. 1122).

6 CONCLUSIONS AND FUTURE WORK

The presented Inverse Distance Weighting based interval field method performs well for the use of complex material models. Compared to a naive interval approach, it increases the realism of the realisations, and simultaneously reduces the dimensionality of the uncertainty problem. Care should be taken to include enough control points to enable covering the full experimental envelope. Also, the definition of the distance measure at the core of the base function definition plays a crucial role in this. It is shown that the proposed method is capable of rigorously determining the bounds on a requested output parameter, even in the presence of a scarce amount of data. The total number of finite element solves is kept within reasonable limits (8), making the approach applicable to transient numerical simulations where single runs can take multiple days. As the proposed method is non-intrusive, the normal workflow of

nr.	principal stress [MPa]	vertex
1	14.76	[0,-1,-1,-1]
2	15.38	[0,-1,-1, 1]
3	16.46	[0,-1, 1,-1]
4	17.41	[0,-1, 1, 1]
5	15.35	[0, 1,-1,-1]
6	17.01	[0, 1,-1, 1]
7	18.21	[0, 1, 1,-1]
8	18.51	[0, 1, 1, 1]

Table 1: Results of the numerical simulation including uncertain material properties.

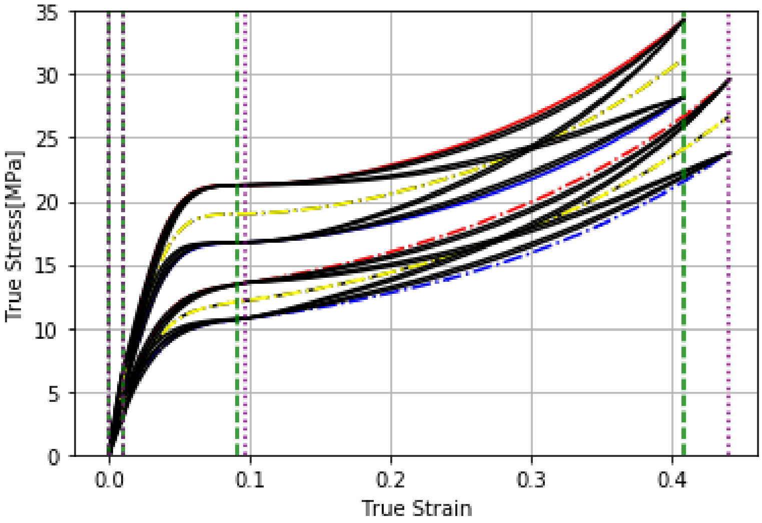


Figure 8: Field realisation for four control points sampled by the vertex method.

preparing and addressing the finite element solver are kept, which also enables to parallelise the simulations.

Acknowledgements

The authors gratefully acknowledge the support of the research foundation Flanders (FWO) under grant 1SA3919N and 12P359N.

REFERENCES

[1] J. Bergstrm. *Mechanics of Solid Polymers: Theory and Computational Modeling*, pages 1–509. Mechanics of Solid Polymers: Theory and Computational Modeling. 2015.

[2] O. H. Yeoh. On the ogden strain-energy function. *Rubber Chemistry and Technology*, 70(2):175–182, 1997.

[3] M. Faes, M. Broggi, E. Patelli, Y. Govers, J. Mottershead, M. Beer, and D. Moens. A multivariate interval approach for inverse uncertainty quantification with limited experimental data. *Mechanical Systems and Signal Processing*, 118:534–548, 2019.

[4] E. Vanmarcke and M. Grigoriu. Stochastic finite element analysis of simple beams. *Journal of Engineering Mechanics*, 109(5):1203–1214, 1983.

[5] W. Verhaeghe, W. Desmet, D. Vandepitte, and D. Moens. Interval fields to represent uncertainty on the output side of a static fe analysis. *Computer Methods in Applied Mechanics and Engineering*, 260:50–62, 2013.

[6] D. Moens, M. De Munck, W. Desmet, and D. Vandepitte. Numerical dynamic analysis of uncertain mechanical structures based on interval fields. In Alexander K. Belyaev and Robin S. Langley, editors, *IUTAM Symposium on the Vibration Analysis of Structures with Uncertainties*, pages 71–83, Dordrecht, 2011. Springer Netherlands.

[7] M. Faes and D. Moens. Recent trends in the modeling and quantification of non-probabilistic uncertainty. *Archives of Computational Methods in Engineering*, pages 1–39, 2019.

[8] C. van Mierlo, W. Dirix, M. Faes, and D. Moens. Impact modeling of hyper-elastic am compliant mechanisms using high-speed digital image correlation. In *Exploring the Design Freedom of Additive manufacturing through Simulation*, pages 72–73, Helsinki, Finland, 2018. NAFEMS.

[9] J.S. Bergstrm, C.M. Rimnac, and S.M. Kurtz. Prediction of multiaxial mechanical behavior for conventional and highly crosslinked uhmwpe using a hybrid constitutive model. *Biomaterials*, 24(8):1365 – 1380, 2003.

[10] Ellen M. Arruda and Mary C. Boyce. A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials. *Journal of the Mechanics and Physics of Solids*, 41(2):389 – 412, 1993.

[11] K. Wegener A.B. Spierings, M. Voegtlin, T. Bauer. Materials Testing Standards for Additive Manufacturing of Polymer Materials :. *Prog Addit Manuf*, 1:9–20, 2015.

[12] Daniel Charles Drucker. A definition of stable inelastic material. Technical report, Brown univ providence RI, 1957.

[13] M. Faes, Y. Wang, P. Lava, and D. Moens. Variability, heterogeneity and anisotropy in the quasi-static response of laser sintered pa-12 components. *Strain*, 53(2), 2017.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

ESTIMATING UNCERTAIN REGIONS ON SMALL MULTIDIMENSIONAL DATASETS USING GENERALIZED PDF SHAPES AND POLYNOMIAL CHAOS EXPANSION

Maurice Imholz[1]**, Dirk Vandepitte**[1]**, and David Moens**[1]

[1]KU Leuven, Department of Mechanical Engineering
Celestijnenlaan 300B, 3001 Heverlee, Belgium
e-mail: maurice.imholz@kuleuven.be

Keywords: interval analysis, uncertainty quantification, polynomial chaos expansion, Bayesian inference

Abstract. *In uncertainty analysis, estimating the degree of uncertainty based on some physical experiments is an essential part of the process to create robust products. Both at the input and the output side of an available model, experiments may be done, which can then be (inverserely) propagated to obtain uncertain results on the other side. In probabilistic analysis, PDF shape, stochastic moments and correlation may be inferred from this data. In possibilistic analysis, these quantities are hard to interpret physically and are therefore difficult to compute. Instead, interval bounds and dependency information can be determined. This paper presents a strategy to infer both interval bounds and dependency information from a (limited) set of data points in a multidimensional space, based on Polynomial Chaos Expansion and a generalized Probability Density Distribution (PDF) shape.*

1 INTRODUCTION

The use of intervals in numerical modelling to represent uncertainty is gaining increased interest. Intervals can be more easily applied in cases of low data availability and where information on extreme cases is of higher inportance. The use of intervals to represent non-deterministic quantities omits the need to predefine a PDF, which may be hard to estimate when little data is available. However, the simplicity of intervals also has a large disadvantage: they are unable te represent dependency between different uncertain quantities. Some solutions to this have been the subject of recent research, such as interval fields [1, 2, 3, 5, 4], interactive fuzzy numbers [6, 7], interval correlation [8], and the use of copulas in interval context [9]. In this paper, a procedure is discussed that, given a small set of multidimensional data points, allows to make an estimate of the total population uncertainty, incorporating the dependency present in the data as well. It makes use of a generalized PDF shape to estimate intervals in a Bayesian inference scheme, and then Polynomial Chaos Expansion (PCE) to convert the hypercubic region spanned by the marginal interval estimates into a uncertain region of arbitrary shape that also incorporates the dependency present. Section 2 elaborates the concept of generalized PDF, also mentioned in [10], section 3 discusses the way the dependency is captured using PCE, and section 4 shows the combined method on a small 2D dataset.

2 Bayesian Interval Estimation based on a generalized PDF shape

This section provides a short summary of the method of estimating interval bounds based on a generalized PDF shape. The reader is refered to [10] for a more elaborate explanation.

2.1 Principle of the generalized PDF shape

Equation 1 shows the general formula for Bayesian analysis, which describes the posterior distribution given the data D $p(\theta \mid D)$ as the product of the likelihood of the data and a prior distribution.

$$p(\theta \mid D) = \frac{p(D \mid \theta)p(\theta)}{p(D)} \tag{1}$$

θ represents a set of stochastic parameters that capture the PDF. In interval context, these parameters become the interval bounds $\overline{x}$ and $\underline{x}$. The data provides a minimum and maximum observed value $\widehat{x_m}$ and $\widehat{x_M}$. Equation 2 again describes Bayes' theorem, now in terms of the observed interval bounds and the true interval bounds that are to be estimated.

$$p(\overline{x}, \underline{x} \mid \widehat{x_M}, \widehat{x_m}) = \frac{p(\widehat{x_M}, \widehat{x_m} \mid \overline{x}, \underline{x})p(\overline{x}, \underline{x})}{p(\widehat{x_M}, \widehat{x_m})} \tag{2}$$

To express the likelihood function $p(\widehat{x_M}, \widehat{x_m} \mid \overline{x}, \underline{x})$, an arbitrary PDF shape S and corresponding stochastic parameter θ is introduced, as the interval bounds are assumed to bound an actual PDF shape, which cannot be identified properly because the dataset is too small. Equation 2 is then rewritten in terms of S and θ as:

$$p(\widehat{x_M}, \widehat{x_m} \mid \overline{x}, \underline{x}) = \int_S \int_\theta p(\widehat{x_M}, \widehat{x_m} \mid \theta) \cdot p(\theta \mid \overline{x}, \underline{x}) d\theta dS \tag{3}$$

$$= \int_S \int_\theta M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta) \cdot p(\theta \mid \overline{x}, \underline{x}) d\theta dS \tag{4}$$

this equation theoretically only holds if the integration is done over all possible PDF shapes S and all values of the corresponding parameter value θ. The first part of the integrand describes the occurence of certain extreme values given the PDF on the quantity x. This equals the extreme value distribution (EVD) $M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta)$, which depends on the number of experiments n. The second part describes the probability on having a certain stochastic parameter value, given the extreme bounds in the total population. Assuming the total population is very large, these extremes are equal to the maximum and minimum values allowed by the PDF that corresponds to a certain value of θ. Given a certain bounded PDF shape S (e.g. the uniform distribution, the 3-σ bounded Gaussian distribution, ...), $p(\theta \mid \overline{x}, \underline{x})$ equals a delta function at that specific parameter value (or combination of values if more than one stochastic parameter is concerned) that puts the maximum and minimum possible values of the PDF at $\overline{x}$ and $\underline{x}$. therefore, the integral needs to be taken over all possible PDF shapes S that are bounded by $\overline{x}$ and $\underline{x}$ (equation 6).

$$p(\widehat{x_M}, \widehat{x_m} \mid \overline{x}, \underline{x}) = \int_S \int_\theta M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta) \cdot \delta(\theta - \theta^*) d\theta dS \tag{5}$$

$$= \int_S M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta^*) dS \tag{6}$$

Putting this into equation 2, the *interval Bayesian inference* equation becomes:

$$p(x^I \mid \widehat{x_M}, \widehat{x_m}) = \frac{\int_S M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta^*) dS \cdot p(x^I)}{p(\widehat{x_M}, \widehat{x_m})} \tag{7}$$

To be able to evaluate the integral, a generalized PDF shape is proposed based on 4 controleable parameters. By definition, $\int_{-\infty}^{+\infty} f_x(x)dx = \int_{\underline{x}}^{\overline{x}} f_x(x)dx = 1$. Many possible parametrizations are possible, and greatly influence the shapes that are considered in evaluating the integral. Since extreme values are of increased interest in the context of this paper, the following four control parameters are proposed (table 1), all focussing on the PDF shape in the extreme values.

symbol	description
p_0	Probability density at $\underline{x}$
p_1	Probability density at $\overline{x}$
$\left.\frac{df_x(x)}{dx}\right\|_0$	first derivative of the PDF at $\underline{x}$
$\left.\frac{df_x(x)}{dx}\right\|_1$	first derivative of the PDF at $\overline{x}$

Table 1: parameters used to determine the PDF shape

A fourth order polynomial is proposed for the explicit representation of the PDF, given by equation 8:

$$f_x(x) = ax^4 + bx^3 + cx^2 + dx + e \tag{8}$$

Using this parameter set and the corresponding fourth order polynomial allows for a large variety of PDF shapes (including nonsymmetrical, sharp tailed, blunt tailed and bipolar shapes), while keeping the integral sufficiently fast to calculate.

The next section discusses two different ways of dealing with the integration defined in equation 7. Next to simply calculating it explicitly, it can also be bounded on the upside by calculating the maximum likelihood.

2.2 Average likelihood and worst-case likelihood estimation

Returning back to equation 7, the following trivial relation can be established for the entire parameter set S (which has been defined above):

$$M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta^*) \leq \max_S M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta^*) \tag{9}$$

Integrating both sides gives:

$$\int_S M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta^*) dS \leq \max_S M_x^{(n)}(\widehat{x_M}, \widehat{x_m}, \theta^*) \cdot \prod_{i=1}^{4} (\theta_{i,max} - \theta_{i,min}) \tag{10}$$

This means that a conservative approximation of the integral can be calculated by determining the maximum likelihood value that occurs within the domain spanned by the parameters in table 1. Essentially, this means that for each test interval x^I, the likelihood is determined by the PDF shape that makes the observed interval most likely, which could be interpreted as the 'worst-case' PDF. Through this, more probability and therefore higher relative importance is given to larger intervals compared to explicitly evaluating the integral, which should lead to larger estimated intervals and therefore more conservative results. This paper refers to the latter approach as the *worst-case likelihood* (WCL) estimate, and the former as the *average likelihood* (AL) estimate. Previous testing of the method shows that the AL estimates tend to be not conservative enough to provide reliable results, but the WCL estimate, giving higher relative importance to larger intervals, does. The combined method discussed in section 4 will therefore use the interval Bayesian inference scheme illustrated here, using the WCL estimate to represent the likelihood function.

3 Interval Polynomial Chaos Expansion

In probabilistic analysis, Polynomial Chaos Expansion (PCE) [11] is used frequently to determine probability distributions on model output quantities. Application of PCE can be done in two ways:

- Given an output $y = f(x)$ as function of a random variable x with known probability density function (PDF), the output distribution can be found by projecting onto a set of polynomial basis functions which are orthogonal w.r.t. the input PDF. Determining the output distribution then comes down to identifying the corresponding PC coefficients.
- Given a quantity u with a known but complex PDF, its probability function can be described more easily by defining u as a function of some *germ variable* ξ with a simple PDF, such as the uniform or Gaussian distribution. Theoretically, by choosing the right functional relation, every type of distribution on u can be obtained.

The first application is well established in the field of numerical modeling [12, 13]. The second application is of particular interest in sampling algorithms, as numerically sampling a complex PDF is nontrivial. Computers can effectively sample from the uniform or Gaussian distribution, but not necessarily from any arbitrary PDF.

3.1 The Inverse Cumulative Density Function transform

Using PCE, the objective is to obtain an explicit expression of:

$$u = f(\xi) \tag{11}$$

such that u has a specific distribution, given the distribution of the germ variable ξ, which is a nontrivial problem, as many definitions of f can lead to the required pdf. The question then becomes to find the most efficient one. It can be proven however that a solution for the above problem is always present, known as the inverse Cumulative Distribution Function (CDF) transform. If the PDF of u is known and u is a continuous variable, its CDF $F_u(u)$ exists and can be determined through:

$$F_u(u) = \int_{-\infty}^{u} f_u(t)dt \tag{12}$$

The domain of $F_u(u)$ is obviously $[0, 1]$ as f_u is always positive and $\int_{-\infty}^{\infty} f_u(t)dt = 1$. We can then obtain the desired distribution of u starting from a uniformly distributed germ between$[0, 1]$ by writing:

$$u = F_u^{-1}(\xi) \tag{13}$$

For an arbitrarily distributed germ with CDF equal to F_ξ we can write:

$$u = F_u^{-1}(F_\xi(\xi)) \tag{14}$$

This expression is especially useful because it holds for any distribution on u or the germ ξ. However, it usually does not give the most efficient mapping.

3.2 Dependent intervals

The use of intervals is of particular interest in the presence of low data availability, as it omits the need of defining and quantifying a suitable PDF. An interval $x^I = \langle \underline{x} | \overline{x} \rangle$ only requires an upper and lower bound to be defined and describes a continuous region of possible values for the quantity x. On the probability of occurence within the interval, no assumption is made, and for the purpose of interval analysis, the probability is assumed to be nonzero of the interval interval, and strictly zero elsewhere. In the multivariate case, an *interval vector* $\mathbf{x^I} = [x_1^I \; x_2^I \cdots x_N^I]$ is used with each entry a simple interval variable. By definition, the entries are assumed independent, so the interval vector defines a set of vectors in the N-dimensional space described by:

$$\mathbf{x^I} = \left[\hat{\mathbf{x}} | \hat{x}_1 \in x_1^I, \hat{x}_2 \in x_2^I, \cdots, \hat{x}_N \in x_N^I \right] \tag{15}$$

Equation 15 describes a hypercube in the N-dimensional domain. Figure 1 illustrates this. If two interval parameters a^I and b^I, are independent, this representation is accurate and introduces no further conservativity. However, in the other case, some degree of conservativity is always introduced through modelling with an interval vector. The higher the dependency, the more conservative this representation will be. This problem cannot be solved within the simple definition of an interval, as not enough parameters are available to represent dependency. Also, this dependency may take a large variety of forms, depending on the shape of the region of possible (a, b)-couples, so the uncertainty model would need a large amount of extra parameters to account for this. Still, the conservativity issue remains and should be addressed.

In structural dynamics, propagation of input interval parameters is usually done through optimization and anti-optimization within the region defined by the input intervals, as the models are usually quite complex and non-monotonous behaviour is possible. Since intervals make no assumption on the probability within the region the define, the optimization is supposed to be unbiased and treat all points in the region as equally probable. For this reason, intervals are usually sampled in a uniform way within the purpose of finding the output optima. So allthough

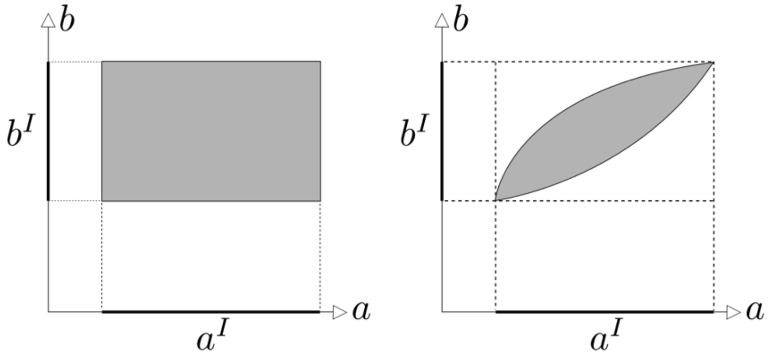


Figure 1: illustration of independent (left) versus dependent (right) interval quantities. The independent case is characterized by the hypercubic region, while the dependent case can theoretically consider any region.

the exact probability is unknown, the uniform distribution is assumed in practise for the purpose of propagating the uncertainty.

The same principle holds in the multivariate case. Given a region with dependency as shown in the right side of figure 1, the sampler is supposed to be unbiased towards any part of the region. This adds an extra requirement to the accurate definition of the uncertain region. Not only does the boundary of the region have to be represented accurately, but also within the region itself uniform sampling has to be possible. The next section describes a PCE-inspired technique that can capture a wide variety of dependencies, starting from a simple interval vector, and incorporating the possibility of uniform sampling, which is called *interval PCE*. The method will first be illustrated in 2D, but can be theoretically expanded to any number of interval components.

3.3 interval PCE

Consider two interval parameters x_1^I and x_2^I, with corresponding interval bounds $\underline{x_1}$,$\underline{x_2}$ and $\overline{x_1}$,$\overline{x_2}$. Assume some dependency is present, which is characterised by a region Ω. Assume the bounds of the intervals itself are perfectly non-conservative, so the square described by the interval vector $\mathbf{x^I} = [x_1^I \; x_2^I]$ is the smallest circumscribed square still fully encapsuling Ω. The means that the far left and far right point part of the region are given by $(\underline{x_1}, x_{2,left})$, $(\overline{x_1}, x_{2,right})$. The upper and lower bounding curve of the region are distinguished as $C_u : x_2 = H(x_1)$ and $C_l : x_2 = h(x_1)$ in between these points (see also figure 2). The PCE is then one of the following form:

$$\begin{cases} x_1^* &= F_\Omega^{-1}\left(\frac{x_1-\underline{x_1}}{\overline{x_1}-\underline{x_1}}\right) \\ x_2^* &= \left(H(x_1)-h(x_1)\right)\frac{x_2-\underline{x_2}}{\overline{x_2}-\underline{x_2}} + h(x_1) \end{cases} \tag{16}$$

with

$$F_\Omega(x_1) = \frac{1}{A}\int_{\underline{x_1}}^{x_1} dt_1 \int_{h(t_1)}^{H(t_1)} dt_2 \tag{17}$$

In equation 17, A is the total surface area of Ω, and the double integral describes the surface area of the part of Ω left of a certain value x_1. Essentially, the first line of equation 16 is an inverse CDF transform: x_1^* is given an artificial distribution, which increases with increasing range of possible x_2-values at a certain value for x_1. This ensures that if x_1 and x_2 are sampled uniformly, the corresponding points are uniformly distributed over the domain Ω. The second line describes a very simple transformation from a uniform distribution between $\overline{x_2}$ and $\underline{x_2}$ to a

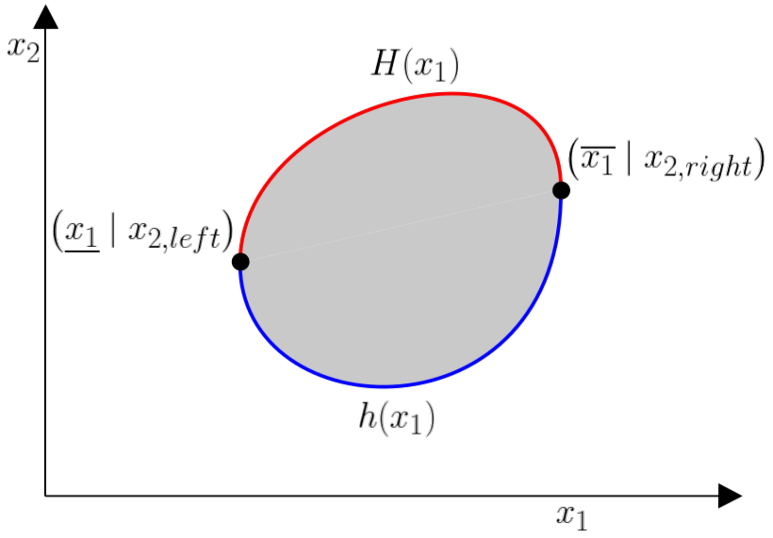

x_2
$H(x_1)$
$(\overline{x_1} \mid x_{2,right})$
$(\underline{x_1} \mid x_{2,left})$
$h(x_1)$
x_1

Figure 2: illustration of the quantities mentioned in the following equations

uniform distribution between the upper and lower bounding curve C_u and C_l, for a certain value of x_1.

In multidimensional space, the PCE is of the following form:

$$\begin{cases} x_1^* & = & F_{\Omega,1}^{-1}\left(\frac{x_1-\underline{x_1}}{\overline{x_1}-\underline{x_1}}\right) \\ x_2^* & = & F_{\Omega,2}^{-1}\left(\frac{x_2-\underline{x_2}}{\overline{x_2}-\underline{x_2}}, x_1\right) \\ \cdots & & \\ x_{n-1}^* & = & F_{\Omega,n-1}^{-1}\left(\frac{x_2-\underline{x_2}}{\overline{x_2}-\underline{x_2}}, x_1, \cdots, x_{n-2}\right) \\ x_n^* & = & (h_u(x_1,\cdots,x_{n-1}) - h_l(x_1,\cdots,x_{n-1}))\,\frac{x_n-\underline{x_n}}{\overline{x_n}-\underline{x_n}} + h_l(x_1,\cdots,x_{n-1}) \end{cases} \tag{18}$$

with

$$\begin{cases} F_{\Omega,1}(x_1) & = & \frac{1}{\Omega_n}\int_{\underline{x_1}}^{x_1} dt_1 \int_{h_1(t_1)}^{H_1(t_1)} dt_2 \int_{h_2(t_1,t_2)}^{H_2(t_1,t_2)} dt_3 \cdots \int_{h_{n-1}(t_1,\cdots,t_{n-1})}^{H_{n-1}(t_1,\cdots,t_{n-1})} dt_2 \\ F_{\Omega,2}(x_2,x_1) & = & \frac{1}{\Omega_{n-1}}\int_{h_1(x_1)}^{x_2} dt_2 \int_{h_2(x_1,t_2)}^{H_2(x_1,t_2)} dt_3 \cdots \int_{h_{n-1}(x_1,t_2,\cdots,t_{n-1})}^{H_{n-1}(x_1,t_2\cdots,t_{n-1})} dt_n \\ \cdots & & \\ F_{\Omega,n-1}(x_{n-1},\cdots,x_1) & = & \frac{1}{A}\int_{h_{n-2}(x_1,\cdots,x_{n-2})}^{x_{n-1}} dt_{n-1} \int_{h_{n-1}(x_1,\cdots,x_{n-2},t_{n-1})}^{H_{n-1}(x_1,\cdots,x_{n-2},t_{n-1})} dt_n \end{cases} \tag{19}$$

In equation 18, the inversion is only done with respect to the first variable inside the braccets, leading to stairwise dependency in the expanded quantities as $x_1^* = f(x_1), x_2^* = f(x_1, x_2), x_3^* = f(x_1, x_2, x_3)$ and so on. This expansion requires an explicit formula for the edge of the region, which can be hard to construct in higher dimensional space, particularly finding explicit descriptions of H_i and h_i in equation 19. Quite often, only 2-way interactions are considered in high dimensional spaces as they tend to have a higher relative impact on the output, which comes down to capturing the dependency in 2D-projections of the total uncertain space. Therefore, the 2D-case is considered in the remainder of this paper.

Usually, an explicit description for $F_\Omega(x_1)$, C_u and C_l is not available, or very difficult to express, so in practise they are expressed by using a truncated PCE based on the univariate legendre polynomials, as they are orthogonal w.r.t. the uniform distribution.

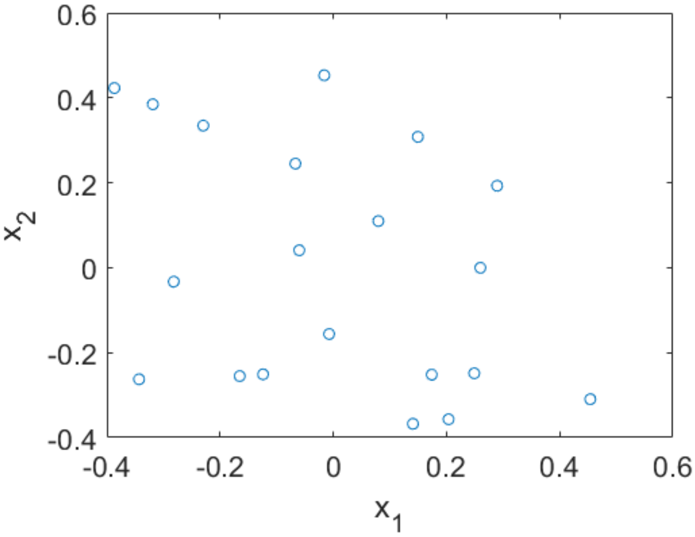


Figure 3: data points used in this example of the combined method

4 The combined method to uncertain regions from small datasets

The method in this section is explained by applying it on the virtual dataset of 20 points, as is given in figure 3.

The data is captured in the 20x2 matrix $\mathbf{X}$. The objective will be to estimate an uncertain region on it in an interval context. The dataset is centered and the eigenvectors $\mathbf{\Phi}$ of $\mathbf{X^TX}$ are identified. The data is projected onto both eigenvectors, leading to two projected sets $\mathbf{u_1} = \mathbf{X}\boldsymbol{\phi}_\mathbf{1}$ and $\mathbf{u_2} = \mathbf{X}\boldsymbol{\phi}_\mathbf{2}$. Next to this two additional projections are done on the vectors $\mathbf{v} = \frac{\lambda_1\boldsymbol{\phi}_\mathbf{1}+\lambda_2\boldsymbol{\phi}_\mathbf{2}}{\sqrt{\lambda_1^2+\lambda_2^2}}$ and $\mathbf{w} = \frac{\lambda_1\boldsymbol{\phi}_\mathbf{1}-\lambda_2\boldsymbol{\phi}_\mathbf{2}}{\sqrt{\lambda_1^2+\lambda_2^2}}$. Projection on these vectors lead to two additional sets $\mathbf{u_3} = \mathbf{Xv}$ and $\mathbf{u_4} = \mathbf{Xw}$. These directions signify the two height lines and diagonals of the smallest circumferential rectangle of the dataset. On all four of the projected datasets, the bayesian inference scheme as was described in section 2 is performed, leading to a total of 4 estimated intervals, which are then multiplied with their corresponding direction, leading to 8 points in the 2D-space that serve as the boundary points of the uncertain region. For the uncertain region, the following parametrization is used (equation 20):

$$\begin{cases} x_1 &= R(\theta)cos(\theta) \\ x_2 &= R(\theta)sin(\theta) \end{cases} \tag{20}$$

with

$$\begin{aligned} R(\theta) &= a_0 + a_1cos(\theta) + a_2sin(\theta) + a_3cos(2\theta) \\ &+ a_4sin(2\theta) + a_5cos(3\theta) + a_6sin(3\theta) + a_7cos(4\theta) \end{aligned} \tag{21}$$

The coefficients a_0 to a_7 can be uniquely determined by the 8 boundary points, leading to the uncertain region in figure 4.

The second part of the method is to apply the theory of interval PCE as described in section 2 to ensure the mapping of the initial variables to the expanded parameters x_1^* and x_2^* ensure that not only the hypercubic spaces is transformed into the uncertain region that was just determined, but also that a uniform sampling on the initial variables also produce a uniform sampling of the uncertain region. The most left and right points are determined by solving $\frac{\partial x_1}{\partial \theta} = 0$, which is

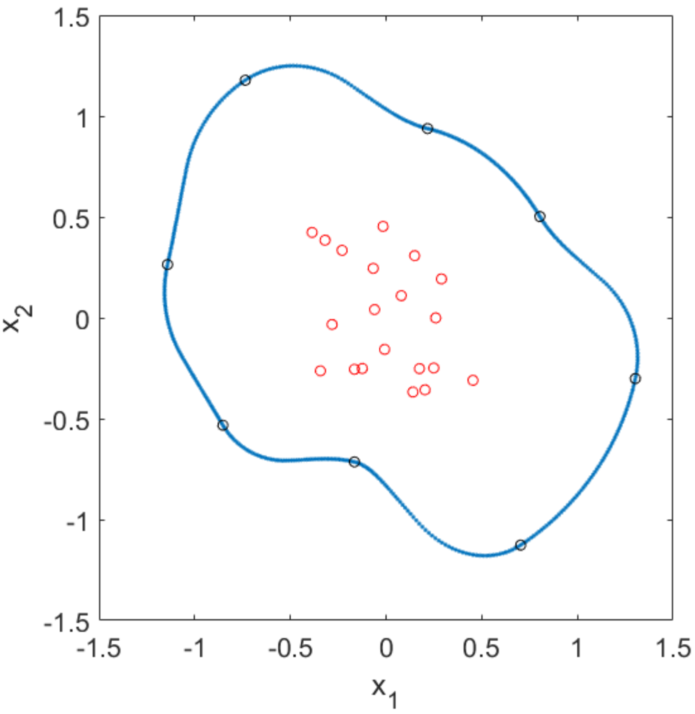


Figure 4: Curve fitted using the 8 boundary points (black circles) and the parametrization in equation 20 and 21

valid for exactly 2 values of θ, referred to as θ_l and θ_r. The upper and lower curve are discretized and fitted using a 10th order Legendre polynomial set, the result of which is given in figure 5.

The CDF as described in equation 17 is determined and inverted numerically and is fitted using a 10th order Legendre polynomial set, the result of which is given in figure 6

This gives an expression for the mapping of x_1 and x_2 on x_1^* and x_2^* given by equation 22. The coefficient values are given in table 2.

$$\begin{cases} x_1^* &= \sum_{i=0}^{10} c_i \cdot P_i(\frac{x_1 - \underline{x_1}}{\overline{x_1} - \underline{x_1}}) \\ x_2^* &= \left(\sum_{i=0}^{10} (H_i - h_i) \cdot P_i(x_1^*)\right) \frac{x_2 - \underline{x_2}}{\overline{x_2} - \underline{x_2}} + \sum_{i=0}^{10} h_i \cdot P_i(x_1^*) \end{cases} \tag{22}$$

Table 2: coefficient values in equation 22 (values are multiplied by 100)

$i =$	0	1	2	3	4	5	6	7	8	9	10
H_i	-67.06	18.67	53.65	-22.16	-7.47	3.98	13.0	7.10	-0.58	-10.82	2.06
h_i	78.36	24.43	-44.17	-16.69	-14.7	-8.45	-3.52	6.62	-5.67	-3.63	-0.45
c_i	50.66	42.69	0.14	4.45	-0.70	1.07	-0.21	0.64	0.23	0.53	-0.08

In equation 22, $P_n(x)$ refers to the nth order Legendre polynomial. The final result of the method given by equation 22 and figure 7 illustrates a uniform sampling within the bounds of the basic variables and the result in the transformed space of x_1^* and x_2^*.

5 Conclusion

This work presented a method to estimate two-dimensional uncertain regions, accounting for possible dependency between the uncertain quantities. From figure 7, it can be seen that

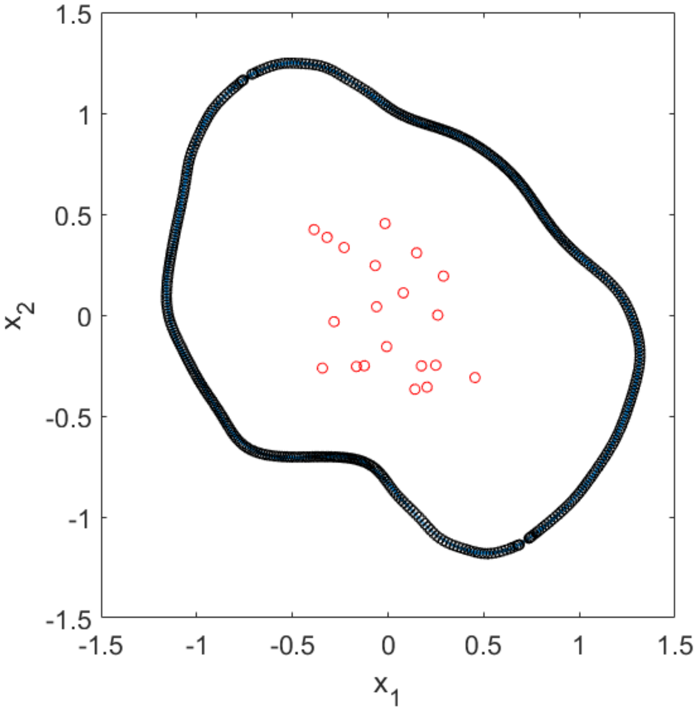


Figure 5: approximation of the upper and lower curve using a 10th order Legendre polynomial set. Blue dots: real curve, black circles: fitted curve

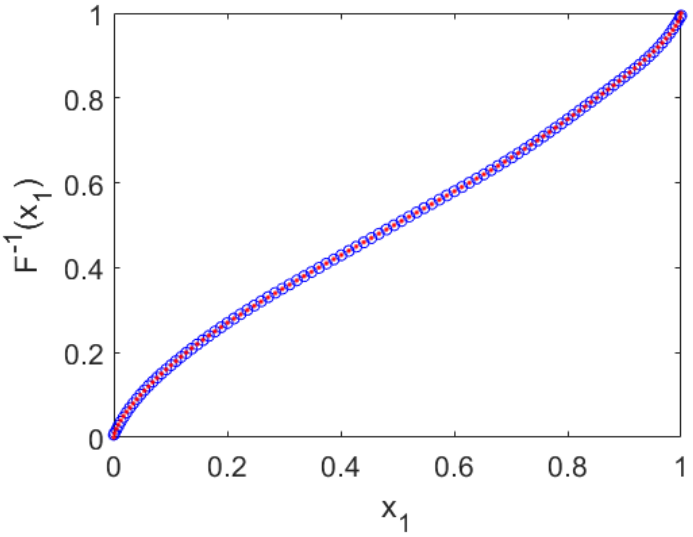


Figure 6: Inverted CDF, red dots: actual curve determined by integral evaluation, blue circles: fitted curve

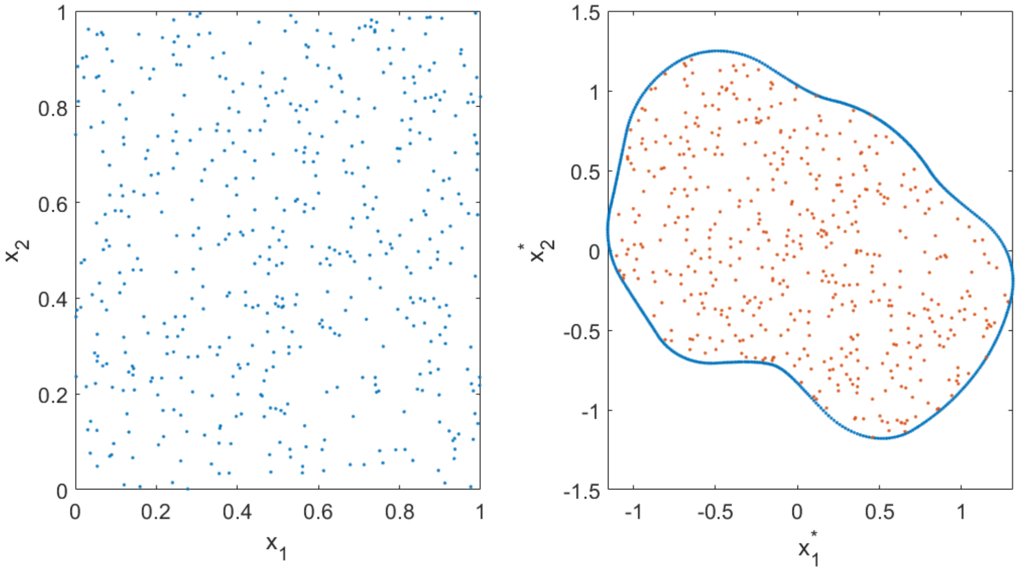


Figure 7: 500 points uniformly sampled from a non-interactive normalized uncertain space (left) and the resulting uncertain region (right)

from a uniform sampling of the initial germ variables, a uniform distribution of the transformed quantities is obtained, and the boundary curve is obeyed as well. Future work will be done on applying the combined method on actual datasets on Finite Element model input parameters and compute resulting output uncertainty, and perform (anti-)optimization to obtain bounds on uncertain output quantities.

Acknowledgements

The authors gratefully acknowledge financial support from FWO-Vlaanderen (the Flemish fund for scientific research) and from the KU Leuven research fund. This work is done with funds provided by FWO project No. G0C2218N, High Dimensional Interval Fields: spatial uncertainty modelling for mechanical reliability (HiDiF).

REFERENCES

[1] D. Moens, M. De Munck, W. Desmet, D. Vandepitte, *Numerical dynamic analysis of uncertain mechanical structures based on interval fields*, IUTAM Symposium on the Vibration Analysis of Structures with Uncertainties. IUTAM Bookseries, vol 27. (2011), pp. 71-83, Springer, Dordrecht

[2] W. Verhaeghe, W. Desmet, D. Vandepitte, D. Moens, *Interval fields to represent uncertainty on the output side of a static FE analysis*, Computer Methods in Applied Mechanics and Engineering, vol. 206 (2013), pp. 50-62

[3] M. Imholz, D. Vandepitte, D. Moens, *Derivation of an input interval field decomposition based on expert knowledge using locally defined basis functions*, Proceedings of the 1st International Conference on Uncertainty Quantification in Computational Sciences and Engineering, UNCECOMP2015, pp. 529-547, May 2015, Crete, Greece

[4] A. Sofi, E. Romeo, O. Barrera, A. Cocks, *An interval finite element method for the analysis of structures with spatially varying uncertainties*, Advances in Engineering Software, vol. 128 (2019), pp. 1-19

[5] W. Gao, D. Wu, K. Gao, X. Chen, F. Tin-Loi, *Structural reliability analysis with imprecise random and interval fields*, Applied Mathematical Modelling, vol. 55 (2018), pp. 49-67

[6] R. Fuller, P. Majlender, *On interactive fuzzy numbers*,Fuzzy Sets and Systems, vol. 143 (2004), pp. 355-369

[7] J.J. Buckley, *On the algebra of intervactive fuzzy numbers*, Fuzzy Sets and Systems, vol. 32 (1989), pp. 291-306

[8] P. Pandian, K. Kavitha, *On correlation between two real interval sets*, Journal of Physics: conf. series 1000 012055

[9] M. Faes, D. Moens, *High dimensional dependence via pair constructions for interval finite models*, USD2018 International conference on uncertainty in Structural Dynamics, September 2018, Leuven

[10] M. Imholz, D. Vandepitte, D. Moens, *Bayesian estimation of interval bounds based on limited data*, USD2018 International conference on uncertainty in Structural Dynamics, September 2018, Leuven

[11] A. O'Hagan, *Polynomial Chaos: A Tutorial and Critique from a Statistician's Perspective*, SIAM/ASA Journal on Uncertainty Quantification, 2013

[12] G. Blatman, B. Sudret, *Adaptive sparse polynomial chaos expansion based on least squares regression*, Journal of Cumputational Physics, vol. 230 (2011), pp. 2345-2367

[13] Z. Ma, J. Wu, Y. Zhang, M. Jiang, *Recursive parameter estimation for load sensing proportional valve based on polynomial chaos expansion*, Engineering Computations, Vol.32 (2015), pp.1343-1371

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

A MACHINE LEARNING APPROACH FOR THE INVERSE QUANTIFICATION OF SET-THEORETICAL UNCERTAINTY

Lars Bogaerts[1], **Matthias Faes**[1], **David Moens**[1]

[1]KU Leuven
Department of Mechanical Engineering
Technology Campus De Nayer, Jan De Nayerlaan 5, St.-Katelijne-Waver, Belgium
{lars.bogaerts, matthias.faes, david.moens}s[at]kuleuven.be

Keywords: Inverse Uncertainty Quantification, Multivariate interval uncertainty, DLR-AIRMOD, Surrogate modelling, dimensionality reduction, machine learning

Abstract. *This paper introduces a machine learning approach for the inverse quantification of set-theoretical uncertainty. Inverse uncertainty quantification (e.g., following Bayesian or interval methodologies) is usually obtained following a process where a distance metric between a set of predicted and measured model responses is iteratively minimized. Consequently, the corresponding computational effort is large and usually unpredictable. Furthermore, often only a limited dataset is available, further complicating the inverse procedure [3]. To overcome these issues, a machine learning approach is proposed to predict the uncertainty in selected model parameters given a limited dataset comprising measured responses.*

To achieve this, machine learning is applied to train a Neural Network that is able to predict model parameter uncertainty, presented a limited set of measured responses, following a set-theoretical approach. This Neural Network is trained by means of a numerically generated data set that captures typical uncertainty in the model parameters. Also, the application of dimension-reduction techniques to aid this inverse quantification are studied. The developed method is applied to the well-known DLR AIRMOD test structure and the results are compared to literature data.

1 INTRODUCTION

Numerical modelling techniques are the backbone in practically all branches of science and technology, from academia to industry. On top of this, it is already proven that non-deterministic approaches are required to cope with the relatively large amount of uncertainty in the input data for these models, such as model parameters, boundary conditions or geometric variables. The ability to include non-deterministic properties is of great value to asses the reliability of a designed structure realistically. This can aid a design to be optimised for robust behaviour under varying external influences. A popular concept in this context is the interval approach, where uncertainties are considered to be contained within a predefined range, consisting of a lower and upper bound. However, a large degree of conservatism on these bounds to prevent premature failure is not the proper solution, but often necessary in case insufficient data are available. This inherently leads to an economic cost, as well as far-from-optimal parameters such as thickness or other weight-affecting parameters which are vital in sectors as automotive, aerospace [6].

In order to assess input data based on observed experimental data, inverse uncertainty quantification (iUQ) aims to quantify the uncertainty in input parameters such that the discrepancies between model output and observed experimental data is minimized. The standard approach for iUQ is still the Bayesian framework, of which the performance is proven numerous times empirically and in special cases even theoretically [13, 18]. A methodology to perform such inverse uncertainty quantification for multivariate interval uncertainty was introduced first in [4], and further extended towards interval fields in [5]. This method is based on the convex hull concept, to represent the dependent uncertain output quantities of an interval FE model. This convex hull is iteratively reconstructed based on iterations on the input interval uncertainty, aiming to minimize the discrepancies with the convex hull over a set of replicated measurement data. This method is illustrated to outperform Bayesian approaches in scarce data conditions [3]. However, the method suffers greatly from the curse of dimensionality due to the required iterative solution of the underlying interval FE problem.

A possible solution to this problem is surrogate modelling, which is typically used to deal with expensive computer codes. A cheap to evaluate surrogate model is constructed to replace the forward model solver. Because such surrogate model is much cheaper to run than the original model solver, it can be used in e.g., a real time monitoring or controlling setting. In the context of inverse uncertainty quantification, Artificial Neural Networks were used in this way in e.g., [3, 13, 14] for both Bayesian and interval iUQ. However, datasets from industrial applicable models often are high dimensional. Since the computation of a convex hull follows an exponential time complexity with its dimension, the dimension should be reduced as to allow a feasible computational time. This requires tools from the fields of big data & machine learning. [10, 15] A broad range of techniques can be adopted, e.g. based on covariance matrix decompositions, active subspace methods, manifold learning or autoencoders have been introduced in recent years. [17]

However for all these methods to perform iUQ, still a considerable computational budget is required to perform the quantification, because still numerous model evaluations are required due to the iterative nature of both the Bayesian and interval iUQ procedures. This paper presents a methodologically new approach for inverse uncertainty quantification in an interval context. The core idea is to train a Neural Network as inverse surrogate model based on the forward FE code for this dataset in combination with deep autoencoder networks to perform dimensionality reduction on the dataset. The DLR AIRMOD test structure with corresponding data set [7] is used to validate the methodology, and the results are compared to those published in literature [3, 13]. This paper is organized as follows. Section 2 elaborates the general setup of the problem in this work. Section 2.2 discusses the task of autoencoders for the dimensionality reduction of the dataset. Section 2.1 describes the proposed technique for a surrogate model, subsequent to the dimensionality reduction in precious section. Section 3 illustrates the performance of this methodology based on the DLR-AIRMOD test structure data. Finaly, Section 4 wraps up the conclusions.

2 METHODOLOGY

The goal in this work is to quantify the interval uncertainty in a set of input parameters of a model, based on limited experimental data. The quantification is performed by the inverse training of a surrogate modelling architecture to predict the model's parameters x based on a set of measured responses y. Let y^m be a set of data on the responses of the structure under consideration. These data are acquired by e.g. an experimental campaign. Since they can be high-dimensional, in general a reduction of their dimension is necessary to be handled by the inverse methodology. Therefore, these values pass through an autoencoder (AE) that is trained on beforehand based on results from a forward FE solver to create a low-dimensional representation y^r of these data. Then, based on this lower-dimensional representaiton of the data, a Neural Network processes the resulting data, aiming at reconstructing the input data $\tilde{x}$ within interval bounds. Both the autoencoder and the artificial Neural Network are offline trained based on a Finite Element model of the structure. The proposed workflow is elaborated in figure 1.

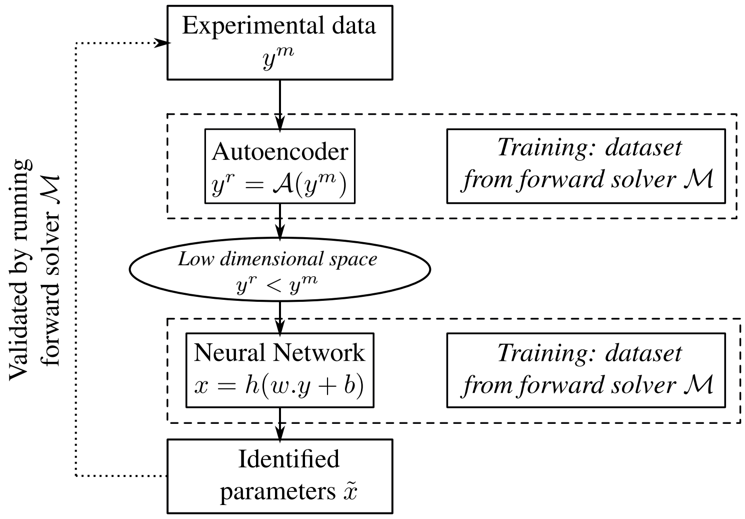

Experimental data
y^m
Autoencoder
$y^r = \mathcal{A}(y^m)$
Training: dataset from forward solver $\mathcal{M}$
Low dimensional space
$y^r < y^m$
Neural Network
$x = h(w.y + b)$
Training: dataset from forward solver $\mathcal{M}$
Identified parameters $\tilde{x}$
Validated by running forward solver $\mathcal{M}$

Figure 1: Flowchart of the proposed methodology

2.1 SURROGATE MODELLING

With this work, we are aiming at the identification of uncertainty in a computationally expensive model $\mathcal{M}$, often based on the Finite Element Method (FEM) including (multi-physics) (partial) differential equations according to:

$$y = \mathcal{M}(x) \tag{1}$$

A forward solver, such as in eq. 1, based on FE solvers is mostly too computationally expensive to solve the inverse problem. A single solution is manageable to compute, but to estimate for an inverse setting, such as the live monitoring and controlling on a production line, the model should first be solved multiple times in an iterative procedure, with iterations up to a magnitude of $\pm 10^5$ before an inverse strategy can be adopted [2]. In this work, an efficient analytical model $\hat{\mathcal{M}}^{-1}$ for inverse uncertainty quantification is introduced to mimic this computational expensive problem. The proposed method is to eliminate the need for fulsome computational iterations by training a surrogate model $\hat{\mathcal{M}}^{-1}$ to quantify the interval uncertainty on the input parameters by means of feeding an Artificial Neural Network (ANN) with measured responses. This results in:

$$\tilde{x} = \left\{ \tilde{x}_i \mid \tilde{x}_i = \hat{\mathcal{M}}^{-1}(y_i), \forall y_i \in y^m \right\} \tag{2}$$

In case $\tilde{x}$ is approximated by an encompassing hypercube, a conservative interval description of the parameter uncertainty is obtained. The ANN surrogate modelling technique is chosen for its effectiveness and versatility [1]. ANN have a guarantee to find a surrogate model description of the output of the network as a close approximation of the real output of the same input value. This universality provides that there is a Neural Network for each possible function [12]. It takes time to set up this approach, due to the choice of several aspects e.g. number of layers, choosing an activation function and the training of the network. Once the network architecture is set and trained, the algorithm can provide responses at a fraction of the required computational cost of running the full numerical model. Furthermore, such prediction is obtained not only on the data used for training, but also on new (experimental) data, which is often referred to as the *out-of-sample extension*.

Neural Networks (NN) are a powerful set of algorithms which can be trained from an input dataset towards an output dataset, regardless of the physical meaning of each individual variable. In an ANN are several layers that are passed by the data in the aim of converging to the output dataset. Each layer in between the first and final layer is defined as hidden layer. By varying the number and size of the hidden layers, an optimum between the accuracy of the output and complexity of the network can be found. The size of the first and last layer is fixed, determined by the dimension of these layers. An ANN defines a function $f : X \rightarrow Y$, where $f(x)$ is a composition of multiple weighted functions $f_n^l(x)$, where l is the layer and n the neuron in the layer l. Often a non-linear weighted sum is used for the composition of the network as follows:

$$\hat{y} = f(x) = K\left(\sum_i w_i.g_i(x)\right) \tag{3}$$

where w_i is a vector of weights to be updated during the training, K the activation function and $g_i(x)$ the elements of the design matrix, which may include other powers or function of x. Since overtraining is a common problem in ANNs, especially when large network achitectures are considered, Bayesian regularization is used in the activation function to attempt to overcome overtraining. Bayesian Regularized ANNs (BRANNs) incorporate Bayes' theorem in to the regularization scheme, and have proven their merit in multivariate interval analysis [6]. The training of the (BR)ANN, also referred to as the learning stage, is typically regulated by back-propagation. This involves comparing the output $\hat{y}$ of the network with the output it is meant to produce y, and using the difference between it to modify the weights w_i of all the connection in the network. Back-propagation ensures that the network learns the correct weights, as to reduce L, the loss function that tells the difference between actual y and intended output $\hat{y}$, equivalent to the mean square error formulation:

$$L(y, \hat{y}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{4}$$

To prevent overfitting, a reguralisation term is added to eq. (4), and both terms are weighted by two hyperparameters α and β:

$$S(w) = \beta \sum_{i=1}^{n} [y_i - f(X_i)]^2 + \alpha \sum_{j=1}^{N_W} w_j^2 \tag{5}$$

with N_W the number of weights. When $\beta >> \alpha$, the network will drive the mean squared error to a lower value. Conversely, when $\alpha >> \beta$, the network weights and biasses will be smaller as compared to a non-regularised performance function, forcing the network response to be smoother. Hence, the former case tends towards a perfect representation of the training data, albeit with the risk of performing bad on new data, whereas the latter aims at a better generalisation performance of the ANN. Specifically, this training is performed following a Bayesian approach, where the weights w and biasses b are modelled as random variables, and identified following a Bayesian approach that minimises L. The regularisation parameters ξ and χ are related to the variances of the random weights and biases, and are also found by performing Bayesian estimation [9].

Since the amount of parameters is rather large in many cases, the number of layers increases most of the times, because the number of weights, which increases when the input layer exists of more parameters and therefore increases the number of connections, should be able to reconstruct the data. This might make it difficult to train the network and often requires also a large amount of samples. This leads to the need of dimensionality reduction techniques, as elaborated in section 2.2. As a conclusion, the main purpose for the choice of a BRANN is the low computational effort, the low risk of overfitting and the required time to test data in this model after it is set up. It also provides an out-of-sample extension for new data, which is an added benefit in several final applications, e.g. live monitoring.

2.2 DIMENSION REDUCTION

An important requirement to achieve the objective of establishing an inverse surrogate model that only requires marginal computational effort and that can process new data, is the ability to embed new high-dimensional data points into an existing low-dimensional data representation. Autoencoders have this important property, which is not lost in combination with the proposed surrogate modelling technique, neural networks [11]. Once the network, consisting of the proposed surrogate model in combination with an autoencoder is trained, new data can easily be projected by the autoencoder from a high-dimensional space into a low-dimensional space, as the trained network defines this transformation. Linear techniques such as PCA lack the ability of a parametric out-of-sample extension. Therefore they need some sort of interpolation based on the linear mapping that they apply on the original data, to guide through the projection onto the lower dimension.

Multilayer autoencoders are feed-forward neural networks with an odd number of hidden layers. The middle layer has d nodes, and the input and output layer have D nodes, with $d < D$.

$$X^D \overset{Encoder}{\rightarrow} Y^d \overset{Decoder}{\rightarrow} X'^D \tag{6}$$

The network is trained such that the mean squared error is minimized between the input and output layer (eq. 4). Ideally the input and output layer from the autoencoder, both with D-dimensions, are equal ($X \approx X'$). Training this network leads to a dataset Y in the middle layer with a d-dimensional representation of the original data, preserving as much structure as possible from the dataset X with D-dimensions. This separates the autoencoder in an input layer, a decoder, the middle layer with $d < D$, an encoder and the reconstructed layer X'. The reconstruction part of the autoencoder makes it of a supervised technique [17]. After the autoencoder is trained, the schematic model can be reduced, resulting in dataset Y:

$$X^D \overset{Encoder}{\rightarrow} Y^d \tag{7}$$

Before reducing the dimensionality, the intrinsic dimension d_i of the dataset should be estimated. This dimension represents how many variables are needed to represent the full dataset, thus $0 \leq d_i \leq D$. Regularly used geometric methods exploit the intrinsic geometry of the dataset and are mostly based on fractal dimensions or nearest neighbour distances [16]. Perhaps the most popular fractal dimension is the correlation dimension. Given a set S$_n$ = $\{$x$_1$, . . . , x$_n\}$ in a metric space, the correlation dimension is defined as:

$$D_C \equiv \lim_{n\to\infty} \lim_{r\to\infty} \frac{log\ C_m(r)}{log\ r} \tag{8}$$

with:

$$C_m(r) = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} I\left\{\|x_j - x_i\| \leq r\right\} \tag{9}$$

where I is the indicator function, r the number of even intervals in the high dimensional hypercube and n the number of data points. The correlation dimension is then estimated by plotting log $C_n(r)$ against log r and estimating the slope of the linear part of the curve until a cut-off is achieved. This leads to a dimension, where data, embedded in a high-dimensional space, can be efficiently summarized in a space of a much lower dimension, without the loss of vital information.

Dimensionality reduction techniques have a high added value for the proposed method, as they do not only reduce the size of the input layer for the ANN, but also the amount of hidden layers is reduced. This leads to a more efficient network, resulting in less complex training and more efficient testing functions.

3 CASE STUDY: DLR-AIRMOD TEST STRUCTURE

3.1 Introduction

The main objective in this work is to illustrate the performance of the developed approach using the challenging DLR AIRMOD data set [7]. The test case is an ideal example to fit into the methodology as described in figure 1. Since both the dimensionality reduction technique and the surrogate model require a set of samples to be trained, a set of input parameters and corresponding results from a Finite Element solver are included. A set of actual measurement data, representing y^m is available to test the methodology and identify the corresponding input parameters $\tilde{x}$.

3.2 Model introduction

The DLR-AIRMOD structure is a scaled replica of the GARTEUR SM-AG19 benchmark air-plane model. A set of 18 parameters including support and joint stiffness values, as well as mass parameters are selected for the identification, in correspondence with literature on the subject ([7, 13]). The DLR-AIRMOD structure, with corresponding dataset is selected due to its challenging nature and the elaborate literature on the subject. The dataset includes 18 input parameters consisting of mass and stiffness values and as an output, 30 eigenmodes corresponding with inital FE estimates and measured eigenfrequencies. According to literature, several techniques are already applied on this case, where results were compared in terms of obtaining information and accuracy [2, 8].

Previous work on the DLR-AIRMOD test structure includes results achieved by interval methods and Bayesian model updating. Interval methods providing the analyst with responses for respectively if the analyst is only interested in bounds on the uncertain parameters. Bayesian model updating is optimal if the analyst is interested in a complete description of the (joint-)plausibility, including correlation and multi-modal descriptors. It can be noted that the model includes some challenges, including asymmetric modal behaviour and closely spaced modes. Based on previous work, the included set of modes used in this methodology includes the 1^{st}-8^{th}, 10^{th}- 12^{th}, 14^{th}, 19^{th} and 20^{th} mode. These 14 modes are selected to be consistent with literature on the subject [7]. More detail on the model and its eigenmodes can be found in [3].

3.3 Inverse Neural Net quantification based on reduced dimensionality

For an inverse UQ with out of sample extension, the pre-selected frequencies resulting from the FEM model are used as input for an autoencoder to reduce its dimension. The FEM model input data is generated by the Monte Carlo sampling technique in a uniform distribution with $\pm 100\%$ bounds on the nominal parameters [7]. The intrinsic dimension is derived based on Eq. 8. Figure 2 illustrates the MSE in function of the dimensions whereto the autoencoder is trained. The reduced dimensionality is chosen to be d =11 due to result of the proposed intrinsic dimensionality, with a MSE of 4,37.10^{-6}. All training computations are made using a single-thread of an Intel Xeon E5 @ 3.7 Ghz, taking a total time of 23.037 s to reach the best training performance, as illustrated in figure 3.

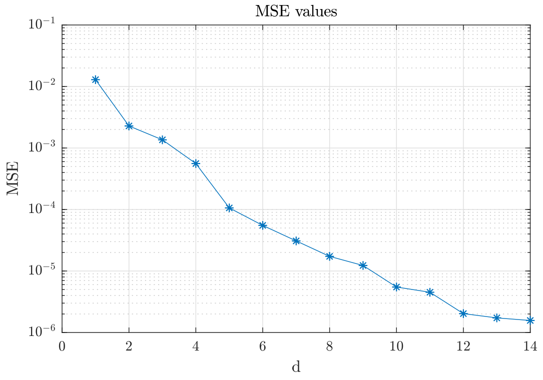


Figure 2: MSE values for each reduced dimension

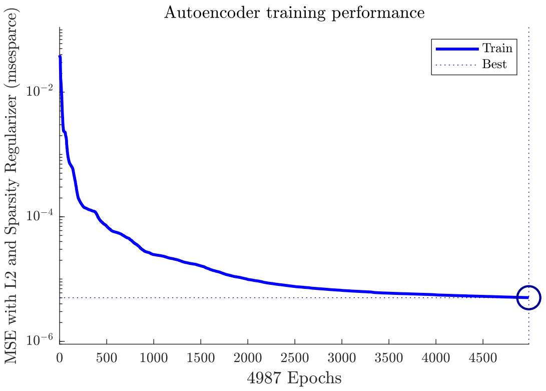


Figure 3: Training performance for $d = 11$

Figure 4 illustrates the correlation matrix from the pre-selected eigenmodes from the DLR-AIRMOD test structure after the supervised learning technique. The matrix illustrates the correlation between the original eigenmodes, being results from an offline FE solver, Y and the reconstructed data Y' for each dimension $D = 14$. The diagonal denotes that the auto correlation is high, which proves the performance of the autoencoder in reconstructing the original responses. The correlation values, based on the complete 2000 sample dataset is provided in

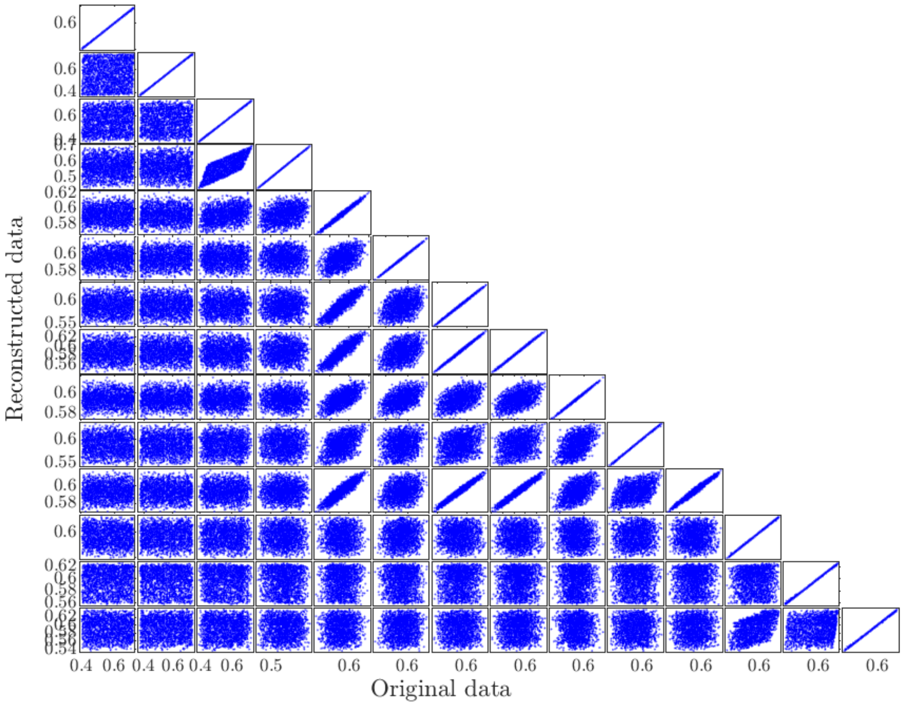


Figure 4: Correlation matrix for the normalised reconstructed results by the autoencoder

table 1. They provide information on how well the reconstructed data Y' resembles the original data Y. The lowest observed correlation is set in bold, with a value of 98,37%. None of the other graphs in the correlation matrix (figure 4) should denote a high correlation for the sake of the autoencoder, because it is only trained on the auto correlation. However it is noted that several frequencies have a rather high correlation with other frequencies. This is an indication that the autoencoder is able to correlate certain responses that are linked to each other due to physical properties in the FE model. This is already a detection for this specific dataset, that the autoencoder will be able to reduce the dimensions, because there is a noticeable correlation with several parameters, which is likely to be retained when the data is reduced. Several notes in figure 4 include: Mode no 7 and 8 have a high correlation, as these are both respectively the asymmetric and symmetric wing torsion. Mode no 19 and 20 have almost no correlation with other eigenmodes, as these eigenmodes are effecting the Horizontal tail piece of the model (respectively horizontal bending and fore-after bending). The autoencoder can therefore successfully reduce the dimensionality, such that the amount of hidden layer connections in the surrogate model will be lower, resulting in a more efficient ANN. The accurately trained autoencoder can perform a dimensionality reduction, according to eq. 7 by only encoding the data into the low dimensional space $d = 11$.

A BRANN surrogate model $\hat{\mathcal{M}}$ is then trained using eq. 2 with the dataset y^r for y (in the lower dimensional space) and $\tilde{x}_\theta$ the input data, of which 16 out of 18 parameters are used as input variables for the FE solver. The selection of these 16 parameters is to be consistent with literature on the subject [3]. The BRANN is trained for each input parameter θ separately and the number of neurons in the hidden layer is increased until the accuracy of the BRANN

Dimension	Correlation
$f1$	99,99
$f2$	99,99
$f3$	99,99
$f4$	99,99
$f5$	99,00
$f6$	99,88
$f7$	99,87
$f8$	99,89
$f10$	99,82
$f11$	99,96
$f12$	**98,37**
$f14$	99,97
$f19$	99,97
$f20$	99,97

Table 1: Correlation from input data en decoded data on the limited set

converges with respect to a separated validation set. The set of trained architectures in the BRANN for each separate AIRMOD model parameter is listed in table 2.

Table 2: Trained BRANN network architectures

$\boldsymbol{\theta}_i$	ANN architecture
1	(11 – 7 – 1)
2	(11 – 7 – 1)
3	(11 – 4 – 1)
4	(11 – 3 – 1)
5	(11 – 10 – 1)
6	(11 – 7 – 1)
7	(11 – 5 – 1)
8	(11 – 4 – 1)
9	(11 – 9 – 1)
10	(11 – 7 – 1)
11	(11 – 7 – 1)
12	(11 – 7 – 1)
13	(11 – 7 – 1)
14	(11 – 7 – 1)
15	(11 – 7 – 1)
16	(11 – 7 – 1)

This yields a complete setup equal to:

$$y \overset{AE}{\rightarrow} y^r \overset{BRANN}{\rightarrow} x. \tag{10}$$

A set of measurement data, achieved from an experimental setup, is used to test the complete trained model and to test its out-of-sample functionality. Figure 5 illustrates the normalised histograms of a Bayesian model updating procedure obtained from literature [3]. The Black arrows indicate the interval bounds obtained by the inverse method of [3]. The red interval bounds indicate the results from the proposed surrogate model. It is noted that for multiple parameters, e.g. 1^{st}-6^{th} and 16^{th}-18^{th}, the surrogate model achieves almost similar results, a few parameters are slightly conservative 8^{th}-11^{th}, 13^{th} and 15^{th}. Also a few parameters are further off, including parameter 7, which is rather conservative, but compared to the normalised histogram, the range from the bounds is even a better approximation with the Bayesian results. Parameter 14 is largely conservative and parameter 12 has a small offset.

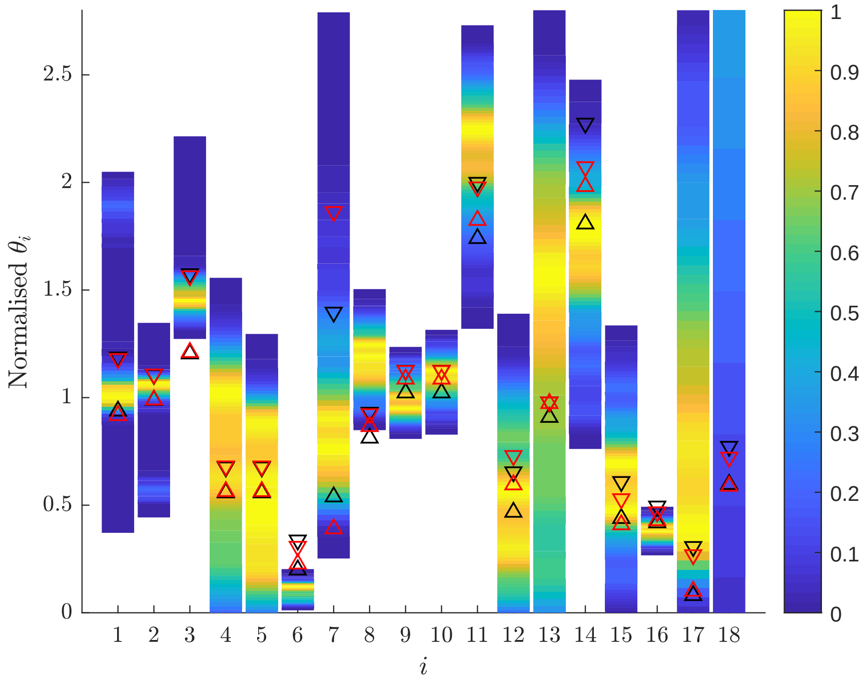


Figure 5: Normalised histograms of the posterior distributions samples, black triangles indicate interval bounds from Multivariate interval quantification, red triangles indicate interval bounds from the surrogate model.

The time required to process 87 data points through the network is marginal. To load new data in the memory 0,409 s is required. The elapsed time to encode the data and passing it through the network is only 0,0284 s. Compared to the inverse method in [3], this approach presents a large increase in numerical efficiency at similar accuracy. Figure 6 illustrates all combinations of the pre-selected eigenmodes obtained by propagating the identified parameters $\tilde{x}$ through their corresponding surrogates of the AIRMOD FE model. They are shown next to the measurement data and the propagated interval fields from [3]. The combination of the considered eigenfrequencies illustrate an almost perfect encapsulation for the 1^{st}-4^{th}. As noted in figure 5, a few parameters have a small offset. This is noted in all the combinations with parameter 6 & 8. All other eigenfrequencies illustrate a good correspondence with the interval bounds of [3].

4 CONCLUSIONS

In this work, an efficient approach for the inverse quantification of set-theoretical uncertainty with black-box numerical simulation models that eliminates the need for iterative procedures is developed. The proposed methodology trains an inverse machine learning architecture, based on input-output pairs that are generated based on a finite element model of the structure or

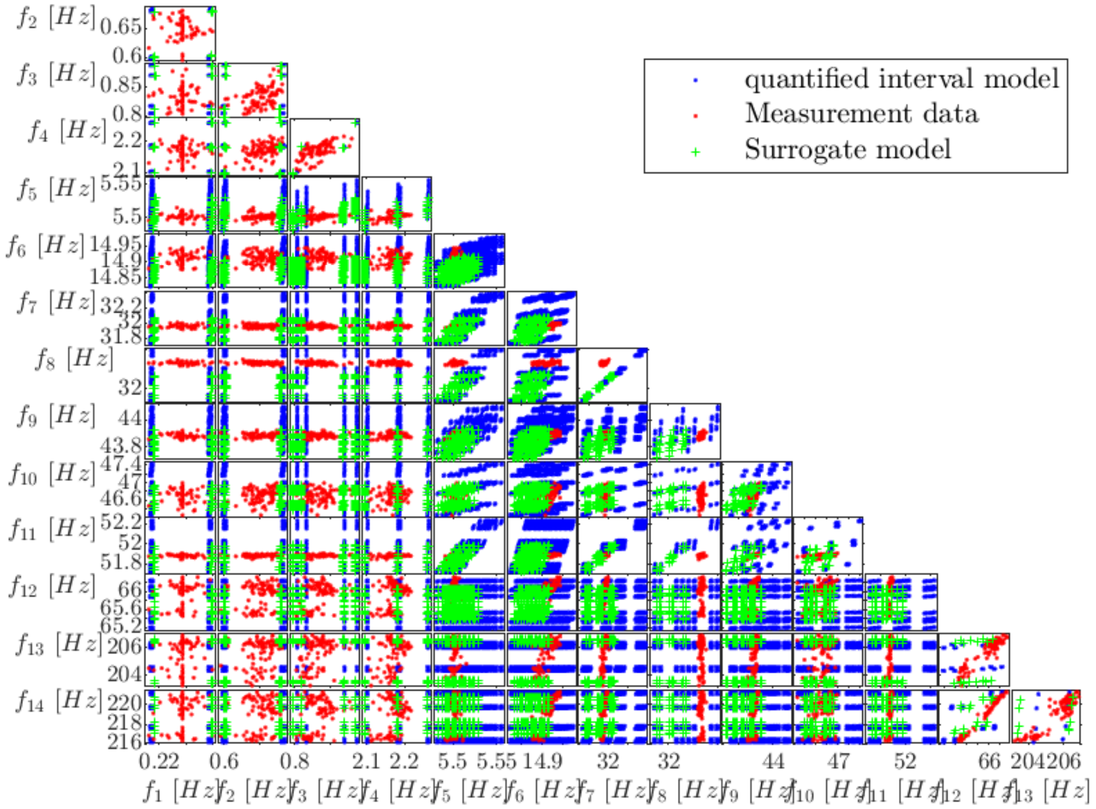


Figure 6: All combinations of considered eigenfrequencies, plotted for experimental data, quantified intervals and the proposed surrogate modelling technique

system under consideration. When the network is trained, experimental data can be processed to identify input parameters. Furthermore, advanced non-linear dimension reduction is used to generalize the method to models have large numbers of inputs and outputs.

The proposed method is applied on a case study to illustrate its performance. The DLR-AIRMOD test structure and the related dataset is used for this application due to challenging nature and availability of benchark solutions. The inverse surrogate modelling method proves to be fast and corresponds well to the original data. The computational time to process experimental time is marginal, making the proposed model ideal for settings such as live monitoring or model updating.

However it is noted that a few ANN were unable to match the model on which the surrogate is trained accurately. Hence the result is more conservative or there is a slight offset between the eigenfrequencies that are obtained when the identified intervals are propated through the FE model and the experimental data. Future work includes the extension of propagating missing data to make the methodology more robust under extremely scarce data conditions.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the Research Foundation Flanders (FWO) for the support in the framework of the research project G0C2218N and the post-doctoral grant 12P359N of Matthias Faes.

REFERENCES

[1] ALANIS, A., ARANA-DANIEL, N., AND LOPEZ-FRANCO, C. *Artificial Neural Networks for Engineering Applications*, 1 ed. 2019.

[2] BROGGI, M., FAES, M., PATELLI, E., GOVERS, Y., MOENS, D., AND BEER, M. Comparison of Bayesian and interval uncertainty quantification: Application to the AIRMOD test structure. *2017 IEEE Symposium Series on Computational Intelligence, SSCI 2017 - Proceedings 2018-Janua* (2018), 1–8.

[3] FAES, M., BROGGI, M., PATELLI, E., GOVERS, Y., MOTTERSHEAD, J., BEER, M., AND MOENS, D. A multivariate interval approach for inverse uncertainty quantification with limited experimental data. *Mechanical Systems and Signal Processing 118* (2019), 534–548.

[4] FAES, M., CERNEELS, J., VANDEPITTE, D., AND MOENS, D. Identification and quantification of multivariate interval uncertainty in finite element models. *Computer Methods in Applied Mechanics and Engineering 315* (2017), 896–920.

[5] FAES, M., AND MOENS, D. Identification and quantification of spatial interval uncertainty in numerical models. *Computers and Structures 192* (2017), 16–33.

[6] FAES, M., AND MOENS, D. Multivariate dependent interval finite element analysis via convex hull pair constructions and the Extended Transformation Method. *Computer Methods in Applied Mechanics and Engineering 347* (2019), 85–102.

[7] GOVERS, Y., HADDAD KHODAPARAST, H., LINK, M., AND MOTTERSHEAD, J. E. Stochastic Model Updating of the DLR AIRMOD Structure. 475–484.

[8] GOVERS, Y., HADDAD KHODAPARAST, H., LINK, M., AND MOTTERSHEAD, J. E. A comparison of two stochastic model updating methods using the DLR AIRMOD test structure. *Mechanical Systems and Signal Processing 52-53*, 1 (2015), 105–114.

[9] MACKAY, D. J. Bayesian interpolation. *Neural Comput. 4*, 3 (1992), 415–447.

[10] MAK, K. K., LEE, K., AND PARK, C. Applications of machine learning in addiction studies: A systematic review. *Psychiatry Research 275*, March (2019), 53–60.

[11] MØLLER, M. F. Efficient Training of Feed-Forward Neural Networks. *DAIMI Report Series 22*, 464 (1993).

[12] NIELSEN, M. A. *Neural Networks and Deep Learning*. Determination Press, 2015.

[13] PATELLI, E., BROGGI, M., GOVERS, Y., AND MOTTERSHEAD, J. E. Model Updating Strategy of the DLR-AIRMOD Test Structure. *Procedia Engineering 199* (2017), 978–983.

[14] PATELLI, E., GOVERS, Y., BROGGI, M., GOMES, H. M., LINK, M., AND MOTTERSHEAD, J. E. Sensitivity or Bayesian model updating: a comparison of techniques using the DLR AIRMOD test data. *Archive of Applied Mechanics 87*, 5 (2017), 905–925.

[15] STETCO, A., NENADIC, G., FLYNN, D., BARNES, M., ZHAO, X., DINMOHAMMADI, F., KEANE, J., AND ROBU, V. Machine learning methods for wind turbine condition monitoring: A review. *Renewable Energy 133* (2018), 620–635.

[16] STRANGE, H., AND ZWIGGELAAR, R. *Open Problems in Spectral Dimensionality Reduction*. 2014.

[17] VAN DER MAATEN, L., POSTMA, E., AND VAN DEN HERIK, J. Dimensionality Reduction: A Comparative Review. *TiCC TR 005*, 1 (2009), 1–35.

[18] WU, X., AND KOZLOWSKI, T. Inverse uncertainty quantification of reactor simulations under the Bayesian framework using surrogate models constructed by polynomial chaos expansion. *Nuclear Engineering and Design 313* (2017), 29–52.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

FATIGUE ANALYSIS OF DISCRETIZED STRUCTURES WITH INTERVAL UNCERTAINTIES UNDER STATIONARY RANDOM EXCITATION VIA SURROGATE MODEL

F. Giunta[1], G. Muscolino[1] and A. Sofi[2]

[1] Department of Engineering, University of Messina
Villaggio S. Agata, 98166 Messina Italy
e-mail: fgiunta@unime.it, gmuscolino@unime.it

[2] Department of Architecture and Territory, University “Mediterranea” of Reggio Calabria, Salita Melissari, Feo di Vito, Reggio Calabria 89124, Italy
e-mail: alba.sofi@unirc.it

Abstract

The fatigue analysis of linear discretized structures with uncertain axial stiffnesses modeled as interval variables subjected to stationary multi-correlated Gaussian stochastic excitation is addressed. The key idea is to estimate the interval expected fatigue life by interval extension of an empirical spectral approach proposed by Benasciutti and Tovo [1], called $\alpha_{0.75}$ *- method.*

The range of the interval expected fatigue life may be significantly overestimated by the classical interval analysis due to the dependency phenomenon which is particularly insidious for stress-related quantities. To limit the dangerous effects of the dependency phenomenon, a novel sensitivity-based procedure relying on the combination of the Improved Interval Analysis via Extra Unitary Interval [2] and the Interval Rational Series Expansion [3] is proposed. This procedure allows one to detect the combinations of the bounds of the interval axial stiffnesses which yield the lower bound and upper bound of the interval expected fatigue life for the stress process at critical points of bar connections.

Keywords: Uncertain-but-bounded axial stiffness, *expected fatigue life*, stationary random excitation, *Improved Interval Analysis, Interval Rational Series Expansion*, sensitivity analysis.

1 INTRODUCTION

Fatigue is recognized as one of the primary causes of failure of many structures and mechanical components. Moreover, fatigue failures may have catastrophic consequences since they happen without any warning. Wind action, which is usually modeled as a stationary multi-correlated Gaussian random process, is perhaps the most important cause of fatigue failure of slender/light structures [4]. The approach commonly adopted to predict fatigue effects is to first convert the load into a set of cycles by using a cycle counting method and then to evaluate the total damage by a proper damage accumulation rule (e.g., Palmgren–Miner rule) as a sum of single cycle damage contributions [5].

In the framework of fatigue failure analysis, it has been recognized that time-domain counting algorithms are very expensive. For this reason, frequency-domain approaches are preferred, especially for stationary stochastic excitations (see e.g., [1],[6]). Indeed, such approaches yield exact or approximate analytical expressions of cycle distribution and fatigue damage under a given counting procedure, without requiring the knowledge of the critical stress or strain time-history.

The *rainflow counting* (*RFC*) method is undoubtedly the most popular and used counting algorithm. In fact, this algorithm extracts cycles on the basis of the material memory mechanisms ([1], [7],[8]). Furthermore, it is well-known that, for a *very narrow-band* (*VNB*) stress process, a Rayleigh distribution can be adopted to represent the cycle distribution (see e.g., [9]). In this case, it is reasonable to state that, for a zero-mean process, a stress cycle is formed by a peak and the following symmetrical valley, and the amplitude equals the value of peak. However, it has been recognized that for not *VNB* stress processes, the Rayleigh distribution yields too conservative results and alternative approximate methods have been proposed (see e.g., [10]-[12]). Recently, by performing both experimental and numerical tests, it has been demonstrated [13] that the approach proposed by Benasciutti and Tovo [1], called $\alpha_{0.75}$-method, is very accurate.

As known, uncertainties affecting structural parameters are commonly modeled resorting to well-established probabilistic approaches. When available data are insufficient to identify a proper probabilistic model for the uncertain variables, non-probabilistic approaches, such as convex models, interval models or fuzzy sets theory, can be alternatively applied. To the best of the authors' knowledge, very few studies have been devoted in the literature to fatigue analysis of structures with uncertain parameters modeled resorting to non-probabilistic approaches (see e.g., [14],[15]).

This study presents the extension of an empirical spectral approach proposed by Benasciutti and Tovo [1] to discretized structures with uncertain material properties modeled as interval variables subjected to stationary multi-correlated random excitation. Due to interval uncertainties, all the response quantities, including the *expected fatigue life*, are described by intervals. To ensure safe design, the lower bound of the interval *expected fatigue life* (worst case scenario) needs to be computed.

It has to be emphasized that the main drawback in the evaluation of the range of selected interval stress components is the so-called *dependency phenomenon* [16],[17] which often leads to an overestimation of the interval solution width unacceptable from an engineering point of view. This phenomenon is due to the inability of the *classical interval analysis* to treat multiple occurrences of the same interval variable in an expression as dependent ones. Interval stresses are more sensitive to the *dependency phenomenon* than displacements since their definition involves double occurrence of the same interval variable. In this paper, to reduce overestimation affecting the bounds of the interval *expected fatigue life* for the critical stress process, a novel sensitivity-based procedure stemming from the combination of the *Improved Interval Analysis* (*IIA*) via *Extra Unitary Interval* (*EUI*) [2] and the *Interval Rational Series Expansion* (*IRSE*) [3] is proposed. The key idea is to perform *Sensitivity Analysis* (*SA*) exploiting the *IRSE* to predict the monotonic increasing or decreasing behavior of the *expected fatigue life* for the critical stress process. This approach, herein referred to as *SA* via *IRSE* [18]-[21], provides explicit closed-form relationships for the sensitivities of the *expected fatigue life* to the uncertain parameters. Unlike the approach to interval fatigue analysis recently developed by the authors themselves [15], the proposed method enables to identify the combinations of the endpoints of the uncertain parameters which give the *Lower Bound* (*LB*) and *Upper Bound* (*UB*) of the *expected fatigue life*.

A truss structure with uncertain axial stiffness subjected to wind excitation is selected as case-study. Since the *expected fatigue life* is a monotonic function of the uncertain parameters, for validation purposes, the proposed bounds of the interval *expected fatigue life* are compared with those provided by the classical combinatorial procedure, the so-called *vertex method*.

2 EXPECTED FATIGUE LIFE OF LINEAR STRUCTURES UNDER STATIONARY STOCHASTIC EXCITATION

Amongst all damage rules, the Palmgren-Miner linear damage model [5] is the most popular and used, due to its simplicity. According to this model, derived for constant amplitude tests, fatigue strength is quantified by the number of cycles to failure, *N*, under repeated sinusoidal cycles with amplitude *y*. For many materials, this relation is explicitly given as a straight line in a double-logarithmic diagram (*Y-N* curve):

$$y^k N = C \tag{1}$$

where *k* and *C* are material parameters. For stationary stress processes, under the Palmgren-Miner rule (ignoring mean-value), the expected damage per unit of time (or mean damage intensity) is a constant quantity and can be evaluated as [9]:

$$\mathrm{E}\langle D_Y \rangle = \nu_{Y,a}\, C^{-1} \int_0^\infty y^k\, p_Y(y)\,\mathrm{d}y \tag{2}$$

where $\mathrm{E}\langle\bullet\rangle$ is the stochastic average operator, $p_Y(y)$ is the amplitude distribution of counted cycles, $\nu_{Y,a}$ is the rate of occurrence of counted cycles (i.e. *counted cycles/s*), given as (see e.g., [9]):

$$\nu_{Y,a} = \frac{1}{2\pi}\sqrt{\frac{\lambda_{Y,4}}{\lambda_{Y,2}}};\ \ \lambda_{Y,m} = \int_0^\infty \omega^m G_Y(\omega)\,\mathrm{d}\omega \tag{3}$$

where $\lambda_{Y,m}$ is the spectral moment of order *m* [22] and $G_Y(\omega)$ is the one-sided *Power Spectral Density* (*PSD*) function of the stationary random stress process $Y(t)$. For stationary excitations, the *expected time to failure* (i.e. the *expected fatigue life*) can be estimated as:

$$T_{\mathrm{F}} = \frac{1}{\mathrm{E}\langle D_Y \rangle}. \tag{4}$$

Hence, for a given stationary random process $Y(t)$ (i.e. for a given spectral density), the *mean damage intensity* $\mathrm{E}\langle D_Y \rangle$ and the *expected fatigue life* T_F depend on the expected rate of occurrence of cycles $\nu_{Y,a}$ as well as on the amplitude distribution $p_Y(y)$ that in turn depends on the counted method adopted. Unfortunately, because of the complicated procedure of peak–valley pairing, at present no explicit analytical solution is available for the amplitude distribution as well as for the *mean damage intensity* and *expected fatigue life*. For this reason, all methods existing in the literature are only approximate. Recently, by performing both experimental and numerical tests [13], it has been demonstrated that the $\alpha_{0.75}$- method proposed by Benasciutti and Tovo [1] is one of the most accurate among approximate procedures. According to this method, the *mean damage intensity* $\mathrm{E}\langle D_Y \rangle$ can be evaluated as:

$$\mathrm{E}\langle D_Y \rangle = \frac{\lambda_{Y,0.75}^2}{\lambda_{Y,0}\lambda_{Y,1.5}} \mathrm{E}\langle D_{Y,VNB} \rangle \tag{5}$$

where $\mathrm{E}\langle D_{Y,VNB} \rangle$ is the mean damage intensity under the *very narrow-band* (*VNB*) approximation, evaluated as [9]:

$$\mathrm{E}\langle D_{Y,VNB} \rangle = \nu_{Y,0}\, C^{-1} \left(\sqrt{2\lambda_{Y,0}}\right)^k \Gamma\left(1+\frac{k}{2}\right); \;\; \nu_{Y,0} = \frac{1}{2\pi}\sqrt{\frac{\lambda_{Y,2}}{\lambda_{Y,0}}} \tag{6}$$

$\Gamma(\bullet)$ being the gamma function. Substituting Eq. (5) into Eq. (4), the following expression of the *expected fatigue life*, T_F, is obtained:

$$T_\mathrm{F} = \frac{\pi C}{\sqrt{2^{(k-2)}}\;\Gamma\left(1+\dfrac{k}{2}\right)} \frac{\lambda_{Y,1.5}}{\lambda_{Y,0.75}^2} \sqrt{\frac{\lambda_{Y,0}^{3-k}}{\lambda_{Y,2}}}. \tag{7}$$

It follows that the evaluation of the *expected fatigue life*, T_F, involves four spectral moments of the stationary stress process $Y(t)$.

3 BOUNDS OF THE INTERVAL EXPECTED FATIGUE LIFE

3.1 Interval model of uncertainties

Over the last decades, the interval model has gained increasing popularity as a simple and effective non-probabilistic approach to represent uncertainties occurring in engineering problems. The basic idea is to describe the $i-$th uncertain parameter as an interval variable $\alpha_i^I = [\underline{\alpha}_i, \overline{\alpha}_i] \in \mathbb{IR}$, denoted by the apex *I*, with $\mathbb{IR}$ indicating the set of all closed real interval numbers, while $\underline{\alpha}_i$ and $\overline{\alpha}_i$ are the *Lower Bound* (*LB*) and *Upper Bound* (*UB*) of α_i^I, respectively. Interval variables are also referred to as uncertain-but-bounded.

According to the *classical interval analysis*, the *i*-th real interval variable $\alpha_i^I = [\underline{\alpha}_i, \overline{\alpha}_i]$ is characterized by the midpoint value (or mean), $\alpha_{0,i}$, and the deviation amplitude (or radius), $\Delta\alpha_i$, given by [16]:

$$\alpha_{0,i} = \frac{1}{2}\left(\underline{\alpha}_i + \overline{\alpha}_i\right); \qquad \Delta\alpha_i = \frac{1}{2}\left(\overline{\alpha}_i - \underline{\alpha}_i\right). \tag{8,a,b}$$

Let $\boldsymbol{\alpha}^I = \left[\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}\right] \in \mathbb{IR}^r$ be a bounded set-interval vector of real numbers collecting r interval variables such that $\underline{\boldsymbol{\alpha}} \le \boldsymbol{\alpha} \le \overline{\boldsymbol{\alpha}}$, with $\underline{\boldsymbol{\alpha}}$ and $\overline{\boldsymbol{\alpha}}$ denoting the *LB* and *UB* vectors. In the sequel, $\boldsymbol{\alpha}_0$ and $\Delta\boldsymbol{\alpha}$ will denote the vectors collecting the midpoint values and the deviation amplitudes, $\alpha_{0,i}$ and $\Delta\alpha_i$, respectively, of the interval variables $\alpha_i^I \;\; (i = 1, \ldots, r)$.

The main limitation of the *classical interval analysis* is the so-called *dependency phenomenon* [16], [17] which often yields over conservative estimates of the interval solution which are useless for design purposes. This phenomenon typically arises when the same interval variable occurs more than once in a mathematical expression. Indeed, the *classical interval analysis* in unable to keep track of interval variables throughout calculations. To reduce conservatism caused by the *dependency phenomenon*, recently the so-called *Improved Interval Analysis* (*IIA*) *via Extra Unitary Interval* (*EUI*) [2] has been proposed. This approach relies on the introduction of a particular unitary interval, called *EUI*, $\hat{e}_i^I \triangleq \left[-1, +1\right]$, $(i = 1, 2, ..., r)$, which does not obey to the rules of the *classical interval analysis*. According to the *IIA* via *EUI*, the following *affine form* definition for the i-th interval variable α_i^I is assumed:

$$\alpha_i^I = \alpha_{0,i} + \Delta\alpha_i \hat{e}_i^I, \qquad (i = 1, 2, \ldots, r). \tag{9}$$

For symmetric interval variables with $\overline{\alpha}_i = -\underline{\alpha}_i$, so that $\alpha_{0,i} = 0$ and $\Delta\alpha_i = \overline{\alpha}_i = -\underline{\alpha}_i$, the previous equation reduces to:

$$\alpha_i^I = \Delta\alpha_i \hat{e}_i^I. \tag{10}$$

3.2 Equations of motion

Let us consider a quiescent *n*-DOF linear structure subjected to a stationary multi-correlated Gaussian stochastic process $\mathbf{F}(t)$. Let $\rho_j = E_j A_j / L_j$ be the axial stiffness of the *j*-th element, where E_j, A_j and L_j are the Young's modulus, cross-sectional area and length of the element, respectively. Without loss of generality, attention is focused on structures with uncertain axial stiffness. Specifically, it is assumed that $r \le m$ elements are characterized by uncertain-but-bounded axial stiffness i.e.:

$$\rho_j^I = \rho_{0,j}(1 + \alpha_j^I) = \rho_{0,j}(1 + \Delta\alpha_j \hat{e}_j^I), \qquad (j = 1, 2, \ldots, r \le m) \tag{11}$$

where $\rho_{0,j} = E_{0,j} A_{0,j} / L_{0,j}$ is the nominal value of the axial stiffness of the *j*-th element; α_j^I is the dimensionless fluctuation of the uncertain axial stiffness around the nominal value, herein modeled as a symmetric interval variable. According to the *IIA*, α_j^I is expressed as in Eq. (10) in terms of the associated *EUI*, $\hat{e}_j^I$, and deviation amplitude $\Delta\alpha_j$ with $\Delta\alpha_j < 1$ in order to ensure always positive values of the uncertain axial stiffness.

The stiffness matrix of the structure is a $n \times n$ interval matrix defined as follows:

$$\mathbf{K}^I \equiv \mathbf{K}(\boldsymbol{\alpha}^I) = \mathbf{S}^{\mathrm{T}} \mathbf{E}(\boldsymbol{\alpha}^I) \mathbf{S} \tag{12}$$

where $\boldsymbol{\alpha}^I$ is the interval vector collecting the fluctuations α_j^I of the axial stiffnesses around the nominal value; $\mathbf{S}^{\mathrm{T}}$ is the $n \times m$ equilibrium matrix; $\mathbf{E}^I \equiv \mathbf{E}(\boldsymbol{\alpha}^I)$ is the $m \times m$ interval diagonal internal stiffness matrix, given by [23]

$$\mathbf{E}^I \equiv \mathbf{E}(\boldsymbol{\alpha}^I) = \mathbf{E}_0 + \sum_{j=1}^{r} \Delta\alpha_j \hat{e}_j^I \, \mathbf{l}_{E,j} \mathbf{l}_{E,j}^{\mathrm{T}}, \tag{13}$$

where $\mathbf{E}_0 = \mathrm{Diag}\begin{bmatrix} \rho_{0,1} & \rho_{0,2} & \cdots & \rho_{0,m} \end{bmatrix}$ is the nominal internal stiffness matrix; $\mathbf{l}_{E,j}$ is a $m-$vector having zero entries except the j-th which is equal to $\sqrt{\rho_{0,j}}$, such that the dyadic product $\mathbf{l}_{E,j}\mathbf{l}_{E,j}^{\mathrm{T}}$ gives a change of rank one to the nominal internal stiffness matrix.

Taking into account Eq. (13), the interval stiffness matrix in Eq. (12) can be rewritten as sum of its nominal value, $\mathbf{K}_0$, plus r rank-one interval modifications, i.e.:

$$\mathbf{K}^I = \mathbf{K}_0 + \sum_{j=1}^{r} \Delta\alpha_j \, \hat{e}_j^I \, \mathbf{K}_j = \mathbf{K}_0 + \sum_{j=1}^{r} \Delta\alpha_j \, \hat{e}_j^I \, \mathbf{w}_j \, \mathbf{w}_j^{\mathrm{T}} \tag{14}$$

where $\mathbf{K}_j = \mathbf{w}_j \, \mathbf{w}_j^{\mathrm{T}}$ is a rank-one matrix and

$$\begin{aligned} \mathbf{K}_0 &= \mathbf{S}^{\mathrm{T}} \, \mathbf{E}_0 \, \mathbf{S}\,; \\ \mathbf{w}_j &= \mathbf{S}^{\mathrm{T}} \, \mathbf{l}_{E,j}. \end{aligned} \tag{15a,b}$$

The equations of motion of the structure with interval axial stiffness subjected to a stationary multi-correlated Gaussian stochastic process $\mathbf{F}(t)$ take the following form:

$$\mathbf{M}\ddot{\mathbf{U}}^I(t) + \mathbf{C}^I\dot{\mathbf{U}}^I(t) + \mathbf{K}^I\mathbf{U}^I(t) = \mathbf{F}(t) \tag{16}$$

where $\mathbf{M}$ is the $n \times n$ mass matrix, herein assumed deterministic; $\mathbf{K}^I \equiv \mathbf{K}(\boldsymbol{\alpha}^I)$ is the interval stiffness matrix given by Eq. (14); $\mathbf{U}^I(t) \equiv \mathbf{U}(\boldsymbol{\alpha}^I, t)$ is the interval stationary Gaussian vector process of displacements; and a dot over a variable denotes differentiation with respect to time t. The Rayleigh model is adopted to define the interval damping matrix, i.e.:

$$\mathbf{C}^I \equiv \mathbf{C}(\boldsymbol{\alpha}^I) = c_0\mathbf{M} + c_1\mathbf{K}^I \tag{17}$$

where c_0 and c_1 are the Rayleigh damping constants herein evaluated setting the uncertain parameters equal to their nominal values. Taking into account the decomposition (14) of the interval stiffness matrix, the interval damping matrix in Eq. (17) can be expressed as sum of the nominal value $\mathbf{C}_0 = c_0\mathbf{M} + c_1\mathbf{K}_0$ plus a superposition of rank one matrices, i.e.:

$$\mathbf{C}^I \equiv \mathbf{C}(\boldsymbol{\alpha}^I) = \mathbf{C}_0 + c_1 \sum_{j=1}^{r} \Delta\alpha_j \hat{e}_j^I \mathbf{w}_j \mathbf{w}_j^{\mathrm{T}}. \tag{18}$$

The external load vector $\mathbf{F}(t)$ in Eq. (16), herein modeled as a stationary multi-correlated Gaussian random process, is fully characterized, from a probabilistic point of view in the frequency domain, by the mean-value vector, $\boldsymbol{\mu}_{\mathbf{F}} = \mathrm{E}\langle \mathbf{F}(t) \rangle$, and the one-sided *PSD* function matrix of the fluctuating part $\mathbf{G}_{\tilde{\mathbf{X}}_{\mathbf{F}}\tilde{\mathbf{X}}_{\mathbf{F}}}$ [3].

The interval stationary Gaussian stochastic response process $\mathbf{U}^I(t)$ ruled by the equations of motion in Eq. (16) is completely characterized in the frequency domain by the mean-value vector, $\boldsymbol{\mu}_{\mathbf{U}}^I \equiv \boldsymbol{\mu}_{\mathbf{U}}(\boldsymbol{\alpha}^I)$, and the one-sided *PSD* function matrix, $\mathbf{G}_{\mathbf{UU}}^I(\omega) \equiv \mathbf{G}_{\mathbf{UU}}(\boldsymbol{\alpha}^I, \omega)$ which have an interval nature [19]-[21]. The generic response quantity, $Y^I(t) \equiv Y(\boldsymbol{\alpha}^I, t)$ (e.g., displacement, strain or stress at a critical point), can be determined from the knowledge of the displacement vector $\mathbf{U}^I(t) \equiv \mathbf{U}(\boldsymbol{\alpha}^I, t)$ as follows:

$$Y(\boldsymbol{\alpha}^I, t) = \mathbf{q}^{\mathrm{T}}(\boldsymbol{\alpha}^I)\,\mathbf{U}(\boldsymbol{\alpha}^I, t) \tag{19}$$

where $\mathbf{q}(\boldsymbol{\alpha}^I) \equiv \mathbf{q}^I$ is a vector collecting the combination coefficients relating the response process $Y^I(t)$ to $\mathbf{U}^I(t)$. Such a vector may depend on the uncertain parameters, as happens, for instance, when stress processes are considered. The complete probabilistic characterization of the interval stationary Gaussian random response process in Eq. (19), expressed as $Y^I(t) = \mu_Y^I + \tilde{Y}^I(t)$, requires the knowledge of the interval mean-value, μ_Y^I, and the interval one-sided *PSD* function, $G_{\tilde{Y}\tilde{Y}}^I(\omega) \equiv G_{YY}^I(\omega)$ of the zero-mean random process $\tilde{Y}^I(t)$. It is worth noting that, due to multiple occurrences of the same interval variable into Eq. (19), the bounds of the interval mean-value and one-sided *PSD* density function of the response process $Y^I(t)$ may be significantly overestimated by the *classical interval analysis*. The latter, indeed, treats multiple occurrences of the same interval variable in an expression as independent ones [16],[17]. By inspection of Eq. (19), it is readily inferred that the number of occurrences of the same interval variable is larger when the vector $\mathbf{q}(\boldsymbol{\alpha}^I) \equiv \mathbf{q}^I$, collecting the combination coefficients, depends on the interval parameters. It follows that interval stress quantities are more vulnerable to the *dependency phenomenon* than displacements. Without loss of generality, attention is herein focused on the stationary normal stress process in the *h*-th element which, according to Eq. (19), can be written as [15]:

$$Y_h(\boldsymbol{\alpha}^I, t) \equiv Y_h^I(t) = \mathbf{q}^{\mathrm{T}}\left(\alpha_h^I\right)\mathbf{U}(\boldsymbol{\alpha}^I, t) = (1 + \Delta\alpha_h \hat{e}_h^I)\,\frac{\rho_{0,h}}{A_{0,h}}\,\mathbf{s}_h^{\mathrm{T}}\,\mathbf{U}(\boldsymbol{\alpha}^I, t) \tag{20}$$

where $\mathbf{s}_h^{\mathrm{T}}$ is the *h*-th row of the compatibility matrix $\mathbf{S}$ (see Eq. (12)).

3.3 Interval expected fatigue life

As shown in the previous sections, the statistics of the response of a randomly excited structure with interval axial stiffness are described by intervals. It follows that, in the context of spectral approaches to fatigue analysis, the *expected fatigue life* of the structure, which depends on certain spectral moments of the critical stress process, turns out to be an interval quantity too.

By interval extension of the solution provided by the $\alpha_{0.75}$ – method [1] reported in Eq. (7), the following expression of the interval *expected fatigue life*, T_{F,Y_h}^I, for the stationary normal stress random process $Y_h^I(t)$ is obtained:

$$T_{\mathrm{F},Y_h}^I = [\underline{T}_{\mathrm{F},Y_h}, \overline{T}_{\mathrm{F},Y_h}] = \frac{\pi C}{\sqrt{2^{(k-2)}}\;\Gamma\left(1+\dfrac{k}{2}\right)}\,\frac{\lambda_{Y_h,1.5}^I}{\left(\lambda_{Y_h,0.75}^I\right)^2}\sqrt{\frac{\left(\lambda_{Y_h,0}^I\right)^{3-k}}{\lambda_{Y_h,2}^I}} \tag{21}$$

where the interval spectral moments of $Y_h^I(t)$ of order $\ell = 0, 0.75, 1.5, 2$ appear. It is readily inferred that Eq. (21) contains multiple instances of the same interval variables and, therefore, the range of the interval *expected fatigue life* may be affected by serious overestimation. In order to efficiently evaluate the bounds, the interval *expected fatigue life* in Eq. (21) may be viewed as a function of the four interval spectral moments $\lambda^I_{Y_h,\ell}$ ($\ell = 0, 0.75, 1.5, 2$). Under this assumption, the *LB* and *UB* of T^I_{F,Y_h} can be estimated from Eq. (21) setting the interval spectral moments $\lambda^I_{Y_h,\ell}$ equal to appropriate combinations of their bounds. However, this approach, recently adopted by the authors [15], often yields very conservative results. To reduce overestimation, in the present paper, a novel procedure based on sensitivity analysis is proposed.

It is worth mentioning that, since the interval *expected fatigue life*, T^I_{F,Y_h}, is a monotonic function of the generic uncertain parameter α_i^I, its “exact” bounds can be computed by applying the *vertex method*. The latter evaluates the *LB* and *UB* of the interval *expected fatigue life* as the minimum and maximum among the values corresponding to all possible combinations of the bounds of the r interval parameters α_i^I, say 2^r. Since 2^r stochastic analyses are needed, the computational costs become unaffordable as the number r of uncertain parameters increases. Conversely, the proposed method is able to handle an arbitrary number of uncertainties since it does not require repeated stochastic analyses, as will be shown in the sequel.

3.4 Proposed sensitivity-based procedure

In order to evaluate the bounds of the interval *expected fatigue life* of a selected stress process of linear structures with uncertain-but-bounded axial stiffness subjected to stationary multi-correlated stochastic excitation, in this section a novel sensitivity-based procedure is proposed. The key idea is to derive the interval one-side *PSD* function of a selected stress random process in approximate explicit form by applying the *Interval Rational Series Expansion* (*IRSE*) [3], [18]-[21]. The latter may be viewed as an effective surrogate model of the interval *frequency response function* (*FRF*) matrix. Then, the bounds of the interval *expected fatigue life* are evaluated performing *Sensitivity Analysis* (*SA*) which allows us to predict the influence of each uncertain parameter on fatigue failure. To this aim, let the dimensionless fluctuations of the interval axial stiffnesses around the nominal value be treated as variable parameters $\alpha_i \in [-\Delta\alpha_i, \Delta\alpha_i]$ collected into the vector $\boldsymbol{\alpha} = [\alpha_1 \quad \alpha_2 \quad \dots \quad \alpha_r]^{\mathrm{T}}$. By direct differentiation of Eq. (21), taking into account that the spectral moments depend on the parameters $\alpha_i \in [-\Delta\alpha_i, \Delta\alpha_i]$, ($i = 1, 2, \dots, r$), the following expression of the sensitivity of the *expected fatigue life* $T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})$ for the stress random process $Y_h(\boldsymbol{\alpha}, t)$ with respect to the $i-$th parameter α_i is obtained:

$$\begin{aligned} S_{T_{\mathrm{F},Y_h},i} = \left.\frac{\partial T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})}{\partial \alpha_i}\right|_{\boldsymbol{\alpha}=\mathbf{0}} &= \frac{\pi C}{2\sqrt{2^{(k-2)}}\,\Gamma\left(1+\frac{k}{2}\right)} \frac{1}{\left(\lambda^{(0)}_{Y_h,0.75}\right)^2} \sqrt{\frac{\left(\lambda^{(0)}_{Y_h,0}\right)^{3-k}}{\lambda^{(0)}_{Y_h,2}}} \left\{ 2S_{\lambda_{Y_h,1.5},i} - \frac{4\lambda^{(0)}_{Y_h,1.5}}{\lambda^{(0)}_{Y_h,0.75}} S_{\lambda_{Y_h,0.75},i} \right. \\ &\left. + \frac{(3-k)\lambda^{(0)}_{Y_h,1.5}}{\lambda^{(0)}_{Y_h,0}} S_{\lambda_{Y_h,0},i} - \frac{\lambda^{(0)}_{Y_h,1.5}}{\lambda^{(0)}_{Y_h,2}} S_{\lambda_{Y_h,2},i} \right\}, \qquad (i = 1, 2, \dots, r) \end{aligned} \tag{22}$$

where $\lambda^{(0)}_{Y_h,\ell}$ are the nominal values of the spectral moments, defined as:

$$\lambda^{(0)}_{Y_h,\ell} = \int_0^\infty \omega^\ell \, G^{(0)}_{Y_hY_h}(\omega) \mathrm{d}\omega, \qquad (\ell = 0\,, 0.75\,, 1.5\,, 2) \tag{23}$$

with $G^{(0)}_{Y_hY_h}(\omega)$ denoting the nominal one-sided *PSD* function of the stress random process $Y_h(\boldsymbol{\alpha},t)$, given by [24]:

$$G^{(0)}_{Y_hY_h}(\omega) = \frac{\rho^2_{0,h}}{A^2_{0,h}} \mathbf{s}^{\mathrm{T}}_h \mathbf{H}^*_0(\omega) \mathbf{G}_{\tilde{\mathbf{X}}_\mathbf{F}\tilde{\mathbf{X}}_\mathbf{F}}(\omega) \mathbf{H}^{\mathrm{T}}_0(\omega) \mathbf{s}_h \tag{24}$$

where $\mathbf{s}^{\mathrm{T}}_h$ is the h-th row of the compatibility matrix $\mathbf{S}$ and $\mathbf{H}_0(\omega)$ is the *FRF* matrix of the nominal system, defined as:

$$\mathbf{H}_0(\omega) = \left[-\omega^2 \mathbf{M} + \mathrm{j}\omega \mathbf{C}_0 + \mathbf{K}_0 \right]^{-1} \tag{25}$$

where $\mathrm{j} = \sqrt{-1}$ denotes the imaginary unit. Furthermore, in Eq. (22), $S_{\lambda_{Y_h,\ell},i}$ denotes the sensitivity of the spectral moment of order ℓ of the stress random process $Y_h(\boldsymbol{\alpha},t)$ to the $i-$th parameter α_i

$$S_{\lambda_{Y_h,\ell},i} = \left. \frac{\partial \lambda_{Y_h,\ell}(\boldsymbol{\alpha})}{\partial \alpha_i} \right|_{\boldsymbol{\alpha}=\mathbf{0}} = \int_0^\infty \omega^\ell S_{G_{Y_hY_h},i}(\omega) \, \mathrm{d}\omega, \quad (\ell = 0, 0.75, 1.5, 2) \tag{26}$$

where $S_{G_{Y_hY_h},i}(\omega)$ is the $i-$th sensitivity of the one-side *PSD* function $G_{Y_hY_h}(\boldsymbol{\alpha},\omega)$, $\boldsymbol{\alpha} \in \boldsymbol{\alpha}^I = [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]$, of the normal stress stationary random process $Y_h(\boldsymbol{\alpha},t)$ which is defined as:

$$G_{Y_hY_h}(\boldsymbol{\alpha},\omega) = \frac{\rho^2_{0,h}}{A^2_{0,h}} (1+\alpha_h)^2 \mathbf{s}^{\mathrm{T}}_h \mathbf{H}^*(\boldsymbol{\alpha},\omega) \mathbf{G}_{\tilde{\mathbf{X}}_\mathbf{F}\tilde{\mathbf{X}}_\mathbf{F}}(\omega) \mathbf{H}^{\mathrm{T}}(\boldsymbol{\alpha},\omega) \mathbf{s}_h. \tag{27}$$

By inspection of the previous equation, it is inferred that the evaluation of the sensitivity of the one-sided *PSD* function $G_{Y_hY_h}(\boldsymbol{\alpha},\omega)$ requires the knowledge of the sensitivity of the interval *FRF* matrix $\mathbf{H}(\boldsymbol{\alpha},\omega)$, $\boldsymbol{\alpha} \in \boldsymbol{\alpha}^I = [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}]$. Such a sensitivity can be efficiently evaluated by direct differentiation of a surrogate model derived by applying the *IRSE* [3], [18]-[21]. According to this model, the interval *FRF* matrix can be approximated as follows:

$$\mathbf{H}(\boldsymbol{\alpha},\omega) \approx \mathbf{H}_0(\omega) - \sum_{i=1}^{r} g_i(\omega,\alpha_i) \mathbf{B}_i(\omega), \qquad \boldsymbol{\alpha} \in \boldsymbol{\alpha}^I = [\underline{\boldsymbol{\alpha}}, \overline{\boldsymbol{\alpha}}] \tag{28}$$

where:

$$\begin{aligned} & g_i(\omega,\alpha_i) = \frac{p(\omega)\alpha_i}{1+p(\omega)\alpha_i b_i(\omega)}; \\ & p(\omega) = 1 + \mathrm{j}\omega c_1; \\ & b_i(\omega) = \mathbf{w}^{\mathrm{T}}_i \mathbf{H}_0(\omega) \mathbf{w}_i; \\ & \mathbf{B}_i(\omega) = \mathbf{H}_0(\omega) \mathbf{w}_i \; \mathbf{w}^{\mathrm{T}}_i \mathbf{H}_0(\omega) \end{aligned} \tag{29a-d}$$

where $\mathbf{w}_i$ is the n-vector defined by Eq. (15b). By direct differentiation of Eq. (28), the following approximate explicit expression of the $i-\text{th}$ sensitivity of the interval *FRF* matrix is obtained

$$\mathbf{A}_i(\omega)=\left.\frac{\partial\mathbf{H}(\boldsymbol{\alpha},\omega)}{\partial\alpha_i}\right|_{\boldsymbol{\alpha}=\mathbf{0}}=-p(\omega)\,\mathbf{B}_i(\omega). \tag{30}$$

Then, the sensitivity of the one-sided *PSD* function $G_{Y_hY_h}(\boldsymbol{\alpha},\omega)$ (see Eq. (27)) to the i-th uncertain parameter can be written in explicit form as:

$$S_{G_{Y_hY_h},i}(\omega)=\left.\frac{\partial G_{Y_hY_h}(\boldsymbol{\alpha},\omega)}{\partial\alpha_i}\right|_{\boldsymbol{\alpha}=\mathbf{0}}=2G^{(0)}_{Y_hY_h}(\omega)+\frac{\rho^2_{0,h}}{A^2_{0,h}}\mathbf{s}_h^{\mathrm{T}}\mathbf{Q}_h(\omega)\mathbf{s}_h,\quad i=h;$$

$$S_{G_{Y_hY_h},i}(\omega)=\left.\frac{\partial G_{Y_hY_h}(\boldsymbol{\alpha},\omega)}{\partial\alpha_i}\right|_{\boldsymbol{\alpha}=\mathbf{0}}=\frac{\rho^2_{0,h}}{A^2_{0,h}}\mathbf{s}_h^{\mathrm{T}}\mathbf{Q}_i(\omega)\mathbf{s}_h,\quad i\neq h \tag{31a,b}$$

where

$$\mathbf{Q}_i(\omega)=\mathbf{A}_i^*(\omega)\mathbf{G}_{\tilde{\mathbf{X}}_\mathbf{F}\tilde{\mathbf{X}}_\mathbf{F}}(\omega)\mathbf{H}_0^{\mathrm{T}}(\omega)+\mathbf{H}_0^*(\omega)\mathbf{G}_{\tilde{\mathbf{X}}_\mathbf{F}\tilde{\mathbf{X}}_\mathbf{F}}(\omega)\mathbf{A}_i^{\mathrm{T}}(\omega). \tag{32}$$

By examining the sign of the i-th sensitivity given by Eq. (22), the monotonic increasing or decreasing behavior of the *expected fatigue life*, $T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})$, with respect to the i-th uncertain parameter α_i can be predicted. Specifically, if $S_{T_{\mathrm{F},Y_h},i}>0$, the *expected fatigue life*, $T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})$, is a monotonic increasing function of α_i and its *LB* and *UB* are obtained setting $\alpha_i=\underline{\alpha}_i$ and $\alpha_i=\bar{\alpha}_i$, respectively; conversely, if $S_{T_{\mathrm{F},Y_h},i}<0$, $T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})$ is a monotonic decreasing function of α_i and its *LB* and *UB* are achieved assuming $\alpha_i=\bar{\alpha}_i$ and $\alpha_i=\underline{\alpha}_i$, respectively. Thus, the combinations of the endpoints of the r uncertain parameters which yield the *LB* and *UB* of the *expected fatigue life*, $T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})$, for the stress random process $Y_h(\boldsymbol{\alpha},t)$, denoted by $\alpha^{(\mathrm{LB})}_{Y_h,i}$ and $\alpha^{(\mathrm{UB})}_{Y_h,i}$, $(i=1,2,\dots,r)$, can be evaluated as follows [25]

$$\text{if}\quad S_{T_{\mathrm{F},Y_h},i}>0,\quad\text{then}\quad \alpha^{(\mathrm{UB})}_{Y_h,i}=\bar{\alpha}_i,\quad \alpha^{(\mathrm{LB})}_{Y_h,i}=\underline{\alpha}_i;$$

$$\text{if}\quad S_{T_{\mathrm{F},Y_h},i}<0,\quad\text{then}\quad \alpha^{(\mathrm{UB})}_{Y_h,i}=\underline{\alpha}_i,\quad \alpha^{(\mathrm{LB})}_{Y_h,i}=\bar{\alpha}_i,\qquad (i=1,2,\dots,r). \tag{33a,b}$$

Such combinations can be collected into the following two vectors:

$$\boldsymbol{\alpha}^{(\mathrm{LB})}_{Y_h}=\begin{bmatrix}\alpha^{(\mathrm{LB})}_{Y_h,1} & \alpha^{(\mathrm{LB})}_{Y_h,2} & \dots & \alpha^{(\mathrm{LB})}_{Y_h,r}\end{bmatrix}^{\mathrm{T}};$$

$$\boldsymbol{\alpha}^{(\mathrm{UB})}_{Y_h}=\begin{bmatrix}\alpha^{(\mathrm{UB})}_{Y_h,1} & \alpha^{(\mathrm{UB})}_{Y_h,2} & \dots & \alpha^{(\mathrm{UB})}_{Y_h,r}\end{bmatrix}^{\mathrm{T}}. \tag{34a,b}$$

Finally, the *LB* and *UB* of the interval *expected fatigue life* for the stress random process $Y_h(\boldsymbol{\alpha}^I,t)$ can be obtained by evaluating Eq. (21) for the structure with assigned values of the uncertain parameters given by those collected into the vectors $\boldsymbol{\alpha}^{(\mathrm{LB})}_{Y_h}$ and $\boldsymbol{\alpha}^{(\mathrm{UB})}_{Y_h}$, respectively:

$$\underline{T}_{\mathrm{F},Y_h}=\frac{\pi C}{\sqrt{2^{(k-2)}}\;\Gamma\left(1+\dfrac{k}{2}\right)}\frac{\lambda_{Y_h,1.5}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{LB})}\right)}{\lambda_{Y_h,0.75}^{2}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{LB})}\right)}\sqrt{\frac{\lambda_{Y_h,0}^{3-k}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{LB})}\right)}{\lambda_{Y_h,2}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{LB})}\right)}};$$

$$\overline{T}_{\mathrm{F},Y_h}=\frac{\pi C}{\sqrt{2^{(k-2)}}\;\Gamma\left(1+\dfrac{k}{2}\right)}\frac{\lambda_{Y_h,1.5}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{UB})}\right)}{\lambda_{Y_h,0.75}^{2}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{UB})}\right)}\sqrt{\frac{\lambda_{Y_h,0}^{3-k}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{UB})}\right)}{\lambda_{Y_h,2}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{UB})}\right)}}. \qquad (35a,b)$$

In the previous equations, $\lambda_{Y_h,\ell}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{LB})}\right)$ and $\lambda_{Y_h,\ell}\left(\boldsymbol{\alpha}_{Y_h}^{(\mathrm{UB})}\right)$, ($\ell=0,0.75,1.5,2$), are the spectral moments of the stress random process $Y_h(\boldsymbol{\alpha}^I,t)$ obtained performing a stochastic analysis of the structure with uncertain parameters $\boldsymbol{\alpha}=\boldsymbol{\alpha}_{Y_h}^{(\mathrm{LB})}$ and $\boldsymbol{\alpha}=\boldsymbol{\alpha}_{Y_h}^{(\mathrm{UB})}$. Notice that the proposed procedure is much more efficient than the *vertex method* since it requires only two stochastic structural analyses. Furthermore, it has to be remarked that the quantity of interest for design purpose is the *LB* of the *interval expected fatigue life*, given by Eq.(35a).

4 NUMERICAL APPLICATION

In order to demonstrate the effectiveness of the proposed method, the truss structure illustrated in Figure 1 subjected to turbulent wind loads has been selected as case-study. The truss structure is composed of 24 pin truss members characterized by nominal Young's modulus $E_0=E_{0,i}=2.1\times10^8$ kN/m^2 , nominal cross-sectional areas $A_0=A_{0,i}=5.76\times10^{-4}$ m^2 and lengths $L_{0,i}=L_i$ ($i=1,2,\ldots,24$) as indicated in Figure 1, where $L=3$.

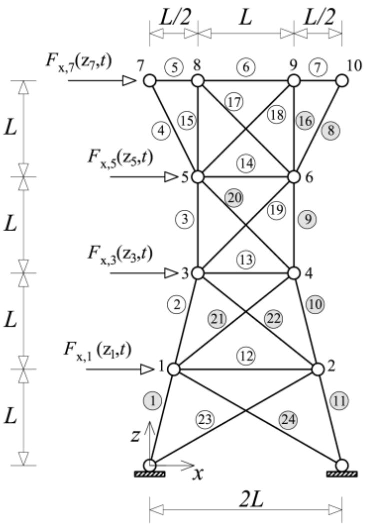


Figure 1: Truss structure under turbulent wind excitation.

The parameters of the *Y-N* curve [5] of the critical points of the pin truss members are set equal to $k=4$ and $C=1934\times10^{12}$ MPa^k [26]. The nominal lumped mass pertaining to nodes 1-6 is $m_{0,j}=600$ kg ($j=1,2,\ldots,6$), while the one pertaining to nodes 7-10 is $m_{0,j}=1200$ kg ($j=7,8,...,10$). The structural damping is modelled by using the Rayleigh damping model with constants $c_0 = 5.96807\ \mathrm{s}^{-1}$ and $c_1 = 0.0004317\ \mathrm{s}$, so that the modal damping ratio for the first and third mode of the nominal structure is $\xi_0 = 0.02$.

As shown in Figure 1, nodes 1, 3, 5 and 7 at levels z_i $(i = 1,3,5,7)$, are excited in the along-wind direction by the nodal forces $F_{x,i}$ $(i = 1,3,5,7)$ [27]:

$$F_{x,i} = \frac{1}{2}\rho C_D A_i W^2(z_i,t) = F_{x,i}^{(s)} + \tilde{F}_{x,i}(z_i,t) \approx \frac{1}{2}\rho C_D A_i w_s^2 + \rho C_D A_i \tilde{W}(z_i,t)\, w_s,\ (i = 1,3,5,7) \quad (36)$$

where $F_{x,i}^{(s)}$ and $\tilde{F}_{x,i}(z_i,t)$ denote the mean and random fluctuating component of wind loads, respectively. The following values of the parameters appearing in Eq. (36) are assumed: air density $\rho = 1.25\ \mathrm{kg/m^3}$, drag coefficient $C_D = 1.2$, and tributary area A_i $(i = 1,3,5,7)$ of nodes 1,3,5,7, given by $A_1 = 9\,\mathrm{m}^2$, $A_3 = 9\,\mathrm{m}^2$, $A_5 = 9\,\mathrm{m}^2$ and $A_7 = 4.5\,\mathrm{m}^2$. Moreover, in Eq. (36) $w_s(z) = w_{s,10}(z/10)^{\beta}$, is the mean wind speed, where $w_{s,10}$ is the mean wind speed measured at height $z = 10\,\mathrm{m}$, and $\tilde{W}(z,t)$ is the fluctuating component of the speed which is modelled as a zero-mean stationary Gaussian multi-correlated random process, fully described from a probabilistic point of view by the one-sided *PSD* function proposed by Davenport [28]:

$$G_{\tilde{W}\tilde{W}}(\omega) = 4K_0 w_{s,10}^2 \frac{\chi^2}{\omega\left(1+\chi^2\right)^{4/3}} \quad (37)$$

where $\chi = b_1\omega/(\pi w_{s,10})$. In this numerical application, when not otherwise specified, it is assumed $w_{s,10} = 25\mathrm{m/s}$, $K_0 = 0.03$, $b_1 = 600\ \mathrm{m}$ and $\beta = 0.3$. The 4-variate zero-mean stationary Gaussian random process $\tilde{\mathbf{X}}(t)$ collecting wind velocity fluctuations at the wind-exposed nodes (1,3,5,7) of the truss structure (see Figure 1) is fully characterized, from a probabilistic point of view by the *PSD* function matrix $\mathbf{G}_{\tilde{\mathbf{X}}\tilde{\mathbf{X}}}(\omega)$, as reported in Ref.[3].

Young's moduli of the material of the $r = 10$ bars highlighted in Figure 1 are modelled as interval variables, i.e. $E_i^I = E_0(1+\Delta\alpha\hat{e}_i^I)$, ($i = 1,8,9,10,11,16,20,21,22,24$). The aim of the analysis is the evaluation of the range of the interval *expected fatigue life* T_{F,σ_1}^I for the normal stress interval random process of bar 1, i.e. $Y_h^I(,t) \equiv \sigma_1^I(t)$.

For validation purpose, in Figure 2 the bounds of the interval *expected fatigue life* T_{F,σ_1}^I obtained by applying the proposed method and the *vertex method* versus the deviation amplitude of the uncertain parameters $\Delta\alpha$ are plotted. An excellent agreement is observed even when the degree of uncertainty increases. Since the interval *expected fatigue life* is a monotonic function of the generic uncertain parameter, the bounds provided by the *vertex method* can be assumed as the "exact" ones. It worth noting, however, that the *vertex method* involves 2^r ($r = 10$ being the number of uncertain parameters) stochastic analyses of the structure, while the proposed procedure requires only 2 stochastic analyses, regardless of the number of uncertain parameters. By inspection of Figure 2, it is observed that the *LB* of the interval *expected fatigue life* is always smaller than the nominal one. This implies that neglecting uncertainties may lead to dangerous overestimation of time to failure.

Figure 3 displays the proposed and "exact" bounds of T_{F,σ_1}^I versus the number *r* of uncertain parameters. Again an excellent agreement is observed. Moreover, it can be noticed that, as the number of uncertainties increases, the truss structure is more exposed to fatigue failure. Indeed, the region of the interval *expected fatigue life*, T^I_{F,σ_1} , enclosed by the *LB* and *UB*, becomes wider.

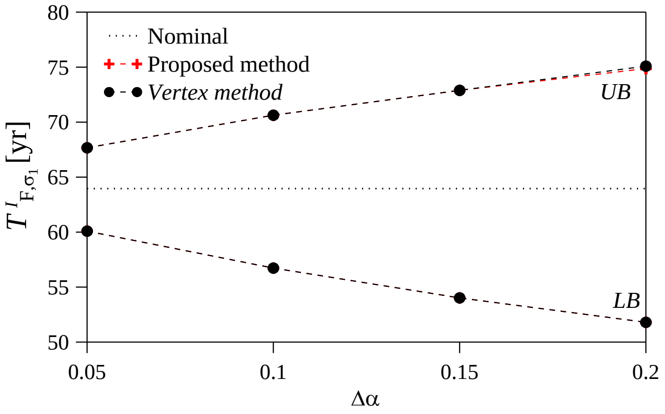


Figure 2: *LB* and *UB* of the interval *expected fatigue life* T^I_{F,σ_1} provided by the *vertex method* and the proposed method versus the deviation amplitude of the uncertain parameters $\Delta\alpha$.

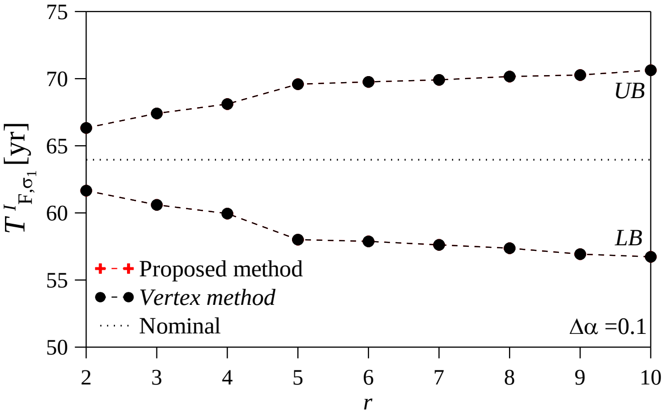


Figure 3: *LB* and *UB* of the interval *expected fatigue life* T^I_{F,σ_1} provided by the *vertex method* and the proposed method versus the number of uncertain parameters r .

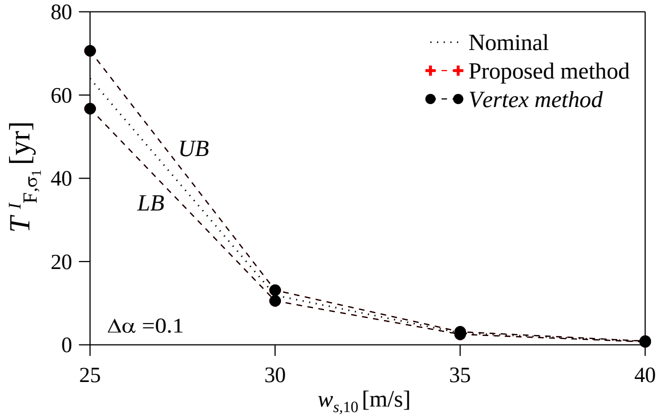


Figure 4: *LB* and *UB* of the interval *expected fatigue life* T^I_{F,σ_1} provided by the *vertex method* and the proposed method versus the mean wind speed.

In Figure 4, the bounds of the interval *expected fatigue life*, T^{I}_{F,σ_1}, versus the mean wind speed $w_{s,10}$ measured at height $z = 10\,\mathrm{m}$ are plotted. Notice that, as the mean wind speed increases, the *LB* and *UB* of T^{I}_{F,σ_1} decrease and the structure becomes more prone to fatigue failure.

In order to identify the most influential uncertain parameters, sensitivities may be ranked by evaluating a percentage measure of the influence of the i-th uncertain parameter on the *expected fatigue life*. For this purpose, the *coefficient of sensitivity* [19] is introduced:

$$\beta_{i,T_{\mathrm{F},Y_h}}(\%) = \frac{1}{T^{(0)}_{\mathrm{F},Y_h}} \left(\left. \frac{\partial T_{\mathrm{F},Y_h}(\boldsymbol{\alpha})}{\partial \alpha_i} \right|_{\boldsymbol{\alpha}=\mathbf{0}} \Delta\alpha_i \right) \times 100 = \frac{1}{T^{(0)}_{\mathrm{F},Y_h}} \left(S_{T_{\mathrm{F},Y_h},i}\, \Delta\alpha_i \right) \times 100 \tag{38}$$

where $\Delta\alpha_i$ denotes the deviation amplitude of the dimensionless fluctuation $\alpha_i \in [-\Delta\alpha_i, \Delta\alpha_i]$, while $T^{(0)}_{\mathrm{F},Y_h}$ is the nominal *expected fatigue life* obtained from Eq. (21) setting $\boldsymbol{\alpha} = \mathbf{0}$. The *coefficient of sensitivity* in Eq. (38) measures the global variability of the *expected fatigue life* with respect to its nominal value. It follows that the most influential uncertain parameters are the ones characterized by higher values of the *coefficient of sensitivity*.

In Figure 5, the *coefficients of sensitivity* of the *expected fatigue life* for the normal stress random process of bar 1 with respect to the fluctuations of the axial stiffness of the twenty-four bars of the truss structure are depicted, for $\Delta\alpha = 0.1$. As expected, the various bars have a different influence on the *expected fatigue life* of bar 1. In particular, it can be observed that the most influential parameter is the axial stiffness of bar 11 followed by that of bar 1.

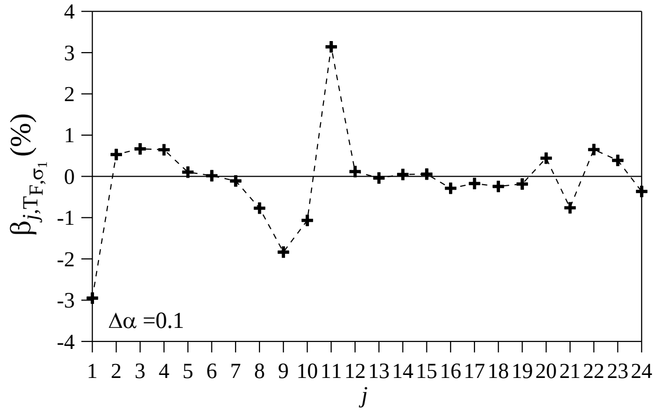


Figure 5: *Coefficients of sensitivity* of the *expected fatigue life* for the normal stress of bar 1 with respect to the fluctuations α_j, $(j = 1, 2, \ldots, 24)$, of the axial stiffness of the twenty-four bars of the truss structure around the nominal value.

The fluctuations α_i ($i = 11, 1, \ldots 6$) of the axial stiffness of the twenty-four bars of the truss, ranked from the most (α_{11}) to the least influential (α_6) one, based on the values of the coefficients $\beta_{i,T_{\mathrm{F},\sigma_1}}(\%)$ reported in Figure 5, are collected into the following vector:

$$\begin{aligned}\mathbf{b} = [&\alpha_{11}, \alpha_1, \alpha_9, \alpha_{10}, \alpha_8, \alpha_{21}, \alpha_3, \alpha_{22}, \alpha_4, \alpha_2, \alpha_{20}, \alpha_{23}, \alpha_{24}, \alpha_{16}, \alpha_{18}, \alpha_{19}, \alpha_{17}, \alpha_{12}, \alpha_7, \\ &\alpha_5, \alpha_{15}, \alpha_{14}, \alpha_{13}, \alpha_6]^{\mathrm{T}}.\end{aligned} \tag{39}$$

The least influential uncertain parameters can be reasonably set equal to their nominal values without significantly affecting the accuracy of the analysis outcomes. In Figure 6a, the bounds of the interval *expected fatigue life*, T^{I}_{F,σ_1}, for the normal stress of bar 1 versus the number $r = 24, 23, \ldots, 2$ of influential parameters retained in the analysis are displayed. Specifically, $r = 24, 23, \ldots, 2$ denotes the number of uncertainties collected into vector $\mathbf{b}$, which progressively decreases from 24 to 2 as an uncertain parameter at a time is neglected, starting from the least influential one (α_6). The smallest number of uncertainties retained in the analysis is $r = 2$ and pertains to the truss with only the axial stiffness of bars 1 and 11 assumed uncertain (see Eq. (39)).

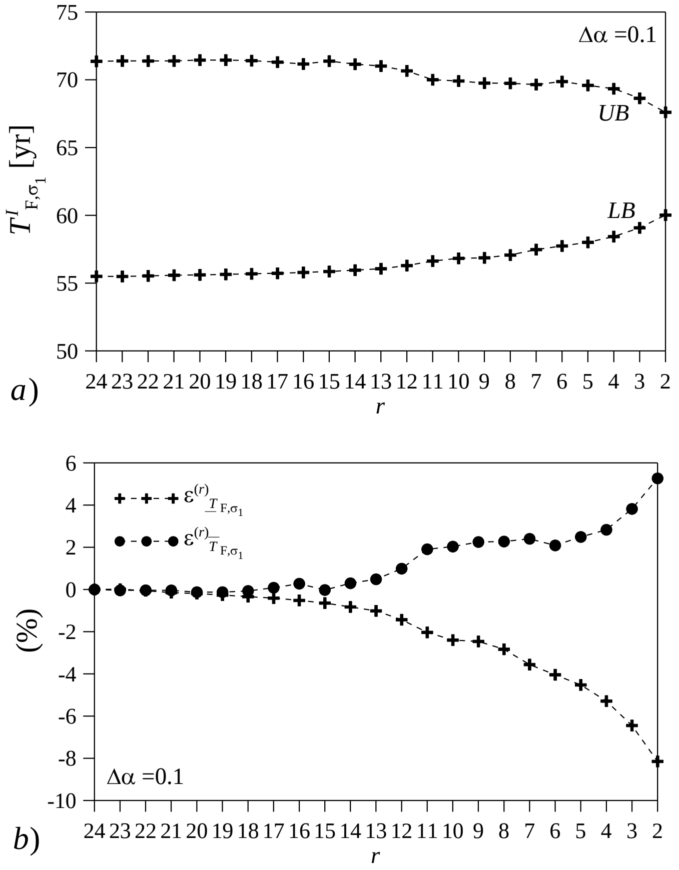


Figure 6: *a*) Proposed bounds of the *expected fatigue life* for the normal stress of bar 1 versus the number of uncertain parameters r sorted as in Eq. (39); *b*) percentage difference with respect to the bounds pertaining to the truss with uncertain axial stiffness of all the bars.

Figure 6b shows the percentage differences between the bounds of T^{I}_{F,σ_1} reported in Figure 6a obtained assuming a decreasing number $r = 24, 23\ldots, 2$ of uncertain parameters with respect to the bounds obtained considering all the twenty-four bars with uncertain axial stiffness, defined as:

$$\varepsilon^{(r)}_{R_{F,\sigma_1}}(\%) = \frac{R^{(24)}_{F,\sigma_1} - R^{(r)}_{F,\sigma_1}}{R^{(24)}_{F,\sigma_1}} \times 100, \quad R_{F,\sigma_1} = \overline{T}_{F,\sigma_1}, \underline{T}_{F,\sigma_1}, \ (r = 24, 23\ldots, 2). \tag{40}$$

Considering only the first $r = 13$ uncertain parameters in Eq. (39), the *LB* of the interval *expected fatigue life*, T^{I}_{F,σ_1}, is overestimated of 1.014%. This allows us to set the remaining parameters equal to their nominal values and thus reduce the computational burden. On the other hand, the *LB* may be significantly overestimated when a larger number of uncertain parameters is set equal to the nominal value.

5 CONCLUSIONS

A novel method for estimating the region of the interval *expected fatigue life* of linear structures with uncertain-but-bounded parameters subjected to stationary multi-correlated stochastic excitations has been proposed. Without loss of generality, attention has been focused on structures with uncertain axial stiffness. The interval *expected fatigue life* is defined by applying the empirical spectral approach proposed by Benasciutti and Tovo [1], called $\alpha_{0.75}$-method. The key idea behind the presented procedure is to derive the interval *Power Spectral Density* function of a selected stress random process in approximate explicit form by applying the so-called *Interval Rational Series Expansion* (*IRSE*) [3], which may be viewed as an effective surrogate model of the *Frequency Response Function* matrix (*FRF*). This allows a straightforward evaluation of the sensitivities of the interval *expected fatigue life* by direct differentiation. Based on the results provided by sensitivity analysis, the combinations of the endpoints of the uncertain parameters which yield the bounds of the uncertain parameters is obtained. In order to limit the overestimation, uncertainties are handled by applying the *Improved Interval Analysis* via *Extra Unitary Interval* [2].

The main features of the proposed method may be summarized as follows: *i*) the *IRSE* allows us to derive the sensitivities of the *expected fatigue life* to the uncertain parameters in approximate explicit form; *ii*) the computational effort is drastically reduced compared to the *vertex method* since only two stochastic analyses of the structure need to be performed whatever the number of uncertain parameters is; *iii*) sensitivity analysis may also be exploited to detect the most influential uncertain parameters.

Numerical results have highlighted the significant influence of uncertainties on the *expected fatigue life*, which can be seriously overestimated when the nominal values of the input parameters are assumed.

REFERENCES

[1] D. Benasciutti, R. Tovo, Comparison of spectral methods for fatigue analysis of broad-band Gaussian random processes. *Probabilistic Engineering Mechanics*, **21**, 287-299, 2006.

[2] G. Muscolino, A. Sofi, Stochastic analysis of structures with uncertain-but-bounded parameters via improved interval analysis. *Probabilistic Engineering Mechanics*, **28**, 152-163, 2012.

[3] G. Muscolino, A. Sofi, Bounds for the stationary stochastic response of truss structures with uncertain-but-bounded parameters. *Mechanical Systems and Signal Processing*, **37**, 163-181, 2013.

[4] M.P. Repetto, G. Solari, Wind-induced fatigue collapse of real slender structures. *Engineering Structures*, **32**, 3888-3898, 2010.

[5] S.H. Crandall, W.D. Mark, *Random Vibration in Mechanical Systems*. Academic Press, 1963.

[6] P.D. Spanos, A. Sofi, J. Wang, B. Peng, A Method for Fatigue Analysis of Piping Systems on Topsides of FPSO Structures. *Journal of Offshore Mechanics and Arctic Engineering*, **128**, 162-168, 2006.

[7] N.E. Dowling, Fatigue failure predictions for complicated stress-strain histories. *Journal of Materials, JMLSA*, **7**, 71-87, 1972.

[8] I. Rychlik, Note on cycle counts in irregular loads. *Fatigue & Fracture of Engineering Materials and Structures*, **16**, 377-390, 1993.

[9] L.D. Lutes, S. Sarkani, *Random Vibrations: analysis of structural and mechanical system*. Elsevier Butterworth-Heinemann, 2004.

[10] P.H. Wirsching, C.L. Light CL, Fatigue under wide band random stresses. *Journal of the Structural Division*, **106**, 1593-607, 1980.

[11] D. Benasciutti, R. Tovo, Cycle distribution and fatigue damage assessment in broad-band non-Gaussian random processes. *Probabilistic Engineering Mechanics*, **20**, 115-27, 2005.

[12] G. Petrucci G, B. Zuccarello, Fatigue life prediction under wide band random loading. *Fatigue & Fracture of Engineering Materials and Structures*, **27**, 1183-1195, 2014.

[13] C.E. Larsen, T. Irvine, A review of spectral methods for variable amplitude fatigue prediction and new results. *Procedia Engineering*, **101**, 243-250, 2015.

[14] Y. Zhu, Y. Tian, Fatigue evaluation of linear structures with uncertain-but-bounded parameters under stochastic excitations. *International Journal of Structural Stability and Dynamics*, **18**, (1850045-1)-(1850045-32), 2018.

[15] F. Giunta, G. Muscolino, A. Sofi, Fatigue analysis of wind excited structures with structural parameters affected by uncertainties described by interval variables. *7th International Conference on Uncertainty in Structural Dynamics (USD2018)*, Leuven, Belgium, September 17-19, 2018.

[16] R.E. Moore, R.B. Kearfott, M.J. Cloud, *Introduction to Interval Analysis*. SIAM, 2009.

[17] D. Moens, D. Vandepitte, A survey of non-probabilistic uncertainty treatment in finite element analysis. *Computer Methods in Applied Mechanics and Engineering,* **194**, 1527-1555, 2005.

[18] G. Muscolino, R. Santoro, A. Sofi, Explicit frequency response functions of discretized structures with uncertain parameters. *Computers & Structures*, **133**, 64-78, 2014.

[19] G. Muscolino, R. Santoro, A. Sofi, Explicit sensitivities of the response of discretized structures under stationary random processes. *Probabilistic Engineering Mechanics,* **35**, 82-95, 2014.

[20] G. Muscolino, R. Santoro, A. Sofi, Explicit reliability sensitivities of linear structures with interval uncertainties under stationary stochastic excitations. *Structural Safety*, **52**, 219-232, 2015.

[21] G. Muscolino, R. Santoro, A. Sofi, Reliability analysis of structures with interval uncertainties under stationary excitations. *Computer Methods in Applied Mechanics and Engineering,* **300**, 47-69, 2016.

[22] E.H. Vanmarcke, Properties of spectral moments with applications to random vibration. *Journal of the Engineering Mechanics Division*, **98**, 425-446, 1972.

[23] N. Impollonia, G. Muscolino, Interval analysis of structures with uncertain-but-bounded axial stiffness. *Computer Methods in Applied Mechanics and Engineering*, **200**, 1945–1962, 2011.

[24] J. Li, J.B. Chen, *Stochastic Dynamics of Structures*. John Wiley & Sons, 2009.

[25] A. Sofi, E. Romeo, A novel Interval Finite Element Method based on the improved interval analysis. *Computer Methods in Applied Mechanics and Engineering*, **311**, 671–697, 2016.

[26] M. Mršnik, J. Slavič, M. Boltežar, Frequency-domain methods for a vibration-fatigue-life estimation – Application to real data. *International Journal of Fatigue*, **47**, 8-17, 2013.

[27] E. Simiu, R. Scanlan, *Wind effects on structures*. John Wiley & Sons, 1996.

[28] A. G. Davenport, The spectrum of horizontal gustiness near the ground in high winds. *Quarterly Journal of the Royal Meteorological Society*, **87**, 194-211, 1961.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

ACTIVE SUBSPACES WITH B-SPLINE SURROGATES ON SPARSE GRIDS

M. F. Rehme[1], **and D. Plüger**[1]

[1]University of Stuttgart
Universitätsstr. 38, 70569 Stuttgart, Germany
e-mail: {michael.rehme, dirk.pflueger}@ipvs.uni-stuttgart.de

Keywords: Active Subspaces, B-Splines, Sparse Grids, Integration.

Abstract. *Many complex model functions allow the reduction of their effective dimension through active subspaces. These are computed by an eigenvalue decomposition of the average of the outer product of the function's gradient with itself. The size of the eigenvalues indicates how much the function changes on average along the direction given by the eigenvectors. This motivates to omit directions belonging to small eigenvalues and therefore to effectively reduce the model's dimension without losing much accuracy. The remaining directions form the active subspace, a linear combination of the input parameters. For real-world applications the required gradients are usually not explicitly known and they are thus commonly approximated with finite differences or ridge functions. The average of the outer product is then calculated using Monte Carlo quadrature. This converges slowly, resulting in long runtimes if the evaluation of the model function is expensive. The differentiation and integration of B-splines is numerically fast and analytically exact. Together with adaptive sparse grid discretization, they can be employed in higher-dimensional approximation. We use this to create a surrogate for the objective function, which provides us with better approximations for the gradient and thus better approximations for the active subspace. Furthermore we present a new integration technique for functions with a one-dimensional active subspace, that is based on a geometric interpretation of B-splines.*

1 INTRODUCTION

Many models in the field of uncertainty quantification rely on high-dimensional input parameters. This makes evaluations take long and calculations based on these models expensive. Active subspaces [1, 2] are an emerging method for the detection of dominant directions in the model's parameter space. If an active subspace is detected, it can be used for sensitivity analysis [3] and to reduce the model's effective dimension.

The detection of active subspaces requires evaluations of the model's derivative. If the derivative is unknown, it has to be approximated. This task has been done with finite differences [4], which require lots of model evaluations, and with linear and quadratic ridge functions [3, 5], which have only limited approximation quality. We were able to improve the detection of active subspaces for models with unknown derivatives, using a surrogate based on spatially adaptive sparse grids [6] and B-splines [7].

Sparse grids overcome the curse of dimensionality [8] to some extent and spatial adaptivity can reduce the necessary amount of grid points even further. B-splines, in contrast to global polynomials, are flexible in regard of their degree and do not suffer from Gibbs and Runge phenomena. Furthermore they directly provide access to gradients and can numerically be integrated exactly.

As a second contribution, we introduce a new approach for the integration of functions with a one-dimensional active subspace. We construct a one-dimensional surrogate for the objective function, again using spatially adaptive sparse grids and B-splines. From this we can calculate an approximation for the integral of the original objective function. The difficulty is to determine the volume of the orthogonal space. We do this based on a geometric interpretation of B-splines, which gives us the exact values for said volume.

2 METHODS

2.1 Active subspaces

Active subspaces are an emerging technique for the detection of important directions in the parameter space of high-dimensional functions. They are linear combinations of the input parameters along which the quantity of interest changes the most on average.

Without loss of generality we consider models given by an objective function defined on the unit cube and uniformly distributed parameters. Let $f : [0,1]^D \to \mathbb{R}$ be such an objective function Further define the symmetric positive semidefinite matrix $C \in \mathbb{R}^{D\times D}$ by

$$C := \int_{[0,1]^D} \nabla f(\mathbf{x}) \nabla f^T(\mathbf{x}) d\mathbf{x}. \tag{1}$$

This matrix admits a real eigenvalue decomposition

$$C = W \Lambda W^T, \tag{2}$$

where $W = [\mathbf{w}_1, \dots, \mathbf{w}_D]$ is the matrix of eigenvectors, and $\Lambda = diag(\lambda_1, \dots, \lambda_D)$ is the matrix of eigenvalues sorted in decreasing order.

The j-th eigenvalue equals the average squared directional derivative of f along $\mathbf{w}_j$ [9],

$$\lambda_j = \mathbf{w}_j^T C \mathbf{w}_j = \mathbb{E}[((\nabla f)^T \mathbf{w}_j)^2]. \tag{3}$$

Consequently, f is constant along $\mathbf{w}_j$, if $\lambda_j = 0$. Furthermore, if $\lambda_n > \lambda_{n+1}$, f changes less, on average, along $\mathbf{w}_{n+1}$ than along $\mathbf{w}_n$. If there is a significant gap between λ_n and λ_{n+1}, we split the eigenpairs accordingly,

$$W := [W_1, W_2] = [[\mathbf{w}_1, \ldots, \mathbf{w}_n], [\mathbf{w}_{n+1}, \ldots, \mathbf{w}_D]], \quad \Lambda = \begin{pmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{pmatrix}, \tag{4}$$

where $\Lambda_1 = diag(\lambda_1, \ldots, \lambda_n)$ and $\Lambda_2 = diag(\lambda_{n+1}, \ldots, \lambda_D)$. The span of the columns of W_1 is called the active subspace. If Λ_2 is relatively small, f varies little along directions in W_2 and f can be approximated by a lower-dimensional function $g : \mathbb{R}^n \to \mathbb{R}$,

$$f(\mathbf{x}) \approx g(W_1^T \mathbf{x}). \tag{5}$$

The detection of active subspaces, that is to say the calculation of the entries of the matrix C, is usually done using Monte Carlo quadrature and evaluations of the objective function's gradient [4]. However if the gradient of the objective function is not known, or only datasets from measurements are available, the gradient must be approximated. This is often done with linear or quadratic ridge functions [3, 5]. But if f is not of linear or quadratic shape these methods naturally are only capable of approximating f and ∇f up to a certain degree. We propose to use B-spline basis functions to approximate f.

2.2 B-splines

B-splines are the canonical basis of the spline space, i.e. the space of piecewise polynomials. They can be quickly evaluated, provide gradients and are numerically exactly integrable. We therefore introduce them as a suitable choice for the calculation of active subspaces.

Let $m, p \in \mathbb{N}_0$ and $\xi := (\xi_0, \ldots, \xi_m + p)$ be a knot sequence, that is a non-decreasing sequence of real numbers. Using the Cox-de-Boor recursion [10, 11], we define $b^p_{i,\xi}$, the B-spline of index i and degree p, as

$$b^p_{i,\xi}(x) = \begin{cases} \dfrac{x - \xi_i}{\xi_{i+p} - \xi_i} b^{p-1}_{i,\xi}(x) + \dfrac{\xi_{i+p+1} - x}{\xi_{i+p+1} - \xi_{i+1}} b^{p-1}_{i+1,\xi}(x) & p \geq 1, \\ \chi_{[\xi_i, \xi_{i+1}]}(x) & p = 0, \end{cases} \tag{6}$$

where $\chi_{[\xi_i, \xi_{i+1}]}(x)$ evaluates to one in the interval $[\xi_i, \xi_{i+1}]$ and zero elsewhere.

Schoenberg introduced B-splines using infinite and uniform knot sequences [12]. The derived B-spline basis spans the corresponding spline space. Restricting the knot sequence to the unit interval invalidates this important property, because the Schoenberg-Whitney conditions are not fulfilled at the intervals boundaries [7]. To revalidate the conditions and obtain full approximation quality we use not-a-knot B-splines [7]. As is common, we only define and use not-a-knot B-splines of odd degree p.

First we define uniform B-splines of level l through the uniform knot sequence $\xi^u_l := (\xi^u_{l,-p}, \ldots, \xi^u_{l,2^l+p})$, where $\xi^u_{l,i} := ih_l$ and $h_l := 2^{-l}$. Then we derive not-a-knot B-splines from uniform B-splines, by requiring continuity of the $p-$th derivatives at the $\frac{p-1}{2}$ left-most and right-most knots inside $D^p_{\xi^u_l} := (\xi^u_{l,0}, \ldots, \xi^u_{l,2^l})$. This requirement is equivalent to removing these knots from the knot sequence ξ^u_l but keeping them in the set of interpolation nodes. Consequently we obtain the uniform not-a-knot sequence of level l and degree p [13], $\xi^{p,nak}_l := (\xi^{p,nak}_{l,0}, \ldots, \xi^{p,nak}_{l,2^l+p+1})$,

$$\xi^{p,\text{nak}}_{l,i} := \begin{cases} \xi^{\text{u}}_{l,i-p}, & i = 0, \ldots, p, \\ \xi^{\text{u}}_{l,i-(p+1)/2}, & i = p+1, \ldots, 2^l, \\ \xi^{\text{u}}_{l,i-1}, & i = 2^l + 1, \ldots, 2^l + p + 1. \end{cases} \tag{7}$$

This definition is only valid if $l \geq \lceil \log_2(p+1) \rceil$. Otherwise we cannot remove $p-1$ knots from the sequence. Therefore, if $l < \lceil \log_2(p+1) \rceil$, we use $\xi_{l,i}^{p,nak} = \xi_{l,i}^{u}$ and Lagrange polynomials

$$L_{l,i}(x) := \prod_{\substack{0 \leq m \leq 2^l, \\ m \neq i}} \frac{x - \xi_{l,m}^{\mathrm{u}}}{\xi_{l,i}^{\mathrm{u}} - \xi_{l,m}^{\mathrm{u}}}, \quad i = 0, \dots, 2^l. \tag{8}$$

This guarantees a basis of the polynomial space for the first levels [13]. Finally, the not-a-knot B-spline basis $b_{l,i}^{p,nak}$ of degree p, level l and index i is given by

$$b_{l,i}^{p,\mathrm{nak}}(x) := \begin{cases} b_{i,\xi_l^{p,\mathrm{nak}}}^{p}(x) & l \geq \lceil \log_2(p+1) \rceil, \\ L_{l,i}(x) & l < \lceil \log_2(p+1) \rceil. \end{cases} \tag{9}$$

In the context of sparse grids, which we will use later on, the boundary points are usually omitted to reduce the overall effort. In order to keep reasonable approximations, the left-most and right-most B-splines are modified, so that they extrapolate towards the boundary [13]. This results in $b_{l,i}^{p,mod}$, the modified not-a-knot B-splines of degree p, level l and index i,

$$b_{l,i}^{p,\mathrm{mod}}(x) := \begin{cases} 1 & l = 1,\ i = 1, \\ b_{l,1}^{p,\mathrm{nak}}(x) - \dfrac{\frac{d^2}{dx^2} b_{l,1}^{p,\mathrm{nak}}(0)}{\frac{d^2}{dx^2} b_{l,0}^{p,\mathrm{nak}}(0)} b_{l,0}^{p,\mathrm{nak}}(x) & l \geq 2,\ i = 1, \\ b_{l,1}^{p,\mathrm{mod}}(1-x) & l \geq 2,\ i = 2^l - 1, \\ b_{l,i}^{p,\mathrm{nak}}(x) & \text{otherwise.} \end{cases} \tag{10}$$

2.3 Sparse grids

The amount of grid points of a full uniform isotropic grid in D dimensions grows like $\mathcal{O}(h_l^{-D})$. This exponential increase makes calculations in higher dimensions impossible. Sparse grids were designed to mitigate this curse of dimensionality. The total amount of grid points of regular sparse grids only grows like $\mathcal{O}(h_l^{-1}(\log_2 h_l^{-1})^{D-1})$. Still, under certain smoothness assumptions, it was shown that for hat functions, i.e. B-splines of degree 1, the L^2-interpolation error decays like $\mathcal{O}(h_l^2(log_2\ h_l^{-1})^{D-1})$ for sparse grids. This is only slightly worse than the full grid error convergence rate of $\mathcal{O}(h_l^2)$ [14, 15].

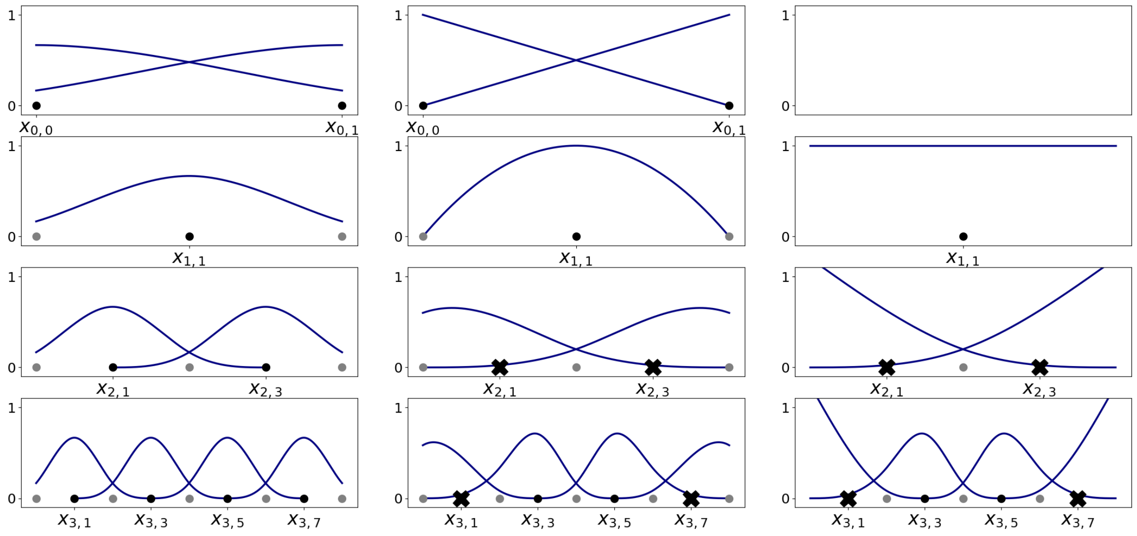
For the definition of sparse grids we need hierarchical basis functions. We define I_l, the hierarchical index set of level l as

$$I_l := \{0 < i < 2^l \mid i \text{ odd}\}. \tag{11}$$

Given univariate basis functions $\varphi_{l,i}$ that are determined by their level l and index i, for example the B-spline basis, we define the corresponding multivariate hierarchical basis functions via tensor products

$$\varphi_{\mathbf{l},\mathbf{i}}(\mathbf{x}) := \prod_{d=1}^{D} \varphi_{l_d,i_d}(x_d),\ \mathbf{l} \in \mathbb{N}_0^D,\ \mathbf{i} \in I_{\mathbf{l}} := I_{l_1} \times \cdots \times I_{l_D}. \tag{12}$$

Without loss of generality we define sparse grids on the unit hypercube $[0,1]^D$. Let $\mathbf{l} := (l_1, \dots, l_D) \in \mathbb{N}_0^D$ be a multi index and $\mathcal{H}_{\mathbf{l}} := \{\mathbf{x}_{\mathbf{l},\mathbf{i}} = (x_{l_1,i_1}, \dots, x_{l_D,i_D})\}$ for $x_{l_d,i_d} = i_d h_{l_d}$



(a) (b) (c)

Figure 1: (a) Hierarchical B-splines, (b) hierarchical not-a-knot B-splines, and (c) hierarchical modified not-a-knot B-splines of degree 3 and levels $0, 1, 2$ and 3 on the unit interval. The not-a-knot change in the knot sequence is illustrated with crosses at $x_{l,1}$ and $x_{l,2^l-1}$.

the anisotropic grid of level $\mathbf{l}$ on $[0,1]^D$. We define hierarchical subspaces of level $\mathbf{l}$ through the basis functions corresponding to $\mathcal{H}_{\mathbf{l}}$,

$$W_{\mathbf{l}} := \text{span}\{\varphi_{\mathbf{l},\mathbf{i}} \mid \mathbf{i} \in I_{\mathbf{l}}\}. \tag{13}$$

Regular nonboundary sparse grids V_l^s of level l are defined as the direct sum of hierarchical subspaces $W_{\mathbf{l}}$,

$$V_l^s := \bigoplus_{|\mathbf{l}'|_1 \leq l+D-1} W_{\mathbf{l}'}, \tag{14}$$

where $|\mathbf{l}'|_1 = \sum_{d=1}^D l'_d$ is the discrete ℓ_1 norm of $\mathbf{l}'$.

In previous work we introduced spatially adaptive sparse grids which can be used to customize the sparse grid to the given objective function [6]. Individual points are added to the sparse grid depending on an a priori guess for their benefit to the approximation. We define the hierarchical children $C(\mathbf{l}, \mathbf{i})$ of a grid point $\mathbf{x}_{\mathbf{l},\mathbf{i}}$ as all grid points $\mathbf{x}_{\mathbf{l}',\mathbf{i}'}$ for which there exists $r \in \{1, \dots D\}$ such that

$$\begin{aligned} &l_d = l'_d,\ i_d = i'_d \quad \forall d \in \{1, \dots D\} \backslash \{r\}, \\ &l'_r = l_r + 1, \\ &i'_r \in \{2i_r - 1, 2i_r + 1\}. \end{aligned} \tag{15}$$

Now let $\mathcal{G}$ be a spatially refined sparse grid, i.e.

$$\mathcal{G} = \{\mathbf{x}_{\mathbf{l},\mathbf{i}} \mid (\mathbf{l}, \mathbf{i}) \in \mathcal{I}\}, \tag{16}$$

for some finite level-index set $\mathcal{I} \subset \{(\mathbf{l}, \mathbf{i}) \mid \mathbf{l} \in \mathbb{N}_0^D, \mathbf{i} \in I_{\mathbf{l}}\}$ and let $\mathcal{I}^{\text{ref}} \subseteq \mathcal{I}$ be the set of refineable grid points, i.e. the set of level-index pairs of grid points for which not all hierarchical children are in $\mathcal{G}$,

$$\mathcal{I}^{\text{ref}} := \{(\mathbf{l}, \mathbf{i}) \in \mathcal{I} : C(\mathbf{l}, \mathbf{i}) \not\subset \mathcal{G}\}. \tag{17}$$

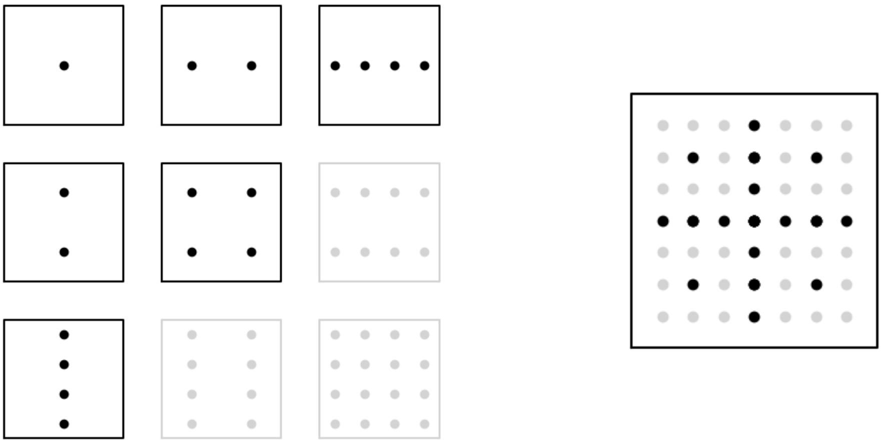


Figure 2: Hierarchical subspace scheme (left) and corresponding regular sparse grid (right).

The grid $\mathcal{G}$ can now be refined by identifying $\mathbf{x}_{\mathbf{l}^*,\mathbf{i}^*} \in \mathcal{G}$, the grid point with the largest influence on the quantity of interest, and by adding all its hierarchical children to the grid. For the identification of $\mathbf{x}_{\mathbf{l}^*,\mathbf{i}^*}$, we use the surplus-based standard criterion. It is based on the heuristic, that larger interpolation coefficients $|\alpha_{\mathbf{i},\mathbf{j}}|$ imply a worse local approximation. Consequently, $\mathbf{x}_{\mathbf{l}^*,\mathbf{i}^*}$ is given by

$$(\mathbf{l}^*, \mathbf{i}^*) = \text{argmax}_{(\mathbf{l},\mathbf{i}) \in \mathcal{I}^{\text{ref}}} |\alpha_{\mathbf{l},\mathbf{i}}|. \tag{18}$$

This refinement procedure is reiterated until the total number of grid points exceeds a given threshold.

2.4 Integration algorithm

If $\lambda_1 > 0$ and $\lambda_2 =, \ldots, = \lambda_D = 0$, then f has an exact one-dimensional subspace $W_1 = \mathbf{w}_1$ and can be represented as

$$f(\mathbf{x}) = g(W_1^T \mathbf{x}) =: g(y), \tag{19}$$

for some function $g : [l, r] \to \mathbb{R}$, with $l = \min_{\mathbf{x} \in [0,1]^D} W_1^T \mathbf{x}$ and $r = \max_{\mathbf{x} \in [0,1]^D} W_1^T \mathbf{x}$. This assumption allows us to introduce our new integration algorithm.

We approximate g using not-a-knot B-splines on a one-dimensional spatially adaptive sparse grid $\mathcal{G}$, including the boundary points l and r, resulting in the surrogate $\hat{g}$,

$$f(\mathbf{x}) = g(y) \approx \hat{g}(y) = \sum_{\mathbf{x}_{\mathbf{l},\mathbf{i}} \in \mathcal{G}} \alpha_{\mathbf{l},\mathbf{i}} \, b_{\mathbf{l},\mathbf{i}}^{p,nak}(y). \tag{20}$$

Because in general g is unknown, we need to approximate it from evaluations of f. For every grid point $y \in \mathcal{G}$ we solve $\mathbf{x} = \text{argmin}_{\mathbf{x} \in [0,1]^D} \|W_1^T \mathbf{x} - y\|_2$ and interpolate in the pairs $(y, f(\mathbf{x}))$, resulting in $\hat{g}$.

Now we want to integrate $\hat{g}$ subject to the volume of the inactive subspace. Calculating this volume numerically is an expensive task, because the shape of the inactive subspace is that of a zonotope [2]. However, combining Ramsay's definition of M-splines [16] and Schoenberg's theorem on simplex volumes [17], we obtain the following corollary

Corollary 1 *The linear density function obtained by projecting orthogonally onto the x-axis the volume of a D-dimensional simplex σ of volume V_σ, so located that its $D+1$ vertices $v_0, \dots, v_D$ project orthogonally into the knot sequence $\xi_\sigma := (\xi_{\sigma,0}, \dots, \xi_{\sigma,D})$, is given by*

$$V_\sigma \cdot M^D_{0,\xi_\sigma}, \tag{21}$$

where the M-spline M^D_{0,ξ_σ} can be represented as a scaled B-spline,

$$M^D_{0,\xi_\sigma} = \frac{D}{\xi_{\sigma,D} - \xi_{\sigma,0}} b^D_{0,\xi_\sigma}, \tag{22}$$

and $V_\sigma = \frac{|\text{det}A|}{D!}$ for $A = [v_1 - v_0, \dots, v_D - v_0] \in \mathbb{R}^{DxD}$.

Let S_D be the group of all permutations of $\{0, \dots, D\}$. Every permutation π in S_D defines a simplex $\sigma_\pi \subset [0,1]^D$ via

$$\sigma_\pi = \{\mathbf{x} = (x_0, \dots, x_D) \in [0,1]^D \mid 0 \leq x_{\pi(0)} \leq x_{\pi(1)} \leq \dots \leq x_{\pi(D)} \leq 1\}. \tag{23}$$

By construction theses simplices triangulate the hypercube $[0,1]^D$ and have equal volume V_1. Consequently, $V(y)$ the volume of the $(D-1)$-dimensional hyperplane $\{\mathbf{x} \in [0,1]^D \mid W_1^T \mathbf{x} = y\}$ is given by

$$V(y) = V_1 \sum_{\pi \in S_D} M^D_{0,\xi_{\sigma_\pi}}(y). \tag{24}$$

Finally we can calculate the desired integral via

$$\int_{[0,1]^D} f(\mathbf{x}) d\mathbf{x} = \int_l^r g(y) V(y) dy \tag{25}$$

$$\approx \int_l^r \sum_{\mathbf{x_{l,i}} \in \mathcal{G}} \alpha_{\mathbf{l,i}} \, b^{p,nak}_{\mathbf{l,i}}(y) V_1 \sum_{\pi \in S_D} M^D_{0,\xi_{\sigma_\pi}}(y) dy \tag{26}$$

$$= V_1 \sum_{\mathbf{x_{l,i}} \in \mathcal{G}} \alpha_{\mathbf{l,i}} \sum_{\pi \in S_D} \int_l^r b^{p,nak}_{\mathbf{l,i}}(y) M^D_{0,\xi_{\sigma_\pi}}(y) dy. \tag{27}$$

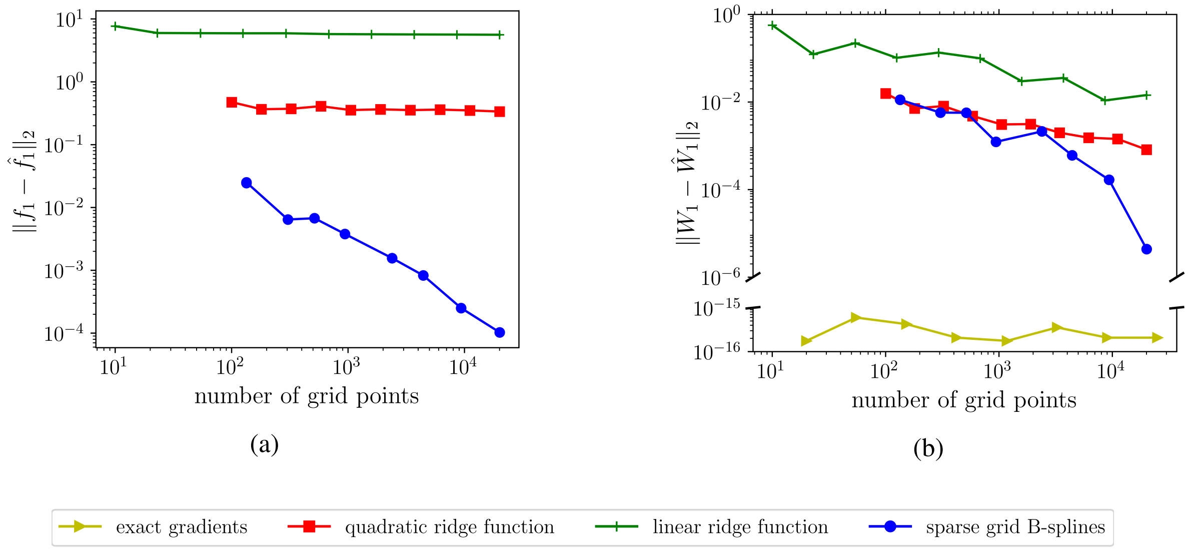
The integral in eq. (27) can be calculated fast and numerically exact using Gaussian quadrature, because both multiplicands are splines.

If the initial assumption $\lambda_2 =, \dots, = \lambda_D = 0$ does not hold, Equation (25) does not hold either. According to Equation (3), Equation (27) still remains a good approximation for the desired integral, if $\lambda_2, \dots, \lambda_d$ are relatively small in comparison to λ_1. However, there is no longer a function g as in Equation (19) and interpolating the pairs $(y = W_1^T \mathbf{x}, f(\mathbf{x}))$ gets unstable. One should be able to overcome this problem using regression with a suitable regularization term.

The restriction to one-dimensional active subspaces for this method seems quite harsh. However, one-dimensional active subspaces have been found in many models for real world applications, among others in models for airfoil shapes [5], solar cells [18] and lithium ion batteries [3].

3 RESULTS

We now show results for our two contributions, the detection of active subspaces and the integration of functions with a one-dimensional active subspace. Throughout this section, we use the common choice of B-spline degree $p = 3$. Adaptive sparse grids are initialized with a regular sparse grid of level 2.



(a)

(b)

Figure 3: Errors of the interpolants of f_1 created with a linear ridge function, quadratic ridge function and modified not-a-knot B-splines on an adaptive sparse grid, and error of the resulting approximations $\hat{W}_1$ for the active subspace W_1.

3.1 Active subspace detection

For the detection of active subspaces we compare our method to the state-of-the-art ”Python Active-Subspaces Utility Library” (PASUL) [19]. This library provides routines for the detection of active subspaces using Monte Carlo quadrature. If the objective function’s derivatives are not known, it provides approximations based on linear or quadratic ridge functions.

Our first demonstration function $f_1 : [0, 1]^8 \to \mathbb{R}$ is given by

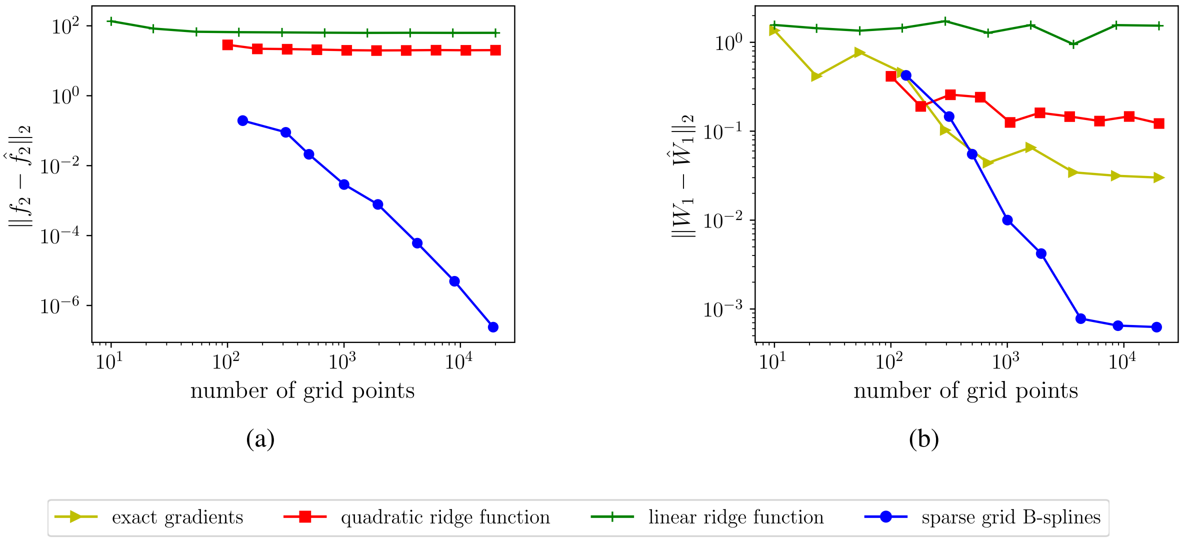
$$f_1(\mathbf{x}) = \frac{\sin(\gamma \sum_{i=1}^{8} x_i + 1)}{\gamma \sum_{i=1}^{8} x_i + 1}, \tag{28}$$

where we choose $\gamma = 0.75$. This function has an exact one-dimensional subspace given by $\mathbf{w}_1 = [1/\sqrt{8}, \dots, 1/\sqrt{8}]$.

In Figure 3b we see, that for any numbers of data points the active subspace is exactly calculated using the analytical derivative and Monte Carlo quadrature. That is explained by the active subspace being one-dimensional. Consider the matrix Λ from Equation (1). It has only one non-zero entry. The actual value of this entry is of no significance, because in practice the eigenvectors W are normalized to unit length. Consequently, one-dimensional subspaces are already detected using a single gradient evaluation, assuming it is not zero by chance.

Let us now assume that we have no access to the objective function’s gradient. Then the active subspace is best approximated by our method. This can be explained by Figure 3a. The approximation $\hat{f}_1$ for f_1 is best for modified not-a-knot B-splines on adaptive sparse grids. B-splines can be differentiated and integrated numerically exact. Therefore the error of $\hat{f}$ directly relates to the error of the entries of the matrix C from Equation (1) and thus to the error of the approximation of the active subspace.

Furthermore the error of our method converges towards zero, while the errors for linear and quadratic ridge functions do not. The number of degrees of freedom of a modified not-a-knot B-spline approximation increases with the number of grid points. This lets it converge towards



(a)

(b)

Figure 4: Errors of the interpolants of f_2 created with a linear ridge function, quadratic ridge function and modified not-a-knot B-splines on an adaptive sparse grid, and error of the resulting approximations $\hat{W}_1$ for the active subspace W_1.

the objective function. The ridge functions have a fixed number of degrees of freedom. Once it is reached, the approximation can no longer improve.

Our second demonstration function $f_2 : [0,1]^8 \to \mathbb{R}$ is given by

$$f_2(\mathbf{x}) = \sin(\beta_1 x_1 + \beta_2 x_2) + \cos(\beta_3 x_3 + \beta_4 x_4) - \sin(\beta_5 x_5 + \beta_6 x_6) - \cos(\beta_7 x_7 + \beta_8 x_8),$$

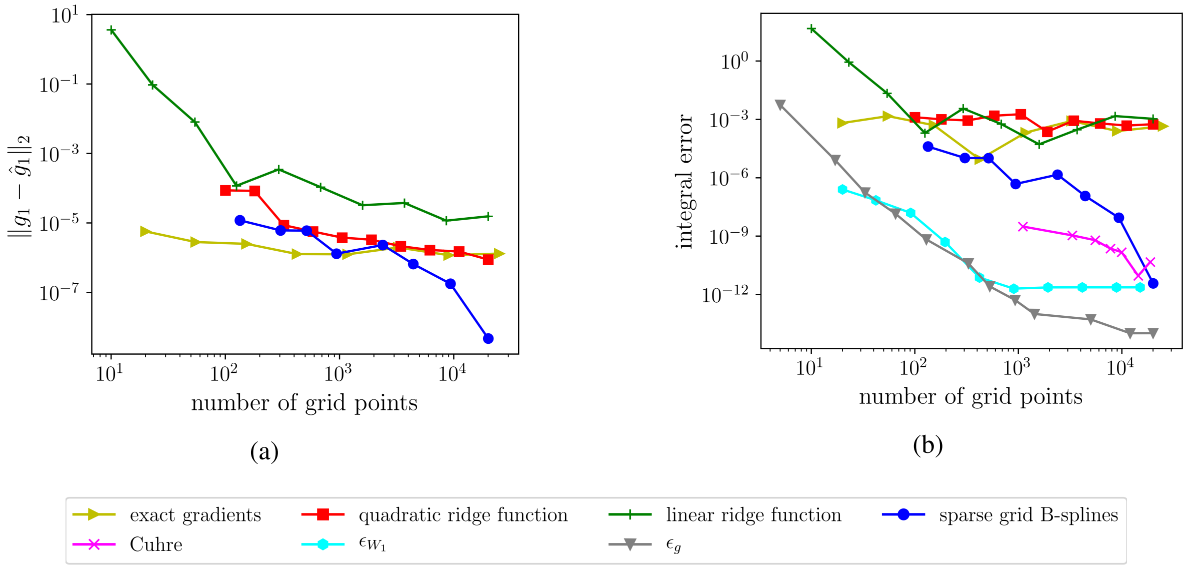
where $\beta = [\beta_1, \ldots, \beta_8] \in \mathbb{R}^8$ is chosen randomly on condition that $\|\beta\|_2 = 2\pi$.

This function has a 4-dimensional active subspace W_1, which we can not anymore state analytically. Therefore we calculate reference values using the analytical derivative of f_2 and 10^7 Monte Carlo points.

In Figure 4b we see, that the active subspace is best approximated with our method. It even outperforms the Monte Carlo scheme, which has access to analytical gradients. This is explained by the slow convergence of Monte Carlo quadrature. The active subspace is not one-dimensional, like in our first example. Therefore the interplay of the entries of the matrix C from Equation (1) is crucial to the calculation of $\hat{W}_1$. When calculating them, the Monte Carlo quadrature only converges with $\mathcal{O}(\sqrt{N})$, where N is the number of points. In contrast, our method uses only approximations for the gradients, but can calculate exact integrals of these. In Figure 4a we see, that the approximation for the gradients converges with $\mathcal{O}(N^{-3})$. For the linear and quadratic ridge function we see the same behavior as for f_1. Their approximation of f cannot converge to zero and thus the approximation quality of the active subspace is limited.

3.2 Integration

We now compute an integral with our new spline-based integration algorithm from Equation (27). We compare the results with the PASUL library, which provides routines for approximating the one-dimensional function $\hat{g}$ with global polynomials. For these we tried all degrees $p \in [1, 10]$ and added the best ones to our comparison. PASUL integrates $\hat{g}$ with Monte Carlo quadrature, where the volume $V(y)$ is approximated with a Monte Carlo Histogram.



(a)

(b)

Figure 5: Errors of the approximations $\hat{g}_1$ calculated from f_1 using an approximation for the active subspace (left). Errors for approximating the integral $I := \int_{[0,1]^8} f_2(\mathbf{x})d\mathbf{x}$ with a univariate integral based on the approximations for $\hat{g}$ via PASUL(gradient, linear and quadratic) and our spline based quadrature, error ϵ_{W_1} for approximating the integral I with our spline based quadrature supplied with the exact active subspace, error for approximating I with the Cuhre algorithm, and error ϵ_g for interpolating g with modified not a knot B-splines on a one-dimensional adaptive sparse grid (right).

Additionally, we calculate the full-dimensional integral using the Cuhre algorithm implemented in the Cuba library [20, 21]. Cuhre is one of the fastest known algorithms for quadrature in moderately high dimensions. It is based on a cubature rule for subregion estimation on a globally adaptive subdivision scheme.

The demonstration function f_1 from Equation (28) has a one-dimensional active subspace. The corresponding function $g_1 : [0, \sqrt{8}] \to \mathbb{R}$ can be stated analytically and is given by

$$g_1(y) = \frac{\sin(y\sqrt{8}\gamma + 1)}{y\sqrt{8}\gamma + 1}. \tag{29}$$

Using this and Equation (25) we calculate a numerically exact reference value for the integral $\int_{[0,1]^D} f_1(\mathbf{x})d\mathbf{x}$.

In Figure 5a we see the error of approximating g_1 from f and the approximated active subspace $\hat{W}_1$. This error is mainly influenced by the quality of the approximation of the active subspace. Modified not-a-knot B-splines on adaptive sparse grids need about 2000 grid points to enter the convergent phase in the active subspace approximation, compare Figure 3b. From then on they outperform the PASUL approximations. These use global polynomials of fixed degree, so again the approximation error does not converge towards zero. This behavior is propagated to the integral error in Figure 5b, where all PASUL methods stagnate around the same error.

The integral error of our spline-based integration converges much faster, and for increasing amount of grid points even gets competitive to the Cuhre algorithm. Also note, that the Cuhre algorithm needs several hundred function evaluations for its first result, while our spline-based integration needs only a small initial grid.

In Figure 5b, we also show the integration error ϵ_{W_1} for our spline-based integration, when provided with the exact active subspace W_1, and the error $\epsilon_g := |\int_l^r g_1(y)dy - \int_l^r \hat{g}_1(y)dy|$, describing the interpolation of the one-dimensional g with modified not-a-knot B-splines on a one-dimensional sparse grid. As expected the two errors ϵ_{W_1} and ϵ_g behave similarly. This confirms that our method reduces the complexity of the high-dimensional integral to that of the corresponding one-dimensional integral, where the error is primarily determined by the approximation of W_1. We conclude, that our method outperforms all other methods if W_1 is known, which holds in particular if the objective function's derivative is known.

4 CONCLUSIONS AND OUTLOOK

In this paper, we introduced a new way to approximate active subspaces using modified not-a-knot B-splines on spatially adaptive sparse grids. We showed how this method outperforms simple ridge functions which are widely used in the field of active subspaces if the objective function's gradient is unknown. We developed an astonishing new algorithm for the integration of functions with a one-dimensional subspace, based on a geometric interpretation of B-splines. If the active subspace is actually known, the algorithm converges quickly towards the correct integral. However, if the active subspace is only approximately known, the method's accuracy is limited by the quality of this approximation.

We plan to extend our sparse grid framework SG^{++} [6] by regression-based computation of $\hat{f}$ and $\hat{g}$ in the context of active subspaces. This will allow us to consider data-driven scenarios. Furthermore, the smoothing aspect of regression makes our new integration method applicable to functions without an exactly one-dimensional active subspace.

The major limitation of our new integration method is the triangulation of the unit cube. We used a simple approach to do so, which relies on $D!$ simplices. Better triangulations would improve the execution time of this algorithm. However, minimal triangulations of the unit cube are highly nontrivial. Optimal triangulations have so far only been found for up to $D = 7$ [22] and a lower bound for the number of simplices is given by $(D+1)^{\frac{D-1}{2}}$ [23].

5 ACKNOWLEDGMENTS

The authors would like to thank the German Research Foundation (DFG) for financial support of the project within the *Cluster of Excellence in Simulation Technology* (EXC310/2) at the University of Stuttgart.

REFERENCES

[1] T. M. Russi, *Uncertainty quantification with experimental data and complex system models*. PhD thesis, UC Berkeley, 2010.

[2] P. G. Constantine, *Active subspaces: Emerging ideas for dimension reduction in parameter studies*, vol. 2. SIAM, 2015.

[3] P. G. Constantine and A. Doostan, "Time-dependent global sensitivity analysis with active subspaces for a lithium ion battery model," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 10, no. 5, pp. 243–262, 2017.

[4] P. G. Constantine and D. Gleich, "Computing active subspaces with Monte Carlo," *arXiv preprint arXiv:1408.0545*, 2014.

[5] Z. J. Grey and P. G. Constantine, “Active subspaces of airfoil shape parameterizations,” *AIAA Journal*, vol. 56, no. 5, pp. 2003–2017, 2018.

[6] D. Pflüger, *Spatially Adaptive Sparse Grids for High-Dimensional Problems*. München: Verlag Dr. Hut, Aug. 2010.

[7] K. Höllig and J. Hörner, *Approximation and Modeling with B-Splines*. Philadelphia: SIAM, 2013.

[8] R. Bellman, *Adaptive Control Processes: A Guided Tour*. Rand Corporation. Research studies, Princeton University Press, 1961.

[9] P. G. Constantine, E. Dow, and Q. Wang, “Active subspace methods in theory and practice: applications to kriging surfaces,” *SIAM Journal on Scientific Computing*, vol. 36, no. 4, pp. A1500–A1524, 2014.

[10] M. G. Cox, “The numerical evaluation of B-splines,” *IMA Journal of Applied Mathematics*, vol. 10, no. 2, pp. 134–149, 1972.

[11] C. De Boor, “On calculating with B-splines,” *Journal of Approximation theory*, vol. 6, no. 1, pp. 50–62, 1972.

[12] I. J. Schoenberg, “Contributions to the problem of approximation of equidistant data by analytic functions,” *Quarterly of Applied Mathematics*, vol. 4, pp. 45–99 and 112–141, 1946.

[13] J. Valentin, *B-Splines on Sparse Grids. Algorithms and Application to Higher-Dimensional Optimization*. PhD thesis, University of Stuttgart, IPVS, 2019. to be published.

[14] H.-J. Bungartz and M. Griebel, “Sparse Grids,” *Acta Numerica*, vol. 13, pp. 147–269, 2004.

[15] J. Garcke, “Sparse grids in a nutshell,” in *Sparse grids and applications* (J. Garcke and M. Griebel, eds.), vol. 88 of *Lecture Notes in Computational Science and Engineering*, pp. 57–80, Springer, 2013.

[16] J. O. Ramsay *et al.*, “Monotone regression splines in action,” *Statistical science*, vol. 3, no. 4, pp. 425–441, 1988.

[17] H. B. Curry and I. J. Schoenberg, “On pólya frequency functions IV: the fundamental spline functions and their limits,” *Journal d’analyse mathématique*, vol. 17, no. 1, pp. 71–107, 1966.

[18] P. G. Constantine, B. Zaharatos, and M. Campanelli, “Discovering an active subspace in a single-diode solar cell model,” *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 8, no. 5-6, pp. 264–273, 2015.

[19] P. G. Constantine, R. Howard, A. Glaws, Z. Grey, P. Diaz, and L. Fletcher, “Python active-subspaces utility library,” *J. Open Source Software*, vol. 1, no. 5, p. 79, 2016.

[20] T. Hahn, “Cuba—a library for multidimensional numerical integration,” *Computer Physics Communications*, vol. 168, no. 2, pp. 78–95, 2005.

[21] J. Berntsen, T. O. Espelid, and A. Genz, “An adaptive algorithm for the approximate calculation of multiple integrals,” *ACM Trans. Math. Softw*, vol. 17, pp. 437–451, 1991.

[22] R. B. Hughes and M. R. Anderson, “Simplexity of the cube,” *Discrete Mathematics*, vol. 158, no. 1-3, pp. 99–150, 1996.

[23] A. Glazyrin, “Lower bounds for the simplexity of the n-cube,” *Discrete Mathematics*, vol. 312, no. 24, pp. 3656–3662, 2012.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24âĂŞ-26 June 2019

CALIBRATION OF A SURROGATE DISPERSION MODEL APPLIED TO THE FUKUSHIMA NUCLEAR DISASTER

N.B.T. Le[1,2], V. Mallet[2], I. Korsakissok[1], A. Mathieu[1] and R. Périllat[1,3]

[1]IRSN—French Institute of Radiation Protection and Nuclear Safety
31 Avenue de la Division Leclerc, 92260 Fontenay-aux-Roses, France
{ngoc-bao-tran.le, irene.korsakissok, anne.mathieu, raphael.perillat-phimecaengineering}@irsn.fr

[2] INRIA—French Institute for Research in Computer Science and Automation, Paris, France
2 Rue Simone Iff, 75012 Paris, France
{ngoc.le, vivien.mallet}@inria.fr

[3] Phimeca Engineering
18/20 boulevard de Reuilly, 75012 Paris, France
perillat@phimeca.com

Keywords: Meta-model, Fukushima, atmospheric dispersion, calibration, Kriging

Abstract. *Calibration methods require a large number of calls to the model, so that the computational time is far too high. To circumvent this issue, we built a meta-model based on statistical emulation, Kriging (Gaussian processes). One of the difficulties was to deal with the uncertain meteorological fields (e.g., time- and space-dependent winds), whose uncertainties are best described by an ensemble of meteorological forecasts. We use an ensemble of 50 alternative meteorological forecasts which are assumed to sample the uncertain meteorological variables. We parameterized the set of admissible meteorological fields as a convex linear combination of the ensemble of forecasts, which raised a number of difficulties in the design of experiment, in the meta-modeling and the calibration.*

Similarly, six alternative emission terms from the Fukushima literature are combined with random weights. The construction time of Gaussian process model is long but it provides in addition a standard deviation of the prediction error. This standard deviation was evaluated on an independent verification set of simulations. The meta-models are then used in an ensemble-based calibration procedure. A deterministic optimization was carried out. The aim was to find the best set of parameters that minimizes the model-to-data scores, like RMSE–Root Mean Square Error. The optimization was made possible thanks to the evaluation speed of the meta-model.

1 INTRODUCTION

Atmospheric dispersion modeling is the mathematical processing of simulating transport, dispersion and transformation of pollutants in a geographical area downwind of the source. Atmospheric dispersion models are used in various fields: forecasting air quality, roadway emissions or coastal fumigation modeling, etc. The French Institute of Radiation Protection and Nuclear Safety–IRSN–develops and uses atmospheric dispersion models in emergency cases that may imply an accidental release of radionuclides in the atmosphere. These models compute air concentration of radionuclides, deposition on the ground and gamma dose rate. These results are used in order to infer mitigation actions.

Using atmospheric dispersion models requires a meteorological forecast and a source term. A weather forecast represents a prediction of atmospheric state during an given time period in the future. It consists of data fields in 2D or 3D varying in time. A source term is a time series of quantities of radionuclides released in the environment. It contains also information about source location and release height. These two elements are the principal uncertainty sources of dispersion models, because the knowledge on these input data is limited and they are the most sensitive inputs, following [2, 3]. There are other uncertain parameters used within the models, e.g. deposition velocity, diffusion coefficient, scavenging coefficient, etc. Values of these parameters are often chosen in a deterministic way. So, simulations of atmospheric dispersion models are subject to considerable uncertainties.

Quantifying these uncertainties can be carried out by calibration methods. They are statistical algorithms which use output measurements to improve the knowledge on inputs. The aim of our work is to build a meta-model which will be used to assign PDFs to the unknown errors on the input variables and to calibrate those PDFs using field observations. A meta-model is necessary because calibration methods require a high number of simulations (or calls to the model) and using directly the full model is too costly in terms of computational time. In this study, Kriging was used to create meta-models.

This paper focuses on the Fukushima nuclear disaster, using an operational dispersion model and measurements collected during or after the disaster. In the evaluation and calibration processes, we made use of radiological observations of activity concentration, deposition and dose rate collected in Japan. Simulations were carried out by ℓdX, also called long-distance model. This study is organized into three principal parts. The first one, section 2, will present the different steps of the meta-model construction, including uncertainty modeling of inputs and derivation of the Kriging model. The second part, section 3, consists in meta-models' evaluation. The last section 4 introduces an application using meta-models for the optimization of deterministic scores.

2 Construction of meta-model

ℓdX is an operational version of the Eulerian transport model Polair3D from the air quality modeling system Polyphemus [7]. Since it is used in emergency cases, efficiency and speed are two important criteria. The computational time depends on many factors: number of grid cells, time step or number of radionuclides, etc. ℓdX simulations often use the same horizontal mesh as that of the weather forecasts. In this study, weather forecasts come from the European Centre for Medium-Range Weather Forecasts–ECMWF. We make use of an ensemble of 50 forecasts instead of just one forecast, in order to account for the uncertainty. Their spatial resolution is 0.25° (about 25 km) and the meteorological time step is 3 hours. This spatial resolution is coarse. A cell can cover many topographies that come with the sub-grid effect [6]. The simulations were carried out on an 40×40 horizontal field with 12 vertical levels, whose center altitudes are 20 m, 100 m, 220 m, 340 m, 500 m, 700 m, 1000 m, 1500 m, 2200 m, 3000 m, 3850 m and 4650 m. The outputs were interpolated to a one-hour time step. The computational cost for one simulation is about five minutes.

Simulation of Eulerian models is only reliable as from three or four cells from the source. Near to the release source, the plumes stay unscattered while the Eulerian models compute average quantities in the surrounding cells. In addition, with our spatial resolution, it is likely that several measurement stations lie in the same cell. In the cells surrounding the source, the plumes probably come across a certain number of stations but the models estimate the same result for all. That is why in the model-to-data comparison of ℓdX, we select the measurement stations whose distance from the source is higher than 100 km.

Calibration algorithms require a high number of simulations that may reach millions of calls to the model. The construction of a meta-model is then essential. The meta-model provides a computationally-efficient approximation of the original, physical model. The building of meta-model consists in learning over an optimized database. Basically, it needs several simulations of the physical model, whose entries constitute an ensemble of data set well distributed in the input space. The outputs of these simulations are used as the response for meta-model learning and this data ensemble, called design of experiment–DOE, must cover as much as possible cases. The meta-model construction can be divided into three parts:

1. Establishment of a design of experiment (DOE),

2. Computing the response of the physical model, i.e., the outputs for each DOE point,

3. Learning a statistical model from the DOE, using Kriging in our case.

Two following subsections will present how to create the DOE. The first one, section 2.1, introduces uncertainty modeling for physical parameters with their variation ranges. The second one 2.2 shows a perturbation method for weather forecast and source term. Finally, in part 2.3, we applied Kriging to construct the meta-model.

2.1 Uncertainty modeling of physical parameters

There are seven parameters in the input space of the meta-model, whose variation ranges were determined by experts. These parameters are considered independent, and each parameter follows its own probability distribution. The table 1 shows all independent parameters, as well as their probability densities, perturbation methods and variation ranges. These parameters are perturbed with different methods, some by adding a random value, some by multiplying by a random value, and others replaced by a random value [6]. These variation ranges were evaluated using experts judgment and bibliographical review, [3].

Variable	Distribution/Method	Variation range
Source elevation $[m]$	Discrete/replace	$[0-40, 40-160, 160-280, 280-400]$
Emission delay $[hours]$	Truncated normal/addition	$[-6, +6]$
Dry deposition velocity $[m/s]$	Uniform/replace	$[5.10^{-4}, 5.10^{-3}]$
Scavenging factor $[hs^{-1}mm^{-1}]$	Uniform/replace	$[10^{-7}, 10^{-4}]$
Scavenging exponent	Uniform/replace	$[0.6, 1]$
Horizontal diffusion $[m^2s^{-1}]$	Uniform/replace	$[0, 1.5] \times 10^4$
Vertical diffusion	Uniform/multiplication	$[1/3, 3]$

Table 1: Variation range of physical parameters.

In order to ensure that the meta-model is representative of the initial model, data sets in DOE must be distributed to cover the entire variation space of inputs. There are many algorithms to generate a DOE. In this study, we use the Latin Hyper-cube Sampling method–LHS. It is based on Latin Square design, which has only one sample in each row and column. A hypercube indicates a cube in dimension three or more. In other words, the aim of Latin Square is to sample from high dimension. Otherwise, some criteria must be respected to optimize the spatial distribution, cf figure A.1.

2.2 Perturbation of weather forecast and source term

The creation of the DOE for meteorological forecast and the source term is more difficult. Recall that we make use of an ensemble of weather forecasts. We also make use of several possible source terms collected in the literature. In [6], a couple of index numbers, corresponding to a weather forecast member and a source term, were randomly generated for each simulation of our DOE. The construction and use of meta-models interpolate the ℓdX response between samples of DOE. Interpolation between discrete indexes is unstable and does not make sense. Hence, the DOE regarding the meteorological ensemble and source terms was formulated using a convex linear combination, one for the meteorology fields, and another one for the source terms. To define one point in the DOE, the same factors k_i are used for all meteorological fields (wind, temperature, pressure, rain, ...) and in all grid cells (in 2D or 3D) and for all times. For any meteorological value p, the linear combination at one DOE reads

$$p_c = \sum_{i=1}^{n_p} k_i p_i, \quad \sum_{i=1}^{n_p} k_i = 1 \tag{1}$$

where n_p is the number of meteorological members, $(p_i)_i$ are the values of the different forecast members, and $(k_i)_i$ are coefficients from the n_p-dimension simplex. The same is applied for the source terms, but at each point of the DOE, there is one set of factors k_i for the meteorology and one independent set of factors for the source terms. Sampling from a unit simplex, e.g., making convex combination, allows to keep physical properties from the construction of weather forecast ensemble. So, the dimension of the DOE for only meteorological members is 50 coefficients. Optimization and calibration on high-dimension space are very difficult and can cause non-uniqueness. As a consequence, we reduced the size of weather ensemble from 50 to 20 by selecting 20 members whose envelop is representative of the entire ensemble. In other words, the standard deviations of the reduced (dimension 20) and that of the entire ensemble must be similar within some error rate (e.g., 5% of original deviation), cf. A.2.

In conclusion, the DOE dimension is 33: 20 coefficients for meteorological ensemble, 6 coefficients for the 6 source terms, and 7 other parameters shown in table 1. To sample data sets from unit simplex, we used the Dirichlet distribution, available in the package *scipy* of `PYTHON`, appendix A.3. Once the main meteorological members are selected, the dimension 20 is still high, uniform sampling by Dirichlet distribution has difficulties to reach the edges and corners of the space. In emergency case, simulations are often run in a deterministic way, which means that only one weather prediction and one source term will be chosen. In our case, a given meteorological member or a given source term corresponds to a corner of the sampling space, with coordinates all set to zeros except for one. In order to take into account these constraints in the DOE, we divided the DOE for the meteorological part into two parts: the first one was sampled by the uniform Dirichlet distribution with $\alpha = (1,1,\dots,1)$ and the second one follows another Dirichlet distribution with $\alpha = (0.1,\dots,0.1)$. $\alpha = (1,\dots,1)$ corresponds to a uniform sampling in the input space. When $\alpha(0.1,\dots,0.1)$, we increase the probability to sample in the corners and edges.

2.3 Outputs' meta-modeling

The ℓdX outputs (concentration, deposition and gamma dose rate) are data fields varying in time. The weather forecasts were temporally interpolated to obtain results every hours from March 12 at 00:00 to March 30 at 18:00 (UTC). There are in total 451 time steps in the ℓdX result. Then, these data were spatially interpolated so as to obtain the forecast at the measurement stations. The table 2 introduces information about the dimensions of ℓdX outputs at the measurement stations.

Output	Data shape
Concentration	108 stations × 451 one-hour time steps
Gamma dose rate	88 stations × 451 one-hour time steps
Deposition	> 30 000 measurement cells of a 0.1° spatial resolution (about 1 km)

Table 2: ℓdX outputs' information

However, meta-modeling is only carried out on scalar outputs. If we build a meta-model for each output element, the meta-model construction time will be very high. Hence, we first built the meta-models for model-to-data scores (RMSE, FMT—figure merit in time—or FMS—figure merit in space) or integrated values (temporally integral dose rate). This solution allows to estimate the general tendency of all stations, e.g. through the RMSE, or a station during the entire release period, e.g. FMT. In case it is necessary to emulate all simulated output instead of just a scalar, an algorithm can be used to reduce data dimension, PCA—Principal Component Analysis. This is a statistical procedure that reduces data by geometrically projecting them onto lower dimensions called principal components, with the goal of finding the best summary of the data using a limited number of principal components. This paper will not make use of this dimension reduction and focuses on scalar outputs.

In this study, an interpolation method is introduced to build the meta-model: Kriging. If Y is the original model we want to replace with a meta-model, the aim is to build an approximate model $\hat{Y}$ with a very low computational time. Let n be the number of samples in DOE and d the dimension of these points (here, $d = 33$), the DOE and learning response are written as follows:

- DOE: $X_{DOE} = \left[\underline{x}_1^T,\dots,\underline{x}_n^T\right], \quad \forall i \in \{1,\dots,n\}\ \underline{x}_i \in \mathbf{R}^d$, then $X \in \mathbf{R}^{d\times n}$,

- Response: $Y_{DOE} = (Y(\underline{x}_1), \ldots, Y(\underline{x}_n)) = (y_1, \ldots, y_n) \in \mathbf{R}^n$, where the $\{y_i\}_i$ are the RMSE or integrated dose rate, etc.

The meta-model consists in a linear regression part and an interpolation part, respectively noted F and Z in equation 2. The coefficients of the regression part are solution of a pseudo-linear system and the interpolation part Z is the residual of F:

$$\hat{Y}(\underline{x}) = F(\underline{x}) + Z(\underline{x}) \tag{2}$$

Kriging—Gaussian process Gaussian process (GP) modeling considers the deterministic response $\{y_i\}_i$ as a realization of a random function $\hat{Y}(\underline{x})$, which consists in a regression part F and a stationary centered stochastic process Z. According to [8], the latter is characterized by its correlation function between two input points x and x': $Cov(Y(\underline{x}), Y(\underline{x}')) = \sigma^2 R(\underline{x} - \underline{x}')$, where σ^2 denotes the variance of Y and R represents the correlation function between the outputs which is formulated as a function of the inputs. This part provides interpolation and spatial correlation properties.

Under the assumption of GP modeling, the joint distribution of response sample Y_{DOE} becomes a multivariate normal distribution:

$$Y_{DOE} \sim \mathscr{N}(F(X_{DOE}), \Sigma_{DOE}) \tag{3}$$

where the covariance matrix is $\Sigma_{DOE} = \left\{Cov\left(Y(\underline{x}_i), Y\left(\underline{x}_j\right)\right) = \sigma^2 R\left(\underline{x}_i - \underline{x}_j\right)\right\}_{ij}$. Let $\underline{x}^*$ be a new point not included in the DOE, the joint probability distribution of Y_{DOE} and $Y^* = Y(\underline{x}^*)$ is written as follows:

$$[Y_{DOE}, Y^*] \sim \mathscr{N}\left(\begin{pmatrix} F(X_{DOE}) \\ F(\underline{x}^*) \end{pmatrix}, \begin{pmatrix} \Sigma_{DOE} & k(\underline{x}^*) \\ k(\underline{x}^*)^T & \sigma^2 \end{pmatrix}\right) \tag{4}$$

where $k(\underline{x}^*) = (Cov(Y(\underline{x}^*), Y(\underline{x}_1)), \ldots, Cov(Y(\underline{x}^*), Y(\underline{x}_n)))^T$. The prediction of GP model on the point $\underline{x}^*$ is characterized by the following condition distribution:

$$Y^*_{|Y_{DOE}} \sim \mathscr{N}\left(F^*, \sigma^{*2}\right) \tag{5}$$

where $F^* = E\left[Y^*_{|Y_{DOE}}\right] = F(\underline{x}^*) + k(\underline{x}^*)^T \Sigma_{DOE}^{-1}(Y_{DOE} - F(X_{DOE}))$ is considered as the prediction value of GP model on $\underline{x}^*$, and $\sigma^{*2} = Var\left[Y^*_{|Y_{DOE}}\right] = \sigma^2 - k(\underline{x}^*)^T \Sigma_{DOE}^{-1} k(\underline{x}^*)$ implies the prediction error.

Our GP model was established using R package DiceKriging. The construction time takes about 3 hours to build, but after that, it takes only 12 ms to make a call to the meta-model (to be compared with the 5 minutes needed with the original, physical model). Let us now evaluate how close the meta-model is from the original model.

3 Evaluation of meta-model

Once the meta-model is built, an evaluation is essential before using it for other applications. Several methods exist to evaluate meta-models, and especially cross validation is applied for a wide range of machine learning approaches, [1]. Their aim is to create many meta-models by using only some points of DOE each time. Remaining points will be used as evaluation sample. A disadvantage of this procedure is that the training algorithm must be rerun several times, the validation can be costly when the fitting time is high. In addition, according to [3], removing some points in DOE can break the good structural properties of the Latin hypercube. In this study, we built a test sample sequentially as suggested by [4], called complementary LHS. The test sample is chosen from a low-discrepancy sequence such as the union of the original LHS and the complementary LHS minimizes the centered discrepancy L^2. In other words, the distribution of complementary-LHS points completes the initial LHS to fill in the space avoiding high closeness between training samples and test samples. This algorithm allows to qualify the surrogate model in low-frequency area.

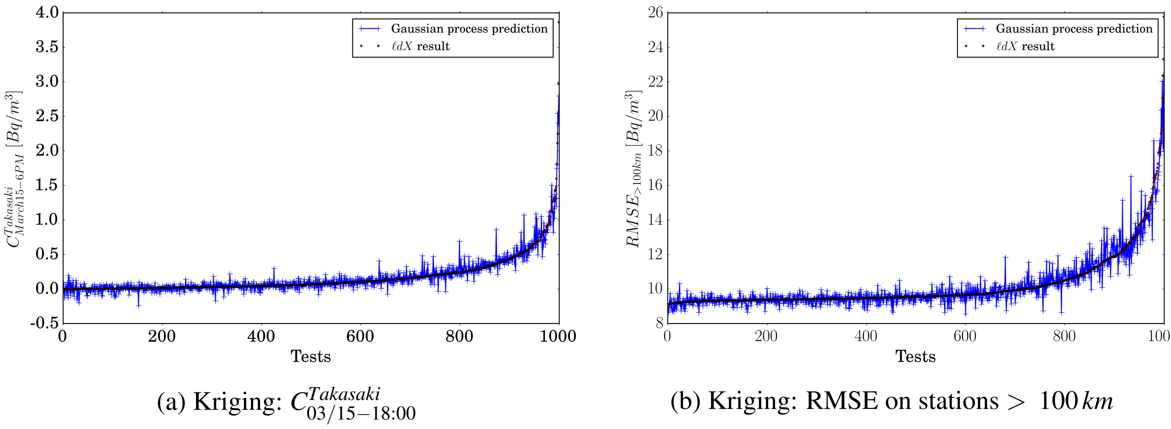
We also need a measure to evaluate the meta-models. The SMSE—standardized mean squared errors—between predictions by meta-model and responses of ℓdX was chosen. The formula for SMSE reads

$$SMSE = \frac{\overline{\left(Y^* - \hat{Y}\left(X^*\right)\right)^2}}{Var\left(Y^*\right)} \tag{6}$$

where X^* is the test sample in complementary LHS, $Y^* = Y\left(X^*\right)$ is the response of ℓdX and $\hat{Y}$ represents the meta-model. Like the other MSE measures, the SMSE is always positive and a value near to zero indicates an efficient meta-model. Additionally, dividing by the responses' variance provides a relative measure of quadratic error compared to ℓdX responses.

3.1 Validation by using SMSE score

Meta-models are built with a 10,000-sample DOE. A complementary sample of size 1000 is generated in addition. These points are chosen so as to be as far as possible to the 10,000 learning samples, hence to verify the quality of the meta-models in the worst cases. The figure 1 shows two emulation examples using Kriging method: the first case emulates the activity concentration in a station (Takasaki) at a given moment in time (March 15 18:00) (figure 1a), the second one calculates the RMSE score on all stations whose distance from the source is higher than 100 km (figure 1b), cf. 2. The 1000 samples were sorted in ascending order with the aim of making easier interpretation. Generally, meta-models predict rather well the output variation. The table 3 allows to further evaluate the results with the SMSE. We can conclude that the meta-models performance is very satisfactory since the errors remain below 9%.



(a) Kriging: $C^{Takasaki}_{03/15-18:00}$

(b) Kriging: RMSE on stations $>$ 100 km

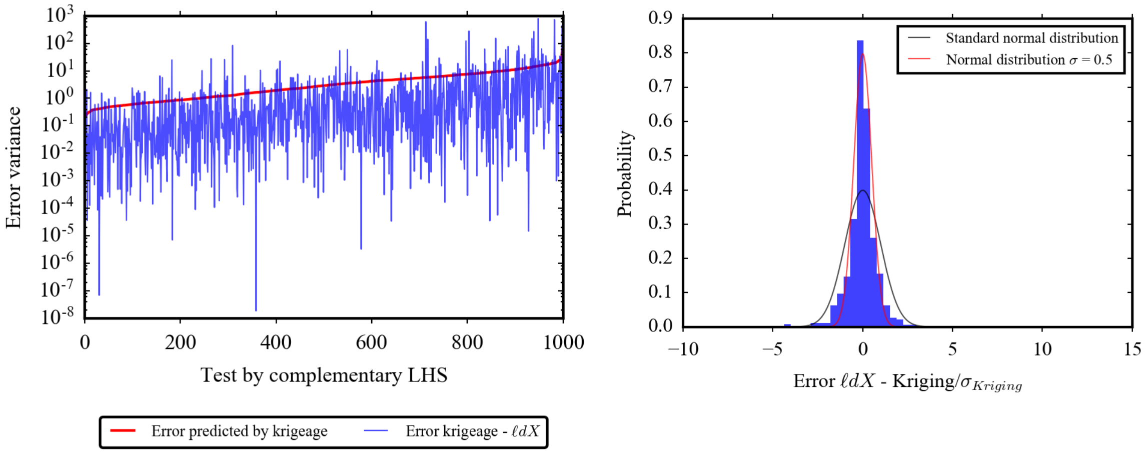
Figure 1: Predictions of Kriging meta-models against the values of the original, physical model.

Emulated scores	SMSE Kriging
Concentration in Takasaki at 18:00 March 15	0.088
Total concentration in Tokyo	0.027
RMSE in stations $>$ 100 km	0.086

Table 3: SMSE of Kriging surrogates. Errors are below 9%.

3.2 Verification of the error predicted by Kriging

According to the GP principle (see section 2.3), the Kriging interpolation of Y^* at point $\underline{x}^*$ follows a normal distribution whose mean is used as the emulator prediction and its standard deviation is regarded as prediction error estimated by the meta-model itself. This value provides a complementary view of the meta-model reliability. So the question arises: how much could we trust this information? A study was carried out on the relationship between the model–meta-model error, called $\varepsilon_{\ell dX-Kriging}$, and the standard deviation of forecast computed by the GP model, called $\sigma_{Kriging}$. The Kriging construction shows that $\varepsilon_{\ell dX-Kriging} \sim \mathcal{N}(0, \sigma_{Kriging})$, then the fraction $\frac{\varepsilon_{\ell dX-Kriging}}{\sigma_{Kriging}}$ must follow a standard normal distribution. Using the test on the complementary LHS, the figure 2a plots $\varepsilon_{\ell dX-Kriging}$ in blue, and $\sigma_{Kriging}$ in red. As in figure 1, the result was sorted in ascending order and the figure was drawn in log scale. The model–meta-model error is often lower than predicted error, which means that the GP model has a tendency to overestimate discrepancy between ℓdX and its surrogate. The figure 2b shows that the distribution of the fraction $\frac{\varepsilon_{\ell dX-Kriging}}{\sigma_{Kriging}}$ looks more like a normal distribution with $\sigma = 0.5$ than the standard normal distribution. Therefore, the GP model doubles its prediction error, which probably comes from a overestimation of the variance σ^2 in the Kriging process, as estimated by DiceKriging. Estimator of $\sigma_{Kriging}$ can be modified in the GP construction but this result is reassuring.



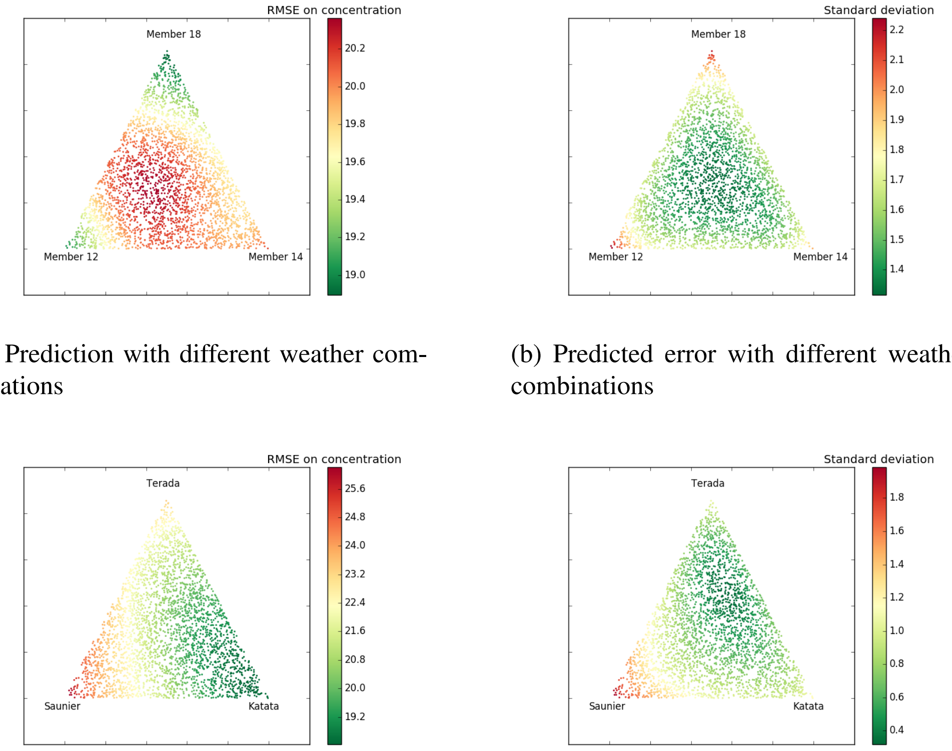
(a) Visualization of the error predicted by Kriging (red) and the actual model–meta-model error (blue)

(b) Histogram of the error fraction $\frac{\varepsilon_{\ell dX-Kriging}}{\sigma_{Kriging}}$

Figure 2: Comparison between predicted and actual errors.

4 Application: minimization of RMSE on activity concentration

An objective of the calibration is to determine the best set of parameters in order to minimize the model-to-data discrepancy. This study requires many runs of the model, hence the meta-model is a useful replacement tool because of its short execution time. The first ℓdX meta-model was built to determine the set of parameters that minimizes the RMSE score of activity concentration. This section introduces firstly an analysis on some cross-sections of the RMSE. It allows to clarify the meta-model performance as some inputs vary. Emulating the RMSE of concentration by Kriging, the figure 3 shows this RMSE when either varying the weights for three meteorological ensemble members (the other members receive a null weight), or for three source terms ([5], [12] and [10]; the other source terms receive a null weight). Also, the prediced error on the RMSE is shown for reference. All other inputs are set to their reference value (i.e., without perturbation). In the left column, the red area corresponds to an important model-to-data discrepancy. This means that the meteorological ensemble members number 12 and 18 perform better than the member 14. Similarly, in that case, the source term [5] performs better than the terms from [12] or [10]. In the right column, we see that the highest meta-modeling errors are in the corners. Because a unit simplex in high dimension (20 or 6) is a very large space, this phenomenon can be explained by low sampling frequency in the corners and boundaries of the input space. In any case, the prediction error is almost always below 10%.



(a) Prediction with different weather combinations

(b) Predicted error with different weather combinations

(c) Prediction with different source term combinations

(d) Predicted error with different source term combinations

Figure 3: Cross-sections of the meta-model on the RMSE of the concentration.

Figure 3 shows that in some directions, the RMSE function seems to reach a minimum value, but we found that an optimization has difficulties in finding a global mimimum. One hypothesis is that the RMSE function is rather flat in vast regions of the input space. We analyzed the Euclidean distance between five points from the DOE with very low RMSE on the activity, see table 4. The points, which correspond to the lowest model-to-data errors, are not located in the same area of the input space. The figure 4 shows that there are many points in the input space, whose RMSE values drop almost to the lowest, while they are far from each other. This explains why optimization algorithm could not converge in this case.

	Best point	Point 2	Point 3	Point 4	Point 5
Best point	0.	1.504	0.943	1.228	0.967
Point 2	-	0.	1.658	1.747	0.967
Point 3	-	-	0.	1.49	1.325
Point 4	-	-	-	0.	1.073

Table 4: Euclidean distance between the five DOE points with the lowest RMSE on the activity. Note that the average distance between the DOE samples is 1.48, which means that the distances of the table are rather high.

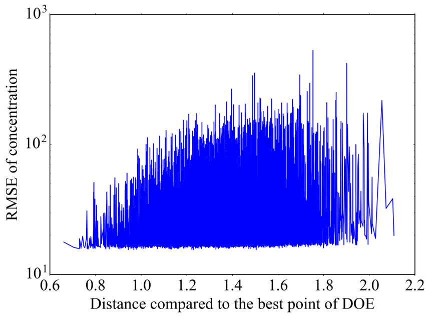


Figure 4: The activity RMSE according to the distance from the DOE point with the lowest RMSE. Even far from the best DOE point, very low RMSE are found.

We created a 1 000 000-points LHS; and compared the minimal RMSE value foretold by Kriging emulator on this denser LHS with the best score of the DOE. The points with the lowest RMSEs tend to give more weight to the source term from [11]. This source term was built *a posteriori*, after the Fukushima nuclear accident by using the concentration measurements and the ℓdX model.

5 Conclusion and perspectives

We were able to build a meta-model of the ℓdX dispersion model, using 10,000 calls to ℓdX. The samples were chosen so as to properly sample the range of variations of the inputs we chose to perturb. A difficulty lied in the treatment of the meteorological ensemble and of the source terms. We managed to sample from convex combinations of those. The meta-model was built unsing Kriging as an interpolation method between the 10,000 samples. Note that one meta-model was built for each output of interest, like the RMSE between the model outputs and the field observations. Besides the cost of running 10,000 simulations, building a meta-model requires 3 hours of computations, but then, any call to the meta-model only costs 12 ms. The speed of the meta-model allowed us to investigate the calibration of its inputs.

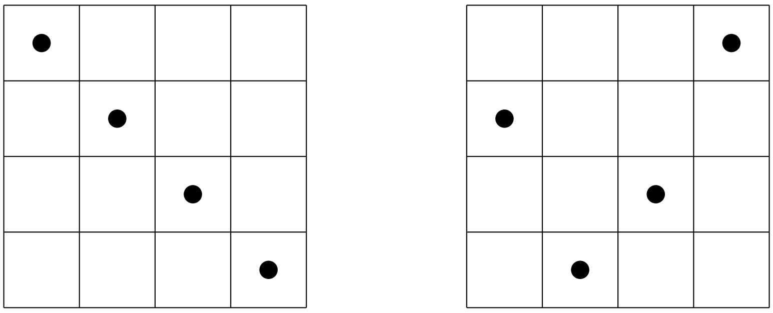
We faced difficulties in finding a global minimum. Deterministic optimization provided many local minima. In the perspective of uncertainty quantification where we look for the distribution of the inputs (not just the most probable value), this means that vast areas of the input space should be associated with a significant probabity density. The Markov chain Monte Carlo (MCMC) methods may be used as a sampling algorithms.

The dimension, which is fairly high in our case (33), may explain some difficulties in the optimization. We could also perturb the meteorological inputs by applying a multiplicative factor k to the ensemble standard deviation and adding this up to the ensemble mean. Here, this factor k would replace the convex combination, which would greatly lower the input dimension.

In the calibration, using different measurement types is essential, e.g. calibrate input parameters using simultaneously concentration and deposition measurements. It allows to compensate for the shortcomings of each measurements. According to [9], deposition was measured on a fine and regular mesh, this is a good spatial cover but it does not provide temporal information. On the contrary, the activity concentration is observed at a limited number of measurement stations, but the observations were collected every hours with a low measurement error.

A Appendices

A.1 LHS



(a) Without optimization

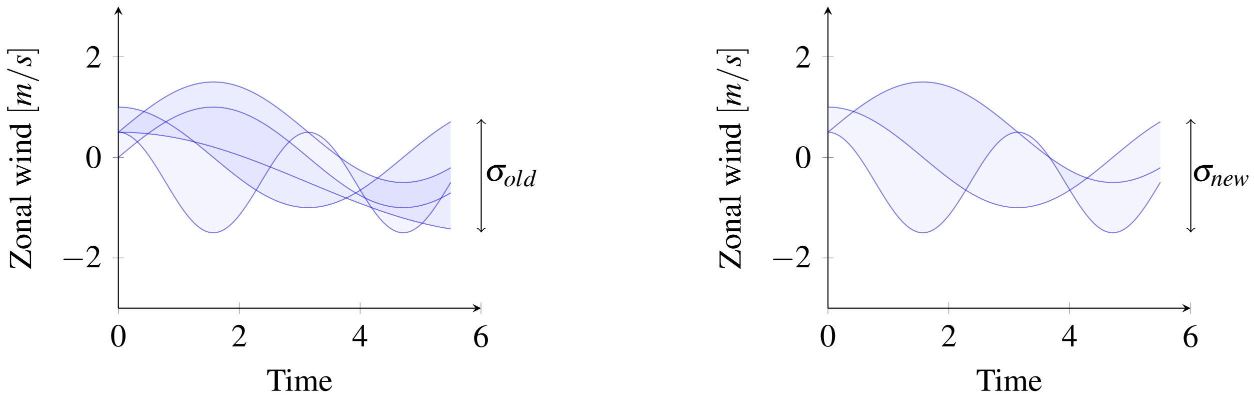
(b) After optimization

Figure 5: Illustration of Latin Hyper-cube Sampling

Algorithm 1 Reduce weather ensemble size

```
Input
    $N_{new}$ The size of new ensemble
    $\varepsilon$ Error rate
    $M$ $= (m_1, \dots, m_N)$ the original ensemble, $m_i \in R_t^N$ where $N - t$ is the number of time steps
Output
    $M_{new}$ $= (m_1, \dots, m_{N_{new}})$ the vector contains indices of selected members
while $\frac{\sigma_{new}}{\sigma_{old}}.100 > \varepsilon$ do
    Compute $\sigma_{old} = \frac{1}{N_t} \sum_{j=1}^{N_t} \sqrt{Var(M^j)}$ where $M^j = \left(m_1^j, \dots, m_N^j\right)$
    Select $N_{new}$ within $N$ members of the original ensemble
    Compute $\sigma_{new} = \frac{1}{N_t} \sum_{j=1}^{N_t} \sqrt{Var\left(M_{new}^j\right)}$
end while
```

A.2 Reduction of weather ensemble size



(a) Original ensemble

(b) New ensemble

Figure 6: Illustration of reduction of ensemble size

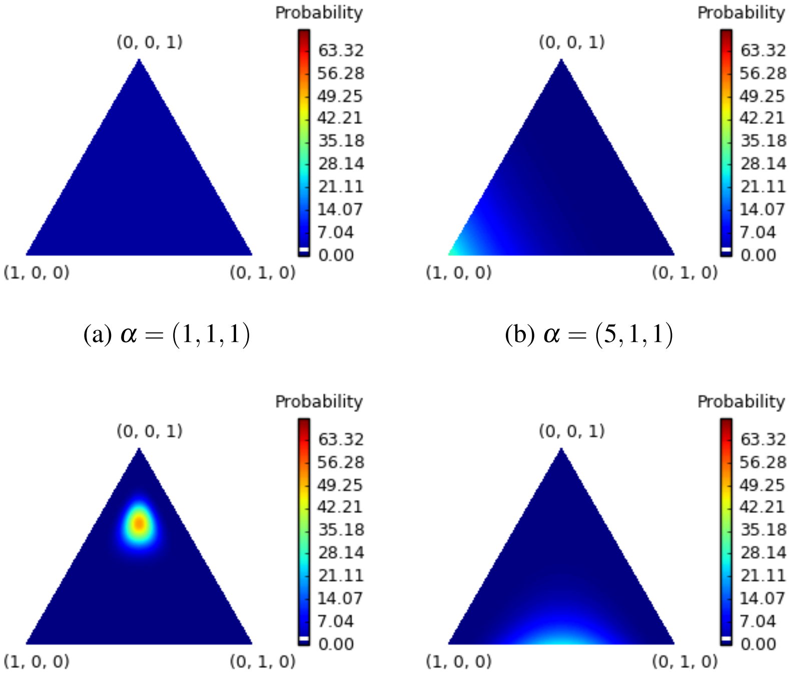
A.3 Dirichlet distribution

The Dirichlet distribution is a family of continuous multivariate probability distributions, parameterized by a n-dimension vector α of positive reals. The distribution samples n-vectors $v = (v_1, \dots, v_n)$ whose entries are real numbers in the interval $[0,1]$ and whose sum is equal to 1. The main feature of a Dirichlet distribution $Dir(\alpha = (\alpha_1, \dots, \alpha_n))$ can be described as in table 5. The figure 7 exposes in dimension three the probability density function (PDF) of some Dirichlet distributions with different α vectors. When $\alpha = (1, \dots, 1)$, the PDF is similar everywhere in the triangle. The blue area implies low probability of appearance and the red one means strong probability. The white line on the color bar indicates the PDF value of $\alpha = (1, \dots, 1)$.

Support	$v = (v_1, \ldots, v_n), \quad (v_i)_i \in [0,1]$ and $\sum_{i=1}^n v_i = 1$
PDF	$f_\alpha(v) = \frac{1}{B(\alpha)} \prod_{i=1}^n v_i^{\alpha_i} - 1$
Mean	$E[V_i] = \frac{\alpha_i}{\alpha_0}$
Variance	$Var(V_i) = \frac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(\alpha_0+1)}$
Covariance	$Cov(V_i, V_j) = \frac{-\alpha_i\alpha_j}{\alpha_0(\alpha_0+1)}, \quad (i \neq j)$

Table 5: Dirichlet distribution description.

where $B(\alpha) = \dfrac{\prod_{i=1}^n \Gamma(\alpha)}{\Gamma(\sum_{i=1}^n \alpha_i)}$ and $\alpha_0 = \sum_{i=1}^n \alpha_i$.



(a) $\alpha = (1,1,1)$

(b) $\alpha = (5,1,1)$

(c) $\alpha = (10,10,30)$

(d) $\alpha = (5,5,1)$

Figure 7: Illustrations of Dirichlet distributions in dimension three.

REFERENCES

[1] H. Blockeel and J. Struyf. Efficient algorithms for decision tree cross-validation. *Journal of Machine Learning Research*, 3:pp.621–650, 2002.

[2] S. Girard, I. Korsakissok, and V. Mallet. Screening sensitivity analysis of a radionuclides atmospheric dispersion model applied to the fukushima disaster. *Atmospheric environment*, 95:pp.490–500, 2014.

[3] S. Girard, V. Mallet, I. Korsakissok, and A. Mathieu. Emulation and sobol' sensitivity analysis of an atmospheric dispersion model applied to the fukushima nuclear accident. *Journal of Geophysical Research: Atmospheres, American Geophysical Union*, Journal of Geophysical Research: Atmospheres, American Geophysical Union:pp.3484 – 3496, 2016.

[4] B. Iooss, L. Boussouf, V. Feuillard, and A. Marrel. Numerical studies of the metamodel fitting and validation processes. *International Journal of Advances in Systems and Measurements*, 3:pp.11–21, 01 2010.

[5] G. Katata, M. Chino, T. Kobayashi, H. Terada, M. Ota, H. Nagai, M. Kajino, R. Draxler, M. C. Hort, A. Malo, T. Torii, and Y. Sanada. Detailed source term estimation of the atmospheric release for the fukushima daiichi nuclear power station accident by coupling simulations of an atmospheric dispersion model with an improved deposition scheme and oceanic dispersion model. *Atmos. Chem. Phys.*, 15:pp.1029–1070, 2015.

[6] N.B.T Le, I. Korsakissok, A. Mathieu, V. Mallet, and R. Périllat. Uncertainty study on atmopsheric dispersion model by using monte carlo perturbation, and applied to the fukushima nuclear accident: Part ii - long-range model. To be submitted, 2019.

[7] V. Mallet, D. Quélo, B. Sportisse, M. Ahmed de Biasi, É. Debry, I. Korsakissok, L. Wu, Y. Roustan, K. Sartelet, M. Tombette, and H. Foudhi. Technical note: The air quality modeling system polyphemus. *Atmospheric Chemistry and Physics*, 7(20):pp.5479–5487, 2007.

[8] A. Marrel, B. Iooss, B. Laurent, and O. Roustant. Calculations of sobol indices for the gaussian process metamodel. *Reliability Engineering and System Safety*, 94:pp.742–751, 2009.

[9] A. Mathieu, M. Kajino, I. Korsakissok, R. Périllat, D. Quélo, A. Quérel, O. Saunier, T. T. Sekiyama, Y. Igarashi, and D. Didier. Fukushima daiichi–derived radionuclides in the atmosphere, transport and deposition in japan: A review. *Applied Geochemistry*, 91:pp.122–139, 2018.

[10] O. Saunier, A. Mathieu, D. Didier, M. Tombette, D. Quélo, V. Winiarek, and M. Bocquet. An inverse modeling method to assess the source term of the fukushima nuclear power plant accident using gamma dose rate observations. *Atmospheric Chemistry and Physics*, 13(22):pp.11403–11421, 2013.

[11] O. Saunier, A. Mathieu, T.T. Sekiyama, M. Kajino, K. Adachi, M. Bocquet, T. Maki, Higarashi, and D. Didier. A new perspective on the fukushima releases brought by newly available 137cs air concentration observations and reliable meteorological fields. 17th International Conference on Harmonisation within Atmospheric Dispersion Modelling for Regulatory Purposes, The publisher, 2016. Budapest, Hungary.

[12] H. Terada, G. Katata, M. Chino, and H. Nagai. Atmospheric discharge and dispersion of radionuclides during the fukushima dai-ichi nuclear power plant accident. part ii: verification of the source term and analysis of regional-scale atmospheric dispersion. *Journal of Environmental Radioactivity*, 112:pp.141–154, 2012.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

SEMI-SUPERVISED REGRESSION USING CLUSTER ENSEMBLE AND LOW-RANK CO-ASSOCIATION MATRIX DECOMPOSITION UNDER UNCERTAINTIES

Vladimir Berikov[1,2], **and Alexander Litvinenko**[3]

[1]Sobolev Institute of Mathematics, Novosibirsk, Russia
e-mail: berikov@math.nsc.ru

[2]Novosibirsk State University, Novosibirsk, Russia

[3] RWTH Aachen, Aachen, Germany
e-mail: litvinenko@uq.rwth-aachen.de

Keywords: Semi-supervised regression, cluster ensemble, co-association matrix, graph Laplacian regularization, low-rank matrix decomposition, hierarchical matrices.

Abstract. *In this paper, we solve a semi-supervised regression problem. Due to the luck of knowledge about the data structure and the presence of random noise, the considered data model is uncertain. We propose a method which combines graph Laplacian regularization and cluster ensemble methodologies. The co-association matrix of the ensemble is calculated on both labeled and unlabeled data; this matrix is used as a similarity matrix in the regularization framework to derive the predicted outputs. We use the low-rank decomposition of the co-association matrix to significantly speedup calculations and reduce memory. Numerical experiments using the Monte Carlo approach demonstrate robustness, efficiency, and scalability of the proposed method.*

1 Introduction

Machine learning problems can be classified as supervised, unsupervised and semi-supervised. Let data set $\mathbf{X} = \{x_1, \ldots, x_n\}$ be given, where $x_i \in \mathbf{R}^d$ is feature vector, d is feature space dimensionality. In a supervised learning context, we are given an additional set $Y = \{y_1, \ldots, y_n\}$ of target feature values (labels) for all data points, $y_i \in D_Y$, where D_Y is target feature domain. In the case of supervised classification, the domain is an unordered set of categorical values (classes, patterns). In case of supervised regression, the domain $D_Y \subseteq \mathbf{R}$. Using this information (which can be thought as provided by a certain "teacher"), it is necessary to find a decision function $y = f(x)$ (classifier, regression model) for predicting target feature values for any new data point $x \in \mathbf{R}^d$ from the same statistical population [5]. The function should be optimal in some sense, e.g., give minimal value to the expected losses.

In an unsupervised learning setting, the target feature values are not provided. The problem of cluster analysis, which is an important direction in unsupervised learning, consists in finding a partition $P = \{C_1, \ldots, C_K\}$ of $\mathbf{X}$ on a relatively small number of homogeneous clusters describing the structure of data. As a criterion of homogeneity, it is possible to use a functional dependent on the scatter of observations within groups and the distances between clusters. The desired number of clusters is either a predefined parameter or should be found in the best way.

We note that since the final cluster partition is uncertain due to random noise in sample data, luck of knowledge about the feature set and the data structure, parameters, weights, and initialization settings, a set of different cluster partitions is calculated. Then a final cluster partition is formed.

In semi-supervised learning problems, the target feature values are known only for a part of data set $X_1 \subset \mathbf{X}$. It is possible to assume that $X_1 = \{x_1, \ldots, x_{n_1}\}$, and the unlabeled part is $X_0 = \{x_{n_1+1}, \ldots, x_n\}$. The set of labels for points from X_1 is denoted by $Y_1 = \{y_1, \ldots, y_{n_1}\}$. It is required to predict target feature values as accurately as possible either for given unlabeled data X_0 (i.e., perform *transductive learning*) or for arbitrary new observations from the same statistical population (*inductive learning*). In dependence of the type of the target feature, one may consider semi-supervised classification or semi-supervised regression problems [31].

The task of semi-supervised learning is important because in many real-life problems only a small part of available data can be labeled due to the large cost of target feature registration. For example, manual annotation of digital images is rather time-consuming. Therefore labels can be attributed to only a small part of pixels. To improve prediction accuracy, it is necessary to use information contained in both labeled and unlabeled data. An important application is hyperspectral image semi-supervised classification [8].

In this paper, we consider a semi-supervised regression problem in the transductive learning setting. In semi-supervised regression, the following types of methods can be found in the literature [18]: co-training [30], semi-supervised kernel regression [26], graph-based and spectral regression methods [27, 12, 28], etc.

We propose a novel semi-supervised regression method using a combination of graph Laplacian regularization technique and cluster ensemble methodology. Graph regularization (sometimes called manifold regularization) is based on the assumption which states that if two data points are on the same manifold, then their corresponding labels are close to each other. A graph Laplacian is used to measure the smoothness of the predictions on the data manifold including both labeled and unlabeled data [29, 1].

Ensemble clustering aims at finding consensus partition of data using some base clustering algorithms. As a rule, application of this methodology allows one to get a robust and effective solution, especially in case of uncertainty in the data model. Properly organized ensemble (even composed of ”weak” learners) significantly improves the overall clustering quality [7].

Different schemes for applying ensemble clustering for semi-supervised classification were proposed in [25, 2]. The suggested methods are based on the hypothesis which states that a preliminary ensemble allows one to restore more accurately metric relations in data in noise conditions. The obtained co-association matrix (CM) depends on the outputs of clustering algorithms and is less noise-addicted than a conventional similarity matrix. This increases the prediction quality of the methods.

The same idea is pivotal in the proposed semi-supervised regression method. We assume a statistical connection between the clustering structure of data and the predicted target feature. Such a connection may exist, for example, when some hidden classes are present in data, and the belonging of objects to the same class influences the proximity of their responses.

To decrease the computational cost and the storage requirement and to increase the scalability of the method, we suggest usage of low-rank (or hierarchical) decomposition of CM. This decomposition will reduce the numerical cost and storage from cubic to (log-)linear [16].

Parametric approximations, given by generalized linear models, as well as nonlinear models, given by neural networks were compared in [6].

In the rest of the paper, we describe the details of the suggested method. Numerical experiments are presented in the correspondent section. Finally, we give concluding remarks.

2 Combined semi-supervised regression and ensemble clustering

2.1 Graph Laplacian regularization

We consider a variant of graph Laplacian regularization in semi-supervise transductive regression which solves the following optimization problem:

find f^* such that $f^* = \arg \min\limits_{f \in \mathbf{R}^n} Q(f)$, where

$$Q(f) := \frac{1}{2}\left(\sum_{x_i \in X_1} (f_i - y_i)^2 + \alpha \sum_{x_i, x_j \in \mathbf{X}} w_{ij}(f_i - f_j)^2 + \beta ||f||^2 \right), \tag{1}$$

$f = (f_1, \ldots, f_n)$ is a vector of predicted outputs: $f_i = f(x_i)$; $\alpha, \beta > 0$ are regularization parameters, $W = (w_{ij})$ is data similarity matrix. The degree of similarity between points x_i and x_j can be calculated using appropriate function, for example from the Matérn family [22]. The Matérn function depends only on the distance $h := \|x_i - x_j\|$ and is defined as $W(h) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)} \left(\frac{h}{\ell}\right)^{\nu} K_\nu\left(\frac{h}{\ell}\right)$ with three parameters ℓ, ν, and σ^2. For instance, $\nu = 1/2$ gives the well-known exponential kernel $W(h) = \sigma^2 \exp(-h/\ell)$, and $\nu = \infty$ gives the Gaussian kernel $W(h) = \sigma^2 \exp(-h^2/2\ell^2)$.

In this paper we also use RBF kernel with parameter ℓ: $w_{ij} = \exp(-\frac{\|x_i - x_j\|^2}{2\ell^2})$.

The first term in right part of (1) minimizes fitting error on labeled data; the second term aims to obtain ”smooth” predictions on both labeled and unlabeled sample; the third one is Tikhonov’s regularizer.

Let graph Laplacian be denoted by $L = D - W$ where D be a diagonal matrix defined by $D_{ii} = \sum\limits_j w_{ij}$. It is easy to show (see, e.g., [1, 29]) that

$$\sum_{x_i, x_j \in \mathbf{X}} w_{ij}(f_i - f_j)^2 = 2f^T Lf. \tag{2}$$

Let us introduce vector $Y_{1,0} = (y_1, \dots, y_{n_1}, \underbrace{0, \dots, 0}_{n-n_1})^T$, and let G be a diagonal matrix:

$$G = diag(G_{11} \dots, G_{nn}),\ G_{ii} = \begin{cases} \beta+1,\ i=1,\dots,n_1 \\ \beta,\ i=n_1+1,\dots,n, \end{cases}. \tag{3}$$

Differentiating $Q(f)$ with respect to f, we get

$$\frac{\partial Q}{\partial f}\ |_{f=f^*} = Gf^* + \alpha L f^* - Y_{1,0} = 0,$$

hence

$$f^* = (G + \alpha L)^{-1}\, Y_{1,0} \tag{4}$$

under the condition that the inverse of matrix sum exists (note that the regularization parameters α, β can be selected to guaranty the well-posedness of the problem). Numerical methods such as Tikhonov or Lavrentiev regularization [24] can also be used to obtain the predictions.

2.2 Co-association matrix of cluster ensemble

In the proposed method, we use a co-association matrix of cluster ensemble as similarity matrix in (1). Co-association matrix is calculated as a preliminary step in the process of cluster ensemble design with various clustering algorithms or under variation across a given algorithm's parameter settings [13].

Let us consider a set of partition variants $\{P_l\}_{l=1}^r$, where $P_l = \{C_{l,1}, \dots, C_{l,K_l}\}$, $C_{l,k} \subset \mathbf{X}$, $C_{l,k} \bigcap C_{l,k'} = \varnothing$, K_l is number of clusters in lth partition. For each P_l we determine matrix $H_l = (h_l(i,j))_{i,j=1}^n$ with elements indicating whether a pair x_i, x_j belong to the same cluster in lth variant or not: $h_l(i,j) = \mathbb{I}[c_l(x_i) = c_l(x_j)]$, where $\mathbb{I}(\cdot)$ is indicator function ($\mathbb{I}[true] = 1$, $\mathbb{I}[false] = 0$), $c_l(x)$ is cluster label assigned to x. The weighted averaged co-association matrix (WACM) is defined as follows:

$$H = (H(i,j))_{i,j=1}^n, \quad H(i,j) = \sum_{l=1}^r w_l H_l(i,j) \tag{5}$$

where $w_1, \dots, w_r$ are weights of ensemble elements, $w_l \geq 0$, $\sum w_l = 1$. The weights should reflect the "importance" of base clustering variants in the ensemble [4] and be dependent on some evaluation function Γ (cluster validity index, diversity measure) [3]: $w_l = \gamma_l / \sum\limits_{l'} \gamma_{l'}$, where $\gamma_l = \Gamma(l)$ is an estimate of clustering quality for the lth partition (we assume that a larger value of Γ manifests better quality).

In the methodology presented in this paper, the elements of WACM are viewed as similarity measures learned by the ensemble. In a sense, the matrix specifies the similarity between objects in a new feature space obtained utilizing some implicit transformation of the initial data. The following property of WACM allows increasing the processing speed.

Proposition 1. Weighted averaged co-association matrix admits low-rank decomposition in the form:

$$H = BB^T,\ B = [B_1 B_2 \dots B_r] \tag{6}$$

where B is a block matrix, $B_l = \sqrt{w_l}\, A_l$, A_l is ($n \times K_l$) cluster assignment matrix for lth partition: $A_l(i,k) = \mathbb{I}[c(x_i) = k]$, $i = 1, \dots, n$, $k = 1, \dots, K_l$.

The proof is fairly straightforward and is omitted here for the sake of brevity. As a rule, $m = \sum_l K_l \ll n$, thus (6) gives us an opportunity of saving memory by storing $(n \times m)$ sparse matrix instead of full $(n \times n)$ co-association matrix. The complexity of matrix-vector multiplication $H \cdot x$ is decreased from $O(n^2)$ to $O(nm)$.

2.3 Cluster ensemble and graph Laplacian regularization

Let us consider graph Laplacian in the form: $L' = D' - H$, where $D' = \text{diag}(D'_{11}, \ldots, D'_{nn})$, $D'_{ii} = \sum\limits_{j} H(i,j)$. We have:

$$D'_{ii} = \sum_{j=1}^{n} \sum_{l=1}^{r} w_l \sum_{k=1}^{K_l} A_l(i,k) A_l(j,k) =$$

$$\sum_{l=1}^{r} w_l \sum_{k=1}^{K_l} A_l(i,k) \sum_{j=1}^{n} A_l(j,k) = \sum_{l=1}^{r} w_l N_l(i) \quad (7)$$

where $N_l(i)$ is the size of the cluster which includes point x_i in lth partition variant.

Substituting L' in (4), we obtain cluster ensemble based predictions of output feature in semi-supervised regression:

$$f^{**} = (G + \alpha L')^{-1}\, Y_{1,0}. \quad (8)$$

Using law-rank representation of H, this expression can be transformed into the form which involves more efficient matrix operations.

Using law-rank representation of H, we get:

$$f^{**} = (G + \alpha D' - \alpha B B^T)^{-1}\, Y_{1,0}.$$

In linear algebra, the following Woodbury matrix identity is known:

$$(S + UV)^{-1} = S^{-1} - S^{-1}U(I + VS^{-1}U)^{-1}VS^{-1}$$

where $S \in \mathbf{R}^{n \times n}$ is invertible matrix, $U \in \mathbf{R}^{n \times m}$ and $V \in \mathbf{R}^{m \times n}$. We can denote $S = G + \alpha D'$ and get

$$S^{-1} = \text{diag}(1/(G_{11} + \alpha D'_{11}), \ldots, 1/(G_{nn} + \alpha D'_{nn})) \quad (9)$$

where G_{ii}, D'_{ii}, $i = 1, \ldots, n$ are defined in (3) and (7) correspondingly.

Now it is clear that the following statement is valid:

Proposition 2. Cluster ensemble based target feature prediction vector (8) can be calculated using low-rank decomposition as follows:

$$f^{**} = (S^{-1} + \alpha S^{-1} B (I - \alpha B^T S^{-1} B)^{-1} B S^{-1})\, Y_{1,0} \quad (10)$$

where matrix B is defined in (6) and S^{-1} in (9).

Note that in (10) we need to invert significantly smaller ($m \times m$) sized matrix instead of ($n \times n$) in (8). The overall computational complexity of (10) can be estimated as $O(nm + m^3)$.

The outline of the suggested algorithm of semi-supervised regression based on the law-rank decomposition of the co-association matrix (SSR-LRCM) is as follows.

Algorithm SSR-LRCM
Input:
$\mathbf{X}$: dataset including both labeled and unlabeled sample;
Y_1: target feature values for labeled instances;
r: number of runs for base clustering algorithm μ;
$\mathbf{\Omega}$: set of parameters (working conditions) of clustering algorithm.

Output:
f^{**}: predictions of target feature for labeled and unlabeled objects.
Steps:
1. Generate r variants of clustering partition with algorithm μ for working parameters randomly chosen from $\mathbf{\Omega}$; calculate weights $w_1, \dots, w_r$ of variants.
2. Find graph Laplacian in law-rank representation using matrices B in (6) and D' in (7);
3. Calculate predictions of target feature according to (10).
end.

In the implementation of SSR-LRCM, we use K-means as base clustering algorithm which has linear complexity with respect to data dimensions.

3 Hierarchical Approximation

In this section we discuss the case if matrices W and H do not have any low-rank decomposition or this low-rank is expensive (e.g., the rank is comparable with n). In that case then one can try to apply, so-called, hierarchical matrices ($\mathcal{H}$-matrices), introduced in [15], [16] or, as an alternative, low-rank tensor techniques [21, 23].

The $\mathcal{H}$-matrix format has a log-linear computational cost[1] and storage. The $\mathcal{H}$-matrix technique allows us to efficiently work with general matrices W and H (and not only with structured ones like Toeplitz, circulant or three diagonal). Another advantage is that all linear algebra operations from Sections 2.1 and 2.2 preserve (or only slightly increase) the rank k inside of each sub-block.

There are many implementations of $\mathcal{H}$-matrices exist, e.g., the HLIB library (http://www.hlib.org/), $\mathcal{H}^2$-library (https://github.com/H2Lib), and HLIBpro library (https://www.hlibpro.com/). We used the HLIBpro library, which is actively supported commercial, robust, parallel, very tuned, and well tested library. Applications of the $\mathcal{H}$-matrix technique to the graph Laplacian can be found in the HLIBpro library[2], and to covariance matrices in [17] and in [20].

The $\mathcal{H}$-matrix technique is defined as a hierarchical partitioning of a given matrix into sub-blocks followed by the further approximation of the majority of these sub-blocks by low-rank matrices. Figure 1 shows an example of the $\mathcal{H}$-matrix approximation $\widetilde{W}$ of an $n \times n$ matrix W, $n = 16000$ and its Cholesky factor $\widetilde{U}$, where $\widetilde{W} = \widetilde{U}\widetilde{U}^\top$. The dark (or red) blocks indicate the dense matrices and the grey (green) blocks indicate the rank-k matrices; the number inside each block is its rank. The steps inside the blocks show the decay of the singular values in $\log$ scale. The Cholesky factorization is needed for computing the inverse, $\widetilde{W}^{-1} = (\widetilde{U}\widetilde{U}^\top)^{-1} = \widetilde{U}^{-\top}\widetilde{U}^{-1}$. This way is cheaper as computing the inverse directly.

To define which sub-blocks can be approximated well by low-rank matrices and which cannot, a so-called admissibility condition is used (see more details in [20]). There are different admissibility conditions possible: weak, strong, domain decomposition based. Each one results in a new subblock partitioning. Blocks that satisfy the admissibility condition can be approximated by low-rank matrices; see [15].

On the first step, the matrix is divided into four sub-blocks. Then each (or some) sub-block(s) is (are) divided again and again hierarchically until sub-blocks are sufficiently small. The procedure stops when either one of the sub-block sizes is n_{min} or smaller (typically $n_{\text{min}} \leq 128$), or when this sub-block can be approximated by a low-rank matrix.

[1] log-linear means $\mathcal{O}(kn \log n)$, where the rank k is a small integer, and n is the size of the data set
[2] https://www.hlibpro.com/

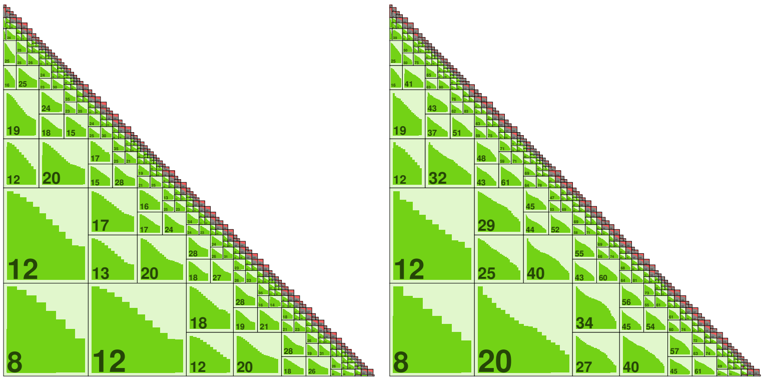


Figure 1: (left) An example of the $\mathcal{H}$-matrix approximation $\widetilde{W}$ of an $n \times n$ matrix W, $n = 16000$. (right) The corresponding Cholesky factor $\widetilde{U}$, where $\widetilde{W} = \widetilde{U}\widetilde{U}^{\top}$.

Another important question is how to compute these low-rank approximations. One (heuristic) possibility is the Adaptive Cross Approximation (ACA) algorithm [16], which performs the approximations with a linear complexity $\mathcal{O}(kn)$ in contrast to $\mathcal{O}(n^3)$ by the standard singular value decomposition (SVD).

The storage requirement of $\widetilde{W}$ and the matrix vector multiplication cost $\mathcal{O}(kn\log n)$, the matrix-matrix addition costs $\mathcal{O}(k^2 n\log n)$, and the matrix-matrix product and the matrix inverse cost $\mathcal{O}(k^2 n\log^2 n)$; see [15]. In Table 1 we show dependence of the two matrix errors on the $\mathcal{H}$-matrix rank k for the Matérn function with parameters $\ell = \{0.25, 0.75\}$, $\nu = 1.5$, and $x_i, x_j \in [0,1]^2$. We can bound the relative error $\|W^{-1} - \widetilde{W}^{-1}\|/\|W^{-1}\|$ for the approximation of the inverse as

$$\frac{\|W^{-1} - \widetilde{W}^{-1}\|}{\|W^{-1}\|} = \frac{\|(I - \widetilde{W}^{-1}W)W^{-1}\|}{\|W^{-1}\|} \leq \|(I - \widetilde{W}^{-1}W)\|.$$

$\|(I - \widetilde{W}^{-1}W)\|_2$ can be estimated by few steps of the power iteration method. The rank $k \leq 20$ is not sufficient to approximate the inverse.The spectral norms of $\tilde{W}$ are $\|\widetilde{W}_{(\ell=0.25)}\|_2 = 720$ and $\|\widetilde{W}_{(\ell=0.75)}\|_2 = 1068$.

Table 1: Convergence of the $\mathcal{H}$-matrix approximation error vs. the $\mathcal{H}$-matrix rank k of a Matérn function with parameters $\ell = \{0.25, 0.75\}$, $\nu = 1.5$, $x_i, x_j \in [0,1]^2$, $n = 16{,}641$, see more in [19]

k	$\Vert W - \widetilde{W}\Vert_2$		$\Vert I - \widetilde{W}^{-1}W\Vert_2$	
	$\ell = 0.25$	$\ell = 0.75$	$\ell = 0.25$	$\ell = 0.75$
20	5.3e-7	2e-7	4.5	72
30	1.3e-9	5e-10	4.8e-3	20
40	1.5e-11	8e-12	7.4e-6	0.5
50	2.0e-13	1.5e-13	1.5e-7	0.1

Table 2 shows the computational time and storage for the $\mathcal{H}$-matrix approximations [19, 20]. These computations are done with the parallel $\mathcal{H}$-matrix toolbox, HLIBpro. The number of computing cores is 40, the RAM memory 128GB. It is important to note that the computing time (columns 2 and 5) and the storage cost (columns 3 and 6) are growing nearly linearly with n. Additionally, we provide the accuracy of the $\mathcal{H}$-Cholesky inverse.

Table 2: Computing times and storage costs of $\widetilde{W} \in \mathbf{R}^{n\times n}$. Accuracy in each sub-block is $\varepsilon = 10^{-7}$.

n	$\widetilde{W}$			$\widetilde{U}\widetilde{U}^\top$		
	time sec	size GB	kB/n	time sec	size GB	$\Vert I - (\widetilde{U}\widetilde{U}^\top)^{-1}W\Vert_2$
128,000	7.7	1.16	9.5	36.7	1.31	$3.8 \cdot 10^{-5}$
256,000	13	2.55	10.5	64.0	2.96	$7.1 \cdot 10^{-5}$
512,000	23	4.74	9.7	128	5.80	$7.1 \cdot 10^{-4}$
1,000,000	53	11.26	11.0	361	13.91	$3.0 \cdot 10^{-4}$
2,000,000	124	23.65	12.4	1001	29.61	$5.2 \cdot 10^{-4}$

3.1 $\mathcal{H}$-matrix approximation of regularized graph Laplacian

We rewrite formulas from Sections 2.1 - 2.3 in the $\mathcal{H}$-matrix format. Let $\tilde{W}$ be an $\mathcal{H}$-matrix approximation of W. The new optimization problem will be:

find $\tilde{f}^*$ such that $\tilde{f}^* = \arg\min_{f\in\mathbf{R}^n} \tilde{Q}(f)$, where

$$\tilde{Q}(f) := \frac{1}{2}\left(\sum_{x_i\in X_1}(f_i - y_i)^2 + \alpha\sum_{x_i,x_j\in\mathbf{X}}\tilde{w}_{ij}(f_i-f_j)^2 + \beta\|f\|^2\right). \tag{11}$$

Using (2) and assuming that the $\mathcal{H}$-matrix approximation error $\|\tilde{L} - L\| \leq \varepsilon$, obtain

$$\|\tilde{Q}(f) - Q(f)\| \leq \alpha\left(f^\top\tilde{L}f - f^\top Lf\right) \leq \alpha\|f\|^2\|\tilde{L}-L\| = \|f\|^2\varepsilon. \tag{12}$$

Let the approximate graph Laplacian be denoted by $\tilde{L} = \tilde{D} - \tilde{W}$ where $\tilde{D}$ be a diagonal matrix defined by $\tilde{D}_{ii} = \sum_j \tilde{w}_{ij}$. Differentiating $\tilde{Q}(f)$ with respect to f, we get

$$\frac{\partial\tilde{Q}}{\partial f}\mid_{f=\tilde{f}^*} = G\tilde{f}^* + \alpha\tilde{L}\tilde{f}^* - Y_{1,0} = 0,$$

hence

$$\tilde{f}^* = (G+\alpha\tilde{L})^{-1}\, Y_{1,0} \tag{13}$$

The impact of the $\mathcal{H}$-matrix approximation error could be measured as follows

$$\|\tilde{f}^* - f^*\| \leq \|(G+\alpha\tilde{L})^{-1} - (G+\alpha L)^{-1}\|\cdot\|Y_{1,0}\| \tag{14}$$

or

$$\|\tilde{f}^* - f^*\| \leq \|(I+\alpha G^{-1}\tilde{L})^{-1} - (I+\alpha G^{-1}L)^{-1}\|\|G\|\cdot\|Y_{1,0}\| \tag{15}$$

Now, if matrix norm (e.g., spectral norm) of $\alpha G^{-1}\tilde{L}$ is smaller than 1, we can write

$$(I+\alpha G^{-1}\tilde{L})^{-1} = I - \alpha G^{-1}\tilde{L} + \alpha^2 G^{-2}\tilde{L}^2 - \alpha^3 G^{-3}\tilde{L}^3 + \ldots \tag{16}$$

and

$$\begin{aligned}&\|(I+\alpha G^{-1}\tilde{L})^{-1} - (I+\alpha G^{-1}L)^{-1}\|\\ &\leq \alpha\|G^{-1}(\tilde{L}-L)\| + \alpha\|G^{-2}(\tilde{L}^2-L^2)\| + \alpha^2\|G^{-3}(\tilde{L}^3-L^3)\| + \ldots\end{aligned}$$

In general, the assumption $\|W - \tilde{W}\| \leq \varepsilon$ is not sufficient to say something about the error $\|(W^{-1} - \tilde{W}^{-1}\|$ because the later is proportional to the condition number of $\tilde{W}$, which could be very large. The reason for a large condition number is that the smallest eigenvalue could lie very close to zero. In this case some regularization may help (e.g., adding a positive number to all diagonal elements, similar to Tikhonov regularization). In this sense, the diagonal matrix G helps to bound the error $\|(G + \alpha\tilde{L})^{-1} - (G + \alpha L)^{-1}\|$. We remind that by one of the properties of the graph Laplacian states $\det(L) = 0$ and L is not invertible. Assume now that instead of Eq. 5 we have an $\mathcal{H}$-matrix approximation $\tilde{H}$ of H. Then the $\mathcal{H}$-matrix approximation of the graph Laplacian will be $\tilde{L}' = \tilde{D}' - \tilde{H}$, where $\tilde{D}' = \text{diag}(\tilde{D}'_{11}, \ldots, \tilde{D}'_{nn})$, $\tilde{D}'_{ii} = \sum\limits_{j} \tilde{H}(i,j)$. It is important to notice that the computational cost of computing $\tilde{D}$ is $\mathcal{O}(kn\log n)$, $k \ll n$.

Substituting $\tilde{L}'$ in (13), we obtain cluster ensemble based predictions of output feature in semi-supervised regression:

$$\tilde{f}^{**} = (G + \alpha\tilde{L}')^{-1}\, Y_{1,0}. \tag{17}$$

Here we cannot apply the Woodbury formula, but we also do not need it since the computational cost of computing $(G + \alpha\tilde{L}')^{-1}$ in the $\mathcal{H}$-matrix format is just $\mathcal{O}(k^2 n \log^2 n)$.

The SSR-LRCM Algorithm requires only minor changes, namely, in the second step we compute an $\mathcal{H}$-matrix representation of the graph Laplacian and on the third step calculate predictions of target feature according to (17). The total computational complexity is log-linear.

4 Numerical experiments

In this section we describe numerical experiments with the proposed SSR-LRCM algorithm. The aim of experiments is to confirm the usefulness of involving cluster ensemble for similarity matrix estimation in semi-supervised regression. We experimentally evaluate the regression quality on a synthetic and a real-life example.

4.1 First example with two clusters and artificial noisy data

In the first example we consider datasets generated from a mixture of two multidimensional normal distributions $\mathcal{N}(a_1, \sigma_X I)$, $\mathcal{N}(a_2, \sigma_X I)$ under equal weights; a_1, $a_2 \in \mathbf{R}^d$, $d = 8$, σ_X is a parameter. Usually such type of data is applied for a classifier evaluation; however it is possible to introduce a real valued attribute Y as a predicted feature and use it in regression analysis. Let Y equal $1+\varepsilon$ for points generated from the first distribution component, otherwise $Y = 2 + \varepsilon$, where ε is a Gaussian random value with zero mean and variance σ_ε^2. To study the robustness of the algorithm, we also generate two independent random variables following uniform distribution $\mathcal{U}(0, \sigma_X)$ and use them as additional "noisy" features.

In Monte Carlo modeling, we repeatedly generate samples of size n according to the given distribution mixture. In the experiment, 10% of the points selected at random from each component compose the labeled sample; the remaining ones are included in the unlabeled part. To study the behavior of the algorithm in the presence of noise, we also vary parameter σ_ε for the target feature.

In SSR-LRCM, we use K-means as a base clustering algorithm. The ensemble variants are designed by random initialization of centroids (number of clusters equals two). The ensemble size is $r = 10$. The wights of ensemble elements are the same: $w_l \equiv 1/r$. The regularization parameters α, β have been estimated using grid search and cross-validation technique. In our experiments, the best results have been obtained for $\alpha = 1$, $\beta = 0.001$, and $\sigma_X = 5$.

For the comparison purposes, we consider the method (denoted as SSS-RBF) which uses

Table 3: Results of experiments with a mixture of two distributions. Significantly different RMSE values (p-value $< 10^{-5}$) are in bold. For $n = 10^5$ and $n = 10^6$, SSR-RBF failed due to unacceptable memory demands.

n	σ_ε	SSR-LRCM			SSR-RBF	
		RMSE	t_{ens} (sec)	t_{matr} (sec)	RMSE	time (sec)
1000	0.01	**0.052**	0.06	0.02	**0.085**	0.10
	0.1	**0.054**	0.04	0.04	**0.085**	0.07
	0.25	**0.060**	0.04	0.04	**0.102**	0.07
3000	0.01	**0.049**	0.06	0.02	**0.145**	0.74
	0.1	**0.051**	0.06	0.02	**0.143**	0.75
	0.25	**0.053**	0.07	0.02	**0.150**	0.79
7000	0.01	**0.050**	0.16	0.08	**0.228**	5.70
	0.1	**0.050**	0.16	0.08	**0.229**	5.63
	0.25	**0.051**	0.14	0.07	**0.227**	5.66
10^5	0.01	0.051	1.51	0.50	-	-
10^6	0.01	0.051	17.7	6.68	-	-

the standard similarity matrix evaluated with RBF kernel. Different values of parameter ℓ were considered and the quasi-optimal $\ell = 4.47$ was taken. The output predictions are calculated according to formula (4).

The quality of prediction is estimated as Root Mean Squared Error: RMSE $= \sqrt{\frac{1}{n}\sum(y_i^{\text{true}} - f_i)^2}$, where y_i^{true} is a true value of response feature specified by the correspondent component. To make the results more statistically sound, we have averaged error estimates over 40 Monte Carlo repetitions and compare the results by paired two sample Student's t-test.

Table 3 presents the results of experiments. In addition to averaged errors, the table shows averaged execution times for the algorithms (working on dual-core Intel Core i5 processor with a clock frequency of 2.8 GHz and 4 GB RAM). For SSR-LRCM, we separately indicate ensemble generation time t_{ens} and law-rank matrix operation time t_{matr} (in seconds). The obtained p-values for Student's t-test are also taken into account. A p-value less than the given significance level (e.g., 0.05) indicates a statistically significant difference between the performance estimates.

The results show that the proposed SSR-LRCM algorithm has significantly smaller prediction error than SSR-RBF. At the same time, SSR-LRCM has run much faster, especially for medium sample size. For a large volume of data ($n = 10^5$, $n = 10^6$) only SSR-LRCM has been able to find a solution, whereas SSR-RBF has refused to work due to unacceptable memory demands (74.5GB and 7450.6GB correspondingly).

4.2 Second example with 10-dimensional real Forest Fires dataset

In the second example, we consider Forest Fires dataset [10]. It is necessary to predict the burned area of forest fires, in the northeast region of Portugal, by using meteorological and other information. Fire Weather Index (FWI) System is applied to get feature values. FWI System is based on consecutive daily observations of temperature, relative humidity, wind speed, and 24-hour rainfall. We use the following numerical features:

- X-axis spatial coordinate within the Montesinho park map;

- Y-axis spatial coordinate within the Montesinho park map;
- Fine Fuel Moisture Code;
- Duff Moisture Code;
- Initial Spread Index;
- Drought Code;
- temperature in Celsius degrees;
- relative humidity;
- wind speed in km/h;
- outside rain in mm/m2;
- the burned area of the forest in ha (predicted feature).

This problem is known as a difficult regression task [11], in which the best RMSE was attained by the naive mean predictor. We use quantile regression approach: the transformed quartile value of response feature should be predicted.

The following experiment's settings are used. The volume of labeled sample is 10% of overall data; the cluster ensemble architecture is the same as in the previous example. K-means base algorithm with 10 clusters with ensemble size $r = 10$ is used. Other parameters are $\alpha = 1$, $\beta = 0.001$, the SSR-RBF parameter is $\ell = 0.1$. The number of generations of the labeled samples is 40.

As a result of modeling, the averaged error rate for SSR-LRCM has been evaluated as RMSE$= 1.65$. For SSR-RBF, the averaged RMSE is equal to 1.68. The p-value which equals 0.001 can be interpreted as indicating the statistically significant difference between the quality estimates.

Conclusion

In this work, we solved the regression problem to forecast the unknown value Y. For this we have introduced a semi-supervised regression method SSR-LRCM based on cluster ensemble and low-rank co-association matrix decomposition. We used a scheme of a single clustering algorithm which obtains base partitions with random initialization.

The proposed method combines graph Laplacian regularization and cluster ensemble methodologies. Low-rank or hierarchical decomposition of the co-association matrix gives us a possibility to speedup calculations and save memory from cubic to (log-)linear.

There are a number of arguments for the usefulness of ensemble clustering methodology. The preliminary ensemble clustering allows one to restore more accurately metric relations between objects under noise distortions and the existence of complex data structures. The obtained similarity matrix depends on the outputs of clustering algorithms and is less noise-addicted than the conventional similarity matrices (eg., based on Euclidean distance). Clustering with a sufficiently large number of clusters can be viewed as Learning Vector Quantization known for lowering the average distortion in data.

The efficiency of the suggested SSR-LRCM algorithm was confirmed experimentally. Monte Carlo experiments have demonstrated statistically significant improvement of regression quality and decreasing in running time for SSR-LRCM in comparison with analogous SSR-RBF algorithm based on standard similarity matrix.

In future works, we plan to continue studying theoretical properties and performance characteristics of the proposed method. Development of iterative methods for graph Laplacian regularization is another interesting direction, especially in large-scale machine learning problems. We will further research theoretical and numerical properties of the $\mathcal{H}$-matrix approximation of W and H. Applications of the method in various fields are also planned, especially for spacial data processing and analysis of genetic sequences.

Acknowledgements

The work was carried on according to the scientific research program "Mathematical methods of pattern recognition and prediction" in The Sobolev Institute of Mathematics SB RAS. The research was partly supported by RFBR grants 18-07-00600, 18-29-09041mk, 19-29-01175 and partly by the Russian Ministry of Science and Education under the 5-100 Excellence Programme. A. Litvinenko was supported by funding from the Alexander von Humboldt Foundation.

REFERENCES

[1] Belkin M., Niyogi P., Sindhwani V. Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. J. Mach. Learn. Res. Vol. 7, no. Nov. 2399-2434 (2006)

[2] Berikov V., Karaev N., Tewari A. Semi-supervised classification with cluster ensemble. In Engineering, Computer and Information Sciences (SIBIRCON), 2017 International Multi-Conference. 245–250. IEEE. (2017)

[3] Berikov V.B. Construction of an optimal collective decision in cluster analysis on the basis of an averaged co-association matrix and cluster validity indices. Pattern Recognition and Image Analysis. 27(2), 153–165 (2017)

[4] Berikov V.B., Litvinenko A., The influence of prior knowledge on the expected performance of a classifier. Pattern recognition letters 24 (15), 2537-2548, (2003)

[5] Berikov V.B., Litvinenko A., Methods for statistical data analysis with decision trees. Novosibirsk, Sobolev Institute of Mathematics, http://www.math.nsc.ru/AP/datamine/eng/context.pdf, (2003)

[6] Bernholdt, D.E., Ciancosa, M.R., Green, D.L., Law, K.J.H., Litvinenko, A. and Park, J.M., Comparing theory based and higher-order reduced models for fusion simulation data, J. Big Data and Information Analytics, 2(3), 41-53, (2018)

[7] Boongoen T., Iam-On N. Cluster ensembles: A survey of approaches with recent extensions and applications. Computer Science Review. 28, 1-25 (2018)

[8] Camps-Valls G., Marsheva T., Zhou D. Semi-supervised graph-based hyperspectral image classification. IEEE Transactions on Geoscience and Remote Sensing. 45(10), 3044–3054 (2007)

[9] https://www.mathworks.com/matlabcentral/fileexchange/41459-6-functions-for-generating-artificial-datasets classification

[10] https://archive.ics.uci.edu/ml/datasets/forest+fires

[11] Cortez P., Morais A. A Data Mining Approach to Predict Forest Fires using Meteorological Data. In J. Neves, M. F. Santos and J. Machado Eds., New Trends in Artificial Intelligence, Proceedings of the 13th EPIA 2007 - Portuguese Conference on Artificial Intelligence, Guimaraes, Portugal, 512–523 (2007)

[12] Doquire G., Verleysen M. A graph Laplacian based approach to semi-supervised feature selection for regression problems. Neurocomputing. Vol. 121, 5-13 (2013)

[13] Fred A., Jain A. Combining multiple clusterings using evidence accumulation. IEEE Transaction on Pattern Analysis and Machine Intelligence. 27, 835–850 (2005)

[14] Grasedyck L. and W. Hackbusch W. Construction and arithmetics of $\mathcal{H}$-matrices. *Computing*, 70(4):295–334, (2003)

[15] Hackbusch W. A sparse matrix arithmetic based on $\mathcal{H}$-matrices. I. Introduction to $\mathcal{H}$-matrices. *Computing*, 62(2):89–108, (1999)

[16] Hackbusch W. *Hierarchical matrices: Algorithms and Analysis*, volume 49 of *Springer Series in Comp. Math.* Springer, (2015)

[17] Khoromskij B.N., Litvinenko A., and Matthies H.G. Application of hierarchical matrices for computing the Karhunen–Loève expansion. *Computing*, 84(1-2):49–67, (2009)

[18] Kostopoulos, Georgios, et al. Semi-supervised regression: A recent review. Journal of Intelligent & Fuzzy Systems. Preprint, 1–18 (2018)

[19] Litvinenko A., Sun Y., Genton M. G., and Keyes D. Likelihood Approximation With Hierarchical Matrices For Large Spatial Datasets. *ArXiv preprint, http://arxiv.org/abs/1709.04419*, (2017)

[20] Litvinenko A. HLIBCov: Parallel Hierarchical Matrix Approximation of Large Covariance Matrices and Likelihoods with Applications in Parameter Identification. *ArXiv preprint, http://arxiv.org/abs/1709.08625*, submitted to Elsevier MethodsX Journal, (2017)

[21] Litvinenko A., Keyes D., Khoromskaia V., Khoromskij B.N., and Matthies H. G. Tucker Tensor analysis of Matérn functions in spatial statistics. *Computational Methods in Applied Mathematics*, (2018) DOI: `https://doi.org/10.1515/cmam-2018-0022`.

[22] Matérn B. *Spatial Variation*, volume 36 of *Lecture Notes in Statistics*. Springer-Verlag, Berlin; New York, second edition edition, (1986)

[23] Nowak W., Litvinenko A., Kriging and Spatial Design Accelerated by Orders of Magnitude: Combining Low-Rank Covariance Approximations with FFT-Techniques. *Mathematical Geosciences*, 45(1):411–435, (2013)

[24] Tikhonov A.N., Goncharsky A., Stepanov V.V., Yagola A.G. Numerical methods for the solution of ill-posed problems (Vol. 328). Springer Science & Business Media (2013)

[25] Yu G. X., Feng L., Yao G. J., Wang, J. Semi-supervised classification using multiple clusterings. Pattern Recognition and Image Analysis. 26(4), 681–68 (2016)

[26] Wang M., Hua X., Song Y., Dai L., Zhang H. Semi-Supervised Kernel Regression. In Sixth International Conference on Data Mining (ICDM06) 1130 -1135 (2006)

[27] Wu M., Scholkopf B. Transductive Classification via Local Learning Regularization. Artificial Intelligence and Statistics. 628-635. (2007)

[28] Zhao M., Chow T. W., Wu Z., Zhang Z., Li B. Learning from normalized local and global discriminative information for semi-supervised regression and dimensionality reduction. Information Sciences. 324, 286-309 (2015)

[29] Zhou D., Bousquet O., Lal T., Weston J., Scholkopf B. Learning with local and global consistency. In Advances in Neural Information Processing Systems. 16, 321-328 (2003)

[30] Zhou Z.-H., Li M. Semi-supervised regression with co-training. Proceedings of the 19th international joint conference on Artificial intelligence. Morgan Kaufmann Publishers Inc. 908-913 (2005)

[31] Zhu X. Semi-supervised learning literature survey. Tech. Rep. Department of Computer Science, Univ. of Wisconsin, Madison. N. 1530 (2008)

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

UNCERTAINTY QUANTIFICATION IN A TWO-DIMENSIONAL RIVER HYDRAULIC MODEL

Siham EL GARROUSSI[1], Matthias DE LOZZO[2], Sophie RICCI[1], Didier LUCOR[3], Nicole GOUTAL[4,5], Cédric GOEURY[4], Sébastien BOYAVAL[5]

[1]CECI, CERFACS/CNRS
42 Av Gaspard Coriolis 31057, Toulouse, Cedex 1
e-mail: {garroussi, ricci}@cerfacs.fr

[2] IRT Saint Exupéry
CS34436, 3 Rue Tarfaya, 31400 Toulouse
e-mail: matthias.delozzo@irt-saintexupery.com

[3] LIMSI
Campus Universitaire bâtiment 507, Rue John Von Neumann, 91400 Orsay
e-mail: didier.lucor@limsi.fr

[4] EDF, LNHE
6 quai Watier, 78400 Chatou
e-mail: nicole.goutal, cedric.goeury@edf.fr

[5] LHSV
6 quai Watier, 78400 Chatou
e-mail: sebastien.boyaval@enpc.fr

Keywords: Open-channel flow, Sensitivity analysis, Surrogate model, Gaussian process, Monte Carlo method.

Abstract. *River hydraulic models are used to assess the environmental risk associated to flooding and consequently inform decision support systems for civil security needs. These numerical models are generally based on a deterministic approach based on resolving the partial differential equations. However, these models are subject to various types of uncertainties in their input. Knowledge of the type and magnitude of these uncertainties is crucial for a meaningful interpretation of the model results. Uncertainty quantification (UQ) framework aims to probabilize the uncertainties in the input, propagate them through the numerical model and quantify their impact on the simulated quantity of interest, here, water level field discretized over an unstructured finite element mesh over the Garonne River (South-West France) between Tonneins and La Réole simulated with a numerical solver, TELEMAC-2D. The computational cost of the sensitivity analysis with the classical Monte Carlo approach is reduced using a surrogate model instead of the numerical solver. The present study investigates one of the machine learning algorithms: A surrogate model based on Gaussian process. This latter was used to represent* *the spatially distributed water level with respect to uncertain stationary flow to the model and friction coefficients. The quality of the surrogate was assessed on a validation set, with small root mean square error and a predictive coefficient equal to 1. Sobol' sensitivity indices are computed and enhance the high impact of the input discharge on the water level variation.*

1 INTRODUCTION

Flood inundation models are central components in any flood risk analysis system as they transform the bulk discharge outputs from flood-frequency analyses or rainfall-runoff models into distributed predictions of flood hazard in terms of water depth, inundation extent and flow velocity. Predictions may be dynamic in time and can be derived from a range of codes which vary in complexity from non-model approaches, such as fitting a planar surface to digital elevation data, through to numerical solutions of fluid dynamics equations derived from considerations of mass and momentum conservation.

Whilst such models are parsimonious in terms of their data requirements and number of unconstrained parameters relative to other environmental physics, their underlying equations may be non-linear. Moreover, the data sets that they do require may be subject to complex, but poorly known errors that may vary markedly in time and space. As a consequence, considerable research has, in recent years, sought to understand and better estimate these uncertainties in order to improve flood risk analysis.

Typically, uncertainties in hydrodynamics models stand are classified as: parametric (or epistemic) uncertainty, arising from incomplete knowledge of the correct settings of the models parameters; input data uncertainty, arising from incomplete knowledge of the true value of the initial state and forcing, usually linked to the aleatory nature of the physics; and structural uncertainty, which is the failure of the model to represent the system, even if the correct parameters and inputs are known. Together, these three components represent a complete probabilistic description of the informativeness of the model for the underlying system. But in practice, all are extremely challenging to specify.

In this study, we consider both epistemic and aleatory uncertainties by investigating the effect of two uncertainty sources on water level calculation for extreme flood event, respectively the roughness coefficient and the upstream discharge.

On the one hand, the estimation of the roughness is difficult because it is a lumped parameter that mostly reflects the flow resistance of the river. Since the roughness coefficient has an extensive effect on flow analysis of a river, including computation of the water level and velocity, its accurate estimation is important for prediction of the water level during flooding. Because of its importance, various efforts have been made to quantify the roughness coefficients of rivers in an objective manner. Among them, an element-based method [9] and empirical equations that relate the roughness coefficient either to bed material [32] or to relative depth [5] are representative. However, owing to the diversity and irregularity of natural rivers, prediction of the roughness coefficient for a specific river reach using these methods is not simple. Thus, until now, field measurements have been made either to directly estimate the roughness coefficient [7] or to provide references [2, 18]. However, there remain uncertainties whether using the methods referred to above or using field measurements. From a practical viewpoint, water level and discharge as variables computed by numerical modeling are influenced by uncertainty in estimating the roughness coefficient. Conducting simulation of dam breakage flow for the Teton Dam, [13] showed that variation in calculated flood flow water depth was less than 5% with a 20% change in the roughness coefficient. He therefore argued that even if uncertainty in Stricklers roughness coefficient is large, its effect on the water depth might be reduced considerably in the process of computation. These conclusions should be deeply investigated in the context of flood simulation, on our own river test case, characterized by long homogeneous friction zones calibrated in high flow.

On the other hand, inundation models require the specification of boundary conditions, which are typically the greatest source of aleatory and epistemic uncertainty when simulating the annual exceeding probability of inundation. Flow at the upstream boundary of the river is often the most important boundary condition, although most applications will require (unless using a kinematic solution) or benefit from downstream-level boundaries (e.g. tidal reaches). In locations where they are available, gauging stations are typically the most accurate source of river flow and level data. However, the ratings at these stations, that convert observed levels to flows, are usually based on low to medium flow observations, necessitating an uncertain extrapolation of the rating for high flows. Rating errors may be especially large when flow is out of bank. Where gauging stations are not available or spatially sparse, hydrological models can be used to simulate upstream discharges. However, despite much effort, rainfall-runoff models are still very uncertain, especially where calibration/validation data are lacking.

Subsequently, once the sources of uncertainties have been identified, they must be propagated in the model. The Monte Carlo (MC) methods are the most common techniques used for uncertainty propagation (UQ) [15]. This framework allows to estimate standard statistics on the model output, e.g. expectation, standard deviation, quantiles or probabilities of exceeding a given threshold. It also makes it possible to estimate sensitivity indices representing the shares of output uncertainty attributable to the different uncertain input parameters, e.g. Sobol' indices where output uncertainty is measured in terms of variance [30]. MC is simple and highly adapted to massively parallel computational resources. Yet, its convergence is slow as it scales inversely to the square root of the sample size and its cost gets prohibitive for expensive models. In this respect, surrogate models such as Gaussian process model (GP) have received tremendous attention in the last few years, as it allows one to replace the original expensive model by a surrogate which is built from an experimental design of limited size [25]. Then the surrogate can be used to compute the UQ study in negligible time. In particular, [27] have shown that, for a 1D hydraulic model, on the Garonne river section that we consider and stationary flow, it features similar performance to estimate statistics by Monte-Carlo random sampling when friction and input forcing uncertainties are taken into account. The accuracy of the water level correlation matrix and sensitivity Sobol' indices estimated with the GP surrogate was assessed with respect to a classical MC estimate based on a large data set. This article is a reference for us because it involves the same section of Garonne river, the same types of uncertain variables (friction and upstream flow) and the same family of surrogate model as those considered in our work. Our work can be seen as an extension to two-dimensional flow modelling and floodplain characterization.

The present study extends the surrogate model approach in hydraulics to 2D modeling taking into account the dynamics of the flood plain. Section 2 presents the hydrodynamics solver TELEMAC-2D, the Garonne test case used in this article and the associated uncertainties. Section 3 presents the GP surrogate strategy based on the reduction of the dimension of the output space with a Proper Orthogonal Decomposition (POD). This section also presents the metrics used to assess the quality of the surrogate along with the sensitivity indices based on output variance decomposition. Results are presented in Section 4. Conclusions and perspectives are finally given in Section 5.

2 MODEL: TWO-DIMENSIONAL FLOW OF THE GARONNE RIVER UNDER UNCERTAINTY

2.1 Physical model

The Shallow Water Equations (SWE), also called depth-averaged free surface flow equations, are commonly used in environmental hydrodynamics modelling [12]. They are derived from the Navier-Stokes equations [31] and express mass and momentum conservation averaged in the vertical dimension. The non-conservative form of the equations are written in terms of the water depth (h) and the horizontal components of velocity (u and v):

$$\frac{\partial h}{\partial t} + \mathrm{div}\,(hu) = 0 \tag{1}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -g\frac{\partial H}{\partial x} + F_x + \frac{1}{h}\mathrm{div}\left(h\nu_e\overrightarrow{\mathrm{grad}}\,(u)\right) \tag{2}$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -g\frac{\partial H}{\partial y} + F_y + \frac{1}{h}\mathrm{div}\left(h\nu_e\overrightarrow{\mathrm{grad}}\,(v)\right) \tag{3}$$

where:

$$\begin{cases} F_x = -\dfrac{g}{{K_s}^2}\dfrac{u\sqrt{u^2+v^2}}{h^{\frac{4}{3}}} - \dfrac{1}{\rho_w}\dfrac{\partial P_{atm}}{\partial x} + \dfrac{1}{h}\dfrac{\rho_{air}}{\rho_w}C_{DZ}U_{w,x}\sqrt{U_{w,x}^2 + U_{w,y}^2} \\ F_y = -\dfrac{g}{{K_s}^2}\dfrac{v\sqrt{u^2+v^2}}{h^{\frac{4}{3}}} - \dfrac{1}{\rho_w}\dfrac{\partial P_{atm}}{\partial y} + \dfrac{1}{h}\dfrac{\rho_{air}}{\rho_w}C_{DZ}U_{w,y}\sqrt{U_{w,x}^2 + U_{w,y}^2} \end{cases}$$

and: ρ_w/ρ_{air} [kg.m^{-3}] is the water/air density, P_{atm} [Pa] is the atmospheric pressure, $U_{w,x}$ and $U_{w,y}$ [m.s^{-1}] are the horizontal wind velocity components, C_{DZ} [-] is the wind influence coefficient, K_s [m$^{\frac{1}{3}}$.s^{-1}] is the river bed and floodplain friction coefficient, using the Strickler formulation [**?**]. F_x and F_y [m.s^{-2}] are the horizontal components of external forces (friction, wind and atmospheric forces), *h* [m] is the water depth, *H* [m] is the water level ($h = H - z_f$ if z_f [m] is the bottom level), *u* and *v* [m.s^{-1}] are the horizontal components of velocity and ν_e [m^2.s^{-1}] is the water diffusion coefficient. div and $\overrightarrow{grad}$ are respectively the divergence and gradient operators.

To solve the system of equations (Eq. (1) to Eq. (3)), initial conditions $h(x, y, t = 0) = h_0(x, y)$; $u(x, y, t = 0) = u_0(x, y)$; $v(x, y, t = 0) = v_0(x, y)$ are provided along with boundary conditions (BC) at surface, at bottom and at upstream and downstream frontiers $h(x_{BC}, y_{BC}, t) = h_{BC}(t)$.

2.2 Study area

The study area extends over a 50 km reach of the Garonne river (France) between Tonneins (upstream), downstream of the confluence with the river Lot, and La Réole (downstream) (see Figure 1). This part of the valley was equipped in the 19th century with infrastructure to protect the Garonne flood plain from flooding such as that occurred in 1875. A system of longitudinal dykes and weirs was progressively constructed after that flood event to protect the floodplains, organize submersion and flood retention areas. Protections on the Garonne river form a system of successive storage areas for the flood plain beyond the dikes.

2.3 Uncertainty characterization

The hydraulic variables are discretized on an unstructured triangular mesh over the two-dimensional study area. We note $\mathbf{h}$ the vector of the water level over the $p = 41416$ nodes of

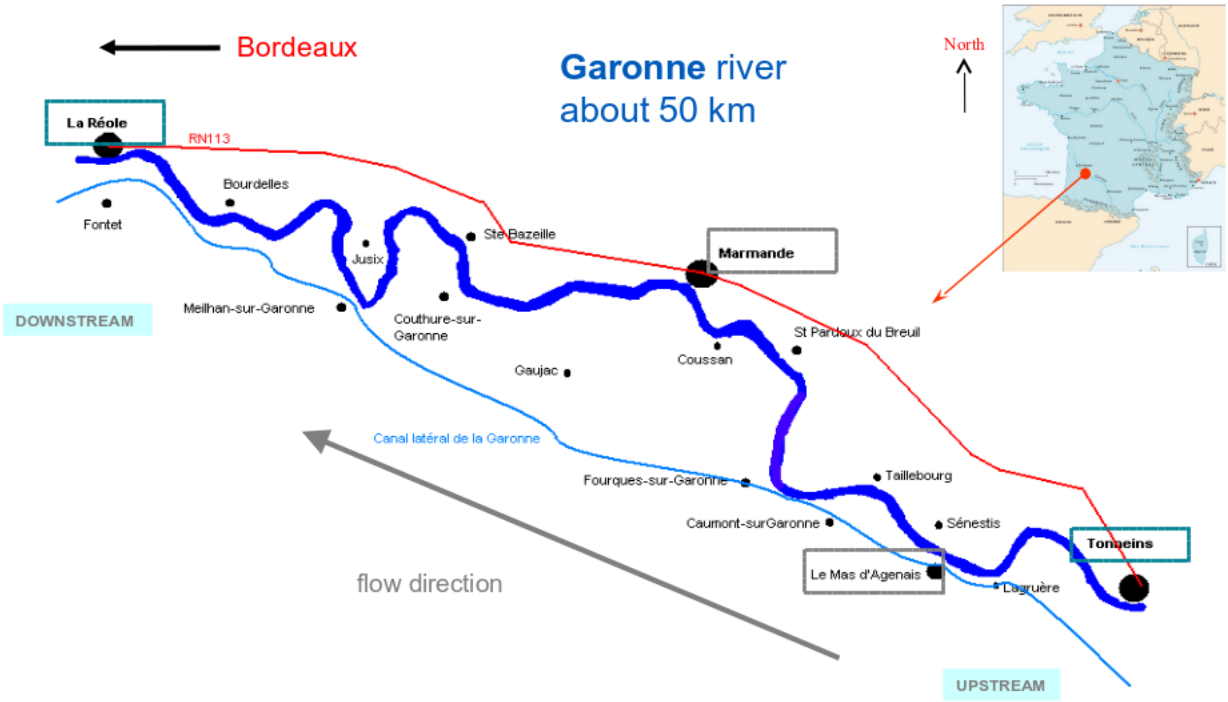


Figure 1: Study area of the Garonne river

the mesh. It represents our quantity of interest (QoI). In this study, the impact of roughness and upstream flow on the discretized water level h is quantified in the context of extreme flood event:

- The roughness coefficient defined according to 4 areas. Indeed, T2D was calibrated for high flow in [4] using steady-state water surface profiles at high discharge, from bank-full discharge in the main channel (2 500 $\mathrm{m}^3.\mathrm{s}^{-1}$) to bank-full discharge in the overbank flow channel between dykes. In the floodplains, the roughness coefficient $K_{s,1}$ is selected as an area with cultivated fields all around the river with a Strickler coefficient of 17 $\mathrm{m}^{1/3}.\mathrm{s}^{-1}$. Classically, according to the available expert knowledge, the friction coefficient is contained in an interval bounded by physical values depending on the roughness of soil material.

  For the main channel, the Strickler roughness coefficient was split into three different areas:

  - from Tonneins to upstream of Mas d'Agenais, $K_{s,2}$: 45 $\mathrm{m}^{1/3}.\mathrm{s}^{-1}$,
  - from upstream of Mas d'Agenais to upstream of Marmande, $K_{s,3}$: 38 $\mathrm{m}^{1/3}.\mathrm{s}^{-1}$,
  - from upstream of Marmande to La Réole, $K_{s,4}$: 40 $\mathrm{m}^{1/3}.\mathrm{s}^{-1}$.

  The distribution of Strickler roughness coefficient is chosen uniform and the interval is set to cover the range of calibration values.

- The upstream discharge is assumed to follow a Gaussian distribution centered around the thousand return period 8 490 $\mathrm{m}^3.\mathrm{s}^{-1}$with a standard deviation of 700 $\mathrm{m}^3.\mathrm{s}^{-1}$. The study is thus focused on extreme flood events that activate the flood plains.

Tab. 1 summarizes the considered input uncertainties.

Variable	Distribution	Units
Q	$\mathcal{U}[2\,500,\ 10\,000]$	$\mathrm{m}^3.\mathrm{s}^{-1}$
$K_{s,1}$	$\mathcal{U}[5,\ 20]$	$\mathrm{m}^{1/3}.\mathrm{s}^{-1}$
$K_{s,2}$	$\mathcal{U}[40,\ 50]$	$\mathrm{m}^{1/3}.\mathrm{s}^{-1}$
$K_{s,3}$	$\mathcal{U}[33,\ 43]$	$\mathrm{m}^{1/3}.\mathrm{s}^{-1}$
$K_{s,4}$	$\mathcal{U}[35,\ 45]$	$\mathrm{m}^{1/3}.\mathrm{s}^{-1}$

Table 1: Distributions of the input variable uncertainties.

2.4 Computing environment

In this work, hydrodynamic is provided using TELEMAC-2D (T2D) depth-averaged hydrodynamic model[1]. It solves the SWE in two dimensions with an explicit first-order time integration scheme, a finite element scheme and an iterative conjugate gradient method. In each point of the mesh, T2D gives the water depth and the vertically average horizontal velocity field [17]. T2D was developed initially by the National Hydraulics and Environment Laboratory (LNHE) of the Research and Development Directorate of EDF, and is now managed by a consortium. The software comes with an API to modify the values of the uncertain parameters in a non-intrusive way.

The surrogate model construction as well as the sensitivity analysis was carried out using the BATMAN-Open-TURNS (BATMAN-OT) library[2]. This library (developed at CERFACS and CECILL-B licensed) provides a convenient, modular and efficient framework for design of experiments, surrogate model and uncertainty quantification [26]. It relies on open source python packages dedicated to statistics (openTURNS[3] [3] and scikit-learn[4] [23]). It also implements advanced methods for resampling, robust optimization and uncertainty visualization.

In terms of infrastructures, CERFACS's cluster, Nemo, has been used to run T2D simulations. The Nemo cluster includes 6,912 cores distributed in 288 compute nodes. The ECU power peak is 277 Tflop/s. On this architecture, simulating the river and flood plain dynamics for the test case presented in Sect. 2.2 over 3 days, takes about 6 minutes on 15 cores.

3 AN EFFICIENT UQ FRAMEWORK FOR COSTLY TWO-DIMENSIONAL SIMULATOR

The Monte Carlo (MC) framework is the most common framework used for uncertainty quantification, due to its simplicity and good statistical results. It is theoretically applicable whatever the complexity of the deterministic model or the desired statistical estimator. However, the required sample size increases squarely with the estimator accuracy and makes this approach rather impracticable when the computational cost of each run of the model, like T2D, is non negligible. One way to lower the computationally demanding is to replace, on one side, the T2D model by a surrogate model [11], on the other side, the pure random sampling by alternative sampling methods such as the Latin Hypercube sampling approach [16].

[1] More information can be found on the website `www.opentelemac.org`.

[2] BATMAN-OT can be downloaded from `https://gitlab.com/cerfacs/batman`.

[3] More information on: `http://www.openturns.org`.

[4] More information on: `https://scikit-learn.org`.

3.1 Build a surrogate model with spatial output

The surrogate model based on Gaussian process regression (GP) [25] has been adopted in the following. We have chosen this metamodel to the detriment of others for two reasons. The first one is its small number of hyperparameters: about one per input parameter. The second one is that it provides a measurement of its model error which would be of interest in our future work, for optimization and data assimilation problems, based on methods as expected improvement [19]. While surrogate models offer a low cost alternative to costly models, their formulation is challenging in high dimension for inputs and outputs. In the present case, the size of the uncertain input space is small and resumes to 5 scalars. Yet, the quantity of interest is 2D and discretized over more than 41 000 points. The output space is reduced in order to limit the cost of the surrogate formulation and the spatial coherence of the later, using a Proper Orthogonal Decomposition (POD) strategy [24]. POD is a post-processing technique that takes a given set of data and extracts basis functions, that contain as much ”energy” as possible. The meaning of ”energy” depends on which kind of POD is used [8]. Here, only POD based on snapshot method [29] is considered.

We propose to build a surrogate model combining POD and GP surrogate model. So we call it “POD+GP surrogate model”. The corresponding algorithm is presented as follows:

1. build a learning dataset $\mathcal{D}_l = \left(\mathbf{x}^{(i)}, \mathbf{h}^{(i)}\right)_{1\leq i\leq N_l}$ of size N_l where the design of experiments $\left(\mathbf{x}^{(i)}\right)_{1\leq i\leq N_l}$ is a Latin hypercube sample (LHS) [21] with $\mathbf{x} = (Q, K_{s,1}, K_{s,2}, K_{s,3}, K_{s,4})$ and $\mathbf{h}^{(i)}$ is the water level computed by T2D over the mesh at $\mathbf{x}^{(i)}$;

2. decompose the sampled output vector $\mathbf{h}$ by achieving a POD on the centered output learning matrix $\mathbf{H} = \left(h_j^{(i)} - N_l^{-1}\sum_{k=1}^{N_l} h_j^{(k)}\right)_{\substack{1\leq i\leq N_l\\1\leq j\leq p}}$ and derive the most significant components; then, any sampled local water level $\mathbf{h}^{(i)}$ can be approximated by a weighted sum of these components where weights depend on the input vector value $\mathbf{x}^{(i)}$;

3. for each component, approximate the relation between its sampled coefficient and the corresponding sampled model inputs by means of a GP surrogate model;

4. formulate the POD+GP surrogate model $\hat{\mathbf{h}}(\mathbf{x})$ as the weighted sum of the more significant POD components where weights are the GP surrogate models depending on $\mathbf{x}$.

3.1.1 Reduction of the output dimension by proper orthogonal decomposition (POD)

The key idea of the snapshot method [29] is to achieve a POD of the centred snapshot matrix $\mathbf{H} = \left(h_j^{(i)} - N_l^{-1}\sum_{k=1}^{N_l} h_j^{(k)}\right)_{\substack{1\leq i\leq N_l\\1\leq j\leq p}} \in \mathbb{M}_{N_l,p}(\mathbb{R})$, which gathers the water level computed at each mesh point for the N_l snapshots, from which the sample mean is substracted.

Based on many observations of a random vector, the POD gives the orthogonal directions of largest variances (or modes) in the probabilistic vector space in order to reduce the vector space dimension [6]. Note that for simplicity purpose, the adjective *centred* is dropped in the following when referring to the centred snapshot matrix $\mathbf{H}$.

The POD of the snapshot covariance matrix $\mathbf{C} = N_l^{-1}\,\mathbf{H}^{\mathrm{T}}\,\mathbf{H} \in \mathbb{M}_p(\mathbb{R})$ is equivalent to the Singular Value Decomposition (SVD) of the snapshot matrix $\mathbf{H}$:

$$\mathbf{H} = \mathbf{U}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathrm{T}} = \sum_{k=1}^{r_p} \lambda_k\,\mathbf{u}_k\,\mathbf{v}_k^{\mathrm{T}}, \tag{4}$$

where $\mathbf{U} \in \mathbb{M}_{N_l}(\mathbb{R})$ is an orthogonal matrix diagonalizing $\mathbf{H}\mathbf{H}^{\mathrm{T}}$ ($\mathbf{u}_k$, the k^{th} column of $\mathbf{U}$, is a left singular vector of $\mathbf{H}$), where $\mathbf{V} \in \mathbb{M}_p(\mathbb{R})$ is an orthogonal matrix diagonalizing $\mathbf{H}^{\mathrm{T}}\mathbf{H}$ ($\mathbf{v}_k$, the k^{th} column of $\mathbf{V}$, is a right singular vector of $\mathbf{H}$), and where $\mathbf{\Lambda} \in \mathbb{M}_{N_l,p}(\mathbb{R})$ is a rectangular diagonal matrix including $r_p = \min(N_l, p)$ singular values on its diagonal. The singular values $\{\lambda_k\}_{1\leq k\leq r_p}$ are the square roots of the eigenvalues of $\mathbf{C}$. Note that in this study, since the size of the training set N_l is lower than the number of mesh points $p = 41\,416$, the rank of $\mathbf{H}$ is here $r_p = N_l$.

At the k^{th} mesh point, the snapshot $h_k(\mathbf{x}^{(i)})$ can then be retrieved as a linear combination of r_p modes $\{\Psi_i\}_{1\leq i\leq r_p}$:

$$h_k\left(\mathbf{x}^{(i)}\right) = \left(\mathbf{U}\,\mathbf{\Lambda}\,\mathbf{V}^{\mathrm{T}}\right)_{ki} = U_{k:}\left(\mathbf{\Lambda}\,\mathbf{V}^{T}\right)_{:i} = \sum_{j=1}^{r_p} \gamma_{k,j}\,\Psi_j\left(\mathbf{x}^{(i)}\right), \tag{5}$$

where for any $j \in \{1,\ldots,N_l\}$, $\gamma_{p,j} := U_{k,j}$ and $\Psi_j\left(\mathbf{x}^{(i)}\right) := \left(\mathbf{\Lambda}\mathbf{V}^{T}\right)_{j,i}$.

From that, we want to approximate each relation $\mathbf{x} \to \Psi_j(\mathbf{x})$ by a GP surrogate model $\Psi_{gp,j}$ from the dataset $\left(\mathbf{x}^{(i)}, \Psi_j\left(\mathbf{x}^{(i)}\right)\right)_{1\leq i\leq N_l}$ in order to propose the following POD+GP surrogate model:

$$\widehat{h}_k(\mathbf{x}) = \sum_{i=1}^{r_p} \gamma_{k,i}\,\Psi_{\text{gp},i}(\mathbf{x}), \tag{6}$$

This POD+GP surrogate model requires the construction of r_p GP surrogate models.

3.1.2 Learning of the significant POD modes by Gaussian process (GP) modelling

As stated by [25], a GP is a random process (here the mode Ψ_i) indexed over a domain (here $\mathbb{R}^d$), for which any finite collection of process values (here $\left\{\Psi_i\left(\mathbf{x}^{(j)}\right)\right\}_{1\leq j\leq N_l}$) has a joint Gaussian distribution. Concretely, let $\widetilde{\Psi}_i$ be a Gaussian random process fully described by its zero mean and its correlation π_i:

$$\widetilde{\Psi}_i(\mathbf{x}) \sim \mathrm{GP}\left(0, \sigma_i^2\,\pi_i(\mathbf{x},\mathbf{x}')\right), \tag{7}$$

with $\pi_i(\mathbf{x},\mathbf{x}') = \mathbb{E}\left[\widetilde{\Psi}_i(\mathbf{x})\widetilde{\Psi}_i(\mathbf{x}')\right]$. In our case, the correlation function π (or kernel) is chosen as a squared exponential:

$$\pi_i(\mathbf{x},\mathbf{x}') = \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{2\,\ell_i^2}\right), \tag{8}$$

where ℓ_i is a length scale describing dependencies of model output between two input vectors $\mathbf{x}$ and $\mathbf{x}'$, and where σ_i^2 is the variance of the output signal. Squared exponential kernel leads to satisfying results but other kernel functions could have been considered, such as a decreasing exponential one or a Matérn one – with their associated hyper-parameters. The choice of the kernel is still an open problem and can be mitigated using the available information on the problem. The square exponential kernel leads to very smooth, thus stable results. Furthermore, it implies that the model is exact at sample points; it does not introduce any other strong assumptions, hence its wide usage among practitioners.

Then the surrogate model of interest is the mean of the GP resulting of conditioning $\widetilde{\Psi}_i$ by the training set $\left\{\Psi_i\left(\mathbf{x}^{(k)}\right)\right\}_{1\leq k\leq N_l}$. For any $\mathbf{x}^* \in \mathbb{R}^d$,

$$\Psi_{\mathrm{gp},i}(\mathbf{x}) = \sum_{k=1}^{N} \beta_{k,i}\, \pi_i\left(\mathbf{x}, \mathbf{x}^{(k)}\right), \tag{9}$$

where $\beta_{k,i} = \left(\mathbf{\Pi}_i + \tau^2\, \mathbf{I}_{N_l}\right)^{-1} \left(\Psi_i\left(\mathbf{x}^{(1)}\right) \ldots \Psi_i\left(\mathbf{x}^{(N_l)}\right)\right)^T$ with $\mathbf{\Pi}_i = \left(\pi_i\left(\mathbf{x}^{(j)}, \mathbf{x}^{(k)}\right)\right)_{1\leq j,k\leq N_l}$, and where τ (referred to as the nugget effect) avoids ill-conditioning issues for the matrix $\mathbf{\Pi}$. The hyperparameters $\{\ell_i, \sigma_i, \tau\}$ are optimized by maximum likelihood applied to the data set $\mathcal{D}_N$ using the L-BFGS-B algorithm [33].

3.1.3 Quality measures for the POD+GP surrogate model

In the present study, two common error metrics are used to assess the quality of the surrogate water level both on the entire mesh (global approach) and at each point of the mesh (local approach): the root mean square error (RMSE) and the predictive coefficient (Q_2). This validation is carried out over an input-output validation dataset $\mathcal{D}_v$ of size N_v.

Root mean square error (RMSE)

The RMSE is used to measure the accuracy of the model, it should be 0 when the model is perfect. At the k^{th} given mesh node, it is defined as the square root of the mean square error (MSE) measuring the square distance between the surrogate model and the reference model:

$$\mathrm{MSE}_k(\mathcal{D}_v) = N_v^{-1} \sum_{i=1}^{N_v} \left(h_k^{(i)} - \widehat{h}_k^{(i)}\right)^2 \quad \text{and} \quad \mathrm{RMSE}_k(\mathcal{D}_v) = \sqrt{\mathrm{MSE}_k(\mathcal{D}_v)} \tag{10}$$

Their global counterpart are: $\mathrm{MSE}(\mathcal{D}_v) = p^{-1} \sum_{k=1}^{p} \mathrm{MSE}_k(\mathcal{D}_v)$ and $\mathrm{RMSE}(\mathcal{D}_v) = \sqrt{\mathrm{MSE}(\mathcal{D}_v)}$.

Predictive coefficient ($\mathbf{Q_2}$)

At the k^{th} mesh node, the Q_2 predictive coefficient is defined as:

$$Q_{2,k} = 1 - \frac{\mathrm{MSE}_k(\mathcal{D}_v)}{\mathrm{MSE}_k(\mathcal{D}_v; \text{mean})} \tag{11}$$

where $\mathrm{MSE}_k(\mathcal{D}_v; \text{mean}) = N_v^{-1} \sum_{i=1}^{N_v} \left(h_k^{(i)} - \overline{h}^{(i)}\right)^2$ is the MSE of the "averaging model" returning the mean of the learning outputs whatever the input parameter value.
The global counterpart of $\mathrm{MSE}_k(\mathcal{D}_v; \text{mean})$ is $\mathrm{MSE}(\mathcal{D}_v; \text{mean}) = p^{-1} \sum_{k=1}^{p} \mathrm{MSE}_k(\mathcal{D}_v; \text{mean})$.
Thus, the global counterpart of $Q_{2,k}$ is:

$$Q_2 = 1 - \frac{\mathrm{MSE}(\mathcal{D}_v)}{\mathrm{MSE}(\mathcal{D}_v; \text{mean})}. \tag{12}$$

The predictive coefficient measures the performance of the surrogate model with respect to the simplest one which consists in averaging the learning output values. When Q_2 is lower than (resp. equal to) zero, the surrogate is worse than (resp. equal to) the learning output values average. When Q_2 is equal to one, the surrogate interpolates the validation dataset. In practice, the surrogate is deemed appropriate when Q_2 is greater than 0.8. The predictive coefficient is also found under the name of Nash-Sutcliffe model efficiency coefficient in the hydrological literature where is assesses the predictive capacity of the simulated discharge over a time window with respect to observed discharges [22].

3.2 Quantify and explain the output uncertainty due to input uncertainty propagation

Once the model is built and validated, it can be used instead of the reference model in an uncertainty quantification study. After propagating the input uncertainties through the surrogate model by means of specific Monte Carlo methods, we can conduct a statistical analysis on the output uncertainty as well as a sensitivity analysis to explain how the uncertain input parameters contribute to this output variability.

3.2.1 Statistical analysis on the output

Using a standard MC approach on the validation data set $\mathcal{D}_v$, the mean value and standard deviation of the water level at the k^{th} mesh point are formulated as:

$$\mu_k = \frac{1}{N_v}\sum_{i=1}^{N_v} \hat{h}_k^{(i)} \quad \text{and} \quad \sigma_k = \sqrt{\frac{1}{N_v - 1}\sum_{i=1}^{N_v}\left(\hat{h}_k^{(i)} - \mu_k\right)^2}. \tag{13}$$

3.2.2 Sensitivity analysis on the output with respect to the inputs

Sobol' indices [30] are commonly used for sensitivity analysis. They provide the shares of the QoI variance $\mathbb{V}$ attributable to the different uncertain inputs. Under the hypotheses that random input variables are independent, here the roughness coefficients and the upstream flow, and the random QoI is square integrable, here the water level h, the decomposition of the QoI reads:

$$\mathbb{V} = \sum_{i=1}^{d} \mathbb{V}_{\{i\}} + \sum_{j=i+1}^{d} \mathbb{V}_{\{i,j\}} + \cdots + \mathbb{V}_{\{1,2,\ldots,d\}} = \sum_{J\subset\{1,2,\ldots,d\}} \mathbb{V}_J, \tag{14}$$

where $\mathbb{V} := \text{Var}\,[\text{QoI}]$, $\mathbb{V}_i := \mathbb{V}\,[\mathbb{E}[\text{QoI}|X_i)]$, $\mathbb{V}_{ij} := \mathbb{V}\,[\mathbb{E}[\text{QoI}|X_iX_j]] - \mathbb{V}_i - \mathbb{V}_j$ and more generally, for any $I \subset \{1,\ldots,d\}$, $\mathbb{V}_I := \mathbb{V}\,[\mathbb{E}[\text{QoI}|x_I]] - \sum_{J\subset I\ \text{s.t.}\ J\neq I} \mathbb{V}_J$. Then, we obtain:

$$1 = \sum_{i=1}^{d} S_{\{i\}} + \sum_{j=i+1}^{d} S_{\{i,j\}} + \cdots + S_{\{1,2,\ldots,d\}} = \sum_{J\subset\{1,2,\ldots,d\}} S_J, \tag{15}$$

where for any $J \subset \{1,2,\ldots,d\}$, $S_J = \frac{\mathbb{V}_J}{\mathbb{V}}$ is called a Sobol' index and belongs to the interval $[0,1]$. $S_{\{i\}}$ is the first order Sobol' index corresponding to the ratio of output variance due to the i^{th} input parameter uniquely, and $S_{\{ij\}}$ is the second-order Sobol' index describing the ratio of output variance due to the i^{th} parameter in interaction with the j^{th} parameter. Also the total Sobol' index that corresponds to the whole contribution of the i^{th} input parameter reads:

$$S_{T_i} = \sum_{\substack{I\subset\{1,\ldots,d\}\\ I\ni i}} S_I. \tag{16}$$

The computation of first order Sobol' indices requires simple integration, those of the second order requires double integration, and so on. Many Monte Carlo techniques exist to estimate these integrals. In this study, the Sobol' indices are estimated using the algorithm proposed in [28].

Lastly, we note that the expression (14) is defined for a scalar QoI, such has the water level h_k at mesh node $k \in \{1, 2, \ldots, p\}$ where p is the mesh size. Consequently, we can easily plot the different Sobol' indices over the mesh on which is defined the model output. Furthermore, this information can be summarized using the generalized Sobol' indices [20]:

$$\forall J \subset \{1, 2, \ldots, d\},\ S_J = \frac{\displaystyle\sum_{k=1}^{p} \mathbb{V}_{[k]} S_{[k],J}}{\displaystyle\sum_{\ell=1}^{p} \mathbb{V}_{[k]}^{(\ell)}}. \tag{17}$$

4 RESULTS

4.1 Learning and test samples

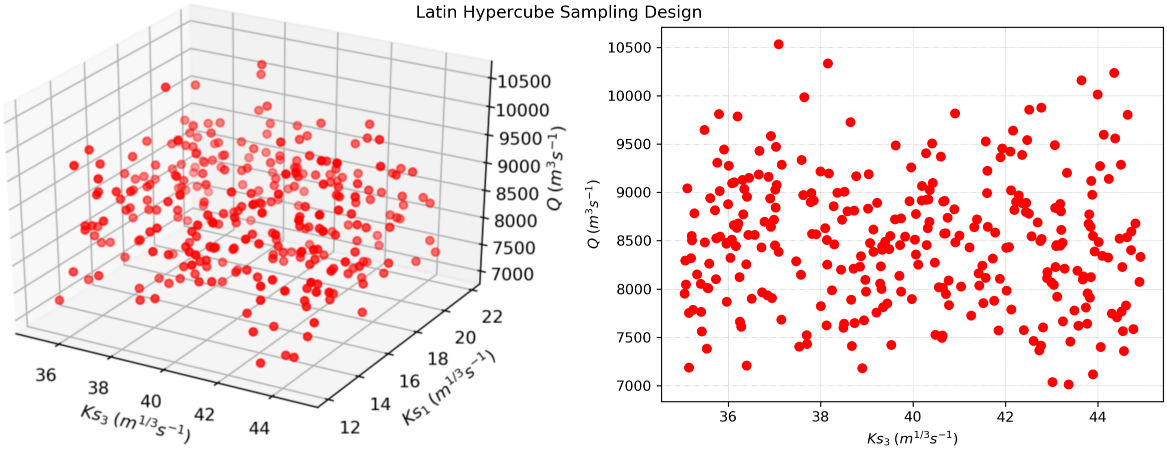


Figure 2: Latin Hypercube Sampling (LHS) DoE for a 300-sample data set, along ($K_{s,1}$, $K_{s,3}$, Q) directions on the left panel and along ($K_{s,3}$, Q) directions on the right panel.

The design of experiment (DoE) for the training and validation data set was generated using Latin Hypercube Sampling (LHS) [21] which is a statistical method for generating a near-random sample of parameter values from a multidimensional distribution. Considering d the number of input variables, LHS strategy scales as $o(d)$ while other strategies require a larger number of samples; for instance to insure the convergence of first order statistics [10]. The LHS space-filling experimental design is shown in Fig. 2, it is associated with a limited computational cost. While more advanced sampling method could be used, LHS strategy was deemed efficient for the present study.

4.2 Surrogate model

The LHS strategy has been applied twice. A first time to build a 2000-sample training set and a second one to create a 1000-sample validation set. Here, the GP kernel was prescribed to a Matern(2.5) function. The validation set was only used to assess the quality of the surrogate model with RMSE and Q_2 error metrics.

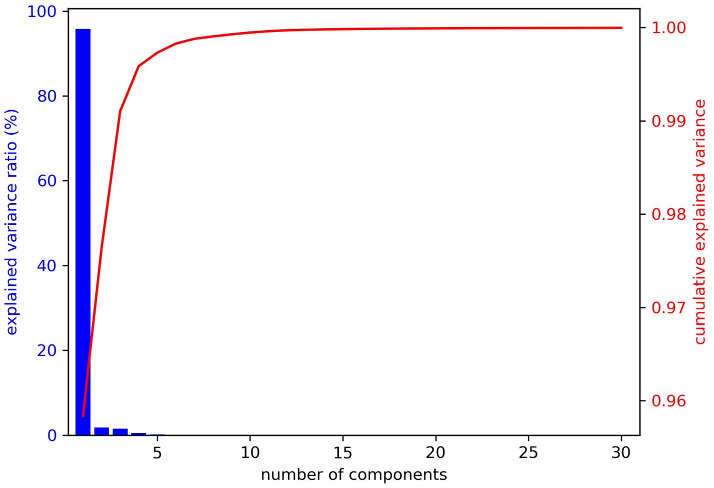


Figure 3: Principal component analysis

The dimension of the output space was reduced with a POD in order to limit the cost of the GP surrogate. The number of modes, also called principal components, to be retained is justified by two criteria taken into account:

- Elbow criterion: on the scree of the POD modes, there is a decrease (elbow) followed by a more regular decrease. In our case, as shown in figure 3, a decrease occurs at the fourth mode, then a regular decrease from the fifth mode. Thus only the first four modes are of interest.
- Kaiser's criterion: only those modes whose inertia is greater than the average inertia should be retained. This criterion leads us to select 4 modes, explaining 99.6% of the total inertia. Indeed, the first principal component explains 95.84% of the total inertia, the second 1.80%, the third 1.46% and the fourth 0.48%.

studyThe cost of the GP surrogate significantly decreases when the dimension of the output is reduced[5] (6 times smaller) applying the POD, as presented in Tab. 2. The output dimension is indeed reduced from $41\,416$ elements to 4 components that explain 99.6% of the variance of the QoI. But, the physical interpretation of the different modes is not always obvious.
As displayed in Fig. 4, the first mode, which explains 95.84% of the output variability, seems to represent the effect of the upstream discharge on the average water level height. Indeed, this component is essentially negative thus its weighting will increase everywhere the average water level height if negative coefficient or decrease everywhere the average water level if positive coefficient. While the second mode, which explains 1.80% of the output variability, seems to

[5] The remaining cost can be considered significant compared to linear surrogate models such as polynomial chaos expansion. This situation is well-known and naturally explained by the learning sample size increasing the cost of inverting the covariance matrix. For prediction, this surrogate model is as fast as the others and also provides a measure of its error.

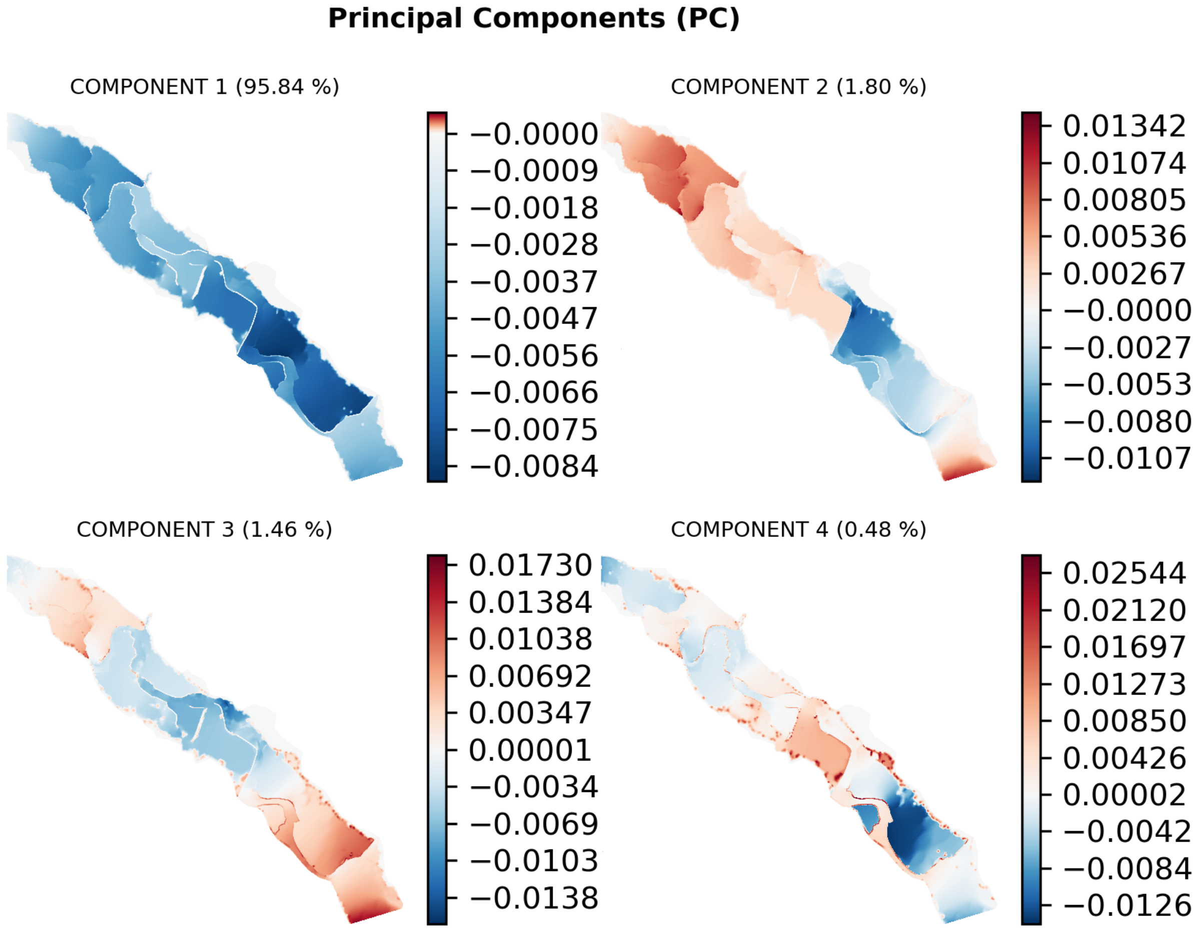


Figure 4: Principal component analysis

represent the effect of main channel flow Strickler friction coefficients on the average water level height as it allows to distinguish the three friction areas defined in section 2.3.
To give more meaning to the principal modes of the decomposition, perspective of our work could stand in the representation of the learning data set on the bi-dimensional sub-spaces spanned by couple of modes, e.g. visualizing the learning data set in a plot with the first mode on the x-axis and the second one on the y-axis.

	GP	POD+GP
CPU run time (h)	≥ 12	2.5

Table 2: CPU run time comparison between GP without and with POD.

The POD+GP surrogate quality is very good with respect to global error metrics RMSE $= 0.8\ cm$ and $Q_2 = 0.99748$. Locally, the quality deteriorates near the boundary of the catchment area as well as along the dikes as shown in Fig. 5. The heterogeneity of the mesh with small cells in the river bed ($\leq\ 40\ m$), near the dikes ($\leq\ 80\ m$) and larger cells in the flood plain ($\leq\ 150\ m$) should be noted and may hide some local failures of the surrogate in the global RMSE and Q_2 criteria.

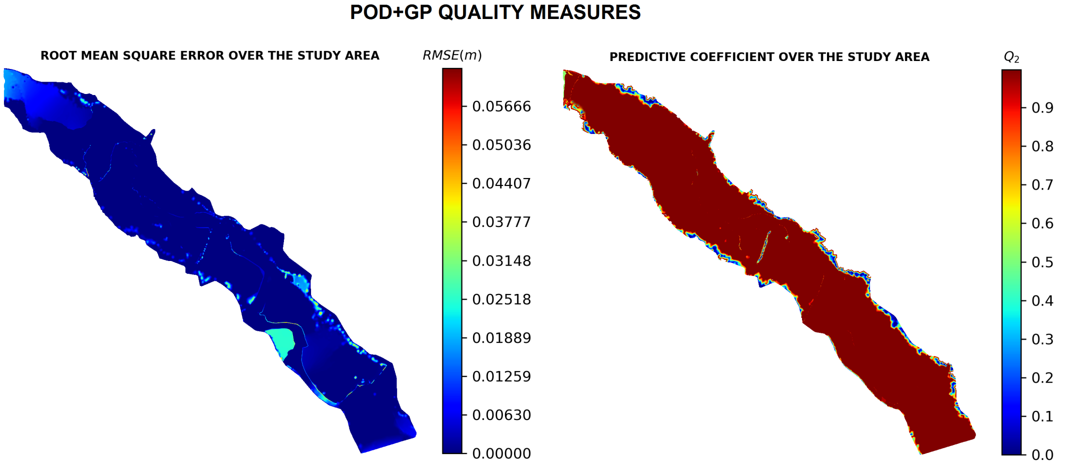


Figure 5: Root mean square error

4.3 Sensitivity analysis with the POD+GP surrogate

The POD+GP surrogate is used to carry out a variance-based sensitivity analysis (SA) over the entire simulated area, with a focus on mesh node $29\ 515$ where Marmande, a city prone to flooding, is located. The POD+GP surrogate allows for a reliable estimation of first and second order statistical moments at Marmande as shown in Tab. 3: the water level mean and standard deviation estimated from the direct model T2D and surrogate are in good agreement with an under estimation of 1.3% for the mean computed with the surrogate.

Given the statistical distributions for the input variables, the SA at Marmande highlights that most of the water level variance is explained by the upstream discharge Q and to a lesser extend, by the Strickler friction coefficient $K_{s,4}$ prescribed between Marmande and La Role as displayed in Fig. 6. At this location, the floodplain friction coefficient $K_{s,1}$ and the friction coefficients upstream of Mas d'Agenais ($K_{s,2}$) and upstream of Marmande ($K_{s,3}$) have barely no impact on the water level. It should be noted that the bootstrap method [1] is used to estimate the variance of the Sobol' indices, this variance is represented by the black error bars in Fig. 6. These indicate that the computation of the SA indices is converged and reliable. It should also be noted that for each input variable, the first (S) and total Sobol' (S_T) indices at Marmande are equal, meaning that, at this location, the multivariate impact of the input on the water level is minimal.

	POD+GP	T2D
Mean (m)	21.57	21.54
Standard deviation (m)	0.24	0.24

Table 3: Statistical moments of the water level height in Marmande.

Fig. 7 displays the mean and the standard deviation of the water level over the 2D domain estimated with the POD+GP surrogate. The mean varies between $0\ m$ near the limits of the domain and $21.57\ m$ at Marmande, where it reaches its maximum. In the floodplain, the mean water level ranges from $3.21\ m$ and $7.8m$ close to the dikes. The water level standard deviation ranges from $0\ m$ to $0.4\ m$ in the floodplain between Mas d'Agenais and Marmande, where the flow is highly bi-dimensional.

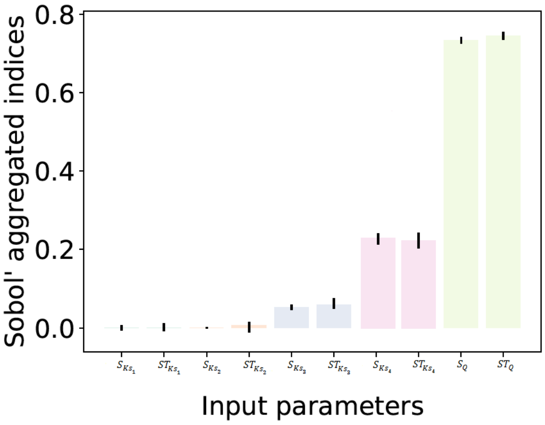


Figure 6: First (S) and total (S_T) Sobol' indices estimated with POD+GP surrogate at Marmande.

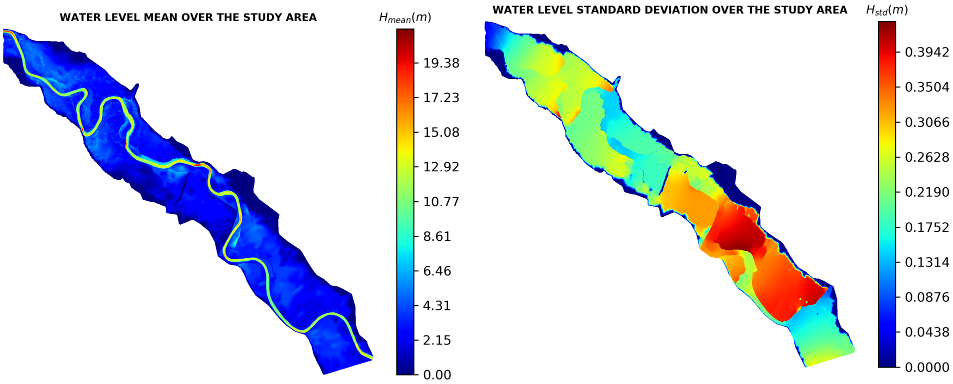


Figure 7: a- Mean water level, b- standard deviation of the water level, estimated with the POD+GP surrogate.

The 2D computation and representation of first Sobol' indices confirms that the variance of the water level is mostly explained (81 % on average) by the input discharge as shown in Fig. 8 and Fig. 9. The floodplain friction coefficient $K_{s,1}$ has no impact on the analysis, the upstream friction coefficient in the river bed $K_{s,2}$ has a small impact on the water level close to Mas d'Agenais, the friction coefficient $K_{s,3}$ between Mas d'Agenais and Marmande explains up to 10 % of the water level variance close to Marmande and the downstream friction coefficient $K_{s,4}$ has an impact over the entire domain with most significance at the upstream and downstream boundaries. As the sum of the first Sobol' indices is smaller than 1, higher order Sobol' indices are non zero, meaning that multivariate effects between Q and K_s explain the remaining part of the water level variance.

5 CONCLUSION AND PERSPECTIVES

In this paper, an uncertainty quantification study was carried out with a 2D numerical solver for the Shallow Water Equations on a section of the Garonne river. It consisted in building a Gaussian process surrogate model on a POD-reduced 2D water level output field.

The surrogate model was formulated with respect to friction coefficients and input discharge, the distribution for friction is supposed to be uniform and centered around calibration values while the discharge distribution is supposed to be Gaussian, centered around a high flood value. The construction of the surrogate was achieved over a 2000-sample training data set and it was

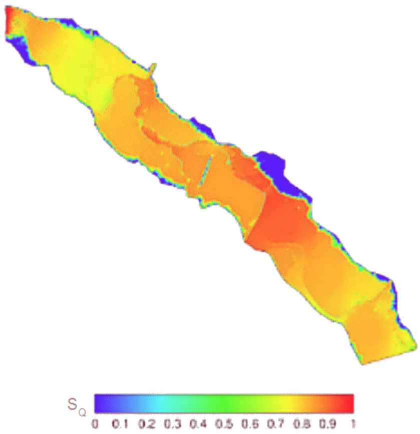


Figure 8: First order Sobol' sensitivity indices computed with the POD+GP surrogate with respect to the input discharge Q.

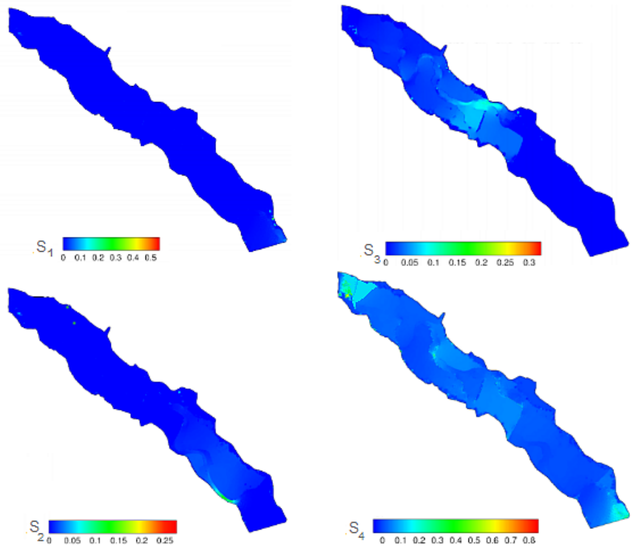


Figure 9: First order Sobol' sensitivity indices computed with the POD-GP surrogate with respect to the friction coefficients $K_{s,1}$ (flood plain), $K_{s,2}$ upstream Mas d'Agenais, $K_{s,3}$ (between Mas d'Agenais and Marmande and $K_{s,4}$ (downstream of Marmande).

validated over a 1000-sample data set. The dimension of the quantity of interest was reduced from 41416 elements to 4 principal components using the POD which has resulted in a significant reduction of the computational cost of the surrogate. The correlation kernel was here prescribed as a Matern(2.5) function. The quality of the POD+GP surrogate model was assessed, the surrogate was deemed satisfying with Q_2 metrics close to 1 for the entire domain and RMSE smaller than $0.01m$. The quality of the surrogate decreases near the dikes. The surrogate was used to perform a global sensitivity analysis based on variance decomposition. It was demonstrated that the upstream discharge is the predominant input variable and explains more than 80 % of the water level variance. The downstream friction coefficient is also a significant input with heterogeneous influence.

It is essential to mention that the conclusions for this study are strongly related to the hypothesis made for the statistical distribution of the inputs. For instance, further study should investigate wider ranges for flood plain coefficients that are highly unknown and may significantly over time as flood events occur on the catchment.

The results of the sensitivity analysis allows for a better understanding of the physics as well as classification of major sources of uncertainty. The latter is of great importance in the context of data assimilation where the control vector should be properly defined to include key factor to improve the model outputs. It was here highlighted that in order to improve water level at Marmande, the control vector should include at least the upstream discharge and the downstream friction coefficient. A perspective for this study thus stands in the implementation of an ensemble-based data assimilation algorithm to improve input discharge and friction assimilation water level observations in the system. Additionally, the cost of the ensemble integration should be reduced using the surrogate model in place of the direct hydraulic solver.

6 ACKNOWLEDGEMENTS

We would like to thank Nicole Goutal from EDF R&D's LNHE department for providing us with the case study, Pamphile Roy from CERFACS for his algorithmic developments in the BATMAN-OpenTURNS library and the EDF R&D's PRISME department for their support in using OpenTURNS. We thank all these people for the enriching discussions we were able to have on uncertainty methods for two-dimensional river hydraulics issues.

REFERENCES

[1] G. Archer, A. Saltelli, I. Sobol, Journal of Statistical Computation and Simulation. *Sensitivity measures, anova-like techniques and the use of bootstrap.*, **58**, 99–120, 1997.

[2] H.H. Barnes, Journal of Hydrology. *Roughness characteristics of natural channels*, **7**, 354, 1969.

[3] M. Baudin, R. Lebrun, B. Iooss, A.L. Popelin, Handbook of Uncertainty Quantification. *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*, 2001–2038, 2017.

[4] A. Besnard, N. Goutal, La Houille Blanche. *Comparaison de modèles 1D à casiers et 2D pour la modélisation hydraulique d´une plaine d´inondation - Cas de la Garonne entre Tonneins et La Réole*, 42–47, 2011.

[5] F.G. Charlton, P.M. Brown, R.W. Benson, Hydraulics Research Station. *The hydraulic geometry of some gravel rivers in Britain*, 1978.

[6] A. Chatterjee, Current Science. *An introduction to the proper orthogonal decomposition*, 808–817, 2000.

[7] W.F. Coon. *Estimates of roughness coefficients for selected natural stream channels with vegetated banks in New York*, U.S. Geological Survey, 1995.

[8] L. Cordier, M. Bergmann. *Proper Orthogonal Decomposition: an overview*, Von Karman Institute for Fluid Dynamics, 2002.

[9] W.L. Cowan, Agricultural Engineering. *Estimating hydraulic roughness coefficients*, **7**, 473–475, 1956.

[10] G. Damblin, Journal of simulation. *Numerical studies of space filling designs: optimization of Latin Hypercube Samples and subprojection properties*, 276–289, 2013.

[11] M. De Lozzo, Journal de la société française de Statistique. *Surrogate modeling and multifidelity approach in computer experimentation*, **156**, 21–55, 2015.

[12] J.C. de Saint-Venant, C. R. Acad. Sc. Paris. *Théorie du mouvement non-permanent des eaux, avec application aux crues des rivières et à l'introduction des marées dans leur lit*, **73**, 147–154, 1871.

[13] D.L. Fread. *BREACH: an Erosion Model for Earthen Dam Failures (Model description and User Manual)*, National Oceanic and Atmospheric Administration, National Weather Service, Silver Spring, MD, 1988.

[14] P. Gauckler, Gauthier-Villars. *Etudes Théoriques et Pratiques sur l'Ecoulement et le Mouvement des Eaux*, 1867.

[15] R. Ghanem, D. Higdon, H. Owhadi, Springer. *Handbook of Uncertainty Quantification*, 2017.

[16] C. Goeury, T. David, R. Ata, Y. Audouin, N. Goutal, A.L. Popelin, M. Couplet, M. Baudin, R. Barate. *Uncertainty quantification on a real case with Telamac-2D*, 2015.

[17] J.M. Hervouet, Wiley. *Hydrodynamics of Free Surface Flows: Modelling with the finite element method*, 2007.

[18] D.M. Hicks, P.D. Mason, Institute of Water and Atmospheric Research (N.Z.). *Roughness characteristics of New Zealand rivers*, 1991.

[19] D.R. Jones, M. Schonlau, W.J. Welch, Journal of Global Optimization. *Efficient Global Optimization of Expensive Black-Box Functions*, **13**, 455–492, 1998.

[20] M. Lamboni, H. Monod, D. Makowski, Reliability Engineering & System Safety. *Multivariate sensitivity analysis to measure global contribution of input factors in dynamic models*, **96**, 450–459, 2011.

[21] M. McKay, J. Beckman, W. Conover, Technometrics. *A comparison of three methods for selecting values of input variables in the analysis of output from a computer code.*, **21**, 239–245, 1979.

[22] J.E. Nash, J.V. Sutcliffe, Journal of Hydrology. *River flow forecasting through conceptual models part I: A discussion of principles*, **10**, 282–290, 1970.

[23] F. Pedregosa, V. Gaël, A. Gramfort, V. Michel and B. Thirion, O. Grisel, Journal of machine learning research. *Scikit-learn: Machine learning in Python*, **12**, 2825–2830, 2011.

[24] J.O. Ramsay, B.W. Silverman, Springer-Verlag. *Functional data analysis*, New York, 1997.

[25] C.E. Rasmussen, C.K.I Williams, MIT Press. *Gaussian Processes for Machine Learning*, 248, Cambridge, MA, USA, 2006.

[26] P.T. Roy, S. Ricci, R. Dupuis, R. Campet, J.C. Jouhaud, C. Fournier, The Journal of Open Source Software. *BATMAN: Statistical analysis for expensive computer codes made easy*, 2018.

[27] P.T. Roy, N. El Moçayd, S. Ricci, J.C. Jouhaud, N. Goutal, M. De Lozzo, M. Rochoux, Stochastic Environmental Research and Risk Assessment. *Comparison of polynomial chaos and Gaussian process surrogates for uncertainty quantification and correlation estimation of spatially distributed open-channel steady flows*, **32**, 1723–1741, 2018.

[28] A. Saltelli, Computer Physics Communications. *Variance based sensitivity analysis of model output. Design and estimator for the total sensitivity index.*, **181**, 259–270, 2010.

[29] A. Siade, M. Putti, W. Yeh, Water resources research. *Snapshot selection for groundwater model reduction using proper orthogonal decomposition*, **46**, 2010.

[30] I.M. Sobol, Math. Modeling Comput. Experiment. *Sensitivity estimates for nonlinear mathematical models*, **4**, 407–414, 1993.

[31] H. Sohr, Birkhuser Basel. *The Navier-Stokes Equations*, 2001.

[32] A. Strickler, Berna. *Beitrge zur Frage der Geschwindigkeitsformel und der Rauhigkeitszahlen fur Strme, Kanle und Geschlossene Leitungen*, 1923.

[33] D.J. Wales, J.P.K. Doye, The Journal of Physical Chemistry A. *Global Optimization by Basin-Hopping and the Lowest Energy Structures of Lennard-Jones Clusters Containing up to 110 Atoms*, **101**, 5111–5116, 1997.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

MODEL INFERENCE FOR ORDINARY DIFFERENTIAL EQUATIONS BY PARAMETRIC POLYNOMIAL KERNEL REGRESSION

David K. E. Green[1,2], **Filip Rindler**[1,2]

[1]The Alan Turing Institute
London, United Kingdom
e-mail: dgreen@turing.ac.uk

[2] Mathematics Institute, University of Warwick
Coventry, United Kingdom
e-mail: f.rindler@warwick.ac.uk

Keywords: Inverse Problems, Model Inference, Machine Learning, Dynamical Systems, Time series Analysis, Artificial Neural Networks, Polynomial Kernel Methods

Abstract. *Model inference for dynamical systems aims to estimate the future behaviour of a system from observations. Purely model-free statistical methods, such as Artificial Neural Networks, tend to perform poorly for such tasks. They are therefore not well suited to many questions from applications, for example in Bayesian filtering and reliability estimation.*

This work introduces a parametric polynomial kernel method that can be used for inferring the future behaviour of Ordinary Differential Equation models, including chaotic dynamical systems, from observations. Using numerical integration techniques, parametric representations of Ordinary Differential Equations can be learnt using Backpropagation and Stochastic Gradient Descent. The polynomial technique presented here is based on a nonparametric method, kernel ridge regression. However, the time complexity of nonparametric kernel ridge regression scales cubically with the number of training data points. Our parametric polynomial method avoids this manifestation of the curse of dimensionality, which becomes particularly relevant when working with large time series data sets.

Two numerical demonstrations are presented. First, a simple regression test case is used to illustrate the method and to compare the performance with standard Artificial Neural Network techniques. Second, a more substantial test case is the inference of a chaotic spatio-temporal dynamical system, the Lorenz–Emanuel system, from observations. Our method was able to successfully track the future behaviour of the system over time periods much larger than the training data sampling rate. Finally, some limitations of the method are presented, as well as proposed directions for future work to mitigate these limitations.

1 INTRODUCTION

Dynamical systems play a crucial role in mathematical modelling across all areas of physics, engineering and applied mathematics. The equations used in some particular application domain are typically derived either phenomenologically [23] or from first principles such as the conservation of energy, mass or momentum (as in mechanics [27]). The structure of the equations should describe the fundamental aspects of the system in question as much as possible. On the other hand, constitutive parameters are often hard to know explicitly and need to be learnt from data. As such, it is necessary to balance rigidity and flexibility when modelling a system.

This paper considers the problem of finding a model of a dynamical system, represented by coupled Ordinary Differential Equations (ODEs), from observations. This is a particular form of inverse problem (as in [25]). The time evolution of many dynamical systems is described by polynomial equations in the system variables and their derivatives. We introduce a form of parametric polynomial kernel regression (related to Radial Basis Function networks [21]). This technique was developed during the search for an algorithm that is able to be trained continuously on streaming data as opposed to complete trajectories. Hidden parameter models (with unobserved variables) are not addressed but the techniques shown here could be extended to such cases in the future, augmenting probabilistic Bayesian filtering methods (as in [16]).

Kernel ridge regression is a nonparametric method for fitting polynomials to data without explicitly calculating all polynomial terms of a set of variables [18, 21]. There are two limitations of this approach when fitting models to time series data. First, as a nonparametric method, the computational time complexity scales cubically with the number of observation points. This is a significant issue when dealing with time series data. Second, it is difficult to compute kernel ridge regression efficiently using streaming data. While it is possible to continually update the inverse of a matrix (see [9]), the roughly cubic scaling of the required matrix operations is not well suited to monitoring high-dimensional systems in a real time data setting. Here, to optimise our parametric polynomial kernel function representations, Stochastic Gradient Descent (SGD) is used along with the Backpropagation method (see [3]). This combination of techniques helps to minimise computational complexity and the amount of explicit feature engineering required to find a good representation of an unknown ODE.

We represent ODE models parametrically as compute graphs. Compute graphs are used in Artificial Neural Network (ANN) theory to model complicated nonlinear structures by the composition of simple functions and are well suited to gradient descent optimisation via the Backpropagation method. It is demonstrated that numerical integration (both explicit and implicit) can be used to discretise ODE time integrals in a way that allows for the inference of continuous-time dynamical system models by gradient descent. This is an extension of an approach that appeared at least as early as [6]. The discretisation procedure is related to the Backpropagation Through Time method [29], which is used for modelling discrete time series with so-called Recurrent Neural Networks.

To demonstrate the findings of this paper, two numerical case studies were carried out. The first is a simple analysis that contrasts the performance of standard ANN techniques with the proposed kernel method. It is shown that our method had the best extrapolation performance. A more extensive analysis of the chaotic spatio-temporal Lorenz–Emanuel dynamical system is also presented. The proposed method is able to recover a maximum likelihood estimate of the hidden polynomial model. For comparison, a parametric model constructed by direct summation of polynomial features (without kernels, of the form used in [26]) was also tested. The parametric polynomial kernel method was able to outperform the direct polynomial expansion, accurately predicting the future evolution of a chaotic dynamical system over periods many times greater than the training interval.

The primary advantage of the technique presented in this paper is that the model representation in parametric form can avoid the curse of dimensionality and poor scaling with training set size associated with nonparametric kernel regression. Further, polynomial kernels avoid the combinatorial explosion that occurs when explicitly computing polynomial series expansions. Interestingly, the accuracy of the proposed parametric kernel method can be tuned by adjusting the dimension of a set of intermediate parameters. The trade-off for increased accuracy is additional training time.

2 BACKGROUND ON COMPUTE GRAPH OPTIMISATION

2.1 Compute graphs and nonlinear function representations

The parametric polynomial regression technique introduced in this paper is built on the framework of so-called compute graphs. This section provides the background theory required for later parts of this work. Compute graphs are very general structures which define the flow of information over a topology and as such provide a convenient parametric representation of nonlinear functions. In particular, compute graphs can be coupled with Automatic Differentiation [20] and the Backpropagation algorithm (an application of the chain rule) to allow for gradient-based optimisation. Stochastic Gradient Descent is the most common form of optimiser used in this context and is briefly described in this section.

Artificial Neural Networks (ANNs) are a subset of compute graphs (in the sense of discrete mathematics [7]). Common ANN terminology such as Deep Neural Networks, Boltzmann Machines, Convolutional Neural Networks and Multilayer Perceptrons refer to different ANN connectivity, training and subcomponent patterns [3, 8]. The choice of an appropriate ANN type depends on the problem being solved. This section works with general compute graph terminology, rather than specific ANN design patterns, as these principles are appropriate for all ANN architectures.

A (real-valued) compute graph consists of a weighted directed graph, i.e. an ordered pair $G = (V, E)$ with the following properties:

- V is the finite set of vertices (or nodes) v_i. Vertices specify an activation function $\sigma_i : \mathbb{R} \to \mathbb{R}$, and an output (or activation) value $a_i \in \mathbb{R}$.

- E is the set of edges e_{ij}. Each edge e_{ij} specifies a start vertex, defined to be v_i, and an end vertex, defined to be v_j. That is, edges are said to start at v_i and terminate at v_j. Edges also specify a weight, $W_{ij} \in \mathbb{R}$.

Edges e_{ij} can be understood as ‘pointing’ from v_i to v_j. Incoming edges to a node v_i are all $e_{jk} \in E$ with $k = i$. Similarly, outgoing edges from a node v_i are all $e_{jk} \in E$ with $j = i$. Parents of a node v_i refer to all nodes v_j such that there is an edge starting at v_j and terminating at v_i. Similarly, children of a node v_i refer to all nodes v_j such that there is an edge starting at v_i and terminating at v_j. A valid path of length m starting at v_1 and terminating at v_m is a set $\{v_1, v_2, \cdots v_m\}$ of at least two nodes such that there exist edges in E from v_i to v_{i+1} for all $i \in [1, m-1]$. A recurrent edge in a compute graph refers to an edge that lies on a valid path from a node v_i to any of its parents. A graph with recurrent edges is said to be a recurrent graph. An example of a (recurrent) compute graph is shown in fig 1.

Inputs to the compute graph are all those nodes with no incoming edges (i.e. no parents), $\{v_i | v_i \in V \wedge \nexists e_{ki} \in E\}$. The activation values a_i for input nodes v_i must be assigned. The

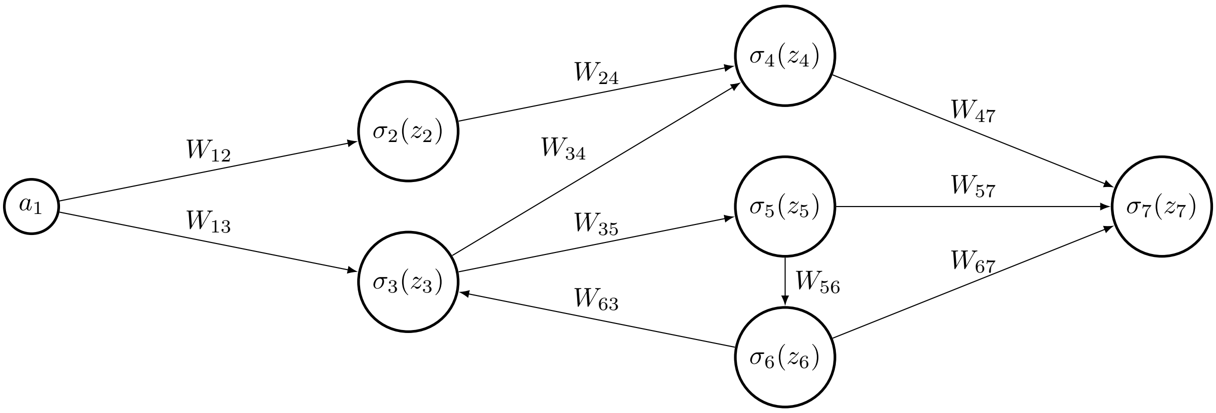


Figure 1: Example of compute graph. The subscript inside each node denotes the node number. Arrowheads indicate the direction of the graph edges. The function inside each node refers to the output function to be applied at the node. Note that node 1 is an input (with value a_1) as it has no parents. Further note that edge W_{63} is recurrent as there is a cycle formed in the graph between nodes $3, 5$ and 6.

values at all other nodes, v_i, in the compute graph are calculated by

$$z_i = \sum_{k\colon v_k \text{ parent of } v_i} W_{ki} a_k, \tag{1}$$

$$a_i = \sigma_i\left(z_i\right), \tag{2}$$

where z_i represents the weighted inputs to a node from all parent nodes and a_i represents the output from a node.

Note that ANNs often define so-called bias units. Bias units allow for inputs to a node to have their mean easily shifted. A bias input to some node v_i can be represented in a compute graph by creating a set of nodes $b_i \in B$, with no parents, that always output a value of 1. Further, each b_i is assigned to be an additional parent of v_i by creating an edge from b_i to v_i with weight B_i so that

$$a_i = \sigma_i\left(\sum_{k\colon v_k \text{ parent of } v_i} W_{ki} a_k + B_i\right). \tag{3}$$

Bias units will not, however, be explicitly indicated in the rest of this section as they can be assumed to be implicitly defined in eqn (1).

The composition of simple functions with a compute graph structure allows for complicated nonlinear functions to be represented parametrically [3].

2.2 Optimisation by Stochastic Gradient Descent and Backpropagation

Optimisation over very large compute graphs representing highly nonlinear functions has become possible using Stochastic Gradient Descent (SGD) coupled with Backpropagation of errors [3]. Advanced forms of SGD such as the Adam optimisation technique [15] are useful for optimising complicated compute graphs. The basic SGD method is described here. Stochastic Gradient Descent finds a locally optimal set of parameters, θ, by iteratively updating the current estimate for the optimal parameters, θ_i. It does so by moving the current estimate in the direction of greatest decreasing error, given by the derivative $\nabla_\theta J(\theta_i)$:

$$\theta_{i+1} := \theta_i - \eta \nabla_\theta J(\theta_i), \tag{4}$$

where η is a small parameter that gives the distance to move in the direction defined by $\nabla_\theta J(\theta)$. Iterations are repeated until a specified error tolerance $\epsilon > 0$ is reached, i.e. until

$$J(\theta_i) \leq \epsilon. \tag{5}$$

Consider the case of approximating some unknown function $f(x)$ by a compute graph that outputs the function $\tilde{f}_\theta(x)$. The weights θ are taken to be the values of the edge weights W_{ij} for all $e \in E$. Let the loss functional in this example be given by

$$J(\theta) := \sum_x |f(x) - \tilde{f}_\theta(x)|^2, \tag{6}$$

for x in some finite set. Thus, $J(\theta)$ is also representable as a compute graph. The graph for $J(\theta)$ contains the graph for $\tilde{f}_\theta(x)$ as a subset. To apply SGD to a compute graph, extended to contain the terms computing the loss functional, the Backpropagation method (an application of the chain rule) can be used if two conditions are met:

- All nodal activation functions, σ_i, must be differentiable.
- The graph must be directed and acyclic, meaning the graph cannot contain any valid paths from a node to any of its parents, i.e. the graph must not have any recurrent edges.

If the above conditions are satisfied, Backpropagation can compute $\nabla_\theta J(\theta)$ via the chain rule. The basic procedure is outlined here, but a more detailed treatment can be found in [3]. In the case that the graph is not acyclic, it can be unrolled via a technique referred to as Backpropagation Through Time [29].

Backwards error derivatives must be computed at all nodes, v_i, in the network:

$$\delta_i := \frac{\partial J}{\partial z_i}. \tag{7}$$

For nodes v_i in the graph that compute the loss functional $J(\theta)$, the derivative δ_i can be computed directly. Otherwise, assume that node v_i has children $\{w_j\}_{j=1}^N$. Using the chain rule, the error derivative δ_i can be calculated by pushing the error derivatives backwards through the graph from children to parents:

$$\delta_i = \sum_{j=1}^N \delta_j \frac{\partial z_j}{\partial a_i} \frac{\partial a_i}{\partial z_i} = \sum_{j=1}^N \delta_j W_{ij} \sigma_i'(z_i). \tag{8}$$

Given the error derivative terms, the desired error gradients $\nabla_\theta J(\theta)$ for $\theta = \{W_{ij}\}_{ij}$ can be computed at node v_j with parents $\{w_k\}_{k=1}^M$ by

$$\frac{\partial J}{\partial W_{ij}} = \delta_j \frac{\partial z_j}{\partial W_{ij}} = \delta_j \frac{\partial}{\partial W_{ij}} \left(\sum_{k=1}^M W_{kj} a_k \right) = \delta_j a_i. \tag{9}$$

Automatic Differentiation [20] can be used to write efficient computer code for Backpropagation. Specifically, Backpropagation is a form of 'reverse accumulation mode' Automatic Differentiation. The above calculations can be organised efficiently by going through the compute graph from output to input nodes. At the time of writing, Tensorflow [1] is a popular implementation of the algorithms described above. Although other (including gradient-free) optimisation procedures can be used that are suitable for general compute graphs, SGD with Backpropagation is typically very computationally efficient when applicable.

3 PARAMETRIC POLYNOMIAL KERNEL REGRESSION

3.1 Overview

Before discussing model inference for ODEs in particular, a parametric polynomial kernel function representation is introduced. Although ANNs and compute graphs are very effective at fitting arbitrary functions, standard ANN methods are poorly suited to polynomial function representation. As typical ANN architectures fit a very large number of parameters, they are unable to perform sensible extrapolation for even low-dimensional polynomial regression problems. Polynomial kernel ridge regression using the so-called kernel trick [21] works well for fitting polynomials but suffers from cubic (that is, $\mathcal{O}(N^3)$) computational time complexity. Gradient-descent compute graph optimisation, as it is a parametric method, provides a way to optimise large data sets without the computational difficulties faced by nonparametric methods. While it is possible to build a compute graph that explicitly includes polynomial basis features, this scales factorially with the number of polynomial features included. In this paper it is shown that polynomial kernels can be inserted into compute graph structures and optimised by SGD, avoiding both the combinatorial explosion of polynomial series expansions and the poor time scaling of nonparametric kernel ridge regression.

3.2 Polynomial kernel ridge regression

Polynomial kernels, typically associated with kernel regression and Support Vector Machines [21, 18], are functions of the form

$$K(x, y) = (b\langle x, y\rangle + c)^d \tag{10}$$

for some $b, c \in \mathbb{R}$, $d \geq 1$. If the values of y are assumed to be some parameters, the expansion of the polynomial kernel (for $d \in \mathbb{N}$) will, implicitly, yield all polynomial combinations up to order d.

Kernel ridge regression is a nonparametric method in the sense that the number of parameters grows with the amount of training data [18]. By contrast, in this paper 'parametric model' refers to a model with a fixed number of parameters. Adopting the notation in [28], the standard form of ridge regression is as follows. Given observations of an unknown function $f \colon \mathbb{R}^D \to \mathbb{R}^E$ at N locations, $\{(x_i, f(x_i))\}_{i=1}^N$, kernel ridge regression finds an approximation, $f_k(x)$, by

$$f(x) \approx f_k(x) = \sum_{i=1}^{N} \alpha_i K(x, x_i), \tag{11}$$

where the values α_i are termed weights and $K(x, x_i)$ is a kernel function. Kernel functions are a form of generalisation of positive definite matrices (see [18] for additional details). Only the (real-valued) polynomial kernel in eqn (10) will be discussed in this paper. The weights

$\alpha = (\alpha_1, \dots, \alpha_N)$ are calculated using $f(x) = (f(x_1), \dots, f(x_N))$ as follows:

$$\alpha = (K + \lambda I)^{-1} f(x), \tag{12}$$

where $K \in \mathbb{R}^{N \times N}$ is the matrix with entries $K_{ji} = K(x_j, x_i)$ and I is the N by N identity matrix. The term $\lambda \in \mathbb{R}$ is a regularisation term that controls overfitting. Note that if $K + \lambda I$ is not invertible, then the inverse must be replaced by a pseudo-inverse. In the sense of Bayesian regression, the term λ represents the scale of Gaussian noise added to observations $f(x_i)$ as a part of the approximation procedure.

The use of kernels for regression as in eqn (11) has the effect of mapping a low-dimensional problem implicitly into a high-dimensional space. This is a very powerful technique for projecting data onto high-dimensional basis functions. Unfortunately, as a (typically) dense matrix must be inverted to calculate α, the computational complexity of standard kernel ridge regression scales cubically with the number of data points, N. This is a severe limitation when considering large data sets such as the time series data considered in later sections of this paper.

3.3 Parametric polynomial kernel representation

Instead of calculating an inner product between known values of x and y as in eqn (10) and inverting a matrix as in eqn (12), this paper demonstrates that a kernel representation can be found in an efficient way using compute graphs and SGD. Consider the following parametric representation of a function $f \colon \mathbb{R}^D \to \mathbb{R}^E$ with parameters $\theta \in \Theta$:

$$f_\theta(x) = W_2 \left[(W_1 x + B_1) \circ (W_1 x + B_1) \right] + B_2, \tag{13}$$

where $\circ$ denotes elementwise matrix multiplication (or Hadamard product), i.e. $A = B \circ C$ means $a_{ij} = b_{ij} c_{ij}$ for the corresponding matrix entries [12]. The remaining terms are defined by $W_1 \in \mathbb{R}^{M \times D}$, $B_1 \in \mathbb{R}^D$, $W_2 \in \mathbb{R}^{E \times M}$ and $B_2 \in \mathbb{R}^E$. The parameters B_1, B_2 are known as bias weights in the ANN literature [3]. The full set of parameters for this model is $\theta = \{W_1, B_1, W_2, B_2\}$. The dimension M is an intermediate representation dimension and is discussed below.

Eqn (13) is a parametric representation of a second-order polynomial kernel. Expanding eqn (13) explicitly would yield a set of second-order polynomials in terms of x_i. However, using SGD the unknown polynomial expression can be found without the need to know the expanded polynomial form. The elementwise matrix product acts like the d-th power in eqn (10). The parameters θ can be trained by SGD and function as parametric representations of Support Vectors. The term M required to complete the definition of eqn (13) is a hyperparameter representing a choice of intermediate representation dimension and is related to the number of Support Vectors required to represent the system (as in Support Vector Regression, see [21]). Increasing the size of M increases the number of parameters but can improve the fit of the regressor (as is demonstrated empirically in Section 5).

An n-th order polynomial could be fit by taking a larger number of Hadamard products. Denote the composition of Hadamard products by $A \circ^n A := A \circ A \circ \cdots \circ A$ (n times). Then, our approach consists of expressing an n-th order representation of $f_\theta \colon \mathbb{R}^D \to \mathbb{R}^E$ as follows:

$$f_\theta(x) = W_2 \left[(W_1 X + B_1) \circ^n (W_1 X + B_1) \right] + B_2 \tag{14}$$

or some similar variation on this theme. The expression in eqn (14) is differentiable in the sense of compute graphs since all of the operations in eqn (14) are differentiable. Comparing with eqns (11) and (12), the parametric form of polynomial kernel regression can be thought of as an approximation to both the α_i and $K(x, x_i)$ terms in a single expression. As the parametric regression form can be optimised by SGD, the cubic scaling of nonparametric kernel ridge regression is avoided.

3.4 Numerical demonstration on simple regression problem

This section demonstrates the proposed method via the approximation of a simple cubic function, namely

$$f(x) := (x-1)(x+1)(x+0.5). \tag{15}$$

The goal of this analysis is to infer the hidden function $f(x)$. Given a set of training data, N pairs $\{(x_i, f(x_i))\}_{i=1}^N$, the problem is to minimise the loss functional

$$J(\theta) := \frac{1}{N}\sum_{i=1}^{N} |f(x_i) - f_\theta(x)|^2. \tag{16}$$

For this test problem, $N = 25$ training data points were sampled uniformly between $x = -2$ and $x = 2$.

First, a standard ANN 'Multilayer Perceptron' (specifically a three-layer deep, 100 unit wide perceptron network) was tested. The reader unfamiliar with these terms can see [21] for definitions, but it is sufficient for the purposes of this paper to understand that this perceptron model computes the function

$$f_\theta(x) = W_4\sigma(W_3\sigma(W_2\sigma(W_1 x + B_1) + B_2) + B_3) + B_4 \tag{17}$$

where $W_1 \in \mathbb{R}^{100\times 1}$, $W_2, W_3 \in \mathbb{R}^{100\times 100}$, $W_4 \in \mathbb{R}^{1\times 100}$, $B_1, B_2, B_3 \in \mathbb{R}^{100}$, and $B_4 \in \mathbb{R}$ such that the parameters of this network are $\theta = \{W_i, B_i\}_{i=1}^4$. Additionally, $\sigma(x)$ denotes the sigmoid function:

$$\sigma(x) := \frac{1}{1+e^{-x}}. \tag{18}$$

In eqn (17), σ is applied to vectors componentwise.

Second, the parametric polynomial method in eqn (14) was tested for polynomial orders $n = 2, 3, 4$. The parameter M was fixed to 20 for all comparisons.

Both the perceptron model and the parametric polynomial kernel model were trained in two stages. The Adam optimiser [15] was first run for 1000 iterations with a learning rate of 0.01 and then for an additional 1000 iterations with a learning rate of 0.001. All ANNs and SGD optimisers were implemented using the Tensorflow software library [1].

Finally, a nonparametric kernel ridge regression estimator of the form in eqn (11) was tested. This was implemented using the SciKit learn 'KernelRidge' function [19] using a third-order polynomial kernel. Note that this function has additional hyperparameters, $\alpha, \mathrm{coef0}$ and γ. These were set to 0.1, 10 and 'None' respectively. The SciKit documentation describes these parameters in detail. As with the parametric estimator, the choice of maximum polynomial degree (d in eqn (12)) is another hyperparameter. For this demonstration, only the known true value ($d = 3$) was tested with the nonparametric regression estimator.

The values of $J(\theta)$ after running SGD are shown in table 1. The third-order parametric polynomial loss is ten orders of magnitude lower than the regression loss of the perceptron

Function representation	$J(\theta)$
Multilayer Perceptron	1.12×10^{-4}
Parametric kernel with $n = 2$	1.70×10^{0}
Parametric kernel with $n = 3$	5.95×10^{-14}
Parametric kernel with $n = 4$	2.34×10^{-1}
Nonparametric polynomial kernel	9.68×10^{-3}

Table 1: Values of $J(\theta)$, defined in eqn (16), after optimisation by SGD for the simple regression task.

network. The lower loss of the $n = 3$ parametric polynomial method compared to $n = 2$ and $n = 4$ is (of course) expected as the hidden function is a third-order polynomial. This indicates that several polynomial orders should be tested when applying the proposed technique to other problems.

The results of the analysis are shown in figs 2 and 3. Each model tested was able to recover the true form of $f(x)$ in the region of the training data. Relative errors for each method are shown in fig 4. Both the parametric and nonparametric polynomial methods were also able to extrapolate well beyond the range of the original data for the $n = 3$ model. This can be best seen in fig 3. The perceptron model, by contrast, almost immediately fails to predict values of the hidden function outside of range of the training data. For inferring hidden polynomial dynamical systems from observations, where the ability to extrapolate beyond the training data is essential, the analysis in this section suggests that the parametric polynomial kernel method can be expected to have performance superior to standard ANN methods.

This analysis also indicates that the loss $J(\theta)$ is an effective indicator of extrapolation performance for polynomial kernel methods (at least in this test case). This is not true for the Multilayer Perceptron model which had a low $J(\theta)$ value but poor extrapolation performance. One must however take care when making assertions about extrapolation performance, as it is easy to make incorrect inferences in the absence of data.

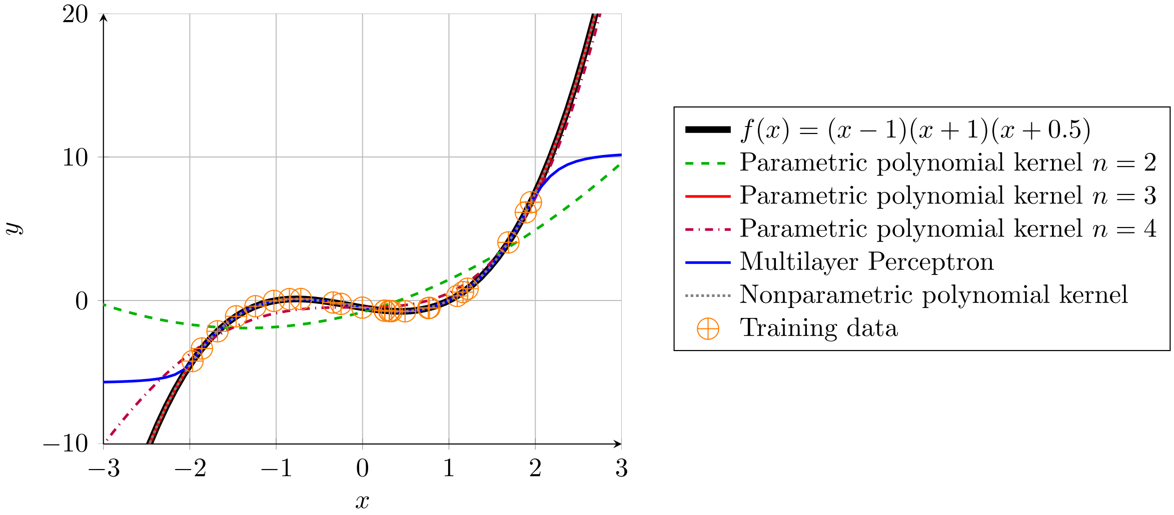


Figure 2: Comparison of performance of the parametric polynomial kernel method on a simple regression task. Note that the true hidden function, from eqn (15), is underneath the function inferred by the $n = 3$ parametric polynomial. The two coincide because of the virtually perfect fit. The nonparametric polynomial kernel ridge estimator also closely coincides with the true $f(x)$. The 25 regression training data points were calculated by sampling uniformly between $x = -2$ and $x = 2$.

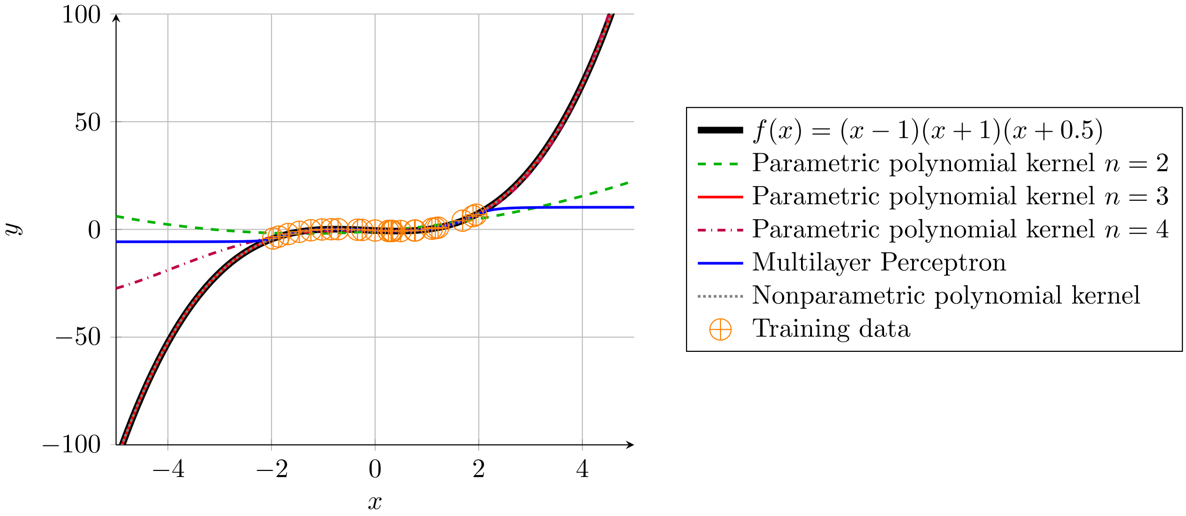


Figure 3: Comparison of performance of the parametric polynomial kernel method on a simple regression task. This is a zoomed out view of fig 2 and shows that the polynomial kernel estimators (both parametric for $n = 3$ and nonparametric) are able to recover the true hidden function in eqn (15) outside of the range of the training data.

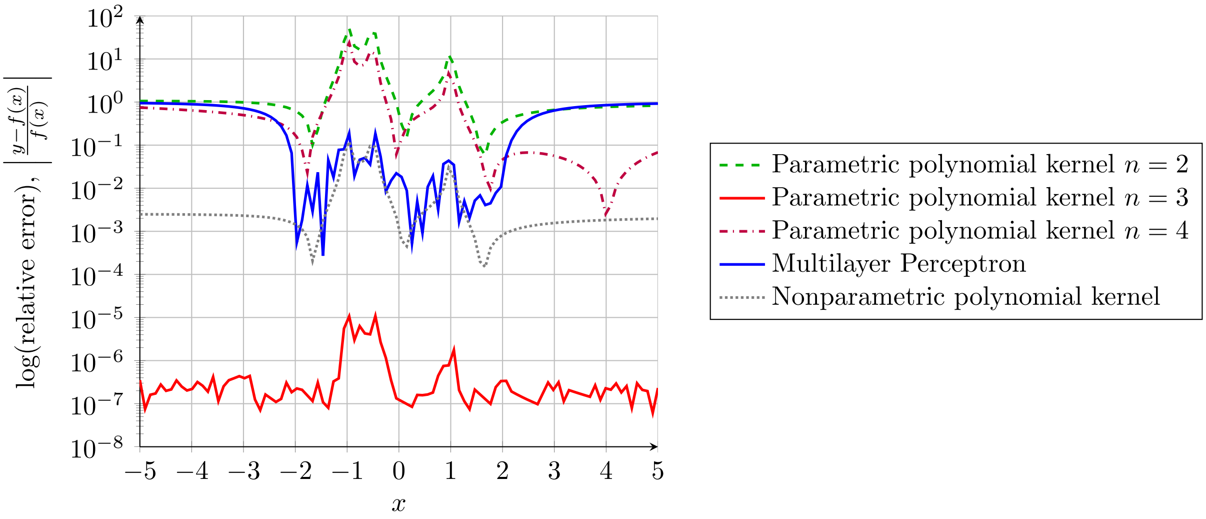


Figure 4: Comparison of pointwise absolute errors for the simple regression task. Errors are computed as $\left|\frac{y-f(x)}{f(x)}\right|$ where $f(x)$ is the true hidden function defined in eqn (15). The parametric polynomial kernel method has the best performance, followed by the nonparametric polynomial kernel ridge method. Note that the training data was restricted to lie within $x = -2$ and $x = 2$.

4 ORDINARY DIFFERENTIAL EQUATION MODEL INFERENCE

4.1 Dynamical Systems

Dynamical systems are classified into either difference equations (discrete-time systems) or differential equations (continuous-time systems) [17]. In this paper, only continuous-time dynamical systems are investigated, although the numerical methods presented could be applied to both continuous-time and discrete-time systems. Continuous-time dynamical systems of the form considered in this paper can be expressed as coupled first-order Ordinary Differential Equations (ODEs):

$$\frac{d}{dt}u(t) = f(t, u(t)), \tag{19}$$

where:

- $t \in [0, \infty)$ represents time;
- $u(t) \in \mathbb{R}^n$ is the vector of values representing the n variables of the system at time t;
- $f(t, u(t)) \in \mathbb{R}^n$ represents the prescribed time derivatives of $u(t)$.

A trajectory of a dynamical system refers to a parameterised path $u(t)$ which returns a value of u for all values of the parameter t. The value of $u(t)$ in eqn (19) can be computed given some initial value, $u(0)$, by integrating $f(t, u(t))$ forward in time:

$$u(t) = u(0) + \int_0^t \frac{d}{d\tau}u(\tau)d\tau = u(0) + \int_0^t f(\tau, u(\tau))d\tau \tag{20}$$

To simplify the solution of ODEs and the implementation of the learning algorithm presented in this paper, we only consider first-order systems. A differential equation of order m of the form

$$\frac{d^m}{dt^m}u(t) = f(t, u(t)) \tag{21}$$

can be converted into a system of first-order coupled ODEs. This is also the standard approach employed in numerical implementations of ODE solvers, for an example, see the SciPy function solve_ivp [14]. The conversion can be achieved by introducing new variables for higher derivatives. Consider an m-th order equation of the form

$$\frac{d^m u}{dt^m} = g\left(t, u, \frac{du}{dt}, \frac{d^2u}{dt^2}, \cdots, \frac{d^{m-1}u}{dt^{m-1}}\right). \tag{22}$$

This can be rewritten by replacing the $\frac{d^iu}{dt^i}$ terms by new variables v_i ($i \in [1, m-1]$) such that:

$$\frac{d}{dt}\begin{bmatrix} u \\ v_1 \\ \vdots \\ v_{m-1} \end{bmatrix} = \begin{bmatrix} v_1 \\ \vdots \\ v_{m-1} \\ f(t, u, v_1, v_2, \ldots, v_{m-1}) \end{bmatrix}. \tag{23}$$

As the value of u at some time depends on the values at infinitesimally earlier times through the derivatives of u, there is a recursive structure present in the equations (this would be even clearer for difference equations or after a discretisation). The model inference technique presented in this paper uses loop unrolling to simplify the derived optimisation problem.

4.2 Model inference for coupled ODEs

Model inference, in this context, is the problem of recovering the form of $f(t, u(t))$ (as in eqn (19)) given observations of $u(t)$ at times from 0 to T. Model inference can be expressed as an optimisation problem:

$$\text{Minimise } J(\theta) := \int_0^T \left| u(t) - \left(u(0) + \int_0^t f_\theta(\tau, u(\tau))d\tau\right) \right|^2 dt, \tag{24}$$

where $J(\theta)$ is a loss functional over some unknown parameters $\theta \in \Theta$. The function $f_\theta(\tau, u(\tau))$ denotes a parametric approximation to the true latent function $f(t, u(t))$. For the purposes of this paper, the parametric representation of f_θ can be assumed to be a directed acyclic compute graph. Denote the trajectories computed using the integral of f_θ by

$$\tilde{u}_\theta(t) := u(0) + \int_0^t f_\theta(\tau, u(\tau))d\tau. \tag{25}$$

Then the loss functional in eqn (24) can be expressed as

$$J(\theta) = \int_0^T |u(t) - \tilde{u}_\theta(t)|^2 \, dt. \tag{26}$$

In this form, it is clear that the $J(\theta)$ measures how closely the observed trajectories $u(t)$ match the predicted trajectories $\tilde{u}_\theta(t)$ for each value of θ. Additionally, although the L^2 norm has been used above, this norm could be changed to any other norm as appropriate. For simplicity, only the L^2 norm will be used in this paper.

If observations of $\frac{du}{dt}$ are available, the optimisation problem can be expressed in an alternative, but not exactly equivalent, differential form:

$$\text{Minimise } K(\theta) := \int_0^T \left| \frac{d}{dt}u(t) - f_\theta(t, u(t)) \right|^2 dt. \tag{27}$$

The loss functional surface for $J(\theta)$ will tend to be smoother over θ when compared to the differential form (since there is an additional integration), potentially altering the behaviour of various optimisation methods. However, the exact minimisers θ^* of both $J(\theta)$ and $K(\theta)$, if they exist so that $J(\theta^*) = K(\theta^*) = 0$, are the same, as can be seen by differentiation.

The choice to optimise over $K(\theta)$ or $J(\theta)$ depends on the chosen representation of f_θ and the availability of observations. Assume that only observations of $u(t)$ are available and not direct observations of $\frac{du}{dt}$. Then it is necessary to either introduce some way to approximate $\frac{du}{dt}$ or to approximate $\int_0^t f_\theta(\tau, u(\tau))d\tau$. In the remainder of this section, it is shown that a discretised form of $J(\theta)$, denoted by $\hat{J}(\theta)$, can be derived. The discretised objective $\hat{J}(\theta)$ can be trained using SGD and Backpropagation as long as $f_\theta(t, u(t))$ can be represented by an acyclic compute graph. The derivation of $\hat{J}(\theta)$ proceeds by first approximating the outer integral in eqn (26) using a finite set of observations of $u(t)$. The derivation of the discretisation is completed by approximating the integral $\int_0^t f_\theta(\tau, u(\tau))d\tau$ using standard numerical time integration techniques.

4.3 Discretisation of the approximate trajectories

The continuous form of the integral in eqn (25) is not amenable to numerical computation and requires discretisation. In particular, if f_θ is to be represented by a compute graph and learnt by SGD, then the entire loss functional $J(\theta)$ must be represented by a differentiable, directed acyclic compute graph. To achieve this, it is useful to first note that the integral in eqn (25) can be decomposed into a series of integrals over smaller time domains. Consider the trajectories from times 0 to t and 0 to $t + h$:

$$\tilde{u}_\theta(t) := u(0) + \int_0^t f_\theta(\tau, u(\tau))d\tau. \tag{28}$$

Then,

$$\begin{aligned} \tilde{u}_\theta(t+h) &= u(0) + \int_0^t f_\theta(\tau, u(\tau))d\tau + \int_t^{t+h} f_\theta(\tau, u(\tau))d\tau & (29) \\ &= \tilde{u}_\theta(t) + \int_t^{t+h} f_\theta(\tau, u(\tau))d\tau, & (30) \end{aligned}$$

giving the trajectory predicted by f_θ from $\tilde{u}_\theta(t)$ to $\tilde{u}_\theta(t + h)$.

The required discretisation can be completed using standard numerical integration techniques. Numerical integration methods such as Euler, Runge-Kutta and Backwards Differentiation (see [13] for an overview) work, roughly, by assuming some functional form for $f(x)$ and analytically integrating this approximation. Numerical integration methods can be expressed as a function of the integrand evaluated at some finite set of m points $\{x_j\}_{j=1}^m$:

$$\int_a^b f(x)dx \approx G\left(a, b, f, \{x_j\}_{j=1}^m\right). \tag{31}$$

Note that the points $a \leq x_j \leq b$ are defined as a part of the specification of a particular numerical integration scheme. The function to be integrated, f, must be able to be evaluated at each x_j.

The trajectories in eqn (30) can then be approximated with a numerical approximation scheme as in eqn (31):

$$\begin{aligned} \tilde{u}_\theta(t+h) \approx \hat{u}_\theta(t+h) &:= \hat{u}_\theta(t_j) + G\left(t, t+h, f_\theta, \{(\tau_j, u(\tau_j))\}_{j=1}^m\right), && (32) \\ \hat{u}_\theta(0) &:= u(0). && (33) \end{aligned}$$

$\hat{u}_\theta(t)$ refers to a trajectory $\tilde{u}_\theta(t)$ with continuous integrals replaced by approximate numerical integrals. The values τ_j are evaluation points and correspond to the values x_j in eqn (31). In general, the smaller the value of h the greater the accuracy of the approximation. Small values of h, however, increase the computational burden required to compute approximate trajectories.

4.4 ODE inference loss functional for observations at discrete times

For practical problems, observations of $u(t)$ will not be available for all times between 0 and T. Typically, the trajectory $u(t)$ will be known only at a finite set of times $t \in \{t_i\}_{i=1}^N$ so that $u(t)$ is known at $\{u(t_i)\}_{i=1}^N$. The finite set $\{(t_i, u(t_i))\}_{i=1}^N$ will be referred to as 'training data' and can be used to discretise the optimisation problem in eqn (24) by the following approximation:

$$\begin{aligned} \text{Minimise } \tilde{J}(\theta) &:= \frac{1}{N}\sum_{i=1}^{N}\left|u(t_i) - \left(u(0) + \int_0^{t_i} f_\theta(\tau, u(\tau))d\tau\right)\right|^2 && (34) \\ &= \frac{1}{N}\sum_{i=1}^{N}\left|u(t_i) - \tilde{u}_\theta(t_i)\right|^2. && (35) \end{aligned}$$

However, the terms $\tilde{u}_\theta(t)$ must also be replaced by a discretisation, as in eqn (32). Assume that a numerical integration scheme is selected that evaluates the integrand at m points. It is convenient to decompose the trajectory integrals $\tilde{u}_\theta(t)$ into a series of integrals over finite subsets of the training data, t_i to t_{i+p} for the window size $p \in \mathbb{N}$ (typically either m or $m-1$), such that

$$\tilde{u}_\theta(t_{i+p}) = \tilde{u}_\theta(t_i) + \int_{t_i}^{t_{i+p}} f_\theta(\tau, u_\theta(\tau))d\tau. \qquad (36)$$

With reference to eqn (32), this can be further approximated by numerical integration:

$$\hat{u}_\theta(t_{i+p}) = \hat{u}_\theta(t_i) + G\left(t_i, t_{i+p}, f_\theta, \{(\tau_j, u(\tau_j))\}_{j=1}^m\right) \qquad (37)$$

such that the value of $u(\tau_j)$ is known (given the training data) for all evaluation points τ_j, $j \in [1, m]$.

Finally, eqn (37) can be modified by using the known value (from the training data) of $u(t_i)$ in place of $\hat{u}_\theta(t_i)$:

$$\hat{u}(t_{i+p}) := u(t_i) + G\left(t_i, t_{i+p}, f_\theta, \{(\tau_j, u(\tau_j))\}_{j=1}^m\right). \qquad (38)$$

Eqn (35) can be approximated by the discretised loss functional $\hat{J}(\theta)$ by inserting $\hat{u}(t)$:

$$\begin{aligned} \hat{J}(\theta) &:= \frac{1}{N-p}\sum_{i=1}^{N-p}\left|u(t_{i+p}) - \hat{u}(t_{i+p})\right|^2 && (39) \\ &= \frac{1}{N-p}\sum_{i=1}^{N-p}\left|u(t_{i+p}) - \left(u(t_i) + G\left(t_i, t_{i+p}, f_\theta, \{(\tau_j, u(\tau_j))\}_{j=1}^m\right)\right)\right|^2. && (40) \end{aligned}$$

As $\hat{J}(\theta)$ is a discrete approximation to $J(\theta)$, the model inference problem in eqn (26) is approximately solved by minimisation of $\hat{J}(\theta)$ over a training data set:

$$\theta^* = \mathrm{argmin}_\theta\, J(\theta) \approx \mathrm{argmin}_\theta\, \hat{J}(\theta). \tag{41}$$

The inferred ODE model then is $f_{\theta^*}(t, u(t))$.

Note that in the above derivation, loss functionals have been computed for time-dependent models of the form $f(t, u(t))$. In practice, optimisation over a single trajectory will only provide useful estimates of f_θ very close to $(t, u(t))$. To find estimates of f_θ away from those points, one would have to observe multiple trajectories and modify $J(\theta)$ to average over these trajectories. Alternatively, in the autonomous case, where f is of the form $f(u(t))$, one trajectory may be enough to infer f_θ, depending on the number of sampling points available.

4.5 Example using Euler integration

To demonstrate concretely how eqn (40) gives a loss functional discretisation, $\hat{J}(\theta)$, for an ODE model that can be optimised by SGD and Backpropagation, an example using simple numerical integration techniques is discussed in this section. Forward Euler (see [13]) computes an approximation to a dynamical system trajectory time integral as follows ($h > 0$):

$$u(t+h) \approx u(t) + hf(t, u(t)). \tag{42}$$

With reference to eqn (31), Forward Euler is a numerical integration scheme with $m = p = 1$, $\tau_1 = a$ and

$$G\left(a, b, f, \{(a, u(a))\}\right) = |b-a| f(a, u(a)). \tag{43}$$

Forward Euler is a so-called explicit method as the approximation of $u(t+h)$ depends only on functions evaluated at times earlier than $t+h$. Backward Euler, conversely, is an implicit method:

$$u(t+h) \approx u(t) + hf(t+h, u(t+h)). \tag{44}$$

With reference to eqn (31), Backward Euler is a numerical integration scheme with $m = p = 1$, $\tau_1 = b$, and

$$G\left(a, b, f, \{(b, u(b))\}\right) = |b-a| f(b, u(b)). \tag{45}$$

Forward time integration using Backward Euler requires solving a system of equations (typically by Newton-Raphson iterations [13]) as $u(t+h)$ appears on both sides of eqn (44). This is characteristic of implicit integration methods. The choice of when to use explicit or implicit integration methods for simulation of a system depends on the form of the dynamical system to be approximated [13]. Implicit methods are more efficient and accurate for so-called ‘stiff’ problems [10, 11].

However, either method can be used to discretise an ODE into a compute graph representation. For example, assume that $f_\theta(t, u(t))$ is represented by an acyclic compute graph. Then, given training data $\{(t_i, u(t_i))\}_{i=1}^N$, the model inference loss functional, $\hat{J}(\theta)$, in eqn (40) can be approximated using Forward Euler as follows:

$$\hat{J}_F(\theta) := \frac{1}{N-1} \sum_{i=1}^{N-1} \left| u(t_{i+1}) - \big(u(t_i) + |t_{i+1} - t_i| f_\theta(t_i, u(t_i)) \big) \right|^2. \tag{46}$$

Implicit integration schemes can be used in essentially the same way as shown above for Forward Euler. As an example, the Backward Euler scheme in (44) can be used to set the model inference loss functional, $\hat{J}(\theta)$, from eqn (40) as follows:

$$\hat{J}_B(\theta) := \frac{1}{N-1} \sum_{i=1}^{N-1} \left| u(t_{i+1}) - \big(u(t_i) + |t_{i+1} - t_i| f_\theta(t_{i+1}, u(t_{i+1}))\big) \right|^2. \tag{47}$$

Note that for the explicit Euler scheme, as in eqn (46), up to time t_N we can infer f_θ only up to time t_{N-1}. Hence, there is a time lag in the learning which is not observed for the implicit Euler scheme.

The loss functionals in eqns (46) and (47) are trivially differentiable and acyclic (as the values of t_i and $u(t_i)$ are just constants that have been taken from observations) as long as the graph representation of f_θ is differentiable and acyclic. Thus, if f_θ is represented by a differentiable and acyclic compute graph, the loss functionals $\hat{J}(\theta)$ can be optimised by SGD.

4.6 Example using linear multistep integration approximation

More sophisticated integration schemes than Backward or Forward Euler can be used to find a differentiable parametric representation of $\hat{J}(\theta)$. Linear multistep integral approximation schemes are briefly described here as they will be used for the numerical simulations presented in the next section of this paper. Any numerical scheme that is differentiable and representable by a directed acyclic compute graph when inserted into the loss functional could be used. Linear multistep methods are a convenient choice when the training data consists of observations of $u(t)$ that have been sampled at constant frequency.

From [10], Adams-Moulton linear multistep integration of order $s = 2$ can be used to approximate a trajectory of a dynamical system from time a to time $b = a + 2h$ for some $h \in \mathbb{R}$ as follows:

$$\hat{u}(b) = u(a+h) + h\left(\frac{5}{12} f_\theta(b, u(b)) + \frac{2}{3} f_\theta(a+h, u(a+h)) - \frac{1}{12} f_\theta(a, u(a))\right). \tag{48}$$

To derive the loss functional $\hat{J}(\theta)$, assume that training data observations of $u(t)$ are given by $\{t_i, u(t_i)\}_{i=1}^N$ and that the times t_i are evenly spaced such that $t_i = (i-1)h$. Inserting eqn (48) into eqn (40) gives the Adams-Moulton approximate loss functional ($m = 3, p = 2$):

$$\hat{J}_A(\theta) := \frac{1}{N-2} \sum_{i=1}^{N-2} \left| u(t_{i+2}) - \hat{u}(t_{i+2}) \right|^2. \tag{49}$$

Note that the full Adams-Moulton integrator (defined in [10]) could also be used to derive a loss functional that approximates a trajectory discretisation using a series of interpolation points between the observations in the training set. For simplicity, only the method shown above (placing the evaluation points at the values in the training data set) is used in this paper.

5 NUMERICAL ANALYSIS OF THE LORENZ–EMANUEL SYSTEM

5.1 Overview

This section demonstrates the application of the parametric polynomial kernel regression technique to the model inference problem for a dynamical system using the discretisation detailed in the previous section of this paper. Simulations of the Lorenz–Emanuel system (see

§7.1 of [23]) were analysed. This dynamical system consists of N variables, u_i for $1 \leq i \leq N$, arranged periodically such that $u_{N+1} = u_1, u_0 = u_N$ and $u_{-1} = u_{N-1}$. Let the full set of variables be denoted by $u := \{u_i\}_{i=1}^N$. The Lorenz–Emanuel system can be highly chaotic, displaying sensitive dependence on initial conditions. The equations of motion for this system are:

$$\frac{du_i}{dt} = (u_{i+1} - u_{i-2})\, u_{i-1} - u_i + F. \tag{50}$$

For the analysis in this section, the following parameters were adopted:

$$N = 8, \quad F = 5. \tag{51}$$

The parameter F represents an external forcing term that prevents the energy in the system from decaying to zero. The value $F = 5$ was chosen to be high enough to cause sensitive dependence on initial conditions.

5.2 Model inference training data and test description

Model inference was performed given the training data shown in fig 5. The training data was generated using the SciPy solve_ivp method [14] with the 'RK45' algorithm (variable 4th-5th order Runge-Kutta [5]) and sampled at a rate of 1000 samples per time unit for times $t = 0$ to $t = 20$. The initial values for the data were generated by sampling each u_i independently from a normal distribution with mean 0 and standard deviation 3:

$$u_i(t = 0) \sim \mathcal{N}(\mu = 0, \sigma = 3). \tag{52}$$

The performance of the proposed method was tested by resampling new initial conditions from the same distribution in eqn (52) and comparing the outputs from the true simulation to simulations generated using an inferred model. All test simulations were again carried out using the SciPy solve_ivp method with 'RK45' integration [14, 5].

We used the Adams-Moulton loss functional, $\hat{J}_A(\theta)$, in eqn (49) to define the model inference task. The specific form of the inferred models is given in Section 5.3. All models were implemented using Tensorflow [1] and optimised with the Adam variant of SGD (see [15] for implementation details). A fixed optimisation training schedule was adopted in all cases and consisted of three phases, P_1, P_2, P_3. Each phase is described by an ordered pair (I_i, η_i) where I_i is the number of gradient descent iterations for that phase and η_i is the 'learning rate' parameter as in eqn (4). The training schedule adopted was:

$$\{P_1 = (1000, 0.1), P_2 = (2000, 0.01), P_3 = (200, 0.001)\}. \tag{53}$$

It was found that this schedule was sufficient to minimise $\hat{J}_A(\theta)$ to approximately the maximum achievable precision for all models tested.

Note that the integrator used to generate trajectories (RK45) and that used for discretisation of the ODE trajectories (Adams-Moulton) are not the same. This was to demonstrate that any ODE solver can be used to generate simulations from the inferred model.

5.3 Model representation with polynomial linearisations and kernels

To complete the specification of the problem, the basic form of f_θ must be provided. If the form of the dynamical system equations are known beforehand, this information can be

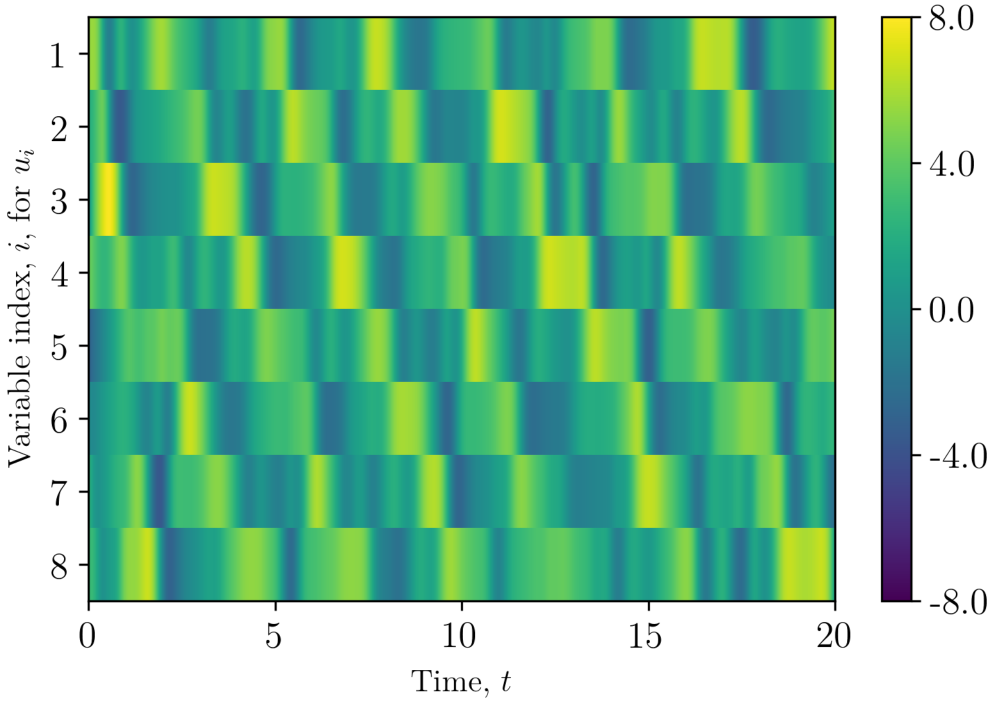


Figure 5: Lorenz–Emanuel system training data, generated using the model defined in eqn (50).

used to simplify the analysis. If no information is available, a search over different types of compute graph architectures must be conducted (as in [24]). For this demonstration, only a polynomial structure is assumed. This is a reasonable assumption that one could make when investigating general interdependent data observations from a dynamical system without any other prior knowledge, as a number of systems have such a structure [23].

For this inference task, the exact form of the polynomial couplings between the various u_i were not provided to the compute graph. Instead, two types of polynomial nonlinearities were tested. First, a linear combination of all second-order polynomial terms that could be constructed using each of the u_i terms was considered, that is, equations of the form

$$\frac{d\hat{u}_i}{dt} = f^i_\theta(u) = \sum_{k=1}^{N}\sum_{j=1}^{k} \alpha^i_{kj} u_k u_j + \beta^i_k u_k + \gamma_i \tag{54}$$

for each $i \in [1, \ldots, N]$. The parameters are $\gamma_i \in \mathbb{R}$, $\beta^i_k \in \mathbb{R}^N$, $\alpha^i_{kj} \in \mathbb{R}$ for $i \in [1, \ldots, N]$, $k = [1, \cdots, N]$, $j = [1, \cdots, k]$. This sort of polynomial is of the traditional form used for polynomial chaos expansions (see [26]).

Second, the parametric polynomial kernel method introduced in this paper and defined in eqn (13) with dimensions $D = E = 8$ was used to represent f_θ. Values of $M = 60, 80$ and 100 were tried to test the effect of this parameter on the accuracy of the results.

5.4 Results

Stochastic Gradient Descent, combined with ODE trajectory discretisation, was successfully applied to model inference for the Lorenz–Emanuel system in eqn (50). Our parametric kernel

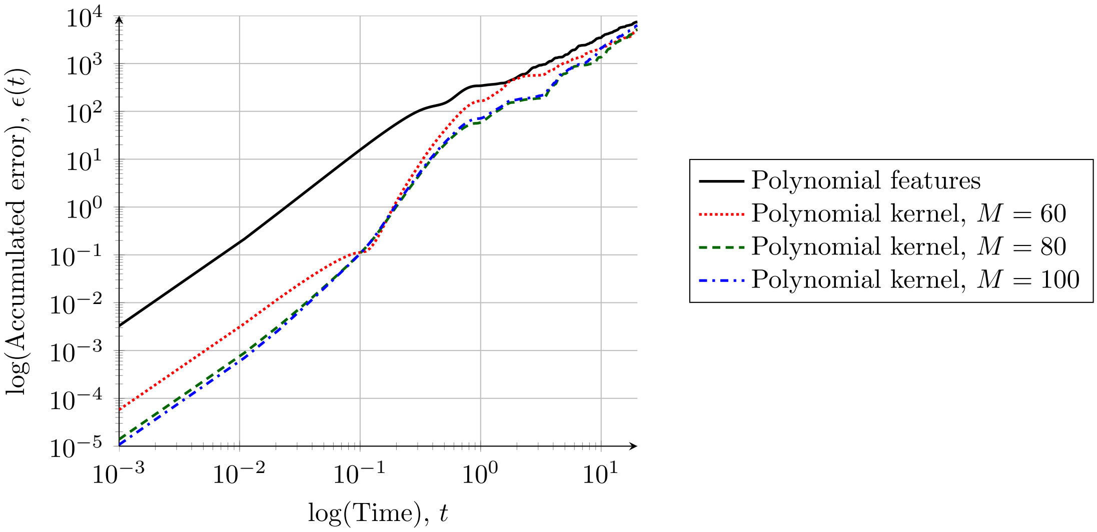


Figure 6: Lorenz–Emanuel system error vs time. Errors are calculated as per eqn (56).

model gave the best accuracy on the inference task. Importantly, the kernel model was able to be tuned to higher accuracies by increasing the number of weights used, M. Although increasing M increases the number of total parameters to be optimised, this trade off may be worthwhile depending on the particular problem.

The performance of the different models is shown in fig 6. The accumulated error, $\epsilon(t)$, was calculated as the sum of squared errors from the true model:

$$\epsilon(t = 0) \quad = \quad 0, \tag{55}$$

$$\epsilon(t + h) \quad = \quad \sqrt{((u(t + h) - \hat{u}(t + h))^2} + \epsilon(t), \tag{56}$$

where $h = 0.001$ (matching the training data sampling rate of 1000 samples per time unit). The errors were calculated for the polynomial feature model in eqn (54) and the polynomial kernel model in eqn (13) for $M = 60$, $M = 80$ and $M = 100$.

From fig 6, the direct polynomial feature mapping had the worst accuracy. The parametric kernel method was able to track the system evolution more accurately. In all cases, the inferred models were able to maintain a small inference error at times up to at least an order of magnitude greater than the training data sampling rate.

Performance on the model inference task for the polynomial kernel method defined by eqn (13) with $M = 100$ is demonstrated in fig 7. This pair of figures shows a comparison between the true model output, $u(t)$, and inferred model output, $\hat{u}(t)$. From fig 7, it can be seen that the overall structure of the equations is captured by the inferred model. Due to the chaotic nature of the system being analysed, once a few errors accumulate, the true and inferred models diverge rapidly.

5.5 Discussion

The parametric polynomial kernel method was able to infer the hidden ODE model with good accuracy given a fixed set of training data. The accumulated errors grow quickly with time. This is reasonable considering the chaotic nature of the Lorenz–Emanuel system. A more mathematically rigorous stability analysis of the numerical scheme would be interesting but is beyond the scope of this paper. A number of possible variations on the numerical example presented could be analysed in future work. For instance, the type of integration method used, the sampling rate of the data, and the effect of different amounts of training data would all be interesting to investigate.

6 CONCLUSIONS

This paper presented a parametric form of polynomial kernel regression, as well as numerical case studies. In particular, the proposed method was applied to the model inference problem for a chaotic dynamical system. Our parametric polynomial kernel method was able to harness the power of kernelised regression without the cubic computational complexity typically incurred by nonparametric polynomial regression, thereby avoiding the curse of dimensionality. Although the method was successfully applied to a test problem, more work will be required to fully understand how best to apply parametric polynomial kernels to real world (rather than simulated) data. As is the case in all regression models, some form of regularisation would need to be included to address overfitting and observational noise.

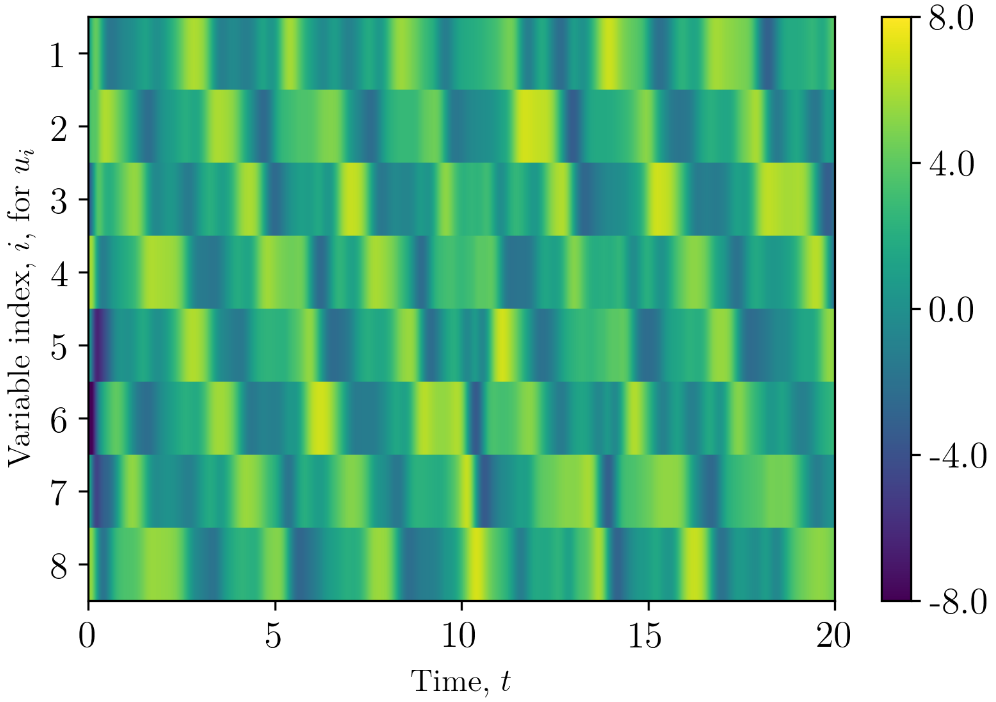
It was assumed for the analysis in this paper that it was known a priori that only certain polynomial couplings are present. Using the wrong polynomial order in the model expansion was found to cause convergence difficulties. This is also the case in nonparametric kernel regression (see [18] and the example in fig 2). As such, this is not considered a serious limitation of the method in that it is possible to test a few different sets of model forms when attempting to find a good fit to a data set. Bayesian model selection methods could be applied to formally assess the quality of different polynomial kernel model dimensions.

It is worth noting that direct projection onto polynomial features was found to perform poorly compared to the polynomial kernel method. Although stochasticity was not considered in this paper, it is quite possible that this finding will impact standard techniques frequently employed for Uncertainty Quantification. A kernel representation of the type introduced in this paper applied to Gaussian and other stochastic features may be useful for improving standard polynomial chaos methods (which are described in [26]).

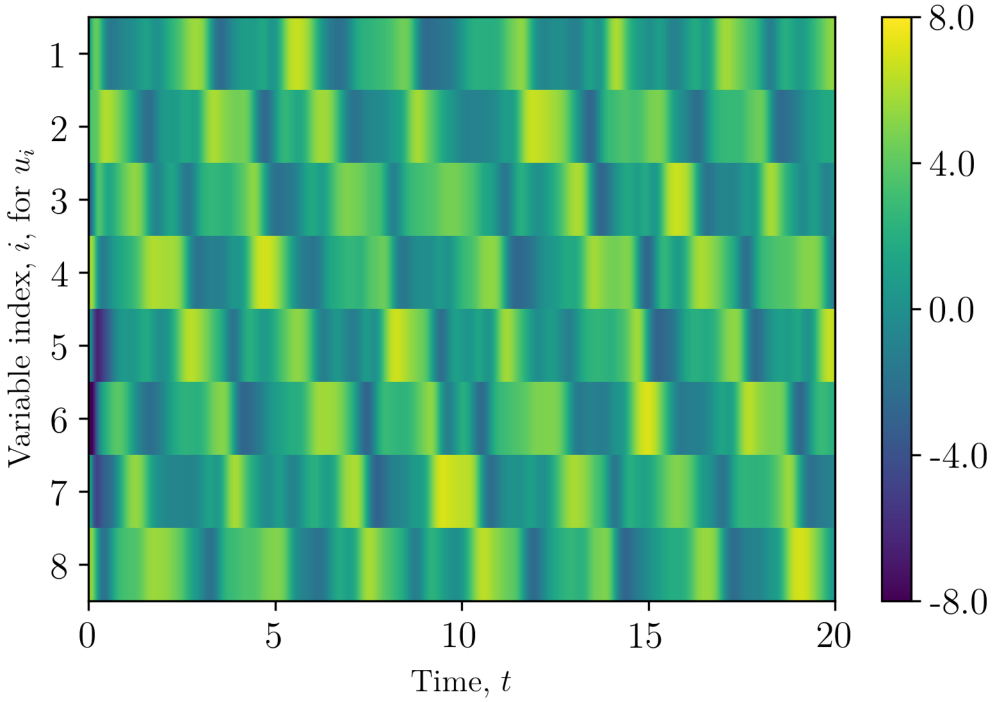
The search for effective compute graph architectures remains a problem that plagues all methods attempting to learn hidden function structures without inserting large amounts of prior knowledge into the inverse problem. Scaling to very high-dimensional problems would be an interesting challenge. Given the partial decoupling from the curse of dimensionality that gradient descent methods can provide, it is hoped that the techniques presented in this paper would be suitable for model inference on large scale dynamical systems in the future.

7 ACKNOWLEDGEMENTS

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, grant agreement No 757254 (SINGULARITY) and a Lloyds Register Foundation grant for the Data-Centric Engineering programme at the Alan Turing Institute.



(a) Data trace from true model, $u(t)$.



(b) Data trace from inferred parametric polynomial kernel model, $\hat{u}(t)$, with $M = 100$.

Figure 7: Comparison of output traces for the Lorenz–Emanuel system, defined in eqn (50): (a) true system simulation, $u(t)$, and (b) most accurate inferred model, $\hat{u}(t)$. The inferred model structure is given by the parametric polynomial kernel in eqn (13) for $M = 100$.

REFERENCES

[1] M. Abadi, A. Agarwal, P. Barham & others, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. `http://www.tensorflow.org/`, 2015.

[2] J. Adler, O. Öktem, Solving ill-posed inverse problems using iterative deep neural networks, *Inverse Problems*. **33(12)**, 2017.

[3] C.M. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[4] B. Carpenter, M.D. Hoffman, M. Brubaker, D. Lee, P. Li, M. Betancourt, The Stan math library: Reverse-mode automatic differentiation in C++. *ArXiv*, `arXiv:1509.07164`, 2015.

[5] R. Dormand, P.J. Prince, A family of embedded Runge-Kutta formulae, *Journal of Computational and Applied Mathematics*. **6(1)**, 19–26, 1980.

[6] P. Eberhard, C. Bischof, Automatic differentiation of numerical integration algorithms, *Neural Networks*. **68**, 717–732, 1999.

[7] S. Epp, *Discrete Mathematics with Applications*. Cengage Learning, 2010.

[8] I. Goodfellow, Y. Bengio, A. Courville *Deep Learning*. MIT Press, 2016.

[9] W.W. Hager, Updating the Inverse of a Matrix, *SIAM Review*, **31(2)**, 221–239, 1989.

[10] E. Hairer, S.P. Nørsett, G. Wanner *Solving Ordinary Differential Equations I: Nonstiff problems*. Springer Science & Business Media, 1993.

[11] E. Hairer, S.P. Nørsett, G. Wanner *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*. Springer Science & Business Media, 1996.

[12] R.A. Horn, C.R. Johnson, *Matrix Analysis*. Cambridge University Press, 2012.

[13] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*. Cambridge University Press, 2009.

[14] E. Jones, T. Oliphant, P. Peterson, *SciPy: Open source scientific tools for Python*. `http://www.scipy.org/`, 2018.

[15] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*. Springer Verlag, 2015.

[16] H.G. Matthies, E. Zander, B.V. Rosič, A. Litvinenko, O. Pajonk, Inverse Problems in a Bayesian Setting, *Computational Methods for Solids and Fluids*. **41**, 245–286, 2016.

[17] J.D. Meiss, *Differential Dynamical Systems, Revised Edition*. SIAM, 2017.

[18] K.P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[19] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in Python, *Journal of Machine Learning Research*. **12**, 2825–2830, 2011.

[20] L.B. Rall, Automatic Differentiation: Techniques and Applications, *Lecture Notes in Computer Science*. **120**, 1981.

[21] S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson, 2016.

[22] M. Schmidt, H. Lipson, Distilling free-form natural laws from experimental data, *Science*. **324**, 81–85, 2009.

[23] J.C. Sprott, *Elegant Chaos: Algebraically Simple Chaotic Flows*. World Scientific Publishing Company, 2010.

[24] K.O. Stanley, R. Miikkulainen, Evolving Neural Networks through Augmenting Topologies, *Evolutionary Computation*. **10(1)**, 99–127, 2002.

[25] A.M. Stuart, Inverse problems: A Bayesian perspective, *Acta Numerica*. **19**, 451–559, 2005.

[26] B. Sudret, *Uncertainty propagation and sensitivity analysis in mechanical models: Contributions to structural reliability and stochastic spectral methods*, Habilitation à diriger des recherches, Université Blaise Pascal, 2007.

[27] J.R. Taylor, *Classical Mechanics*. University Science Books, 2005.

[28] K. Vu, J.C. Snyder, L. Li, M. Rupp, B.F. Chen, T. Khelif, K.-R. Müller, K. Burke, Understanding kernel ridge regression: Common behaviors from simple functions to density functionals, *International Journal of Quantum Chemistry*. **115(16)**, 1115–1128, 2015.

[29] P. Werbos, Generalisation of Backpropagation with application to a recurrent gas market model, *Neural Networks*. **1(4)**, 339–356, 1988.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

CONCRETE GRAVITY DAMS FE MODELS PARAMETERS UPDATING USING AMBIENT VIBRATIONS

G. Sevier[1], A. De Falco[2]

[1] Dept. of Civil and Industrial Engineering, University of Pisa
Largo Lucio Lazzarino, 1 – 56122 Pisa (Italy)
giacomo.sevieri@unifi.it

[2] Dept. of Energy, Systems, Territory and Constructions Engineering, University of Pisa
Largo Lucio Lazzarino, 1 – 56122 Pisa (Italy)
a.defalco@ing.unipi.it

Abstract

Most of the dams around the world were designed before the introduction of seismic regulations and without concerns about their dynamic behavior. The failure of a large gravity dam might have catastrophic effects putting at risk a large number of human lives, not counting the considerable economic consequences. Since there are no case histories of concrete gravity dams failed after seismic events, numerical models assume great importance for the evaluation of the seismic performance of such structures or to control them within a SHM framework. Several different sources of uncertainty are involved in numerical models of concrete gravity dams, their effects can be reduced by exploiting all available information about the structure. Ambient vibrations are an important source of information because they can be used to characterize the dynamic behavior of the structure. In this paper, a procedure, defined in the Bayesian framework, which allows calibrating the dynamic model parameters using ambient vibration is presented. Ambient vibrations are used to determine the modal characteristics of the system, by applying the Operational Modal Analysis (OMA), which are used in the updating process. The use of meta models based on the general Polynomial Chaos Expansion (gPCE) and a modified version of Markov Chain Monte Carlo (MCMC) allows both considering the SSI in the numerical model of the dam and solving the problem of coherence between experimental and numerical modes. Finally, the proposed procedure is applied to the case of an Italian dam showing the applicability to real cases.

Keywords: concrete dams, gPCE, OMA, Bayesian Updating, MCMC, UQ.

1 INTRODUCTION

Concrete dams are fundamental infrastructures due to their use for energy production, floods control and industrial supply. However, the largest part of existing concrete dams located in developed countries have been designed by following only static concepts. In light of the revaluation of some areas as seismic and the higher reliability levels required by the community, nowadays a large number of existing concrete dams are outdated [1]. Therefore, the evaluation and the mitigation of the seismic risk of concrete dams is a task of primary importance for our society [2].

Structural Health Monitoring (SHM) is a powerful tool both to control the structural behaviour (Diagnosis phase) and to predict the remaining life expectancy of the dam (Prognosis phase) [3]. In this context, one or more Quantities of Interest (QI) of the dam are monitored in order to detect abnormal behaviour of the structure. Predictive models, which reproduce the selected QI, must be defined in order to forecast the dam behaviour considering the effects of the uncertainties and those of the errors [4]. Numerical models are the only way to investigate the dynamic behaviour of concrete dams. Indeed, as discussed by Hall [5] there are no case histories on concrete dams failed after seismic events.

Numerical models commonly used in dam engineering field are particularly complex due to the presence of three different interacting domains, dam, basin and soil. In this context, De Falco et al. [6–8] showed the influence of modelling strategy on the solution of dynamic analyses of concrete dams. Once a deterministic model has been defined, the uncertainties related to the model parameters lead to a biased result, which must be quantified (UQ) and reduced as much as possible in order to perform a reliable numerical model of the dam. All available information must be used for this purpose. In particular, the observations recorded by the monitoring system can be used to calibrate the model parameters and not only to control the health state of the structure.

The largest part of existing concrete dams is equipped by static monitoring systems, which record the displacements of few points on the structure and the environmental conditions, i.e. reservoir level, air and water temperatures. De Falco et al. [9] showed how to use information coming from the static monitoring system to update the mechanical parameters of the model materials in a Bayesian framework. Despite static SHM can provide useful information both for the structural control during normal operations and for the calibration of the model parameters, dynamic monitoring systems seem to be more appropriate when the seismic behaviour of a structure is investigated. With the aim to perform a permanent dynamic SHM system the only practicable choice is the registrations of ambient vibrations [10]. The observations recorded by a dynamic monitoring system based on ambient vibrations can be directly used in the updating process or elaborated through Operational Modal Analysis (OMA) [11] in order to obtain the modal characteristics of the system, i.e. frequencies and mode shapes. In this latter case, the updating process is defined with regard to the modal characteristics of the system, which are then the QI of the problem. The use of modal characteristics as QI is the commonly adopted approach in civil engineering field, because it leads to a simplification from the numerical point of view. Indeed, in this way, modal analyses are used within the updating process, instead of transient ones needed in the case of the direct use of ambient vibrations.

In dam engineering field dynamic SHM systems are very rare, even though some applications are available in the literature [12]. Most of the available research works aim to verify the feasibility of the installation of dynamic monitoring systems on concrete dams, but they do not discus the use of the observations for structural control or model calibration purposes. The numerical complications in the modal analysis of concrete dams, related to the SSI, has led to a broader use of static SHMs rather than dynamic ones.

In this paper, the effects of the uncertainties related to the mechanical parameters of the materials on the dynamic behaviour of concrete dams are investigated and discussed. Subsequently, the hierarchical Bayesian procedure for the updating of dynamic model parameters, proposed by Sevieri et al. [13], is applied in order to verify its effect on real cases. The numerical problems, discussed next, are solved by using a modified version of MCMC which allows both selecting and reordering the numerical mode shapes. A large concrete gravity dam, located in the centre of Italy, is used to investigate these two topics.

2 DAM DESCRIPTION

In this work, a large Italian concrete gravity dam is used as benchmark for the quantification of the effects of the epistemic uncertainty on the modal behaviour, and for the application of the hierarchical procedure to reduce them. The dam, showed in Figure 1, is composed by 26 monoliths for a total crest length around 450 m, and a maximum height of 65 m. The monoliths are connected each other through vertical contraction joints, which show an opening-closing movement during the year. This behaviour, recorded by the static monitoring system, is related to the variation of the environmental conditions, and in particular that of temperatures. Despite this movement has quasi-static nature, and then it cannot be directly used as source of information for dynamic properties updating, it must be considered in the updating procedure.

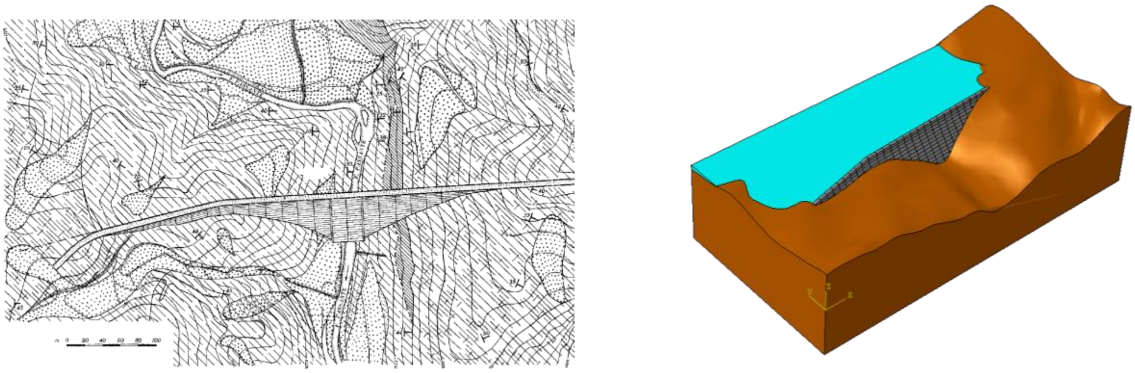


Figure 1: Dam drawing and FE model.

The mechanical parameters of the materials have been deduced from the experimental campaigns conducted in the past. The values of specific weight ρ, Young modulus E, Poisson's ratio υ, compressive and tensile strength, f_t and f_c, of the concrete and the soil (subscript C and S, respectively) are reported in Table 1.

	ρ_C [kg/m^3]	E$_C$ [MPa]	ν_C	$f_{t,C}$ [MPa]	$f_{c,C}$ [MPa]	ρ_S [kg/m^3]	E$_S$ [MPa]	ν_S	$f_{t,S}$ [MPa]	$f_{c,S}$ [MPa]
mean	2500.0	25000.0	0.20	1.85	15.3	2600.0	15000.0	0.22	1.7	51.2
s. d.	87.5	5875.0	0.069	0.629	3.443	725.4	7185.0	0.105	0.613	19.661

Table 1: Mechanical parameters of the materials.

3 UQ IN THE MODAL ANALYSIS OF A CONCRETE DAM

In this section the effect of the uncertainties related to the mechanical parameters of the materials on the modal characteristics of the dam are investigated. There are only few research works on the Uncertainty Quantification (UQ) in dam engineering field [14,15], but none of them addresses the problem of the concrete gravity dam modal characteristics considering the SSI and the FSI. FE models which consider SSI and FSI are characterized by a high computational burden, so the use of probabilistic procedure for UQ could be prohibitive. In this application, the computational burden is strongly reduced by using the general Polynomial Chaos Expansion (gPCE) [16,17] to approximate both numerical frequencies $\boldsymbol{f}^{\mathrm{FEM}}$ and mode shapes $\boldsymbol{\Phi}^{\mathrm{FEM}}$. Only the uncertainty related to the elastic parameters are considered in this application due to the elastic nature of the modal analysis. The elastic parameters of the materials are collected in $\boldsymbol{\theta}_{\mathrm{el}}$, while deterministic measurable variables, e.g. the basin level, are collected in $\mathbf{x}$, that is $\boldsymbol{f}^{\mathrm{FEM}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ and $\boldsymbol{\Phi}^{\mathrm{FEM}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$. The uncertain output of the FEA can be described in a probabilistic space defined by the triplet $(\Omega,\mathfrak{F},\mathbb{P})$: where Ω is the space of all events, $\mathfrak{F}$ is the σ-algebra and $\mathbb{P}$ the probability measure. Assuming that $\boldsymbol{f}^{\mathrm{FEM}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ and $\boldsymbol{\Phi}^{\mathrm{FEM}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ are smooth enough to be represented in terms of simple random variables $\boldsymbol{\theta}_{\mathrm{el}}(\omega)$ corresponding to the Askey scheme [18], they can be approximated through the gPCE,

$$\begin{aligned}\hat{\boldsymbol{f}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}) &= \sum_{\alpha\in\mathbf{I}} f^{(\alpha)}\psi_{(\alpha)}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})\\ \hat{\boldsymbol{\Phi}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}) &= \sum_{\beta\in\mathbf{J}} \phi^{(\beta)}\gamma_{(\beta)}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}).\end{aligned} \tag{1}$$

In the previous equations $\hat{\boldsymbol{f}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ and $\hat{\boldsymbol{\Phi}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ are the gPCE approximations, $\psi_{(\alpha)}$ and $\gamma_{(\beta)}$ are the multivariate orthogonal polynomials with finite multi-index sets $\mathbf{I}$ and $\mathbf{J}$, while $f^{(\alpha)}$ and $\phi^{(\beta)}$ are the polynomials coefficients. Assuming that the dam concrete and the foundation soil are isotropic and heterogeneous, their elastic tensors $\mathbb{C}$ are fully described by the bulk modulus K and the shear modulus G [19]. A parametrization of the forward problem in K and G is more convenient than the use of Young modulus E and Poisson's ratio υ, because $\mathbb{C}$ is linear in K and G. The prior distributions of K and G are defined starting from Table 1 by assuming them log-normally distributed as reported in Table 2.

	K_C [MPa]	G_C [MPa]	K_S [MPa]	G_S [MPa]
distribution	LN	LN	LN	LN
mean	14880.0	10424.0	19210.0	10446.0
s. d.	5824.3	2520.5	19590.0	5203.0

Table 2: Prior distributions of $\boldsymbol{\theta}_{\mathrm{el}}$.

The only measurable variable in this application is the basin level, which oscillates between 29 m and 63 m, which are respectively the minimum and maximum regulation level. The opening-closing behavior of the vertical contraction joints can be investigated in modal analysis by considering two limit cases:

- the vertical joints are completely closed, then the contacts between monoliths are modeled as "bonded";
- the vertical joints are decompressed and by assuming that the monoliths can have relative displacements, the contacts are modeled as "frictionless".

In both cases the contacts are defined as relationship among two surfaces, then the two FE models have the same numbers of elements and nodes. Starting from the orography of the soil and the structural drawings of the dam, a 3D model of the system dam-soil-basin was imported from a CAD program to ABAQUS® v 6.14 [20]. The FE model (Figure 2) is composed by 40638 quadratic tetrahedral mechanical elements C3D10 for the soil, 14397 quadratic tetrahedral mechanical elements C3D10 for the dam body, 28707 linear tetrahedral acoustic elements AC3D4 for the basin and 1550 linear hexahedral one-way infinite elements as boundary conditions for the soil domain. An acoustic impedance is placed at the end of the reservoir in order to avoid the reflection of incident waves [8].

The solutions are calculated for different sets of the mechanical parameters, whose values are sampled from the prior distributions. They are used to train the gPCE of the "bonded" case, i.e. $\hat{\boldsymbol{f}}_{\mathrm{b}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ and $\hat{\boldsymbol{\Phi}}_{\mathrm{b}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$, and the "frictionless" one, i.e. $\hat{\boldsymbol{f}}_{\mathrm{f}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$ and $\hat{\boldsymbol{\Phi}}_{\mathrm{f}}(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}})$.

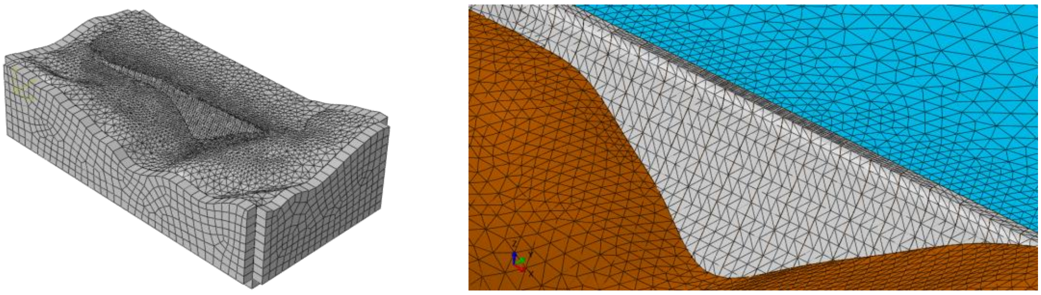


Figure 2: Mesh of the FE model.

The outputs of 350 analyses, for each model, are used to determine the gPCE coefficients, by using the approach proposed by Rosić and Matthies [16], while the polynomial expansion degrees are selected in order to minimize the errors in terms of mean and variance. The attention has been focused on the frequency range 2-20 Hz, where the fundamental modes of the system can be found. The presence of the SSI leads to a large number of numerical modes related to the soil mass. In this work, we refer to this issue as "coherence problem of the numerical modes", and in the context of the forward problem the order of numerical modes could change, due to the variation of the mechanical parameters set.

In this paper, the coherence problem has been tackled by reordering three fundamental numerical modes before they are used in the gPCE coefficients calculation. The Modal Assurance Criterion (MAC) [21] is used for this purpose. Let's consider two mode shapes $\boldsymbol{\phi}_{\mathrm{i}}$ and $\boldsymbol{\phi}_{\mathrm{j}}$, the MAC coefficient which allows measuring the difference between them is defined as

$$\mathrm{MAC}(i,j)=\frac{\left|\boldsymbol{\phi}_i^T\boldsymbol{\phi}_j^*\right|^2}{\left(\boldsymbol{\phi}_i^T\boldsymbol{\phi}_i^*\right)\left(\boldsymbol{\phi}_j^T\boldsymbol{\phi}_j^*\right)}, \tag{2}$$

where T and * indicate the transposed vector and the complex conjugated vector respectively. The MAC coefficient is always a real number, ranging from 0, in the case of no correlation, to 1 in the case of full correlation.

Only the modes related to the dam body are significant from the updating point of view, because experimental modes are usually recorded on the structure. In this paper, the 3 first modes which mobilize the largest amount of dam mass (Figure 3) are chosen as reference for the forward problem.

Hermite polynomials are used as basis functions, the relative relationships between errors and expansion degree are shown in Figure 4. In the end, a 5th order expansion degree is chosen for both frequencies and mode shapes, in order to have a small error both in terms of mean values and variances.

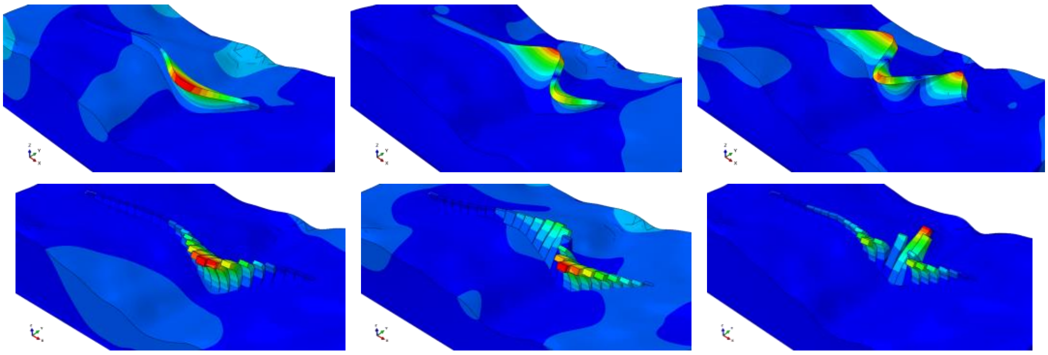
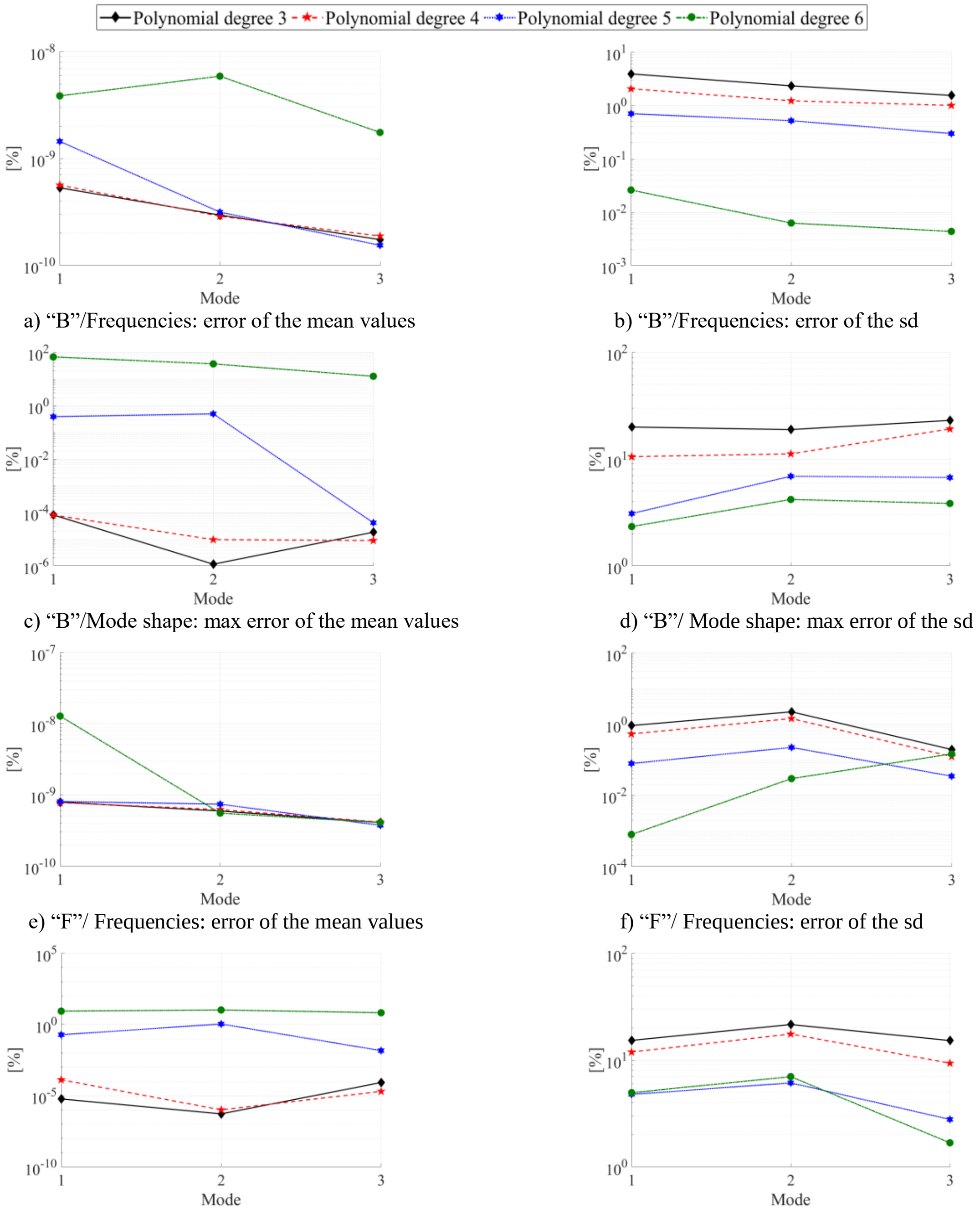


Figure 3: Reference modes for the UQ of bonded model (first line) and frictionless model (second line).



a) “B”/Frequencies: error of the mean values

b) “B”/Frequencies: error of the sd

c) “B”/Mode shape: max error of the mean values

d) “B”/ Mode shape: max error of the sd

e) “F”/ Frequencies: error of the mean values

f) “F”/ Frequencies: error of the sd

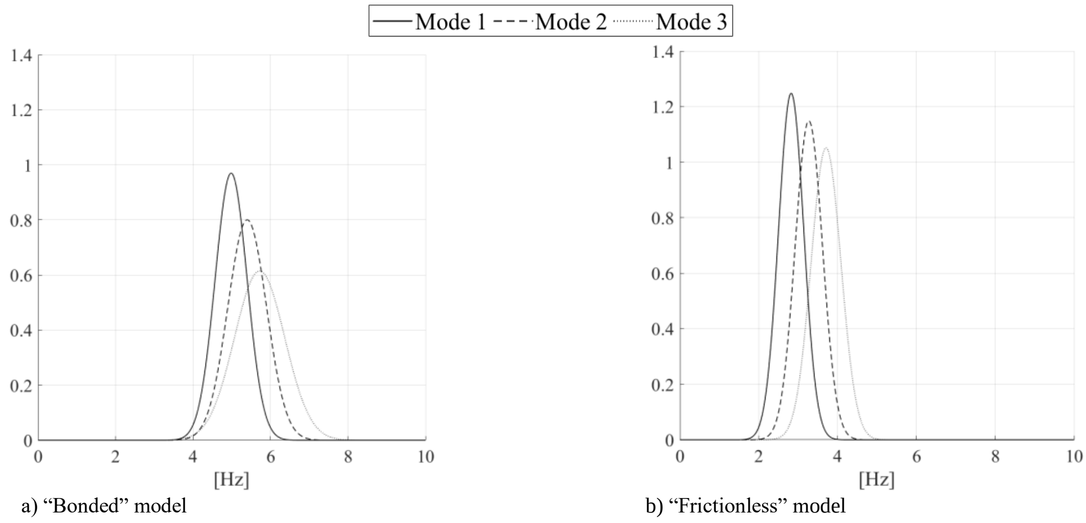
g) “F”/Mode shape: max error of the mean values

h) “F”/Mode shape: max error of the sd

Figure 4: Meta model errors. B = bonded model; F = frictionless model.

Figure 5 shows the distributions of the frequencies of both models. In the “bonded” case the mean values of the first three frequencies are in the range 5-6 Hz, while in the “frictionless” one they are in the range 3-4 Hz. The lack of stiffness of the “frictionless” model, due to the absence of the interaction among monoliths in U-D direction, leads to smaller frequencies values. In both cases the standard deviations increase toward higher frequency and they are relatively smaller in the “frictionless” case.

The gPCE based meta models $\hat{\boldsymbol{f}}_{\mathrm{b}}\left(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}\right)$, $\hat{\boldsymbol{\Phi}}_{\mathrm{b}}\left(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}\right)$, $\hat{\boldsymbol{f}}_{\mathrm{f}}\left(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}\right)$ and $\hat{\boldsymbol{\Phi}}_{\mathrm{f}}\left(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}\right)$ are used instead of the FE models to solve the inverse problem. In this way, the computational burden is strongly reduced, thus making possible the solution of the inverse problem without needing High Performance Computing (HPC).



a) “Bonded” model

b) “Frictionless” model

Figure 5: Distributions of the first three frequencies.

4 HIERARCHICAL BAYESIAN PROCEDURE FOR DYNAMIC MODEL PARAMETERS UPDATING

In this Section, the procedure proposed by Sevieri et al. [13] for the dynamic model parameters updating is applied to the case of study. The procedure, defined in a hierarchical Bayesian framework, allows solving the inverse problem by using experimental modal characteristics of the system determined through the elaboration of ambient vibrations with OMA.

Let’s consider the meta models previously defined (Section 3), the relationships between the *i*-th observation of the *k*-th experimental frequency $f_{\mathrm{k,i}}$ and the corresponding numerical prediction $\hat{f}_{k,i}$ is expressed by a multi-variate additive probabilistic model [22]

$$\ln\left(f_{k,i}\left(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}},\boldsymbol{\Sigma}_{f}\right)\right)=\ln\left(\hat{f}_{k,i}\left(\mathbf{x},\boldsymbol{\theta}_{\mathrm{el}}\right)\right)+\sigma_{f_k}\varepsilon_{f_{k,i}}. \tag{3}$$

All entries in Equation 3 have been already defined except for $\sigma_{f_k}\varepsilon_{f_{k,i}}$ which is the error term composed by random variables normally distributed $\varepsilon_{f_{k,i}}$ and their standard deviations σ_{f_k}, while $\boldsymbol{\Sigma}_f$ is the covariance matrix. The logarithmic function is used to stabilize the variance and to satisfy the homoskedasticity assumption [23].

Let’s consider *q* modes of a system characterized by *m* dynamic d.o.f. ($q \leq m$). The corresponding mode shapes matrix $\boldsymbol{\Phi}=\left[\boldsymbol{\phi}_1,\ldots,\boldsymbol{\phi}_k,\ldots,\boldsymbol{\phi}_q\right]^{\mathrm{T}}$ (*m* x *q* dimension), can be reorganized in only one column vector, thus obtaining $\boldsymbol{\phi}^{\mathrm{total}}$ with dimension *m·q* x *1*. Finally, by defining a global index *h* such that $1 \le h \le m \cdot q$, the relationship between the *i*-th observation of the *h*-th component of $\boldsymbol{\phi}^{\text{total}}$, i.e. $\phi_{h,i}$, and its corresponding numerical simulation $\hat{\phi}_{h,i}$ can be expressed as

$$\phi_{h,i}\left(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\boldsymbol{\Sigma}_{\phi}\right)=\hat{\phi}_{h,i}\left(\mathbf{x},\boldsymbol{\theta}_{\text{el}}\right)+\sigma_{\phi_h}\varepsilon_{\phi_{h,i}}, \tag{4}$$

where $\boldsymbol{\Sigma}_{\phi}$ is the covariance matrix and $\sigma_{\phi_h}\varepsilon_{\phi_{h,i}}$ is the error term composed by a normally distributed random variable $\varepsilon_{\phi_{h,i}}$ and its standard deviation σ_{ϕ_h}.

Despite the use of the gPCE based meta models, the computational burden could be still very high due to the large number of random variables. Indeed, both the elastic parameters $\boldsymbol{\theta}_{\text{el}}$ and the terms of the covariance matrices, $\boldsymbol{\Sigma}_f$ and $\boldsymbol{\Sigma}_{\phi}$, are updated. However, by assuming that only the components of a same mode are correlated, the number of variables considerably decreases, that is

$$\begin{aligned}\boldsymbol{\Sigma}_f &= \begin{bmatrix}\sigma^2_{f_1} & 0 & \cdots & 0\\ 0 & \sigma^2_{f_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \sigma^2_{f_q}\end{bmatrix}\\ \boldsymbol{\Sigma}_{\phi} &= \begin{bmatrix}\left[B_1\right] & & & \\ & \left[B_2\right] & & \\ & & \ddots & \\ & & & \left[B_q\right]\end{bmatrix}.\end{aligned} \tag{5}$$

In this way, $\boldsymbol{\Sigma}_{\phi}$ assumes a block form, as indicated in Equation 5, where each block indicates the covariance matrix of each mode, otherwise the terms are zero. The computational burden could be further reduced by assuming a covariance function for each block $\left[B_k\right]$. In this application, two components ϕ_r and ϕ_s of the same mode *k* are correlated through an exponential covariance function of the Euclidean distance d_{ϕ_r,ϕ_s}

$$\text{COV}\left(\phi_r,\phi_s\right)=\frac{1}{\lambda_k}\exp\left(-w_k d_{\phi_r,\phi_s}\right). \tag{6}$$

In this way, the terms of each block $\left[B_k\right]$ are described by two coefficients λ_k and w_k, which are respectively collected in the vectors $\boldsymbol{\lambda}$ and $\mathbf{w}$.

The procedure proposed by Sevieri et al. [13] is based on a hierarchical Bayesian model [24] with two levels. This particular architecture of the Bayesian process allows updating both mean values and standard deviations of the random variables and inserting information on more than one level. These two features are particularly advantageous in the prognosis phase of a dynamic SHM. The first level is represented by the hyper-parameters Ξ_{el}, i.e. mean values and standard deviations, of the elastic random variables $\boldsymbol{\theta}_{\text{el}}$. Whereas, the second level is composed by the elastic parameters $\boldsymbol{\theta}_{\text{el}}$ themselves and the terms of the covariance matrix $\boldsymbol{\Sigma}_f$, in the case of frequencies, or the coefficients $\boldsymbol{\lambda}$ and $\mathbf{w}$, in the case of mode shapes. For the sake of simplicity, let’s collect the parameters of the second level in the vector $\boldsymbol{\Theta}$. Once a set of new observations $\mathbf{y}$ is available, the prior distribution $p(\boldsymbol{\Theta},\Xi_{\text{el}})=p(\boldsymbol{\Theta}\,|\,\Xi_{\text{el}})\,p(\Xi_{\text{el}})$ can be updated through the likelihood function $L(\mathbf{x},\boldsymbol{\Theta},\Xi_{\text{el}}\,|\,\mathbf{y})$, thus obtaining the posterior distribution $p(\boldsymbol{\Theta},\Xi_{\text{el}}\,|\,\mathbf{y})$

$$p(\boldsymbol{\Theta},\Xi_{\text{el}}\,|\,\mathbf{y})=\kappa L(\mathbf{x},\boldsymbol{\Theta},\Xi_{\text{el}}\,|\,\mathbf{y})\,p(\boldsymbol{\Theta}\,|\,\Xi_{\text{el}})\,p(\Xi_{\text{el}}). \tag{7}$$

In the previous Equation, κ is the normalizing factor. By exploiting the large amount of data, which will be available due to the integration of the present procedure within a SHM system, and the Central Limit Theorem, the likelihood function can be written as

$$L(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\Xi_{\text{el}},\boldsymbol{\Sigma}_f)\propto\prod_{i=1}^{l}\frac{\exp\left[-\frac{1}{2}\mathbf{r}_i^{f^T}(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\Xi_{\text{el}})\boldsymbol{\Sigma}_f^{-1}\mathbf{r}_i^f(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\Xi_{\text{el}})\right]}{\sqrt{|2\pi\boldsymbol{\Sigma}_f|}}, \tag{8}$$

in the case of frequencies, and as

$$L(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\Xi_{\text{el}},\boldsymbol{\Sigma}_\phi)\propto\prod_{i=1}^{l}\frac{\exp\left[-\frac{1}{2}\mathbf{r}_i^{\phi^T}(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\Xi_{\text{el}})\boldsymbol{\Sigma}_\phi^{-1}\mathbf{r}_i^\phi(\mathbf{x},\boldsymbol{\theta}_{\text{el}},\Xi_{\text{el}})\right]}{\sqrt{|2\pi\boldsymbol{\Sigma}_\phi|}}, \tag{9}$$

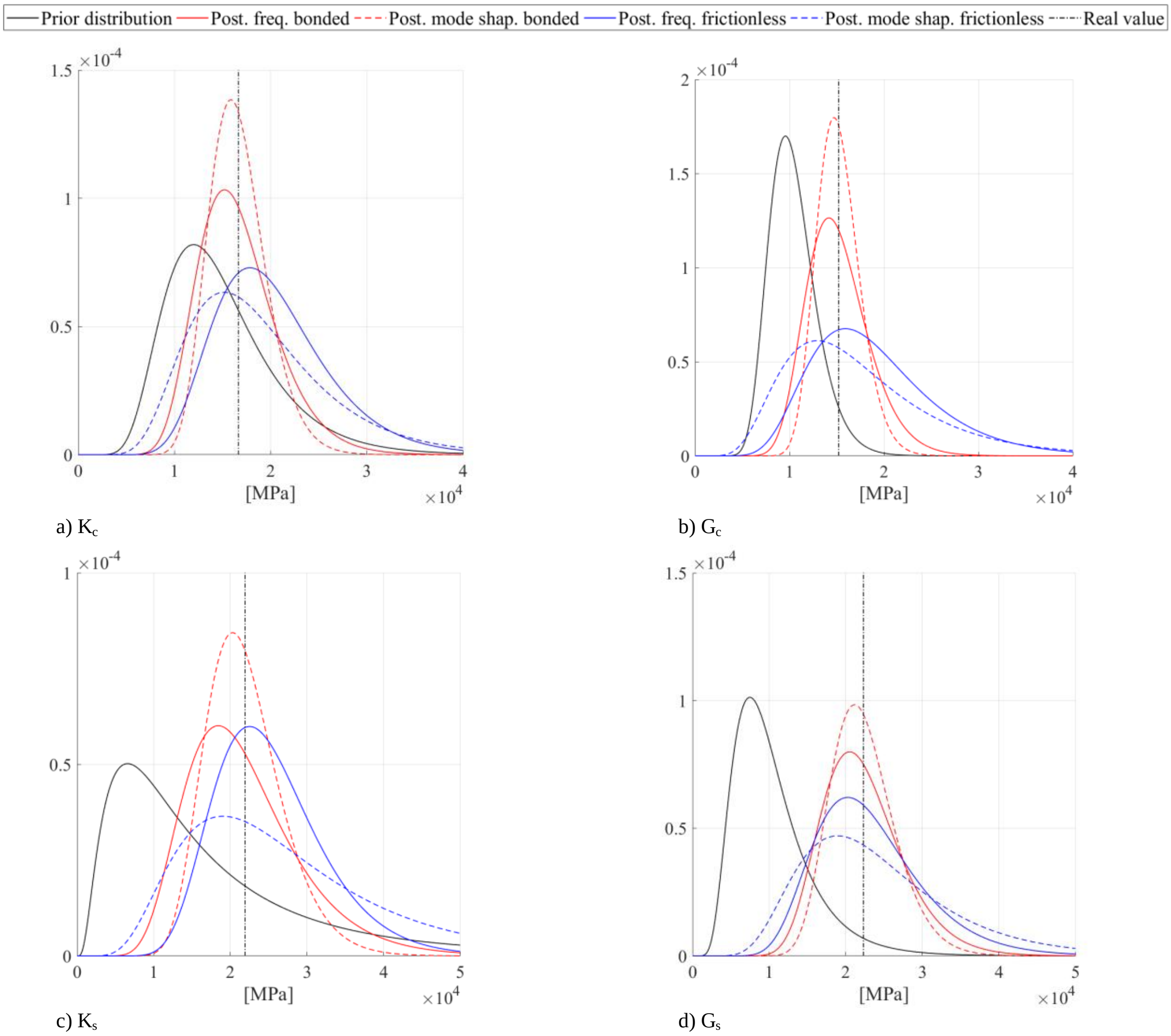
in the case of mode shapes. All entries in Equation 8 and 9 have been already defined except for $\mathbf{r}_i^f$ and $\mathbf{r}_i^\phi$ which are respectively the residuals of the frequencies and of the mode shapes. The residuals are the difference between the *i*-th observation and the corresponding prediction.

In the context of inverse problem, the coherence problem, previously mentioned (Section 3), is faced by modifying the numerical algorithm Markov Chain Monte Carlo (MCMC) [25] by introducing a reordering step based on the MAC matrix. More specifically, once the deterministic model, or its approximation, is solved (*i*-th step), the resulting numerical mode shapes are used with the experimental observations to calculate the *i*-th MAC matrix. Therefore, the numerical results are reordered coherently with the experimental ones, thus moving the highest MAC coefficients on the diagonal. The reordered numerical results are used to compute the residuals and to solve the inverse problem. In civil engineering field, the coherence between experimental and numerical modes is usually guaranteed by defining suitable objective functions or by using the concept of *system mode shapes* [26]. However, in the former case the predictive model of the modal characteristics can not be explicitly defined, while in the latter case numerical modes related to the soil are not discarded. Therefore, the modifications of MCMC, proposed by Sevieri et al. [13] seems to be more efficient in dam engineering field. The hyper-prior distributions (Table 3) are defined by using the material test results (Table 1). The distributions of the mean values are directly derived from Table 1, while for those of the standard deviations a C.o.V. equal to 10% is assumed. Non-informative prior distributions [22] are used for the terms of $\boldsymbol{\Sigma}_f$ and the coefficients collected in $\boldsymbol{\lambda}$ and $\mathbf{w}$, since no information about them are available. In this case of study, since no records of ambient vibrations are available, a high-fidelity model is used to simulate the experimental behavior of the dam. In this application, the high-fidelity model is a more refined version of the “bonded” one presented in Section 3. It is composed by 81276 quadratic tetrahedral mechanical elements C3D10 for the soil, 28794 quadratic tetrahedral mechanical elements C3D10 for the dam body, 57414 linear tetrahedral acoustic elements AC3D4 for the basin and 3100 linear hexahedral one-way infinite elements as boundary conditions for the soil domain. The elastic parameters of the high-fidelity model are reported in Table 3.

	K_C [MPa]	G_C [MPa]	K_S [MPa]	G_S [MPa]
s. d.	16667.0	15217.0	21930.0	22321.0

Table 3: Elastic parameters of the high-fidelity model.

The Bayesian procedure is applied separately for frequencies (Equation 3) and mode shapes (Equation 4), both in the case of "bonded" and "frictionless" model. The results in terms of comparisons between prior and posterior distributions of the parameters $\boldsymbol{\theta}_{el}$ are shown in Figure 6. In the case of "bonded" model the updating leads to the same values assumed for the high-fidelity model, either using frequencies or mode shapes predictive models. In the "frictionless" case the use of different predictive models leads to different results which are in contrast with the correct one, because of the lack of stiffens due to the absence of interaction between adjacent monoliths. These results highlight the need to perform predictive models which consider the state of the vertical contraction joints, i.e. close or open. An idea is to model the state of each vertical joint as random variable, and to correlate these states with the environmental conditions, i.e. reservoir level and temperatures.



a) K_c

b) G_c

c) K_s

d) G_s

Figure 6: Comparison between prior and posterior distributions.

5 CONCLUDING REMARKS

Dynamic SHM systems, based on ambient vibrations, are powerful tools to control the health state of the structures and to reduce the uncertainties in the predictive models. However, due to the amount of uncertainties and the numerical complications which affect the FE models of concrete dams, dynamic SHM system are very rare in dam engineering field. Indeed, the dam-soil-reservoir interaction as well as the epistemic uncertainties lead to a large number of numerical modes with no experimental correlations.

In this paper, an Italian concrete gravity dams is used first to investigate the effect of the epistemic uncertainties on the modal behavior of the structure and then to apply a hierarchical Bayesian procedure to reduce them. Particular attention has been placed on the contribution of the vertical contraction joints behavior which must be considered in order to perform a reliable predictive model of the structure. Indeed, the opening-closing movement of the vertical contraction joints during the year leads to a strong variation of the modal behavior of the dam, which cannot be neglected. This paper shows how a hierarchical Bayesian framework, if integrated within a dynamic SHM of concrete dams, can successfully improve the performance of the SHM itself.

REFERENCES

[1] ASDSO. State and federal oversight of dam safety must be improved. Mag Assoc State Dam Saf Off , 2011.

[2] M. Hariri-ardebili, Risk , Reliability , Resilience (R3) and Beyond in Dam Engineering : A State-of- the-Art Review. *International Journal of Disaster Risk Reduction*, **31**, 806–831, 2018.

[3] E. Chatzi, *Identification Methods for Structural Health Monitoring. 1st ed.*, Springer International Publishing, 2016.

[4] P. Gardoni, A. Der Kiureghian, K. M. Mosalam, Probabilistic models and fragility estimates for bridge components and systems, *PEER Rep. No. 2002/13*, 2002.

[5] J. F. Hall, The dynamic and earthquake behaviour of concrete dams: review of experimental behaviour and observational evidence. *Soil Dyn Earthq Eng*, **7**, 58–121, 1988.

[6] M. Andreini, A. De Falco, G. Marmo, M. Mori, G. Sevieri, Modelling issues in the structural analysis of existing concrete gravity dams, *Proceedings of the 85th ICOLD Annual Meeting*, 363-383, 2017.

[7] A. De Falco, M. Mori, G. Sevieri, Simplified Soil-Structure Iinteraction models for concrete gravity dams. *Proc. 6th Eur. Conf. Comput. Mech. 7th Eur. Conf. Comput. Fluid Dyn.*, 2269–80, Glasgow, 2018.

[8] A. De Falco, M. Mori, G. Sevieri, FE models for the evaluation of hydrodynamic pressure on concrete gravity dams during earthquakes. *Proc. 6th Eur. Conf. Comput. Mech. 7th Eur. Conf. Comput. Fluid Dyn.*, 1731–42, Glasgow, 2018.

[9] A. De Falco, M. Mori, G. Sevieri, Bayesian updating of existing concrete gravity dams model parameters using static measurements. *Proc. 6th Eur. Conf. Comput. Mech. 7th Eur. Conf. Comput. Fluid Dyn.*, Glasgow, 2018.

[10] C. Rainieri, G. Fabbrocino, *Operational Modal Analysis of Civil Engineering Structures. 1st ed.*, Springer-Verlag New York, 2014.

[11] R. Brincker, C. E. Ventura, *Introduction to Operational Modal Analysis*, John Wiley & Sons, Ltd, 2015.

[12] P. Bukenya, P. Moyo, H. Beushausen, C. Oosthuizen. Health monitoring of concrete dams: A literature review, *Jour. of Civil Structural Health Monitoring*, **4**, 235–244, 2014.

[13] G. Sevieri, A. De Falco. A hierarchical Bayesian procedure for the updating of gravity dam model parameters via ambient vibrations, *Computer-Aided Civil and Infrastructure Engineering*, under review, 2019.

[14] M. Hariri-Ardebili, V. Saouma. Sensitivity and uncertainty quantification of the cohesive crack model, *Engineering Fracture Mechanics*, **155**, 18-35, 2016.

[15] M. Hariri-Ardebili, S. Mahdi Seyed-Kolbadi, V. Saouma, J. Salamon, L. Nuss, Anatomy of the vibration characteristics in old arch dams by random field theory. *Engineering Structures*, **179**, 460–75, 2019.

[16] B. Rosić, H. Matthies, Sparse bayesian polynomial chaos approximations of elasto-plastic material models, *XIV Int. Conf. Comput. Plast. Fundam. Appl.*, 256–67, Barcelona, 2017.

[17] D. Xiu, *Numerical Methods for Stochastic Computations*, Princeton University Press, 2010.

[18] D. Xiu, G. Karniadakis, The Wiener-Askey Polynomial Chaos for Stochastic Differential Equations, *SIAM Journal on Scientific Computing*, 2002.

[19] S. Timoshenko, J. Goodier, *Theory of Elasticity*, McGraw-Hill College, 1970.

[20] ABAQUS, *ABAQUS documentation*, 2014.

[21] R. Allemang, The modal assurance criterion - Twenty years of use and abuse, *Sound and Vibration*, **37**, 14–21, 2003.

[22] G. Box, G. Tiao, *Bayesian Inference in Statistical Analysis*, Wiley-Interscience, 1992.

[23] Z. Yang, A modified family of power transformations, *Economics Letters*, **92**, 14–9, 2006.

[24] A. Gelman, J. Carlin, H. Stern, D. Rubin, *Bayesian Data Analysis*, Chapman and Hall/CRC, 2004.

[25] D. Gamerman, H. Lopes, *Markov Chain Monte Carlo-Stochastic Simulation for Bayesian Inference*, CRC Press, 2006.

[26] Y. Huang, C. Shao, B. Wu, J. Beck, H. Li, State-of-the-art review on Bayesian inference in structural system identification and damage assessment, *Adv Struct Eng*, 2018.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

A BAYESIAN HIERARCHICAL MODEL FOR CLIMATIC LOADS UNDER CLIMATE CHANGE

P. Croce[1], P.Formichi[1] and F. Landi[1]

[1] Department of Civil and Industrial Engineering – Structural Division, University of Pisa
Largo Lucio Lazzarino 1, Pisa, Italy
e-mail: name@e-mail.address

Abstract

In the mid-term future, climate change could determine significant alterations of the frequency and magnitude of climate extremes, so affecting the design of new structures and infrastructures, and the reliability of existing ones designed according to the provisions of present or past Codes.

In this work, a Bayesian hierarchical model for the characterization of climate extremes under non stationary climate conditions is presented starting from the analysis of an ensemble of future climate projections. The Bayesian Hierarchical Model is formulated through the classical three-level formulation, in which the standard extreme value representation at each site is combined with a spatial latent process, and collects the main sources of uncertainties regarding climate projections.

A Metropolis Hastings algorithm within a Gibbs sampler is implemented to update model parameters, and from the posterior probability density functions of the extreme value distribution parameters, return levels that serve as basis for structural design are estimated. The implementation of the model in different time windows combined with the Bayesian framework allows the probabilistic assessment of time evolution of extreme value parameters and return levels.

The results obtained for a relevant case study demonstrate the possibilities of the proposed methodology to describe climate extremes under climate change and to provide guidance for potential amendments in the current definition of climatic actions on structures.

Keywords: Climate Change, Climatic Actions, Structural Design, Bayesian Hierarchical Model, MCMC algorithm.

1 INTRODUCTION

In the mid-term future, climate change could determine significant alterations of the frequency and the magnitude of climate extremes. Since structural design is often governed by climatic actions such as thermal, wind, snow and ice loads, alteration of them caused by climate change could significantly affect the design of new structures and infrastructures as well as the reliability of the existing ones designed in accordance to the provisions of current or past Codes [1]. Indeed, the current definition of climatic actions on structures is based on the extreme value analysis of the underlying natural phenomena (daily temperatures, ground snow load, wind velocities) under the assumption of stationary climate conditions [2].

As consequence of global warming this assumption is becoming more and more arguable and a better evaluation of climate extremes and their evolution over time is needed to evaluate the potential consequences for infrastructures and buildings.

Dealing with climate extremes, generally recorded at a spatial scale, a key strategy in extreme value analysis to overcome difficulties caused by the scatter of data is the spatial modelling [3]. The main advantage in spatial modelling is the pooling of information but it can be also useful for interpolation to sites where little or no data may have been collected. Then, the implementation in a Bayesian framework, enables inferences and predictions to incorporate uncertainties in process variation and parameter estimates.

In order to characterize the spatial behavior of the extreme value process, a Bayesian hierarchical model for climate extremes derived from the analysis of Regional Climate Model (RCM) output is proposed. The model is able to incorporate physical and spatial information through covariates and random effects and is implemented on different time windows of forty years long to assess the time evolution of extreme value parameters. From the posterior PDFs of extreme value parameters, the characteristic values of climatic loads, used for structural design, are evaluated assessing their changes with time and considering the uncertainty in the predictions.

The proposed methodology will be presented showing the results obtained for extreme ground snow loads in the Italian Mediterranean region [4], considering an ensemble of six different RCMs for the period 1951-2100 and two different emission scenarios.

2 METHODOLOGY

There has been considerable recent interest in spatial hierarchical models to characterize the spatial behavior of climate data. Aim of these models is to describe how the marginal distribution of a quantity of interest varies with its location. The key idea is that rather than applying a spatial model directly to the data, it is assumed that there is a latent spatial process characterized by a spatial model for the parameters of the marginal distributions at each location. An extensive review of such models for spatial data can be found in [5].

Hierarchical spatial modelling for extremes has begun to be studied recently, one of the first work in this field is found in [3], while successive developments and applications are available in [6] for extreme precipitations and in [7] for extreme precipitations obtained by regional climate models. They are increasingly used for the capability to borrow strength from neighboring locations when estimating parameters in extreme value analysis, usually characterized by small amount of data. The Bayesian Hierarchical Model is formulated through what has now become the standard three-level hierarchical formulation [8]:

- Data Layer, which is the base layer where data, e.g. the yearly maxima of the investigated climate variables, are modelled at each location according to the Extreme Value theory;

- Process Layer, where the latent process that drives the extremes for the study region is formulated;
- Prior Layer, where information about the parameters controlling the latent process are given in terms of prior distributions.

The model is flexible and able to incorporate covariate information, variability due to spatial effects and micro-scale variability due to climate model uncertainty. Each layer of the model will be fully described in the next paragraphs.

2.1 Data Level

At data level, series of yearly maxima derived from the analysis of climate projections provided by each RCM *r*, are available for each cell *i* in the study region. In order to evaluate the evolution in time of the extreme value process, data are divided in subsequent time windows of 40 years shifted by ten years, thus obtaining eleven time window *t* (1951-1990, 1961-2000,..., 2041-2080 and 2051-2090). The time window length is set to 40 years to be consistent with the actual definition of climatic loads on structural codes, which is based on the analysis of observed data series of climate extremes of about forty years [9], while the shift of ten years is defined to properly evaluate the evolution in time of climatic loads.

For each time window *t*, *N*=40 yearly maxima are thus given at each cell *i* in the study region and assuming an Extreme Value Distribution Type I as marginal distribution, the random variable Y_{itr} is described by the cumulative distribution function *F(y)*

$$F(Y_{i,t,r} < y) = \exp\left\{-\exp\left[-\frac{y-\mu_{i,t,r}}{\sigma_{i,t,r}}\right]\right\} \tag{1}$$

and the probability density function *f(y)* is

$$f(Y_{i,t,r} < y) = \frac{1}{\sigma_{i,t,r}}\exp\left\{-\left[\frac{y-\mu_{i,t,r}}{\sigma_{i,t,r}} + \exp\left(-\frac{y-\mu_{i,t,r}}{\sigma_{i,t,r}}\right)\right]\right\} \tag{2}$$

where $\mu_{i,t,r}$ and $\sigma_{i,t,r}$ are the location and scale parameter for cell *i*, time window *t* and RCM *r*. The first level of the hierarchical model structure, for each climate model *r*, will be described by

$$Y_t(s) \mid \theta_t \sim \mathrm{EVI}(\mu_\mathrm{t}(\mathrm{s},\omega),\exp(\log(\sigma_\mathrm{t})(\mathrm{s},\omega))) \tag{3}$$

with

$Y_t(s)$ are the yearly maxima of climate data at the location *s* in the study region for the time window *t;*

θ_t are the random parameter of the model in the time window *t*;

$\mu_t(s,\omega)$ is a random field describing the spatial variation of location parameter of EV Type I distribution in the time window *t*, where ω ∈ Ω express the random event;

$log(\sigma_t)(s,\omega)$ is a random field describing the spatial variation of the log-scale parameter of EV Type I distribution in the time window *t*, where ω ∈ Ω express the random event.

If $Y_{\mathrm{i,t}}$ is a vector of the yearly maxima in the investigated time window t for the cell *i* in the study region and $Y_t = (Y^{\mathrm{T}}_{1,t}, ..., Y^{\mathrm{T}}_{D,t})$ contains all the maxima for the *D* cells in the region, then assuming the conditional independence of Y_i for all location, common assumption in hierarchical modelling [5], starting from eq. 2 the likelihood function becomes

$$p(Y_t \mid \theta_t) = \prod_{i=1}^{D} \prod_{k=1}^{40} \frac{1}{\sigma_i} \exp\left\{ -\left[\frac{y_{ik} - \mu_i}{\sigma_i} + \exp\left(-\frac{y_{ik} - \mu_i}{\sigma_i} \right) \right] \right\} \tag{4}$$

2.2 Process Level

In the hierarchical model, at the process level, the latent spatial process is formulated by constructing a structure that relates the parameter of the data level to the characteristics of the region. In particular, a Gaussian random field is proposed to model spatial variation of location and log-scale parameters according the following formulas

$$\mu_t(s,\omega) \sim N(X\beta_{\mu,t} + W_{\mu,t}(s,\omega), \tau^2_{\mu,t}) \tag{5}$$

$$\log(\sigma_t)(s,\omega) \sim N(X\beta_{\sigma,t} + W_{\sigma,t}(s,\omega), \tau^2_{\sigma,t}) \tag{6}$$

with

$W_{\mu,t}(s,\omega)$ is a spatial random effect described by a zero mean Gaussian random field $N(0,\sum_{\mu}(l_{\mu,t}, s_{\mu,t}))$ with covariance matrix $\sum_{\mu}$.

$W_{\sigma,t}(s,\omega)$ is a spatial random effect described by a zero mean Gaussian random field $N(0,\sum_{\sigma}(l_{\sigma,t}, s_{\sigma,t}))$ with covariance matrix $\sum_{\sigma}$.

X is a matrix of covariate information;

$\beta_{\mu,t}$ and $\beta_{\sigma,t}$ are vectors of regression coefficients for μ_t and σ_t given X;

τ_{μ}^2 and τ_{σ}^2 are precision terms for the location and the log-scale fields

Different models may be set for the covariance structure, considering stationarity or non stationarity in the covariance function as described in [10]. In this work an exponential model with parameter correlation length $l_{\mu,t}$ and sill $\underline{s_{\mu,t}}$ has been considered. ;

Covariate information are spatially-varying, physical features or observable quantities that can either be collected at all prediction locations of interest or in some way interpolated from nearby observations [11] (for example, elevation, or geographical feature such latitude or longitude but also wind speed or direction).

The precision terms, $\tau_{\mu,t}^2$ and $\tau_{\sigma,t}^2$ in eq. 5 and 6, can be viewed as a noise associated with replication of measurements at location s, and in this case represents the variability of the data related to internal climate model uncertainty. However, the availability of few realizations of climate model run, often only one, due to the enormous computational demand doesn't allow a direct assessment of this source of uncertainty.

A possibility to assess the uncertainty related to the RCM internal variability is the methodology described by the authors in [12], where an ad hoc weather generator is proposed able to generate new consistent climate projections directly from RCM output. Analyzing the generated series, an evaluation of the noise associated to the EV parameters becomes possible and the constant precision terms $\tau_{\mu,t}^2$ and $\tau_{\sigma,t}^2$, associated at each investigated climate model r, depending on the cell i and the time window t, are defined.

2.3 Prior Level

Prior distribution are finally assigned to the hyperparameters of the model at each time window t, $\theta_t(\beta_{\mu,t}, \beta_{\sigma,t}, l_{\mu,t}, s_{\mu,t}, l_{\sigma,t}, s_{\sigma,t})$. Where possible, uninformative priors are assigned to these parameters and conjugate priors are used to facilitate the use of Gibbs sampling in the model implementation.

Normal distribution with mean defined as the mean of the point estimates of parameters in the region and large variance are set for the intercept terms of the regression coefficients ($\beta_{0,\mu}$

and *$\beta_{0,\sigma}$)*, while normal distribution with zero mean and large variance are set for the other regression coefficients *β*.

However, informative priors are generally needed for the sill ($s_{\mu,t}$, $s_{\sigma,t}$) and correlation length ($l_{\mu,t}$, $l_{\sigma,t}$) parameters to avoid improper posteriors [5]. Since these parameters are not observable quantities, a preliminary analysis should be carried out to characterize the behavior of the experimental semi-variogram for μ and σ. Following the procedure proposed in [13] maximum likelihood estimates of μ and σ are computed at each location in the study region, and prior distributions for the parameters are chosen to define a wide envelope around the experimental semi-variogram given by the ML estimates.

2.4 Implementation of the model

In order to update each parameter θ_t of the described model a Metropolis–Hastings algorithm within a Gibbs sampler has been implemented. This hybrid MCMC algorithm [14] consists of a Gibbs sampler where a Metropolis step is used in order to sample from conditional distributions which are not known. Parameters of the model, which will be implemented for each time window t, are collected at each step i of the algorithm in the vector $\theta_t^{(i)}(\beta_{\mu,t}^{(i)}, \beta_{\sigma,t}^{(i)}, l_{\mu,t}^{(i)}, s_{\mu,t}^{(i)}, l_{\sigma,t}^{(i)}, s_{\sigma,t}^{(i)})$. Then, applying the Gibbs sampler, we partition the sampling for location μ and log-scale log*(σ)* parameters and the next point in the chain $i + 1$, is generated in the following steps:

- Updating of correlation length parameter;
- Updating of sill parameter;
- Updating of regression parameters;
- Updating of EV parameter at each site;
- Repetition of the previous four steps for log-scale parameter.

A complete description of each step of the algorithm can be found in [15]. The algorithm is iterated checking the convergence for each parameters and finally, posterior densities of parameters θ_t are obtained. Implementing the model in the subsequent time windows *t*, the variation over time of posterior densities can be easily assessed, especially for EV parameters and consequently for return levels. In particular, for the definition of climatic actions on structures, we are interested in the evaluation of climate change impact on characteristic values c_k, i.e. value having a probability of 2% to be exceeded in one year (mean return period of 50 years) [16]. Therefore, posterior samples are easily computed for c_k according to

$$c^i_{k,t} = \mu^i_t + \sigma^i_t \left\{-\log\left[-\log(1-0.02)\right]\right\} \tag{7}$$

and updated return level maps for characteristic loads can be easily drawn evaluating changes in the different time windows.

3 APPLICATION FOR GROUND SNOW LOADS

3.1 Study area and dataset

This section shows an application of the methodology presented in the previous section, on extreme ground snow loads considering the Zone 3-4 of the Italian Mediterranean climatic region defined by the Annex C to EN1991-1-3 [4]. The study region is shown in Figure 1 and comprises *D*=272 cells at which climate projections are provided by the highest resolution Regional Climate Models developed by the EUROCORDEX initiative [17].

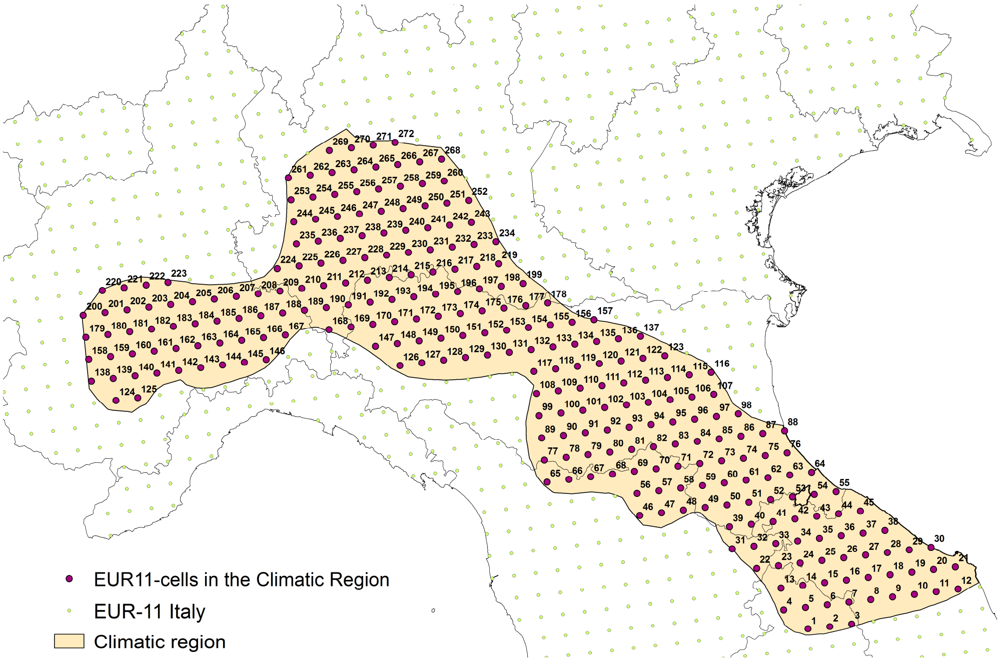


Figure 1: Investigated region.

Climate projections provided by an ensemble of r=6 RCMs for the period 1951-2100 have been analyzed, considering a medium emission scenario RCP4.5 and the highest emission scenario RCP8.5 [18]. The main characteristics of the investigated climate projections are reported in Table 1.

Institute	RCM	GCM	Period	Experiment
DMI	HIRHAM5	EC-EARTH	1951-2100	Historical,RCP4.5,RCP8.5
CLMcom	CCLM4-8-17	CNRM-CM5-LR	1951-2100	Historical,RCP4.5,RCP8.5
CLMcom	CCLM4-8-171	EC-EARTH	1951-2100	Historical,RCP4.5,RCP8.5
KNMI	RACMO22E	EC-EARTH	1951-2100	Historical,RCP4.5,RCP8.5
MPI-CSC	REMO2009	MPI-ESM-LR	1951-2100	Historical,RCP4.5,RCP8.5
IPSL-INERIS	WRF331F	CM5A-MR	1951-2100	Historical,RCP4.5,RCP8.5

Table 1: Overview on the analyzed climate projections and their main characteristics.

3.2 Implementation and results

In order to derive ground snow loads from regional climate models output such as daily temperatures and precipitation, the procedure described in [1] and [15] has been implemented deriving series of N=140 yearly maxima snow load for each cell in the study region.

Among possible covariate information, altitude shows most significant influence on extreme snow loads, it has been then considered as the only covariate and a quadratic model has been chosen as defined in the Eurocode EN1991-1-3 [4] for characteristic ground snow load in Mediterranean region. Then, covariate matrix X and the vectors of regression coefficients $\beta_{\mu,t}$ and $\beta_{\sigma,t}$ in eq. 5 and 6 become

$$X = \begin{bmatrix} 1 & alt_1 & alt_1^2 \\ \vdots & \vdots & \vdots \\ 1 & alt_D & alt_D^2 \end{bmatrix}; \beta_{\mu,t} = \begin{bmatrix} \beta_{\mu,0,t} \\ \beta_{\mu,1,t} \\ \beta_{\mu,2,t} \end{bmatrix}; \beta_{\sigma,t} = \begin{bmatrix} \beta_{\sigma,0,t} \\ \beta_{\sigma,1,t} \\ \beta_{\sigma,2,t} \end{bmatrix} \tag{7}$$

The model have been implemented for each time window *t*, and the MCMC algorithm has been iterated 40 000 times, obtaining posterior densities of random parameters $\theta_t(\beta_{\mu,t}, \beta_{\sigma,t}, l_{\mu,t}, s_{\mu,t}, l_{\sigma,t}, s_{\sigma,t})$. As an example in Figure 2 the results in terms of posterior densities of location μ and scale σ EV parameters, but also q_k, are presented for one cell, i=160, in the study region in different time windows (t = 1, 4, 8, 10) according to one of the investigated climate model (first RCM in Table 1).

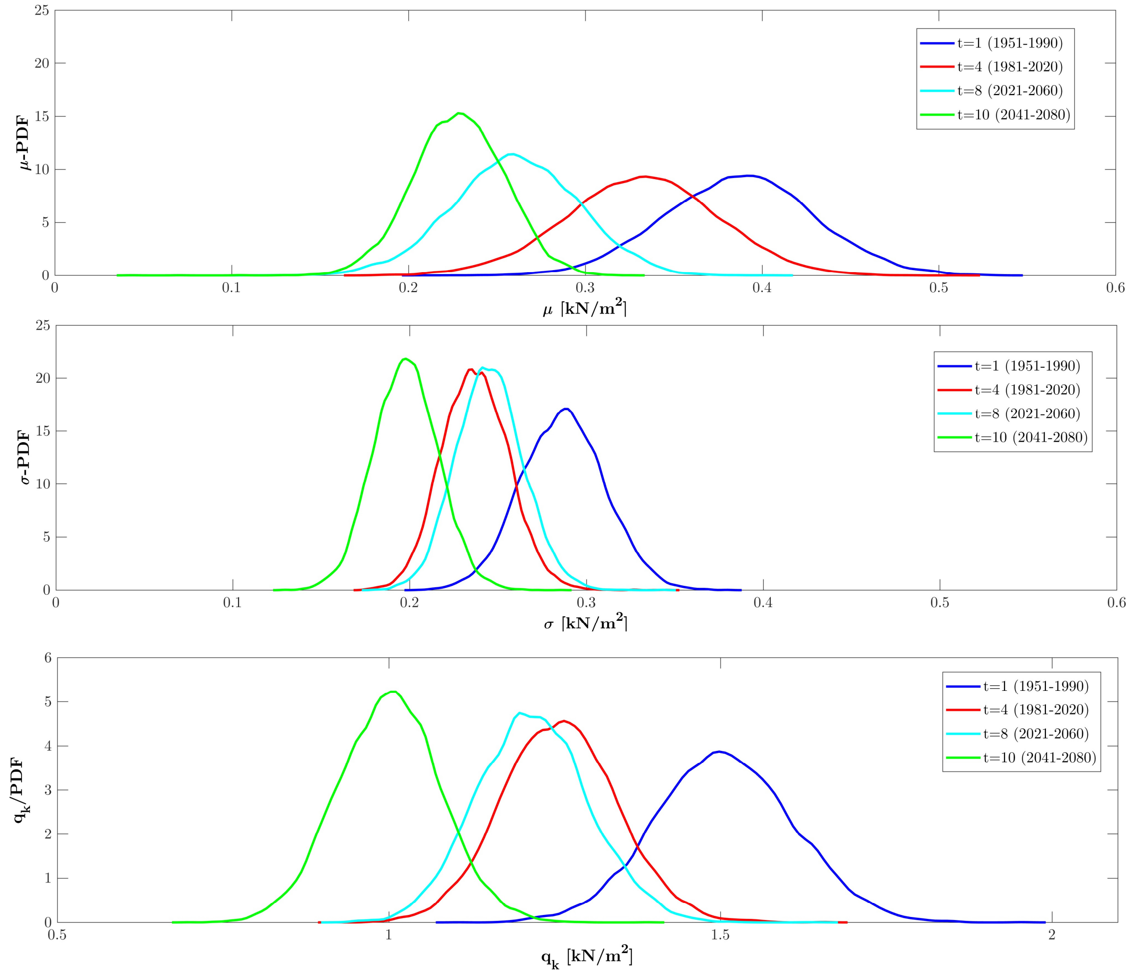


Figure 2: Changes in posterior PDFs of μ , σ and q_k with time *t*.

The hierarchical model combined with the Bayesian approach enables a direct assessment of the uncertainties affecting the extreme value process using the posterior distribution of parameters and return values, as shown in Figure 2. Moreover, the implementation of the model in subsequent time window allows a direct estimation of the effect of climate change on extreme ground snow loads by means of the analysis of changes in posterior densities of EV parameters and return values.

The spatial pooling of the data provides an added value in comparison with classical approach based on maximum likelihood estimates at point level leading to more precise and less variable estimates [3]. The reduced uncertainty in the estimation is shown in Figure 2 where

q_k estimates obtained by the presented spatial model are compared with the classical site by site analysis according the maximum likelihood method, for some cells at increasing distance in the study region. The results in terms of 95% confidence interval clearly show the reduced uncertainty for the illustrated spatial model confirming the advantages of spatial pooling for tail estimation.

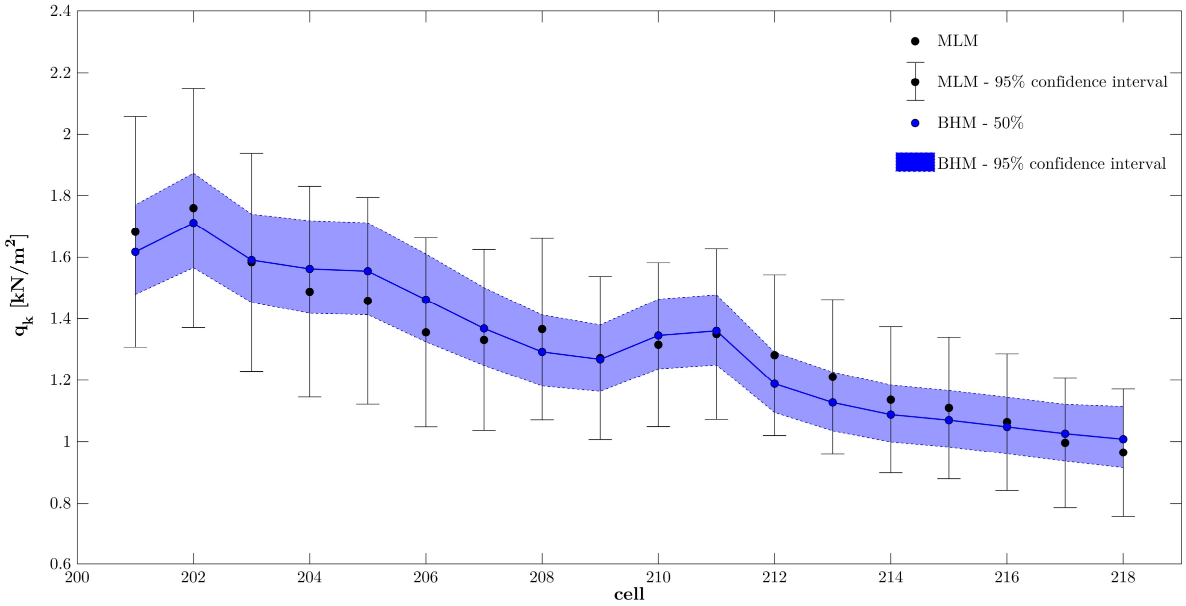


Figure 3: q_k estimates by maximum likelihood method (MLM) and Bayesian hierarchical model (BHM) with 95% confidence intervals.

3.3 Return level and Factor of Change Maps

Return level maps can also be drawn from the posterior samples of q_k obtained according the investigated climate models and scenarios. However, more information about climate change impact can be derived by the definition of factors of change (FC) as the difference or the ratio of predictions from RCM in the future period and the historical period.

The factor of change approach has a long history in climate change impact studies, it is based on the assumption that changes in the observed climate variables form present to future are the same than changes predicted by the climate models not requiring to apply bias correction methods.

Factor of change maps represent a good solution for the assessment and the visualization of future trends in climatic actions since the estimated changes can be easily applied to the current version of climatic load maps in structural Codes.

Therefore, from the posterior samples of q_k, mean and standard deviation for FC are computed

$$\overline{FC}(q_{k,t}) = \frac{\overline{q_{k,t}}}{\overline{\overline{q_{k,1}}}};\sigma_{FC(q_{k,t})} = \frac{\sigma_{q_{k,t}}}{\overline{\overline{q_{k,1}}}} \tag{8}$$

As an example maps for mean FC and standard deviation are reported in Figure 4 and 5 respectively, considering t=8 (2021-2060) and the six RCMs in Table 1 run according the RCP4.5 scenario.

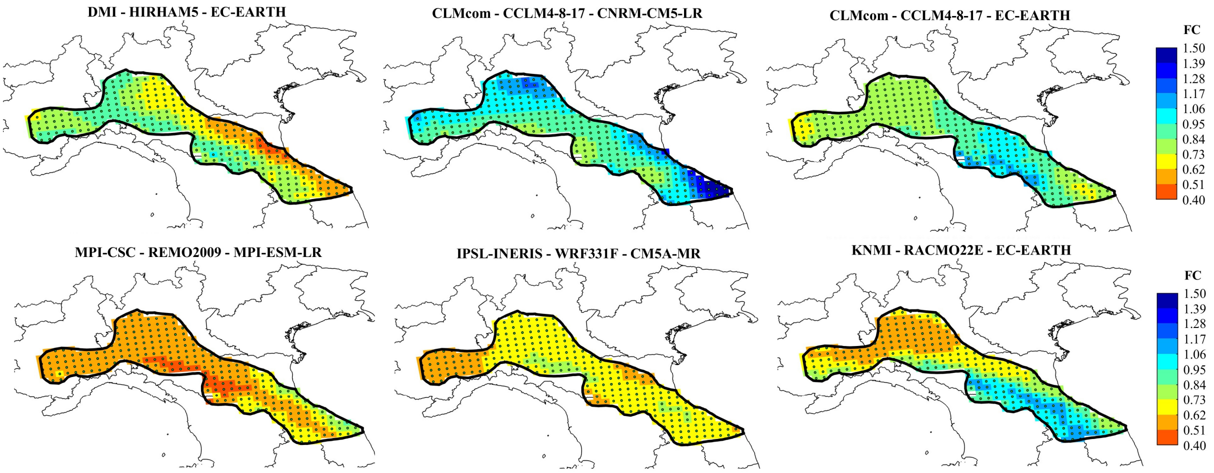

DMI - HIRHAM5 - EC-EARTH
CLMcom - CCLM4-8-17 - CNRM-CM5-LR
CLMcom - CCLM4-8-17 - EC-EARTH
MPI-CSC - REMO2009 - MPI-ESM-LR
IPSL-INERIS - WRF331F - CM5A-MR
KNMI - RACMO22E - EC-EARTH
FC
1.50
1.39
1.28
1.17
1.06
0.95
0.84
0.73
0.62
0.51
0.40

Figure 4: Posterior mean of q_k Factor of Change for 2021-2060 w.r.t. 1951.1990 according to the climate models in Table 1, Scenario RCP4.5.

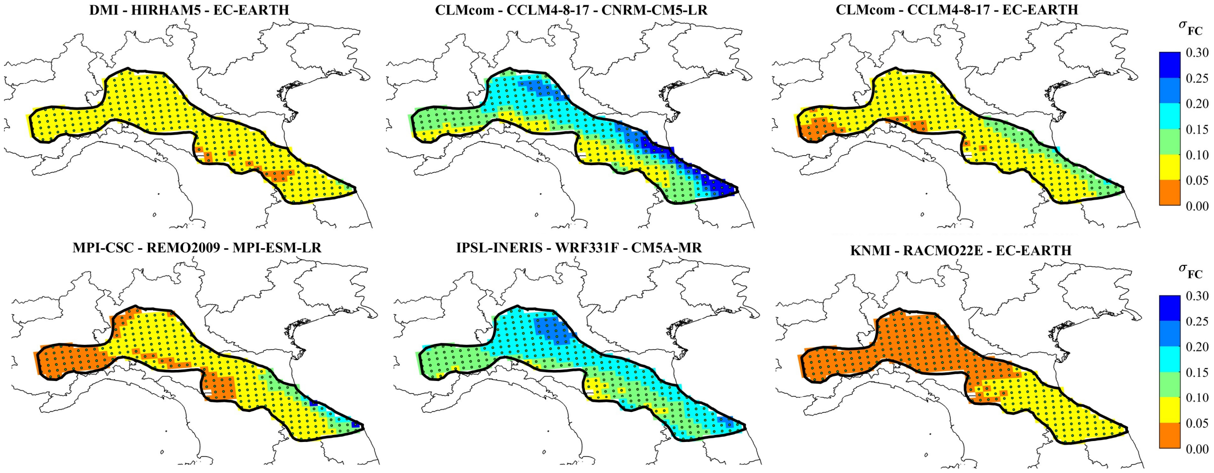

DMI - HIRHAM5 - EC-EARTH
CLMcom - CCLM4-8-17 - CNRM-CM5-LR
CLMcom - CCLM4-8-17 - EC-EARTH
MPI-CSC - REMO2009 - MPI-ESM-LR
IPSL-INERIS - WRF331F - CM5A-MR
KNMI - RACMO22E - EC-EARTH
σ_{FC}
0.30
0.25
0.20
0.15
0.10
0.05
0.00

Figure 5: Posterior standard deviation of q_k Factor of Change for 2021-2060 w.r.t. 1951.1990 according to the climate models in Table 1, Scenario RCP4.5.

The results obtained for the different climate models can be finally combined considering each climate model of the ensemble as an equally likely representation of future climate. In this way, a complete probabilistic description of future changes in characteristic loads is obtained providing guidance for potential amendments of the current version of climatic load maps in structural Codes.

In Figure 6, the results in terms of factor of change maps for characteristic ground snow load q_k are presented in a bivariate map, which consider the 25-75% prediction interval for FC, for three time windows (1991-2030, 2011-2050, and 2031-2080) according the RCP4.5 and RCP8.5 scenario (second and third row respectively). In the same Figure, on the top row, the current snow load map for the study region, obtained implementing the load altitude relationship given in the Annex C of EN1991-1-3 [4] and based on the results of the European Snow Load Research Project[19], which analyzed observed data series of ground snow loads in the period 1951-1990, is also reported.

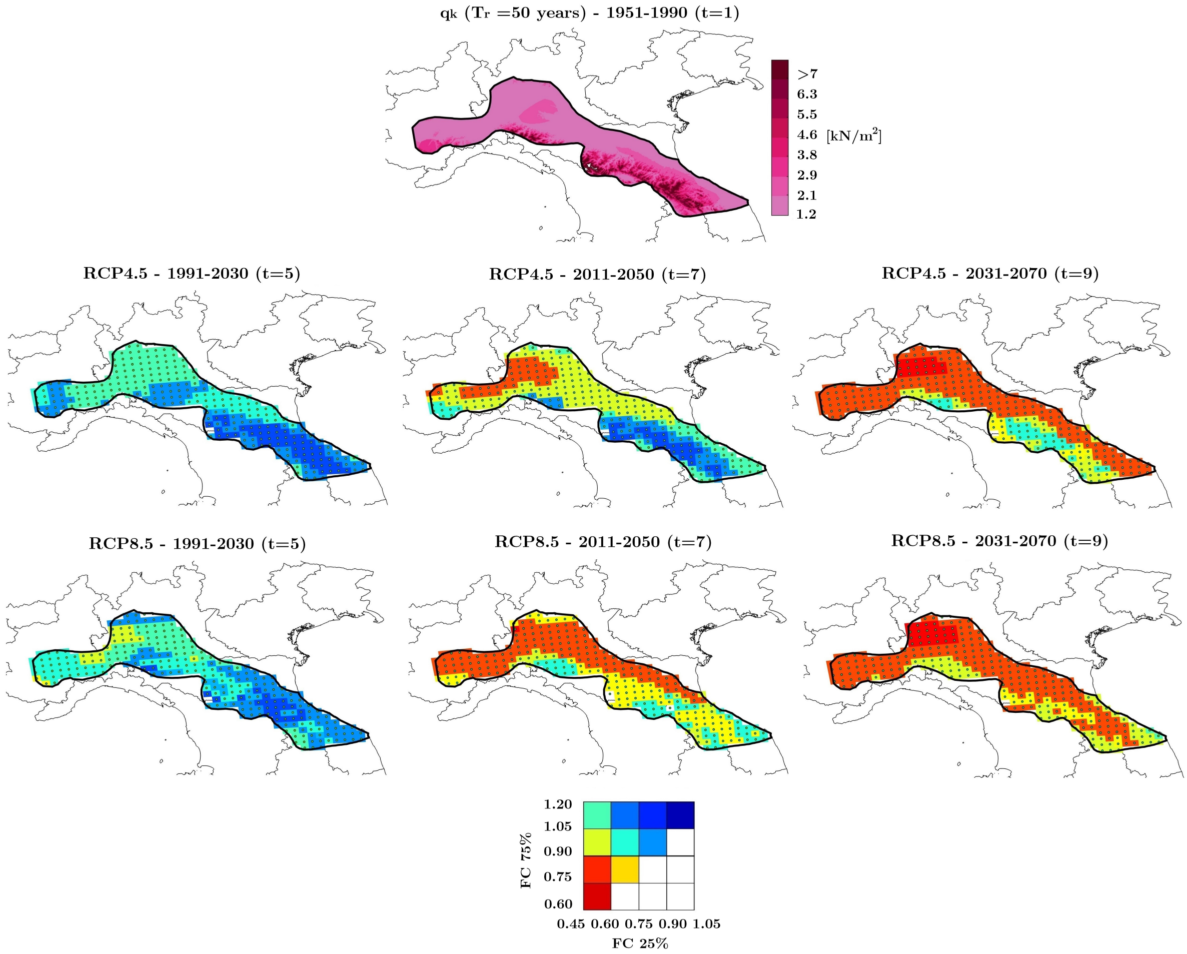


Figure 6: Factors of Change for q_k - Confidence interval [25-75%] Map (Scenario RCP4.5).

4 CONCLUSIONS

In order to estimate future changes in climatic actions on structures, a methodology based on the construction of a Bayesian hierarchical model for the characterization of climate extremes derived from the analysis of high-resolution climate model output has been presented.

The model is formulated through the classical three-level formulation, in which the standard extreme value representation at each site is combined with a spatial latent process, and it is implemented in different time windows to assess climate change effects on the extreme value process.

An application on ground snow loads has been carried out to illustrate the capabilities of the proposed methodology. The results shows that the Bayesian framework enables a direct assessment of the uncertainties affecting the prediction of the extreme value parameters and return levels. Moreover, the spatial pooling of the data leads to more precise and less variable estimates with respect to classical approaches based on maximum likelihood estimates at point level.

Finally, combining the results obtained for each climate model, suitable factors of change uncertainty maps are drawn providing guidance for potential amendments of the climatic load maps in structural Codes.

REFERENCES

[1] P. Croce, P. Formichi, F. Landi, F. Marsili, Climate change: Impact on snow loads on structures. *Cold Regions Science and Technology*, 150, 35–50, 2018.

[2] P. Croce, P. Formichi, F. Landi, F. Marsili, Evaluating the effect of climate change on thermal actions on structures. In *Life-Cycle Analysis and Assessment in Civil Engineering: Towards an Integrated Vision*, 1751–1758, 2019.

[3] E. Casson, S. Coles, Spatial Regression Models for Extremes. *Extremes*, 449–468, 1999.

[4] CEN. EN 1991-1-3:2003 - Eurocode 1: Actions on structures - Part 1-3: General actions - Snow loads. 2003.

[5] S. Banerjee, B. P. Carlin, A. E. Gelfand, *Hierarchical modeling and analysis for spatial data*. CRC Press LLC, 2004.

[6] D. Cooley, D. Nychka, P. Naveau, Bayesian Spatial Modeling of Extreme Precipitation Return Levels. *Journal of the American Statistical Association*, 479, 824–840, 2007.

[7] D. Cooley, S. R. Sain, Spatial hierarchical modeling of precipitation extremes from a regional climate model. *Journal of Agricultural, Biological, and Environmental Statistics*, 15, 381–402, 2010.

[8] S. R. Sain, R .Furrer, N. Cressie, A spatial analysis of multivariate output from regional climate models. *Annals of Applied Statistics*, 1, 150–175, 2011.

[9] P. Formichi, et al.; *Eurocodes: background and applications. Elaboration of maps for climatic and seismic actions for structural design with the Eurocodes*, JRC Science for Policy Report.

[10] H. Xu, P. Gardoni, Improved latent space approach for modelling non-stationary spatial–temporal random fields, *Spatial Statistics*, 23, 160–181, 2018.

[11] M. D. Risser, C. A. Calder, Regression-based covariance functions for nonstationary spatial modeling, *Environmetrics*; 26, 284–297, 2015.

[12] P. Croce, F. Landi, P. Formichi, R. Castelluccio, Use of weather generators to assess impact of climate change : thermal actions on structures. In *Proc. of the Fifth Intl. Conf. Advances in Civil, Structural and Mechanical Engineering - CSM 2017*, 32–36, 2017.

[13] D. Cooley, D. Nychka, P. Naveau. Bayesian Spatial Modeling of Extreme Precipi-tation Return Levels. J*ournal of the American Statistical Association*, 102, 824–40, 2007.

[14] C. P. Robert, , G. Casella. *Monte Carlo Statistical Methods*, Springer, New York., 2004.

[15] F.Landi, A General Methodology for the Assessment of the Impact of Climate Change on Snow Loads on Structures, Phd Thesis, University of Pisa – TU Braunschweig.

[16] CEN. EN1990 Eurocode - Basis of structural design. 2002.

[17] D. Jacob, et al. EURO-CORDEX: New High-Resolution Climate Change Projections for European Impact Research. *Regional Environmental Change*, 14, 563–78, 2014.

[18] Van Vuuren, et al., The representative concentration pathways: an overview, *J. Climatic Change*, 109, 5-31, 2011.

[19] L. Sanpaolesi, et al. *Phase 1 Final Report to the European Commission, Scientific Support Activity in the Field of Structural Stability of Civil Engineering Works: Snow Loads*. Tech. rep., Department of Structural Engineering, University of Pisa, 1998.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

KRIGING IN TENSOR TRAIN DATA FORMAT

Sergey Dolgov[1], **Alexander Litvinenko**[2], **and Dishi Liu**[3]

[1] University of Bath
Claverton Down, Bath, BA2 7AY, United Kingdom
e-mail: s.dolgov@bath.ac.uk

[2]RWTH Aachen
Kackertstr. 9C, 52072, Aachen, Germany
e-mail: litvinenko@uq.rwth-aachen.de

[3] Institute of Scientific Computing, Technische Universität Braunschweig
Mühlenpfordtstrasse 23, D-38106 Braunschweig, Germany.
e-mail: d.liu@tu-bs.de

Keywords: low-rank tensor approximation; tensor train; geostatistical estimation; geostatistical optimal design, kriging, circulant, Toeplitz, FFT

Abstract. *Combination of low-tensor rank techniques and the Fast Fourier transform (FFT) based methods had turned out to be prominent in accelerating various statistical operations such as Kriging, computing conditional covariance, geostatistical optimal design, and others. However, the approximation of a full tensor by its low-rank format can be computationally formidable. In this work, we incorporate the robust Tensor Train (TT) approximation of covariance matrices and the efficient TT-Cross algorithm into the FFT-based Kriging. It is shown that here the computational complexity of Kriging is reduced to $\mathcal{O}(dr^3n)$, where n is the mode size of the estimation grid, d is the number of variables (the dimension), and r is the rank of the TT approximation of the covariance matrix. For many popular covariance functions the TT rank r remains stable for increasing n and d. The advantages of this approach against those using plain FFT are demonstrated in synthetic and real data examples.*

This paper is dedicated to our wonderful colleague Prof. Hermann G. Matthies on the occasion of his 68th birth anniversary.

Contents

1 Introduction

Kriging is an interpolation method that makes estimates of unmeasured quantities based on (sparse) scattered measurements. It is widely applied in the estimation of some spatially distributed quantities such as daily moisture, rainfall intensities, temperatures, contaminant concentrations or hydraulic conductivities, etc. [40, 22]. Kriging is also used as a surrogate of some complex physical models for the purpose of efficient uncertainty quantification (UQ), in which it estimates the model response under some random perturbation of the parameters. In the first case the estimation grids are usually in two or three dimensions [60, 9, 18] or four dimensions in a space-time Kriging [3, 34, 21], while in the latter the dimension number could be much larger (equals to the number of uncertain parameters). When considering finely resolved estimation grids (which is often the case for UQ jobs), Kriging can easily exceed the computational capacity of modern computers. In this case estimation variance of Kriging or solving the related geostatistical optimal design problems incurs even higher computational costs [41, 43, 55]. Kriging mainly involves three computational tasks. The first is solving a $N \times N$ system of equations to obtain the

Kriging weights, where N is the number of measurements. Despite its $\mathcal{O}(N^3)$ complexity this task is better manageable since N is usually much smaller than the number of estimates on a fine grid, $\bar{N} = \bar{n}^d$, d the dimensionality, especially when the measurement is expensive like for complex physical models. The second task is to compute the $\bar{N}$ Kriging estimates by multiplying the weights vector to the $\bar{N} \times N$ cross-covariance matrix between measurements and unknowns. The third task is to evaluate the $\bar{N}$ estimation variances as the diagonal of a $\bar{N} \times \bar{N}$ conditional covariance matrix. If we take the optimal design of sampling into account, there is an additional task to repeatedly evaluate the $\bar{N} \times \bar{N}$ conditional covariance matrix for the purpose of a high-dimensional non-linear optimization [32, 54, 51].

Remarkable progress had been made in speeding up Kriging computations by Fast Fourier transform (FFT) [11]. The low-rank tensor decomposition techniques brought a further possible reduction in the time cost, since d-dimensional FFT on a tensor in low-rank format can be made at the cost of a series of 1-dimensional FFT's, as exemplified in [59] by using canonical, Tucker and Tensor Train formats of tensors. The work in [44] brought a significant further reduction of computational cost for the second and third Kriging tasks as well as the task for the optimal design of sampling by applying a low-rank canonical tensor approximation to the vectors of interest.

In this paper, we enhance the methodology proposed in [44] by employing a more robust low-rank Tensor Train (TT) format instead of the canonical format. We apply the TT-cross algorithm for efficient approximation of tensors, which is a key improvement compared to the method introduced in [44] where the low-rank format of the covariance matrix was assumed to be given. We also consider a more broad Matérn class of covariance functions.

The current work improves the applicability of the use of low-rank techniques in the FFT-based Kriging. We achieve a reduction of the computational complexity of Kriging to the level of $\mathcal{O}(dr^3\bar{n})$, where r is the considered TT rank of the approximation, and $\bar{n}$ is the number of grid points in *one* direction, such that $\bar{N} = \bar{n}^d$ is the total number of estimated points.

We assume second-order stationarity for the covariance function and simple Kriging on a rectangular, equispaced grid parallel to the axes.

We also discuss possible extensions to non-rectangular domains and to general (scattered) measurement points. In such cases, the tensor ranks may significantly increase, up to the full rank. For the cases when FFT technique is not applicable the authors of [52, 37, 35, 29] applied the hierarchical matrix technique ($\mathcal{H}$-matrices). A parallel implementation of Kriging was done in [50].

1.1 State of the art for FFT-based Kriging

Let us assume that the covariance function is second-order stationary and is discretized on a tensor (regular and equispaced grid) mesh with $\bar{N} = \bar{n}^d$ points. Then the $\bar{N} \times \bar{N}$ auto-covariance matrix of the unknowns has a symmetric (block-) Toeplitz structure (Section 3.1), which can be extended to a (block-) circulant matrix by a periodic embedding in which the number of rows and columns is enlarged, for example, from $\bar{N}$ to $\check{N} = 2\bar{N}+1$ [49, 22, 31]. It is known [11] that only the first column of the circulant matrix has to be stored. This reduces the computing cost from quadratic to log-linear [61] in $\bar{N}$. The key in the FFT-based Kriging is the fact that the multiplication of a circulant matrix and a vector is a discrete convolution which can be computed swiftly through FFT algorithm so that the quadratic computational complexity is also reduced to a log-linear one [12].

If the measurements are given on a regular equispaced grid, the first Kriging task is solving a system also with a symmetric positive-definite Toeplitz matrix [11, 4]. Further development of methods handling measurements that are on a subset of a finer regular grid have been made in [49, 11].

The work in [44] combined the power of FFT and the low-rank canonical tensor decomposition. It was assumed that the covariance matrix and the vector of interest (of size $\check{N}$) are available in a low-rank canonical tensor format which is a sum of r Kronecker products of vectors of size $\check{n}$ each, with $\check{n}^d = \check{N}$. Separable covariance functions (e.g. Gaussian, separate exponential) can be decomposed exactly with $r = 1$. For smooth non-separable covariance functions, a small r value can usually give a good approximation.

The canonical tensor representation can not only greatly reduce the memory storage size of the circulant matrix, but also speed up the Fourier transform since the d-dimensional FFT applied on the Kronecker product of matrices can be implemented by computing the 1-dimensional FFT on the first direction of each matrix. This reduces the complexity to $\mathcal{O}(dr\check{n}\log\check{n})$. For $r \ll \check{n}$ this is a significant reduction from the complexity of FFT on the full tensor, which is $\mathcal{O}(d\check{n}^d\log\check{n})$.

1.2 Goals, approach and contributions

However, converting a full tensor to a well approximating low-rank tensor format can be computationally formidable. Simply generating the full tensor itself might be beyond the memory capacity of a desktop computer. To make the low-rank FFT-based method practical, we need an efficient way to obtain a low-rank approximation directly from the multi-dimensional function that underlies the full tensor. It could be a challenging task though to approximate the first column of the Toeplitz (circulant) matrix in the canonical tensor format for $d \geq 3$. This is due to the fact that the class of rank-k canonical tensors is a nonclosed set in the corresponding tensor product space (pp 91-92 in [28]). The Tucker format tensor decomposition [27, 17, 15] adopted in [36] could be too costly to use for problems with $d \geq 3$.

In this paper, we adopt an alternative tensor format, namely, the Tensor Train (TT) format [47, 17] (introduced in Section 4.1) which can be obtained from a full tensor in a stable direct way by a sequence of singular value decompositions of auxiliary matrices, or, more importantly, it can be computed iteratively by the *TT-cross* method [48] which has the complexity in the order of $\mathcal{O}(dr^3\bar{n})$, see Section 4.2 for more details. Often this is the most time-consuming stage of Kriging operations. Once the tensors are approximated in the TT format, the FFT can be carried out with a modest $\mathcal{O}(dr^2\bar{n}\log\bar{n})$ complexity. This makes the overall low-rank FFT-based Kriging practical for high dimensions. We test the efficiency of the method in terms of computational time and memory usage in Section 5.

Thus, our paper is novel in three aspects: (i) we approximate the covariance matrix in the low-rank TT tensor format using only the given covariance function as a black box (this part was missing in [44]), (ii) we extend the methodology to Matérn, exponential and spherical covariance functions (in addition to Gaussian functions), and (iii) we demonstrate that the low-rank approach enables high-dimensional Kriging.

1.3 Notation

We denote vectors by bold lower-case letters (e.g., $\mathbf{c}$, $\mathbf{u}$, $\boldsymbol{\xi}$) and matrices by bold upper-case letters (e.g., $\mathbf{C}_{ss}$, $\mathbf{M}$, $\mathbf{H}$). Letters decorated with an overbar represent the size of the tensor grid of estimates. Embedded matrices, vectors and their sizes are denoted by letters with a check accent (e.g., $\check{\mathbf{C}}$, $\check{\mathbf{c}}$, $\check{n}$, $\check{n}_i$). $\mathcal{F}^{[d]}$ stands for d-dimensional Fourier transform (FT), $\mathcal{F}_i$ for one-dimensional FT along the i-th dimension. $\mathcal{F}^{[-d]}$ and $\mathcal{F}_i^{-1}$ are their inverse operators.

2 Kriging and geostatistical optimal design

Like in [44], we work with the *function estimate form* [30, 31] of Kriging (introduced in Section 2.2). We take simple Kriging in which the estimates are assumed to have zero mean.

2.1 Matérn covariance

A low-rank approximation of the given function or a data set is a key component of the tasks formulated above. Among of the many covariance models available, the Matérn family [39] is widely used in spatial statistics and geostatistics.

The Matérn covariance function is defined as

$$C_{\nu,\ell}(r) = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}r}{\ell}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}r}{\ell}\right). \tag{1}$$

Here $r := \|p_1 - p_2\|$ is the distance between two points p_1 and p_2 in $\mathbb{R}^d$; $\nu > 0$ defines the smoothness. The larger is parameter ν, the smoother is the random field. The parameter $\ell > 0$ is called the covariance length and measures how quickly the correlation of the random field decays with distance. $\mathcal{K}_\nu$ denotes the modified Bessel function of order ν. It is known that setting $\nu = 1/2$ we obtain the exponential covariance model. The value $\nu = \infty$ corresponds to a Gaussian covariance model.

In [36], the authors provided the analytic sinc-based proof of the existence of low-rank tensor approximations of Matérn functions. They investigated numerically the behavior of the Tucker and canonical ranks across a wide range of parameters specific to the family of Matérn kernels. It could be problematic to extend the results of this work to $d > 3$, since one of the terms in the Tucker decomposition storage cost $\mathcal{O}(drn + r^d)$ is growing exponentially with d.

2.2 Computational tasks in Kriging and optimal sampling design

The computation of a simple Kriging process and optimal sample design involve mainly these tasks:

Task-1. Let $\mathbf{y}$ denote a N-size vector containing the sampled values, $\mathbf{C}_{yy}$ denote the auto-covariance matrix. If the measurements are not exact and the covariance matrix $\mathbf{R}$ of the random measurement error is available, $\mathbf{R}$ is to be added to $\mathbf{C}_{yy}$. The first task is to solve the below system for the Kriging weights $\boldsymbol{\xi}$:

$$\mathbf{C}_{yy}\boldsymbol{\xi} = \mathbf{y} \tag{2}$$

Task-2. With the weights $\boldsymbol{\xi}$ we can obtain the Kriging estimates $\hat{\mathbf{s}}$ (sized $\bar{N} \times 1$) by a superposition of columns of the cross-covariance matrices $\mathbf{C}_{sy}$ (sized $\bar{N} \times N$) weighted by $\boldsymbol{\xi}$, i.e. the Kriging estimate $\hat{\mathbf{s}}$ is given by [31]:

$$\hat{\mathbf{s}} = \mathbf{C}_{sy}\boldsymbol{\xi}\,. \tag{3}$$

Task-3. The variance $\hat{\boldsymbol{\sigma}}^2_{\mathbf{s}}$ of the estimates $\hat{\mathbf{s}}$ is to be obtained from the diagonal of the conditional covariance matrix $\mathbf{C}_{ss|y}$:

$$\begin{aligned}\hat{\boldsymbol{\sigma}}^2_{\mathbf{s}} = \operatorname{diag}(\mathbf{C}_{ss|y}) &= \operatorname{diag}\left(\mathbf{C}_{ss} - \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys}\right) \\ &= \operatorname{diag}\left(\mathbf{C}_{ss}\right) - \sum_{i=1}^{N}\left(\mathbf{C}_{sy}\boldsymbol{\zeta}_i\right)^{\circ 2}, \end{aligned} \tag{4}$$

where $\boldsymbol{\zeta}_i$ is the i-th column of $\mathbf{L}^{-T}$ with $\mathbf{L}$ the lower triangular Cholesky factor matrix of $\mathbf{C}_{yy}$, and the superscript $\circ 2$ denotes Hadamard square.

Task-4. The goal of geostatistical design is to optimize sampling patterns (or locations) for $\mathbf{y}$. There two most common objective functions to be minimized, which are also called A- and C- criteria of geostatistical optimal design [41, 43, 5]:

$$\begin{aligned}\phi_A &= \bar{N}^{-1}\operatorname{trace}\left[\mathbf{C}_{ss|y}\right] \\ \phi_C &= \mathbf{z}^\top\mathbf{C}_{ss|y}\mathbf{z} = \mathbf{z}^\top(\mathbf{C}_{ss} - \mathbf{C}_{sy}\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys})\mathbf{z}\,, \end{aligned} \tag{5}$$

where $\mathbf{z}$ is a data vector [43].

3 Interface from Kriging to FFT-based methods

In this section we give a brief introduction to the basics of FFT-based Kriging [11]. We assume that the measurement points are a subset of the estimate grid points. The simplest version of Kriging is a direct *injection*: the estimated values are set equal to the measurement values at the corresponding locations, and to zeros at all other points. Equivalently, we say that we inject a (small) tensor of measurements into a (larger) tensor of estimations.

For the FFT-based Kriging we use a regular, equispaced grid which leads to a (block) Toeplitz covariance matrix that can be augmented to a circulant one (Section 3.1). An embedding operation augments the injected tensor to the size that is compatible with the circulant covariance matrix. The (pseudo-)inverse of embedding is called extraction (Section 3.2).

3.1 Embedding Toeplitz covariance to circulant matrices

A Toeplitz matrix is constant along each descending diagonal (from left to right). A block Toeplitz matrix has identical sub-matrices in each descending diagonal block and each sub-matrix Toeplitz. If the covariance function is stationary and the estimates are made on a d-dimensional regular, equispaced grid, the covariance matrix $\mathbf{C}_{ss}$ is symmetric level-d block Toeplitz [2]. Since submatrices are repeating along diagonals the required storage could be reduced from $\mathcal{O}(\bar{N}^2)$ to $\mathcal{O}(\bar{N})$ elements [61, 23].

A circulant matrix $\check{\mathbf{C}}$ is a Toeplitz matrix that has its first column $\check{\mathbf{c}}$ periodic. This type of matrices come from covariance functions that are periodic in the domain. A circulant matrix-vector product can be computed efficiently by FFT [57]. The eigenvalues of $\check{\mathbf{C}}$ can be computed as the Fourier transform of its first column $\check{\mathbf{c}}$ [58, 2, pp. 350-354]. These properties lead us to the fast FFT-based kriging methods.

A Toeplitz matrix $\mathbf{C}_{ss}$ can always be augmented to a circulant matrix $\check{\mathbf{C}}$. This process is called *embedding*. Let $\mathbf{C}(:,1)$ be the first column of $\mathbf{C}_{ss}$. Embedding is often done by appending the second through the last but one element of $\mathbf{C}(:,1)$ to the end of $\mathbf{C}(:,1)$ in reverse order, which makes a periodic vector $\check{\mathbf{c}}$. For the cases $d > 1$, this augmentation has to be done recursively in every level for the d-level Toeplitz covariance matrix. An equivalent way of doing this is to augment the domain (to be 2^d times larger) and extend the covariance function to be periodic on the domain, as illustrated in [33, 45]. In [42, 6, 45] the authors have addressed the issue of the minimum embedding size.

3.2 Injection, embedding and extraction of data tensors

Suppose we obtained the Kriging weights $\boldsymbol{\xi}$ for the measurements by solving (2). The injection of $\boldsymbol{\xi}$ means to insert it in a larger all-zero tensor that has the same size of the estimate tensor, i.e. the *injected* tensor has non-zero entries only at the measurement sites.

Suppose we have N measurements indexed by $j = 1, \cdots, N$, each associated with a weight ξ_j and a site index vector $\boldsymbol{\ell}_j$, then the injection of $\boldsymbol{\xi}$ results in a tensor $\bar{\boldsymbol{\xi}} \in \mathbb{R}^{\bar{n}_1 \times \bar{n}_2 \times \cdots \times \bar{n}_d}$ with entries:

$$\bar{\boldsymbol{\xi}}(i_1, i_2, \cdots, i_d) = \begin{cases} \xi_j & \text{if } \boldsymbol{i} = \boldsymbol{\ell}_j, \forall j \in [1, \cdots, N] \\ 0 & \text{otherwise} \end{cases} . \tag{6}$$

We denote the injection operation by $\mathcal{H} : \boldsymbol{\xi} \to \bar{\boldsymbol{\xi}}$.

Embedding an injected weight tensor enhances its mode size from $\bar{n}$ to $\check{n} = 2\bar{n}$ by padding zeros to the extra entries so that the tensor is of 2^d times the original size. The embedded weight tensor $\check{\boldsymbol{\xi}} \in \mathbb{R}^{\check{n}_1 \times \check{n}_2 \times \cdots \times \check{n}_d}$ has entries:

$$\check{\boldsymbol{\xi}}(i_1, i_2, \cdots, i_d) = \begin{cases} \bar{\boldsymbol{\xi}}(i_1, i_2, \cdots, i_d) & \text{if } i_\ell \leq \bar{n}_\ell,\ 1 \leq \ell \leq d \\ 0 & \text{otherwise} \end{cases} . \tag{7}$$

We denote the embedding operation by $\mathcal{M} : \bar{\boldsymbol{\xi}} \to \check{\boldsymbol{\xi}}$.

The extraction is the inverse operation of embedding, we denoted it by $\mathcal{M}^\dagger$. By $\mathcal{M}^\dagger(\boldsymbol{\eta})$ we take only the first half of $\boldsymbol{\eta}$ in every dimension, which results in a new tensor of only $\frac{1}{2^d}$ of the size of $\boldsymbol{\eta}$.

3.3 Matrix-vector multiplication via FFT

With the circulant covariance matrix $\check{\mathbf{C}}$ obtained as explained in Section 3.1, the Task-2 in (3) becomes a discrete convolution which can be computed by using FFT[57], this is written as (e.g., Fritz, Nowak and Neuweiler, [11]):

$$\begin{aligned} \mathbf{C}_{sy}\boldsymbol{\xi} = \mathbf{C}_{ss}\mathcal{H}(\boldsymbol{\xi}) &= \mathcal{M}^\dagger \check{\mathbf{C}} \mathcal{M}(\mathcal{H}(\boldsymbol{\xi})) \\ &= \mathcal{M}^\dagger \mathcal{F}^{[-d]} \left(\mathcal{F}^{[d]} \left(\check{\mathbf{c}} \right) \circ \mathcal{F}^{[d]} \left(\check{\boldsymbol{\xi}} \right) \right) . \end{aligned} \tag{8}$$

where the operation $\mathcal{M}(\mathcal{H}(\cdot))$ injects and embeds $\boldsymbol{\xi}$ into $\check{\boldsymbol{\xi}}$. The $\mathcal{F}^{[d]}$ is evaluated by the Fast Fourier Transformation (FFT) [10]. Without using tensor approximations the computational complexity for Kriging is reduced to $\mathcal{O}\left(\check{N} \log \check{N}\right)$, and the storage size reduced to $\mathcal{O}\left(\check{N}\right)$.

For the variance estimation (Task-3) in (4) the FFT method also applies. We first need to do a Cholesky decomposition $\mathbf{C}_{yy} = \mathbf{L}\mathbf{L}^T$, and inject and embed each column $\boldsymbol{\zeta}_i$ of

$\mathbf{L}^{-T}$ to get the corresponding $\check{\boldsymbol{\zeta}}_i$. Then (4) can be computed as

$$\hat{\boldsymbol{\sigma}}_{\mathbf{s}}^2 = \sigma_s^2 \mathbf{1}_{\bar{N}} - \textstyle\sum_{i=1}^{N} \left[\mathcal{M}^{\dagger}\mathcal{F}^{[-d]}\left(\mathcal{F}^{[d]}\left(\check{\mathbf{c}}\right) \circ \mathcal{F}^{[d]}\left(\check{\boldsymbol{\zeta}}_i\right)\right)\right]^{\circ 2}, \tag{9}$$

where σ_s^2 is the prior variance, $\mathbf{1}_{\bar{N}}$ is a $\bar{N}$-length vector of all ones.

4 FFT-based Kriging accelerated by low-rank tensor decomposition

In addition to the efficient FFT-based method enabled by the Teoplitz structure of covariance matrices, the Kriging process can be further sped up by low-rank representations of the embedded covariance matrices. Since the covariance functions are usually smooth, large covariance matrices could be well approximated by a low-rank tensor format. A literature survey of low-rank tensor approximation techniques is available in [27, 15].

In this section, we approximate the first column of the circulant covariance matrix in tensor train (TT) format and then rewrite 8 also in the TT format. We start with a brief reviewing of the TT technique.

4.1 TT decomposition

We assume that the data vectors ($\mathbf{c}$, $\boldsymbol{\xi}$, etc.) can be associated to a function discretised on a structured grid in d dimensions, for example, if $\xi(x, y, z)$ is sampled on a Cartesian 3-dimensional grid,

$$\boldsymbol{\xi} = \{\xi(x_{i_1}, y_{i_2}, z_{i_3})\}_{i_1,i_2,i_3=1}^{n_1,n_2,n_3}. \tag{10}$$

Then we can enumerate the entries of the vector via sub-indices $i_1, i_2, \dots, i_d$, thereby seeing it as a *tensor* with elements $\boldsymbol{\xi}(i_1, \dots, i_d)$. We approximate such tensors, and, consequently, associated data vectors, in the Tensor Train (TT) decomposition [47],

$$\boldsymbol{\xi}(i_1, i_2, \dots, i_d) \approx \tilde{\boldsymbol{\xi}}(i_1, i_2, \dots, i_d) := \sum_{\alpha_0,\dots,\alpha_d=1}^{r_0,\dots,r_d} \xi^{(1)}_{\alpha_0,\alpha_1}(i_1)\xi^{(2)}_{\alpha_1,\alpha_2}(i_2)\cdots\xi^{(d)}_{\alpha_{d-1},\alpha_d}(i_d). \tag{11}$$

Here $\xi^{(k)}$, $k = 1, \dots, d$, are called *TT blocks*. Each TT block $\xi^{(k)}$ is a three-dimensional tensor of size $r_{k-1} \times n_k \times r_k$, $r_0 = r_d = 1$. The efficiency of this representation relies on the *TT ranks* $r_0, \dots, r_d$ being bounded by a moderate constant r. For simplicity we can also introduce an upper bound of the univariate grid sizes $n_k \leq n$. Then we can notice that the TT format (11) contains at most dnr^2 elements. This is much smaller than the number of entries in the original tensor which grows exponentially in d. Using Kronecker products, one can rewrite (11) as follows,

$$\tilde{\boldsymbol{\xi}} = \sum_{\alpha_0,\dots,\alpha_d=1}^{r_0,\dots,r_d} \xi^{(1)}_{\alpha_0,\alpha_1} \otimes \xi^{(2)}_{\alpha_1,\alpha_2} \otimes \cdots \otimes \xi^{(d)}_{\alpha_{d-1},\alpha_d},$$

i.e. we see each TT block as a set of vectors of length n_k.

Of course, one can think of any other scheme of sampling a function, e.g. at random points, but the TT decomposition requires independence of sub-indices $i_1, \dots, i_d$, and therefore the Cartesian product discretisation. The rationale behind using this, on the first glance excessive, scheme, is the fast convergence of the approximation error ε with the TT ranks. If $\xi(x, y, z)$ is analytic, the TT ranks often depend logarithmically on ε [56, 26, 53]. Combining the TT approximation with collocation on the Chebyshev grid, which allows to take $n = \mathcal{O}(|\log \varepsilon|)$ for analytic functions, one arrives at $\mathcal{O}(d|\log \varepsilon|^3)$ overall cost of interpolation or integration using the TT format. This can be significantly cheaper than the $\mathcal{O}(\varepsilon^{-2})$ cost of Monte Carlo quadrature or Radial Basis function interpolation. Moreover, TT ranks depend usually very mildly on the particular univariate discretisation scheme, provided that it can resolve the function. We can use any univariate grid in each variable instead of the Chebyshev rule. For example, a uniform grid yields Toeplitz or circulant covariance matrices, which are amenable to fast FFT-based multiplication/diagonalisation.

However, it is difficult to obtain sharp bounds for the TT ranks theoretically. Therefore, we resort to robust numerical algorithms to compute a TT approximation of given data.

4.2 TT-cross approximation

A full tensor can be compressed into a TT format quasi-optimally for the desired tolerance via the truncated singular value decomposition (SVD) [47]. However, the full tensor might even be impossible to store. In this section we recall the practical TT-cross method [48] that computes the representation (11) using *only a few* entries from $\boldsymbol{\xi}$. It is based on the skeleton decomposition of a matrix [14], which represents an $n \times m$ matrix A of rank r as the *cross* (in Matlab-like notation)

$$A = A(:, \mathcal{J}) A(\mathcal{I}, \mathcal{J})^{-1} A(\mathcal{I}, :) \tag{12}$$

of r columns and rows, where $\mathcal{I}$ and $\mathcal{J}$ are two index sets of cardinality r such that $A(\mathcal{I}, \mathcal{J})$ (the intersection matrix) is invertible. If $r \ll n, m$, the right-hand side requires only $(n + m - r)r \ll nm$ elements of the original matrix.

In order to describe the TT-cross method, we introduce the so-called *unfolding* matrices $\Xi_k = [\boldsymbol{\xi}(i_1, \dots, i_k; i_{k+1}, \dots, i_d)]$, that have the first k indices grouped together to index rows, and the remaining indices grouped to index columns. Let us now consider Ξ_1 and apply the idea of the matrix cross (12). Assume that there exists a set of r_1 index tuples, $\mathcal{I}_{>1} = \{i_2^{\alpha_1}, \dots, i_d^{\alpha_1}\}_{\alpha_1=1}^{r_1}$, such that the $\mathcal{I}_{>1}$-"columns" of the original tensor $\boldsymbol{\xi}(:, \mathcal{I}_{>1})$ form a "good" basis for all columns of Ξ_1. The reduction (12) may be formed for r_1 rows at positions $\mathcal{I}_{<2} = \{i_1^{\alpha_1}\}_{\alpha_1=1}^{r_1}$, which are now *optimized* by choosing the $r_1 \times r_1$ submatrix $\boldsymbol{\xi}(\mathcal{I}_{<2}, \mathcal{I}_{>1})$ such that its *volume* (modulus of determinant) is maximal. This can be done by the *maxvol* algorithm [13] in $\mathcal{O}(nr_1^2)$ operations. Now we construct the first TT block $\xi^{(1)}$ as the $n \times r_1$ matrix $\boldsymbol{\xi}(:, \mathcal{I}_{>1}) \boldsymbol{\xi}(\mathcal{I}_{<2}, \mathcal{I}_{>1})^{-1}$. In a practical algorithm, the inversion is performed via the QR-decomposition for numerical stability. Next, we reduce the tensor onto $\mathcal{I}_{<2}$ in the first variable, and apply TT-cross inductively to $[\Xi_{>1}(\alpha_1, i_2, \dots, i_d)] = [\boldsymbol{\xi}(i_1^{\alpha_1}, i_2, \dots, i_d)]$.

In the k-th step, assume that we are given the reduction $\Xi_{>k-1}(\alpha_{k-1}, i_k, \dots, i_d)$, a "left" index set $\mathcal{I}_{<k} = \{i_1^{\alpha_{k-1}}, \dots, i_{k-1}^{\alpha_{k-1}}\}_{\alpha_{k-1}=1}^{r_{k-1}}$, and a "right" set $\mathcal{I}_{>k} = \{i_{k+1}^{\alpha_k}, \dots, i_d^{\alpha_k}\}_{\alpha_k=1}^{r_k}$. The $r_{k-1}n \times r_k$ reduced unfolding matrix $[\Xi_{>k-1}(\alpha_{k-1}, i_k;\ \mathcal{I}_{>k})]$ is again feasible for the *maxvol* algorithm, which produces a set of row positions $\ell_k = \{\alpha_{k-1}^{\alpha_k}, i_k^{\alpha_k}\}_{\alpha_k=1}^{r_k}$. The next left set $\mathcal{I}_{<k+1}$ is constructed from ℓ_k by replacing α_{k-1} with the corresponding indexes $i_1^{\alpha_{k-1}}, \dots, i_{k-1}^{\alpha_{k-1}}$ from $\mathcal{I}_{<k}$. Continuing this process until the last variable, where we just copy $\xi^{(d)} = \Xi_{>d-1}$, we complete the induction.

This process can be also organized in a form of a binary tree, which gives rise to the so-called hierarchical Tucker cross algorithm [1]. In total, we need $\mathcal{O}(dnr^2)$ evaluations of $\boldsymbol{\xi}$ and $\mathcal{O}(dnr^3)$ additional operations in computations of the maximum volume matrices.

Algorithm 1 TT cross algorithm with rank adaptation.

```
Require: Initial index sets $\mathcal{I}_{>k}$, rank increasing parameter $\rho \geq 0$, stopping tolerance
    $\delta > 0$ and/or maximum number of iterations $\text{iter}_{\max}$.
Ensure: TT blocks of an approximation (11) to $\boldsymbol{\xi}$.
while $\text{iter} < \text{iter}_{\max}$ and $\|\tilde{\boldsymbol{\xi}}_{\text{iter}} - \tilde{\boldsymbol{\xi}}_{\text{iter}-1}\| > \delta\|\tilde{\boldsymbol{\xi}}_{\text{iter}}\|$ do
    for $k = 1, 2, \dots, d$ do                                          ▷ Forward iteration
        (Optionally) prepare an auxiliary enrichment set $\mathcal{I}^{aux}_{>k}$.
        Compute the $r_{k-1}n \times r_k$ unfolding matrix $\boldsymbol{\xi}(\mathcal{I}_{<k}, i_k;\ \mathcal{I}_{>k})$.
        Compute $\mathcal{I}_{<k+1}$ by the maxvol algorithm and (optionally) truncate.
    end for
    for $k = d, d-1, \dots, 1$ do                                        ▷ Backward iteration
        (Optionally) prepare an auxiliary enrichment set $\mathcal{I}^{aux}_{<k}$.
        Compute the $r_{k-1} \times nr_k$ unfolding matrix $\boldsymbol{\xi}(\mathcal{I}_{<k}\ ; i_k, \mathcal{I}_{>k})$.
        Compute $\mathcal{I}_{>k-1}$ by the maxvol algorithm and (optionally) truncate.
    end for
end while
```

The TT-cross method requires some starting index sets $\mathcal{I}_{>k}$. Without any prior knowledge, it seems reasonable to initialize $\mathcal{I}_{>k}$ with independent realizations of any easy to sample reference distribution (e.g. uniform or Gaussian). If the target tensor $\boldsymbol{\xi}$ admits an *exact* TT decomposition with TT ranks not greater than $r_1, \dots, r_{d-1}$, and all unfolding matrices have ranks not smaller than the TT ranks of $\boldsymbol{\xi}$, the cross iteration outlined above reconstructs $\boldsymbol{\xi}$ *exactly* [48]. However, practical tensors can usually only be *approximated* by a TT decomposition with low ranks. Nevertheless a slight *overestimation* of the ranks can deliver a good approximation, if a tensor was produced from a regular enough function [1, 7].

However, it might be necessary to refine the sets $\mathcal{I}_{<k}, \mathcal{I}_{>k}$ by conducting *several* TT cross iterations, going back and forth over the TT blocks and optimizing the sets by the maxvol algorithm. For example, after computing $\xi^{(d)} = \Xi_{>d-1}$, we "reverse" the algorithm and apply the maxvol method to the *columns* of a $r_{d-1} \times n$ matrix $\xi^{(d)}$. This gives a *refined* set of points $\mathcal{I}_{>d-1} = \{i_d^{\alpha_{d-1}}\}$. The recursion continues from $k = d$ to $k = 1$, optimizing the right sets $\mathcal{I}_{>k}$, while taking the left sets $\mathcal{I}_{<k}$ from the previous (forward) iteration. After several iterations, both $\mathcal{I}_{<k}$ and $\mathcal{I}_{>k}$ can be optimized to the particular target function, even if the starting sets were inaccurate.

This adaptation of points can be combined with the *adaptation of ranks*. If the initial ranks $r_1, \dots, r_{d-1}$ were too large, they can be reduced to quasi-optimal values for the desired accuracy via SVD. However, we can also *increase* the ranks by computing the unfolding matrix $\left[\boldsymbol{\xi}(\mathcal{I}_{<k}, i_k;\ i_{k+1}^{\alpha_k}, \dots, i_d^{\alpha_k})\right]$ on an *enriched* index set: we take $\{i_{k+1}^{\alpha_k}, \dots, i_d^{\alpha_k}\}$ from $\mathcal{I}_{>k}$ for $\alpha_k = 1, \dots, r_k$, and also from an *auxiliary* set $\mathcal{I}^{aux}_{>k}$ for $\alpha_k = r_k + 1, \dots, r_k + \rho$. This increases the k-th TT rank from r_k to $r_k + \rho$. The auxiliary set can be chosen at random [46] or using a surrogate for the error [8]. The pseudocode of the entire TT cross method is listed in Algorithm 1, where we let $\mathcal{I}_{<1} = \mathcal{I}_{>d} = \emptyset$ for uniformity. Empowered with the enrichment scheme, we are not limited to just truncating ranks from above. Instead, we can start with a low-rank initial guess and increase the ranks until the desired accuracy is met.

4.3 TT representation of general and structured matrices

Let us now consider how the TT format (11) can be generalised to matrices $\mathbf{C} \in \mathbb{R}^{n^d \times n^d}$, such as the $\mathbf{C}_{ss}$ matrix from (4). Using sub-indices $i_1, \ldots, i_d$, we can think of a matrix as a $2d$-dimensional tensor with elements $\mathbf{C}(i_1, \ldots, i_d;\ j_1, \ldots, j_d)$. However, most matrices in our applications have full ranks, and a straightforward $2d$-dimensional TT decomposition would be inefficient. Instead, we consider a permuted, or *matrix* TT decomposition [47]:

$$\mathbf{C}(i_1, \ldots, i_d;\ j_1, \ldots, j_d) = \sum_{\beta_0, \ldots, \beta_d = 1}^{R_0, \ldots, R_d} C^{(1)}_{\beta_0, \beta_1}(i_1, j_1) C^{(2)}_{\beta_1, \beta_2}(i_2, j_2) \cdots C^{(d)}_{\beta_{d-1}, \beta_d}(i_d, j_d), \qquad (13)$$

or in the Kronecker form,

$$\mathbf{C} = \sum_{\beta_0, \ldots, \beta_d = 1}^{R_0, \ldots, R_d} C^{(1)}_{\beta_0, \beta_1} \otimes C^{(2)}_{\beta_1, \beta_2} \otimes \cdots \otimes C^{(d)}_{\beta_{d-1}, \beta_d}. \qquad (14)$$

The identity matrix can be trivially represented in matrix TT format $I_{n^d} = I_n \otimes \cdots \otimes I_d$ with $R_0 = \cdots = R_d = 1$. Furthermore, we can quickly assemble block Toeplitz and circulant matrices if their first column/row is given in the TT format [24]. Let us introduce the operation $\mathcal{T} : \mathbb{R}^{2n} \to \mathbb{R}^{n \times n}$ which assembles a Toeplitz matrix from a vector of its first column and row stacked together, and the operation $\mathcal{C} : \mathbb{R}^{n} \to \mathbb{R}^{n \times n}$ which assembles a circulant matrix from its first column. Assume that a vector $\mathbf{c}$ of size $(2n)^d$ or a vector $\check{\mathbf{c}}$ of size n^d are given in the TT format (11),

$$\mathbf{c} = \sum_{\alpha_0, \ldots, \alpha_d = 1}^{r_0, \ldots, r_d} c^{(1)}_{\alpha_0, \alpha_1} \otimes \cdots \otimes c^{(d)}_{\alpha_{d-1}, \alpha_d}, \quad \check{\mathbf{c}} = \sum_{\alpha_0, \ldots, \alpha_d = 1}^{r_0, \ldots, r_d} \check{c}^{(1)}_{\alpha_0, \alpha_1} \otimes \cdots \otimes \check{c}^{(d)}_{\alpha_{d-1}, \alpha_d} \qquad (15)$$

Then the block Toeplitz or circulant matrix, respectively

$$\mathbf{C} = \left(\bigotimes_{k=1}^{d} \mathcal{T} \right) \mathbf{c}, \qquad \check{\mathbf{C}} = \left(\bigotimes_{k=1}^{d} \mathcal{C} \right) \check{\mathbf{c}},$$

can be written in the matrix TT formats (13) with the same TT ranks,

$$\mathbf{C} = \sum_{\alpha_0, \ldots, \alpha_d = 1}^{r_0, \ldots, r_d} \left(\mathcal{T} c^{(1)}_{\alpha_0, \alpha_1}\right) \otimes \cdots \otimes \left(\mathcal{T} c^{(d)}_{\alpha_{d-1}, \alpha_d}\right), \qquad \check{\mathbf{C}} = \sum_{\alpha_0, \ldots, \alpha_d = 1}^{r_0, \ldots, r_d} \left(\mathcal{C} \check{c}^{(1)}_{\alpha_0, \alpha_1}\right) \otimes \cdots \otimes \left(\mathcal{C} \check{c}^{(d)}_{\alpha_{d-1}, \alpha_d}\right).$$

Similarly we can apply the multivariate Fourier transform without changing TT ranks:

$$\left(\bigotimes_{k=1}^{d} \mathcal{F} \right) \mathbf{c} = \sum_{\alpha_0, \ldots, \alpha_d = 1}^{r_0, \ldots, r_d} \left(\mathcal{F} c^{(1)}_{\alpha_0, \alpha_1}\right) \otimes \cdots \otimes \left(\mathcal{F} c^{(d)}_{\alpha_{d-1}, \alpha_d}\right), \qquad (16)$$

where $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$ is the univariate FFT. This reduces the complexity of FFT from $\mathcal{O}(N \log N) = \mathcal{O}(dn^d \log n)$ to $\mathcal{O}(dr^2 n \log n)$.

In general, the TT format allows to represent the product of any matrix given in (13) and a compatible vector given in (11) in another TT format [47] with multiplied ranks,

$$\mathbf{C}\boldsymbol{\xi} = \sum_{\gamma_0, \ldots, \gamma_d = 1}^{(r_0 R_0), \ldots, (r_d R_d)} \left(C^{(1)}_{\beta_0, \beta_1} \xi^{(1)}_{\alpha_0, \alpha_1} \right)_{\gamma_0, \gamma_1} \otimes \cdots \otimes \left(C^{(d)}_{\beta_{d-1}, \beta_d} \xi^{(d)}_{\alpha_{d-1}, \alpha_d} \right)_{\gamma_{d-1}, \gamma_d}, \qquad (17)$$

where $\gamma_k = \alpha_k + (\beta_k - 1) r_k$, $k = 0, \ldots, d$.

4.4 Kriging operations in TT format

To rewrite the Kriging estimation (8) in low rank format, we first find a TT approximation (15) of $\mathbf{c}$ by using the TT-cross algorithm introduced in Section 4.2. With the rest of the operations we can proceed in two ways.

4.4.1 Small number of scattered samples

If we assume N to be small, the Task-1 of computing Kriging weights, $\mathbf{C}_{yy}\boldsymbol{\xi} = \mathbf{y}$, can be computed directly at low cost. Now we inject the scattered values into a TT tensor of desired size as introduced in (6). Suppose $\boldsymbol{\ell}_j \in \mathbb{N}^d$ is the position of the jth sample, $j = 1, \dots, N$, we can define

$$\mathbf{H}_j = \bigotimes_{k=1}^{d} \mathbf{e}_j^{(k)}, \quad \text{where} \quad \mathbf{e}_j^{(k)}(i_k) = \begin{cases} 1, & i_k = \ell_j(k) \\ 0, & \text{otherwise,} \end{cases}$$

i.e. the injection operation (6) per sample. Now the injected tensor is written in the CP format as

$$\bar{\boldsymbol{\xi}} = \sum_{j=1}^{N} \xi_j \mathbf{H}_j, \tag{18}$$

which can be converted to TT format directly by the formula in [16, pp. 380] or using the Alternating Least Squares (ALS) [19] approximation.

Similarly, we can use the direct truncation or the ALS method for summing columns of $\mathbf{C}_{sy}$ with the weights $\boldsymbol{\zeta}_i$ in (4), as well as the summation of different vectors $(\mathbf{C}_{sy}\boldsymbol{\zeta}_i)^{\circ 2}$.

Embedding operation (7) is simpler and more efficient: we just need to pad every TT block with zeros. Assuming we are given a vector $\boldsymbol{\xi}$ in the form (11), we construct the following new TT blocks of a vector $\check{\boldsymbol{\xi}}$:

$$\check{\xi}^{(k)}_{\alpha_{k-1},\alpha_k}(i_k) = \begin{cases} \xi^{(k)}_{\alpha_{k-1},\alpha_k}(i_k), & i_k = 1, \dots, \bar{n}_k, \\ 0, & i_k = \bar{n}_k + 1, \dots, n_k, \end{cases} \qquad k = 1, \dots, d. \tag{19}$$

Similarly, Extraction operation is performed by truncating the range of i_k in each TT block from n_k back to $\bar{n}_k$. Most importantly, embedding and extraction can be performed very efficiently without changing the TT ranks, similarly to FFT (16).

Finally, we need to compute the Hadamard products of TT tensors, e.g. $\mathcal{F}^{[d]}(\check{\mathbf{c}}) \circ \mathcal{F}^{[d]}(\check{\boldsymbol{\xi}})$ in (8). The Hadamard product can be constructed exactly via (17) by noticing that

$$\mathbf{s} := \mathbf{c} \circ \boldsymbol{\xi} = \mathbf{C}\boldsymbol{\xi}, \quad \text{for} \quad \mathbf{C} = \operatorname{diag}(\mathbf{c}),$$

or approximately by applying the TT-Cross algorithm to a tensor given elementwise by the formula $\mathbf{s}(i_1, \dots, i_d) = \mathbf{c}(i_1, \dots, i_d)\boldsymbol{\xi}(i_1, \dots, i_d)$. The direct multiplication requires $\mathcal{O}(dnR^2r^2)$ operations, and the truncation afterwards has an even higher cost $\mathcal{O}(dnR^3r^3)$. In contrast, the TT-Cross approach needs computing $\mathcal{O}(dnr^2)$ samples of the target tensor $\mathbf{s}$, which means taking samples of the TT decompositions for $\mathbf{c}$ and $\boldsymbol{\xi}$ and multiplying them. Sampling another TT tensor requires in total $\mathcal{O}(dnR^2r)$ operations, which, assuming that the ranks are comparable, $R \sim r$, results in a total of $\mathcal{O}(dnr^3)$ operations in the TT-Cross computation of Hadamard products, which is thus preferred in this paper.

For geostatistical optimal design (Task-4) we need to compute the trace of $\mathbf{C}_{ss|y}$. Since in the Task-3 we obtain already the diagonal of $\mathbf{C}_{ss|y}$ in the TT format, the trace can be evaluated swiftly by computing a dot product with the all-ones tensor.

4.4.2 Large number of structured samples

When N is large, the summation (18) can be a difficult operation in the TT format, potentially leading also to the TT ranks being in the order of N. However, a large number of samples usually means that these samples are distributed fairly uniformly in the domain of interest. In this case, we switch to the TT computations even before Task-1 in equation (2). First, we interpolate the given samples onto a uniform Cartesian grid with the mesh interval being in the order of the average distance between the original samples. In the remaining operations, we assume that $\mathbf{y}$ is structured in this way, i.e. it can be seen as a tensor $\mathbf{y}(i_1, \dots, i_d)$, $i_k = 1, \dots, \bar{m}_k$, $k = 1, \dots, d$. Thus, we can approximate $\mathbf{y}$ in the TT format.

The solution for weights (2) becomes a rather difficult operation for a large N. However, given the TT decompositions for $\mathbf{y}$ and $\mathbf{C}_{yy}$, the linear system can be solved more efficiently by employing ALS and similar tensor algorithms [19, 8]. Similarly, we can compute $\mathbf{C}_{yy}^{-1}\mathbf{C}_{ys}$ for (4) by treating $\mathbf{C}_{ys}$ as the right hand side, and expanding $\mathbf{C}_{yy}$ accordingly.

If we interpolate $\mathbf{y}$ onto a periodic uniform Cartesian grid, the matrix $\mathbf{C}_{yy}$ becomes circulant, similarly to $\check{\mathbf{C}}$. In this case we can approximate only its first column in the TT format, perform the Fourier transform to obtain the eigenvalues, and apply again the TT-Cross method to approximate the pointwise division $\mathcal{F}^{[d]}(\mathbf{y})(i_1, \dots, i_d)/\mathcal{F}^{[d]}(\mathbf{c})(i_1, \dots, i_d)$.

5 Numerical tests

We used the Matlab package *TT-Toolbox* (`https://github.com/oseledets/TT-Toolbox`) for Tensor Train algorithms. The codes used for numerical experiments are available at `https://github.com/dolgov/TT-FFT-COV`. All computations are done on a MacBook Pro produced in 2013, equipped with 16GB RAM and an 2.7 GHz Intel Core i7 CPU.

We consider three test cases: 1) a 2-dimensional problem with $N = \prod_{i=1}^{2} n_i = 600^2$ (it is easy to visualize); 2) a 3-dimensional problem with $N = 10^{15}$ and 3) 10-dimensional problem with $N = \prod_{i=1}^{10} n_i = 100^{10}$. One of these parameters could be, for example, time. The daily soil moisture data set, used below, is taken from [20, 37, 38], where only one replicate, sampled at N locations, is used.

5.1 Kriging of daily moisture data

Numerical models play important role in climate studies. These numerical models are complicated and high-dimensional, including such variables as pressure, temperature, speed, and direction of the wind, level of precipitation, humidity, and moisture. Many parameters are uncertain or even unknown. Accurate modeling of soil moisture finds applications in the agriculture, weather prediction, early warnings of flood and in some others. Since the underlined geographical areas are usually large and high spatial resolutions are required, the involved data sets are huge. This could make the computational process in dense matrix format unfeasible or very expensive. By involving efficient low-rank tensor calculus, we can increase the spatial and time resolution and consider more parameters. It is clear that utilization of the rank k tensor approximation introduces an additional numerical error in quantities of interest (QoIs). By increasing tensor ranks we reduce this approximation error.

We consider high-resolution soil moisture data from January 1, 2014, measured in the topsoil layer of the Mississippi River basin, U.S.A (Fig. 1).

Figure 2 shows an example of daily moisture data. On the left picture we used 2000

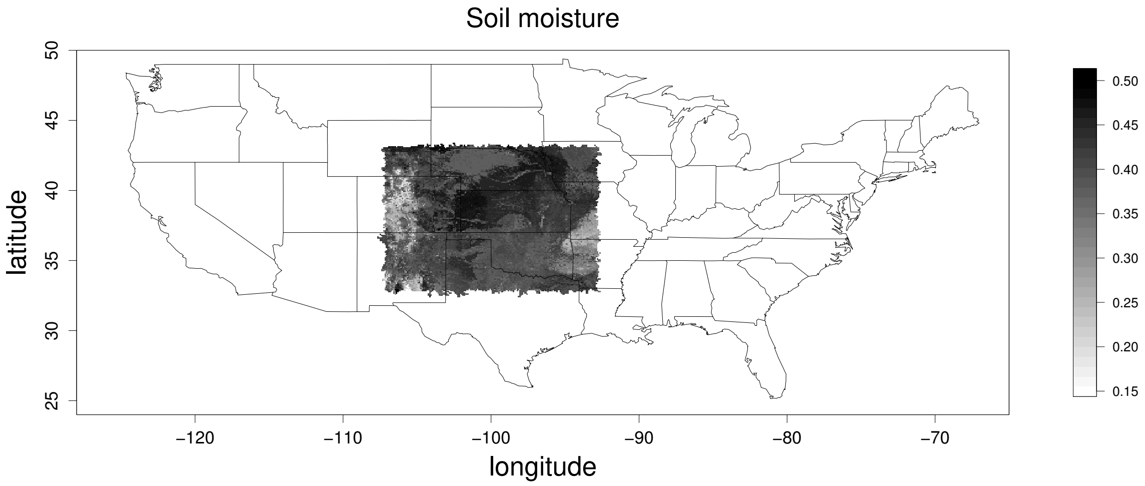


Figure 1: The area where the daily soil moisture data were measured, Mississippi River basin, U.S.A.

points $(x, y, v)_{i=1}^{N}$, $N = 2000$ for interpolation, and on the right 4000 points. The third picture shows two set of locations: one with 2000 points, marked with the blue symbol + and with 4000 points, marked with red dot.

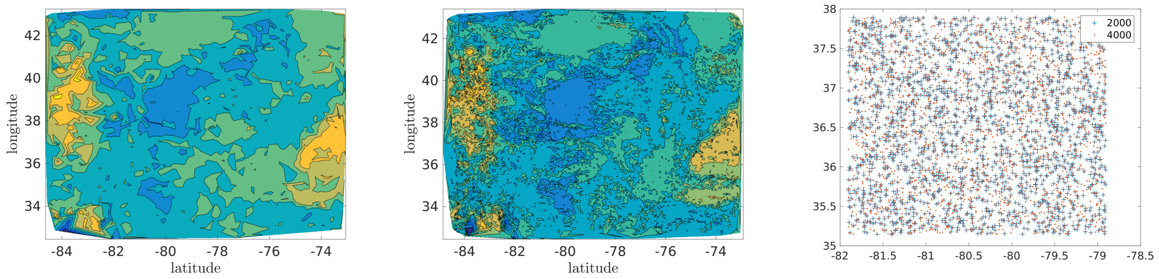


Figure 2: Daily moisture data. Interpolated from (left) 2000 and (center) 40000 measurement points. (right) Two sets of sampling points, 2000 and 4000.

The spatial resolution is 0.0083 degrees, and the distance of one-degree difference in this region is approximately 87.5 km. The grid consists of $1830 \times 1329 = 2.432.070$ locations with 2.000.000 observations and 432.070 missing values. Therefore, the available spatial data are not on a regular grid.

The tensor product Kriging is performed as described in Sec. 4.4.2. First, we interpolate the given measurements (Fig. 3, left) onto a (coarse) Cartesian grid with the mesh interval being approximately equal to the average distance between the measurements. Specifically, we ended up with a 65×65 grid (Fig. 3, center). Then the tensor of values on this coarse grid is approximated into a TT decomposition. Finally, the Kriging estimate (2)–(3) on a fine grid with 257×257 points (Fig. 3, right) is computed in the TT format using FFT and TT-Cross algorithms.

5.2 High-dimensional field generation: computational benchmark

To generate the following 2D, 3D and 10D random fields we used the Matlab script test_generate_y_tt.m in `https://github.com/dolgov/TT-FFT-COV`.

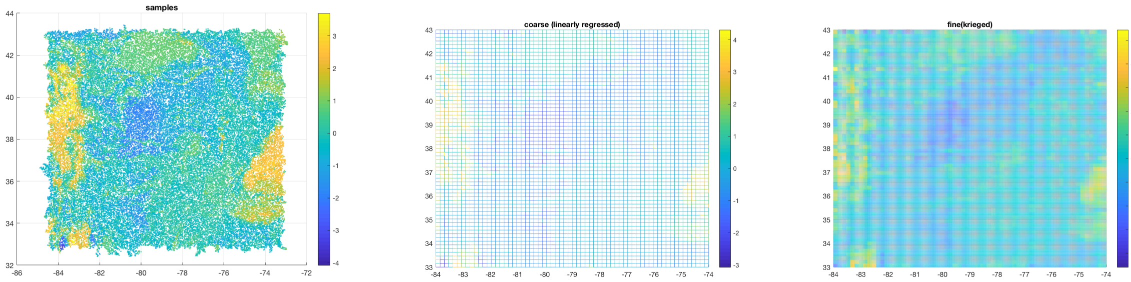


Figure 3: (left) 64000 measurements of the moisture; (center) regression on a coarse 65 × 65 Cartesian mesh; (right) TT-Kriging approximation on a fine mesh.

2D example. In this example we generated a high-resolution 2-dimensional Matérn random field in $[0, 2000]^2$. One realization is presented in Fig. 4. The smoothness of the Matérn field is $\nu = 0.4$, covariance lengths in x and y directions $(1, 1)$ and the variance 10. This realization is computed by the following formula in the TT format

$$\boldsymbol{u}' = \mathbf{C}^{1/2}\boldsymbol{\xi} = \sqrt{\frac{1}{n}}\mathbf{F}^\top \boldsymbol{\Lambda}^{1/2}\boldsymbol{\xi} = \sqrt{\frac{1}{n}}\mathcal{F}^{-1}(\boldsymbol{\lambda}^{1/2} \circ \boldsymbol{\xi}), \tag{20}$$

where the inverse Fourier $\mathcal{F}^{-1}$, the square root of eigenvalues $\boldsymbol{\lambda}^{1/2}$, and tensor product $\boldsymbol{\xi}$ of two Gaussian random vectors are approximated in the TT format. Particularly, $\boldsymbol{\xi} = \boldsymbol{\xi}_1 \otimes \boldsymbol{\xi}_2$ is a tensor product of two Gaussian vectors. The size of the first column $\check{\mathbf{c}}$ of $\check{\mathbf{C}}$ is 3200×3600 and the computing time was 1 sec. With TT procedures one can create very fine resolved random fields in large domains. For instance, generation of a random field in the domain $[0, 1.000.000]^2$ with $1.600.000 \times 1.800.000$ locations takes less than 1 minute.

3D example. This example is very similar to the previous 2D example. The difference is only that the domain is $[0, 100.000]^3$ and the size of the first column of $\mathbf{C}$ is $160.000 \times 180.000 \times 160.000 = 4.608 \cdot 10^{15}$. The computing time was 3 minutes.

10D example. In this example, we generated a 10-dimensional Matérn random field. One of the dimensions could be time, for example. Table 1 contains all model parameters and the number of unknowns in (hypothetical) full tensor and in the TT decomposition of the final field $\hat{\mathbf{s}}$. In this example we computed TT approximation of the first column of the multilevel circulant covariance matrix (cf. [24, 25]). Then we diagonalized this circulant matrix via FFT and computed square root of diagonal elements. After that we generated a random field by multiplying the square root with a random vector of the following structure $\boldsymbol{\xi} := \bigotimes_{\nu=1}^{10} \boldsymbol{\xi}_\nu$, where $\boldsymbol{\xi}_\nu$ is a normal vector. We note that we never store the whole vector $\boldsymbol{\xi}$ explicitly, but only it's tensor components $\boldsymbol{\xi}_\nu$. Also, note that $\boldsymbol{\xi}$ is not Gaussian.

The TT approximation tolerance is set to 10^{-4}. In the 10-dimensional case above the maximal rank was 143, and the total computing time 118 sec. In the similar 8-dimensional case the maximal rank was 138, and the total computing time 96 sec. Of course, one should observe tensor ranks not only of $\hat{\mathbf{s}}$, but of other steps such as the TT approximation of the measurement vector and of the first column of the covariance matrix. These TT ranks were smaller than the TT ranks of the final solution though.

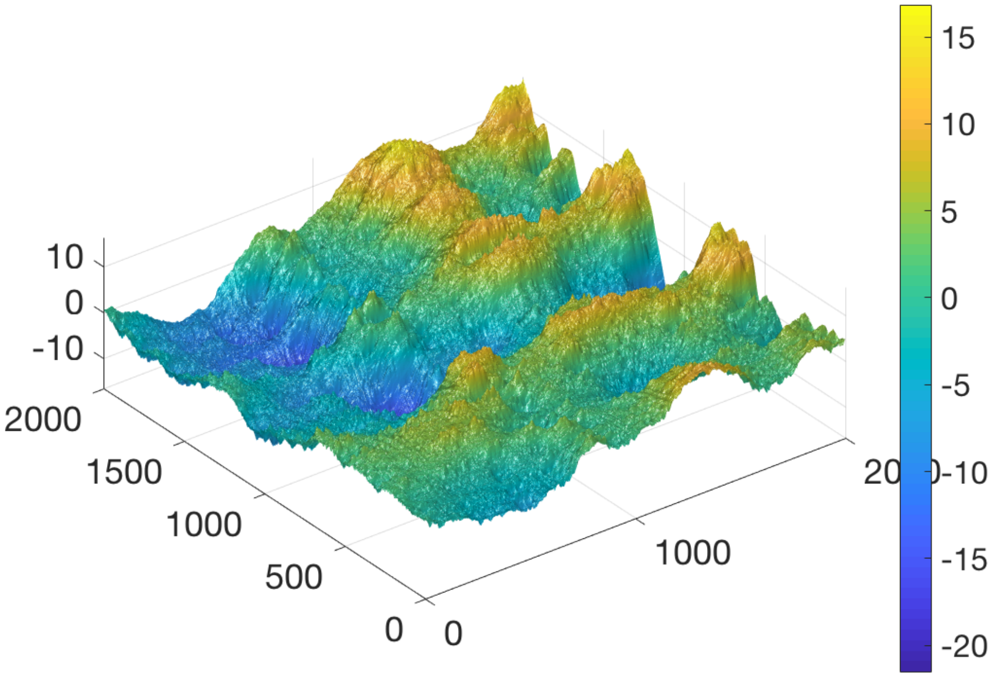


Figure 4: High-resolution realization of 2D Matérn random field, computed with TT tensor format in $[0, 2000]^2$.

Table 1: Parameters of the 10-dimensional problem.

parameter	value
variance of model	10
vector of correlation length in $x_1, \ldots, x_{10}$-direction	$[1, 5, 10, 15, 20, 25, 30, 35, 40, 45]$
length of domain in $x_1, \ldots, x_{10}$-direction	$[10, 50, 100, 150, 200, 250, 300, 350, 400, 450]$
number of elements in $x_1, \ldots, x_{10}$-direction	$[100, 100, 100, 100, 100, 100, 100, 100, 100, 100]$
number of elements in original tensor	$100^{10} = 10^{20}$
number of elements in TT tensor	10^7

6 Discussion and Conclusions

In this paper, we proposed an FFT-based Kriging that utilizes a low-rank Tensor Train (TT) approximation of the covariance matrix. We apply the TT-Cross algorithm to generate a low-rank decomposition avoiding full tensors which could be well beyond the memory capacity of a desktop PC.

The low-rank format reduces the storage of the embedded circulant covariance matrix from exponential to linear in the number of variables. The circulant matrix can be diagonalized by FFT. Furthermore, due to the linearity of the Fourier transform, the TT format allows to implement the d-dimensional FFT at the cost of $\mathcal{O}(dr^2)$ one-dimensional FFT operations.

We then use the same technique to generate large Matérn random fields since the diagonalized covariance matrix gives eigen pairs for the spectral expansion of the underlying random field. We show in numerical examples that this method can generate very large random fields with a commonly affordable computational resource.

We demonstrated how to utilize the TT tensor format to speed up such geostatistical tasks as the generation of large random fields, computing kriging coefficients, kriging estimates, conditional covariance, and geostatistical optimal design. We used the fact that after discretization on a tensor grid the obtained matrix could be extended to a circulant one. Then, much expensive linear algebra operation could be done via d-dimensional FFT. From the definition, one can see that FFT has tensor rank 1. After approximating the first column of the circulant matrix in the TT format (we assumed that such approximation exists) we were able to apply efficient TT tensor arithmetics and speedup expensive calculations even more. Utilizing TT format in FFT calculus allowed us to decrease computational cost and storage from $\mathcal{O}(\bar{N}\log\bar{N})$ to $\mathcal{O}(dr^3\bar{n})$, where $r \geq 1$ is the tensor rank, d the dimensionality of the problem and $\bar{n}$ is the number of points along the single longest edge of the estimation grid.

The presented numerical techniques have memory requirements as low as $\mathcal{O}\left(d\bar{n}r^2\right)$. Thus, we achieved log-complexity in the total number of lattice points. The resulting methods allow much better spatial resolution and significantly reduce the computing time.

The fundamental assumptions are: the covariance matrix is separable or has a TT-rank $r \ll n$, the interpolation grid is a rectangular tensor grid, and the measurements also lie in the tensor grid. The random vector used to generate the random field is a Kronecker product of smaller random vectors.

Acknowledgments

The research reported in this publication was supported by funding from the Alexander von Humboldt Foundation. We also would like to thank Wolfgang Nowak for sharing his Matlab code.

REFERENCES

[1] J. Ballani and L. Grasedyck. Hierarchical tensor approximation of output quantities of parameter-dependent PDEs. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):852–872, 2015.

[2] S. Barnett. *Matrices Methods and Applications.* Oxford Applied Mathematics and Computing Science Series. Clarendon Press, Oxford, 1990.

[3] P. Bogaert. Comparison of kriging techniques in a space-time context. *Mathematical Geology*, 28(1):73–86, 1996.

[4] R. H. Chan and M. K. Ng. Conjugate gradient methods for Toeplitz systems. *SIAM Review*, 38(3):427–482, 1996.

[5] O. A. Cirpka and W. Nowak. First-order variance of travel time in non-stationary formations. *Water Resour. Res.*, 40(3):W03507, 2004.

[6] C. R. Dietrich and G. N. Newsam. Fast and exact simulation of stationary Gaussian processes through: Circulant embedding of the covariance matrix. *SIAM J. Sci. Comput.*, 18(4):1088–1107, 1997.

[7] S. Dolgov and R. Scheichl. A hybrid Alternating Least Squares – TT Cross algorithm for parametric PDEs. arXiv preprint 1707.04562, 2017.

[8] S. V. Dolgov and D. V. Savostyanov. Alternating minimal energy methods for linear systems in higher dimensions. *SIAM J. Sci. Comput.*, 36(5):A2248–A2271, 2014.

[9] P. A. Finke, D. J. Brus, M. F. P. Bierkens, T. Hoogland, M. Knotters, and F. De Vries. Mapping groundwater dynamics using multiple sources of exhaustive high resolution data. *Geoderma*, 123(1):23–39, 2004.

[10] M. Frigo and S. G. Johnson. FFTW: An adaptive software architecture for the FFT. In *Proc. ICASSP*, volume 3, pages 1381–1384, IEEE, Seattle, WA, 1998. http://www.fftw.org.

[11] J. Fritz, W. Nowak, and I. Neuweiler. Application of FFT-based algorithms for large-scale universal Kriging problems. *Math. Geosci.*, 41(5):509–533, 2009.

[12] M. G. Genton. Separable approximations of space-time covariance matrices. *Environmetrics*, 18:681–695, 2007.

[13] S. A. Goreinov, I. V. Oseledets, D. V. Savostyanov, E. E. Tyrtyshnikov, and N. L. Zamarashkin. How to find a good submatrix. In V. Olshevsky and E. Tyrtyshnikov, editors, *Matrix Methods: Theory, Algorithms, Applications*, pages 247–256. World Scientific, Hackensack, NY, 2010.

[14] S. A. Goreinov, E. E. Tyrtyshnikov, and N. L. Zamarashkin. A theory of pseudoskeleton approximations. *Linear Algebra Appl.*, 261:1–21, 1997.

[15] L. Grasedyck, D. Kressner, and C. Tobler. A literature survey of low-rank tensor approximation techniques. *GAMM-Mitt.*, 36(1):53–78, 2013.

[16] W. Hackbusch. *Elliptic differential equations*, volume 18 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1992. Theory and numerical treatment, Translated from the author's revision of the 1986 German original by Regine Fadiman and Patrick D. F. Ion.

[17] W. Hackbusch. *Tensor Spaces and Numerical Tensor Calculus*. Springer Series in Computational Mathematics. Springer Verlag, 2012.

[18] M. R. Haylock, N. Hofstra A. M. G. Klein Tank, E. J. Klok, P. D. Jones, and M. New. A european daily high-resolution gridded data set of surface temperature and precipitation for 1950–2006. *J. Geophys. Res*, 113:D20119, 2008.

[19] S. Holtz, T. Rohwedder, and R. Schneider. The alternating linear scheme for tensor optimization in the tensor train format. *SIAM J. Sci. Comput.*, 34(2):A683–A713, 2012.

[20] H. Huang and Ying S. Hierarchical low rank approximation of likelihoods for large spatial datasets. *Journal of Computational and Graphical Statistics*, 27(1):110–118, 2018.

[21] S. De Iaco, S. Maggio, M. Palma, and D. Posa. Toward an automatic procedure for modeling multivariate space-time data. *Computers & Geosciences*, 41:1–11, 2011.

[22] A. G. Journel and C. J. Huijbregts. *Mining Geostatistics.* Academic Press, New York, 1978.

[23] T. Kailath and A. H. Sayed. Displacement structure: Theory and applications. *SIAM Review*, 37(3):297–386, 1995.

[24] V. Kazeev, B. Khoromskij, and E. Tyrtyshnikov. Multilevel Toeplitz matrices generated by tensor-structured vectors and convolution with logarithmic complexity. *SIAM J. Sci. Comput.*, 35(3):A1511–A1536, 2013.

[25] V. Khoromskaia and B. N. Khoromskij. Block circulant and toeplitz structures in the linearized hartree–fock equation on finite lattices: Tensor approach. *Computational Methods in Applied Mathematics*, 17(3):431–455, 02 2017.

[26] B. N. Khoromskij. Structured rank-$(r_1, \ldots, r_d)$ decomposition of function-related operators in $\mathbb{R}^d$. *Comput. Methods Appl. Math*, 6(2):194–220, 2006.

[27] B. N. Khoromskij. Tensor-structured numerical methods in scientific computing: Survey on recent advances. *Chemom. Intell. Lab. Syst.*, 110(1):1–19, 2012.

[28] B. N. Khoromskij. *Tensor numerical methods in scientific computing.* Walter de Gruyter GmbH & Co KG, 2018.

[29] B. N. Khoromskij and A. Litvinenko. Data sparse computation of the karhunen-loève expansion. *AIP Conference Proceedings*, 1048(1):311–314, 2008.

[30] P. K. Kitanidis. Analytical expressions of conditional mean, covariance, and sample functions in geostatistics. *Stoch. Hydrol. Hydraul.*, 12:279–294, 1996.

[31] P. K. Kitanidis. *Introduction to Geostatistics.* Cambridge University Press, Cambridge, 1997.

[32] J. B. Kollat, P. M. Reed, and J. R. Kasprzyk. A new epsilon-dominance hierarchical bayesian optimization algorithm for large multiobjective monitoring network design problems. *Adv. in Water Res.*, 31(5):828 – 845, 2008.

[33] B. Kozintsev. *Computations with Gaussian random fields.* PhD thesis, Institute for Systems Research, University of Maryland, 1999.

[34] P. Kyriakidis and A. Journel. Geostatistical space–time models: A review. *Mathematical Geology*, 31:651–684, 1999. doi:10.1023/A:1007528426688.

[35] A. Litvinenko. HLIBCov: Parallel Hierarchical Matrix Approximation of Large Covariance Matrices and Likelihoods with Applications in Parameter Identification. *arXiv 1709.08625*, Sep 2017.

[36] A. Litvinenko, D. Keyes, V. Khoromskaia, B. N. Khoromskij, and H. G. Matthies. Tucker tensor analysis of matérn functions in spatial statistics. *Computational Methods in Applied Mathematics*, 19(1):101–122, 2019.

[37] A. Litvinenko, Y. Sun, M. G. Genton, and D. E. Keyes. Likelihood approximation with hierarchical matrices for large spatial datasets. *Computational Statistics & Data Analysis*, 137:115 – 132, 2019.

[38] A. Litvinenko, Y. Sung, H. Huang, M. G. Genton, and D. E. Keyes. Github repository: daily moisture data, https://github.com/litvinen/HLIBCov.git, 2017.

[39] B. Matérn. *Spatial variation.* Springer, Berlin, Germany, 1986.

[40] G. Matheron. *The Theory of Regionalized Variables and Its Applications.* Ecole de Mines, Fontainebleau, France, 1971.

[41] W. G. Müller. *Collecting spatial data. Optimum design of experiments for random fields.* Springer, Berlin, Germany, 3 edition, 2007.

[42] G. N. Newsam and C. R. Dietrich. Bounds on the size of nonnegative definite circulant embeddings of positive definite Toeplitz matrices. *IEEE Transactions on Information Theory*, 40(4):1218–1220, 1994.

[43] W. Nowak. Measures of parameter uncertainty in geostatistical estimation and geostatistical optimal design. *Math. Geosciences*, 42(2):199–221, 2010.

[44] W. Nowak and A. Litvinenko. Kriging and spatial design accelerated by orders of magnitude: Combining low-rank covariance approximations with fft-techniques. *Mathematical Geosciences*, 45(4):411–435, May 2013.

[45] W. Nowak, S. Tenkleve, and O. A. Cirpka. Efficient computation of linearized cross-covariance and auto-covariance matrices of interdependent quantities. *Math. Geol.*, 35(1):53–66, 2003.

[46] I. V. Oseledets. DMRG approach to fast linear algebra in the TT–format. *Comput. Meth. Appl. Math.*, 11(3):382–393, 2011.

[47] I. V. Oseledets. Tensor-train decomposition. *SIAM J. Sci. Comput.*, 33(5):2295–2317, 2011.

[48] I. V. Oseledets and E. E. Tyrtyshnikov. TT-cross approximation for multidimensional arrays. *Linear Algebra Appl.*, 432(1):70–88, 2010.

[49] G. G. S. Pegram. Spatial interpolation and mapping of rainfall (SIMAR) Vol.3: Data merging for rainfall map production. *Water Research Commission Report*, (1153/1/04), 2004.

[50] L. Pesquer, A. Cortés, and X. Pons. Parallel ordinary kriging interpolation incorporating automatic variogram fitting. *Computers & Geosciences*, 37(4):464–473, 2011.

[51] P. Reed, B. Minsker, and A. J. Valocchi. Cost-effective long-term groundwater monitoring design using a genetic algorithm and global mass interpolation. *Water Resour. Res.*, 36(12):3731–3741, 2000.

[52] A. K. Saibaba and P. K. Kitanidis. Efficient methods for large-scale linear inversion using a geostatistical approach. *Water Resour. Res.*, 48(W05522), 2012.

[53] R. Schneider and A. Uschmajew. Approximation rates for the hierarchical tensor format in periodic sobolev spaces. *Journal of Complexity*, 2013.

[54] R. Shah and P. M. Reed. Comparative analysis of multiobjective evolutionary algorithms for random and correlated instances of multiobjective d-dimensional knapsack problems. *European Journal of Operational Research*, 211(3):466 – 479, 2011.

[55] G. Spöck and J. Pilz. Spatial sampling design and covariance-robust minimax prediction based on convex design ideas. *Stochastic Environmental Research and Risk Assessment*, 24:463–482, 2010.

[56] E. E. Tyrtyshnikov. Tensor approximations of matrices generated by asymptotically smooth functions. *Sbornik: Mathematics*, 194(6):941–954, 2003.

[57] C. F. van Loan. *Computational Frameworks for the Fast Fourier Transform.* SIAM Publications, Philadelphia, PA, 1992.

[58] R. S. Varga. Eigenvalues of circulant matrices. *Pacific J. Math.*, 4:151–160, 1954.

[59] J. Vondřejc, D. Liu, M. Ladecký, and H. G. Matthies. FFT-based homogenisation accelerated by low-rank approximations. *arXiv e-prints*, page arXiv:1902.07455, Feb 2019.

[60] S. M. Wesson and G. G. S. Pegram. Radar rainfall image repair techniques. *Hydrological and Earth Systems Sciences*, 8(2):8220–8234, 2004.

[61] D. L. Zimmerman. Computationally exploitable structure of covariance matrices and generalized covariance matrices in spatial models. *J. Stat. Comput. Sim.*, 32(1/2):1–15, 1989.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

BAYESIAN UPDATING OF CABLE STAYED FOOTBRIDGE MODEL PARAMETERS USING DYNAMIC MEASUREMENTS

C. Pepi[1], M. Gioffré[2], M.D. Grigoriu[3], and H.G. Matthies[4]

[1,2]Dept. of Civil and Environmental Engineering, Univ. of Perugia
via G. Duranti 93, 06125 Perugia, Italy
e-mail: {chiara.pepi,massimiliano.gioffre}@unipg.it

[3] Dept. of Civil and Environmental Engineering, Cornell University
363 Hollister Hall, Ithaca, NY 14853
e-mail: mdg12@cornell.edu

[4] Institute of Scientific Computing, Technische Universität Braunschweig
Braunschweig, 38106 - Germany
e-mail: wire@tu-bs.de

Keywords: Cable stayed footbridge, Bayesian Inference, Uncertainty Quantification, Polynomial Chaos Expansion, Surrogate Models.

Abstract. *The topic of model updating has been the focus of intensive research since it is a useful mean for reliable predictions of the structural performance of dynamic systems. The differences between the output of the Finite Element (FE) model and the modal parameters estimated using Ambient Vibration Tests (AVT) can be due to both model and measurement uncertainties. The need for taking uncertainties into account has been widely recognized and several approaches have been developed by the two main schools of probability interpretations: the frequentist and the Bayesian interpretation. In the latter, probability is not interpreted as the relative occurrence of a random phenomena but as the plausibility of an hypothesis. The main scope of the interpretation of probability in the Bayesian context leads to the fact that the reason of uncertainty of the structural parameters is seen in the incomplete available information/data. In this work, the Bayesian updating of cable stayed footbridge model parameters using dynamic measurements is discussed. The quantification of model uncertainties is carried out by means of the prediction error when the numerical model updating is performed using two different reference Data Sets: the first one consists in the experimental natural frequencies and the second one consists in both natural frequencies and corresponding modal vectors. In practice, when incomplete measurements of vibration modes are available, including the modal vectors in the reference data set is not an easy task. For this reason, the Modal Assurance Criterion (MAC) is used in order quantify the modal vector prediction error. In addition, the numerical model output is replicated by means of Polynomial Chaos (PC) based surrogate model in order to reduce the computational burden related to the posterior distribution evaluation at each step of Markov Chain Monte Carlo sampling.*

1 INTRODUCTION

A physical model may be described by a *forward problem*, which predicts some Quantities of Interest (QoI) of the system given a set of unknown/uncertain input set of parameters [1, 2]. The corresponding *inverse problem* consists in estimating the set of these parameters from a set of measured/observed data, taking into account that in realistic applications the data are noisy, incomplete and characterized by a significant level of uncertainty [3].

A classical inverse problem in structural engineering is Finite Element (FE) model updating aiming to invert the standard forward relation between the unknown parameters and the predicted response of a model using experimentally observed data. Usually, incomplete modal data (e.g. natural frequencies and vibration modes) are used to calibrate model parameters in order to minimize the distance between the model predictions and the observed quantities [4, 5, **?**].

The FE model updating can be divided into two main approaches: deterministic and probabilistic model updating [7]. The former is well established in literature with several successful applications to strategic and historic structures. In practice, the modal data identified from the measurements are very sensitive to measurement noise, environmental conditions and level of excitation occurring during the tests. Furthermore, the numerical model is always a simplified representation of a real structure and therefore a large number of uncertainties arise because of uncertain geometry, material properties, boundary conditions as well as for simplifications and idealizations.

Therefore the role of measurement and model uncertainty in model updating is crucial and probabilistic FE model updating methods such as Bayesian methods have become popular allowing for explicitly accounting for all the sources of errors involved in the updating process [7, 8, 9]. In the Bayesian updating framework the unknown model input parameters are taken to be uncertain and modeled as Random Variables (RVs) described by their posterior marginal distributions, obtained from prior information and measurements of QoIs that are observable and depend on the unknown parameters. The main limitation of Bayesian updating is the high computational cost related to the posterior distributions computation especially when several updating parameters are modified during the process or when a large data set is used as target. The acceleration of the Bayesian updating framework can be achieved with surrogate models able to reproduce the numerical FE solution with the surrogate solution [3, 10].

In this paper, a Bayesian robust framework for the calibration of a FE numerical model describing an actual steel cable-stayed footbridge in Terni (Umbria Region) is defined using dynamic incomplete modal data (natural frequencies and vibration modes) obtained via Ambient Vibration Tests (AVT). Two updating parameters are selected whose effects on both natural frequencies and vibration modes are significant. The evaluation of the posterior marginal distributions is carried out using Markov Chain Monte Carlo (MCMC) method [11, 12].

The deterministic solution at each step of the chain replaced by the solution obtained via Polynomial Chaos (PC) based surrogate models for reducing the high computational costs [13, 14, 15].

When mode shape are used as reference the formulation of the Bayesian updating framework is not an easy task since mode shape matching is usually required. For this reason the Modal Assurance Criterion (MAC) is used for ensuring mode shape matching and to represent the mode shape vector prediction error as the difference between the measured and the predicted modal data.

Section 2 introduces the general probabilistic model while Section 3 briefly reviews the

Bayesian updating framework with a special focus on the computational aspects and on the likelihood function formulation. Section 4 briefly reviews the PC expansion method and finally the procedure is applied to the cable-stayed footbridge case and the main results are presented and discussed.

2 UNCERTAINTY IN FINITE ELEMENT MODEL PARAMETER ESTIMATION

A numerical FE model $\mathcal{M} : \mathbb{R}^N \to \mathbb{R}^M$ provides a mapping from the parameters $\mathbf{\Theta} = \{\Theta_1, ..., \Theta_N\} \in \mathbb{R}^N$ to an output vector $\mathbf{u} = \{u_1, ..., u_M\} \in \mathbb{R}^M$ so that:

$$\mathbf{u} = \mathcal{M}(\mathbf{\Theta}) \tag{1}$$

In the ideal case, the model output $\mathbf{u}$ corresponds perfectly to the true system output $\mathbf{D}$, i.e. $\mathbf{D} = \mathcal{M}(\mathbf{\Theta})$. This latter equality is the starting point for the deterministic FE model parameter estimation using incomplete modal data, where the main objective is to estimate the model parameters $\Theta_i, i = 1, ..., N$ for a given set of measured system output.

Actually, a numerical mechanical model is not able to perfectly reproduce the real behavior of the true structural system [16]. Therefore, a modeling error $\mathbf{e}_M$ defined as the difference between the real behavior of the true system and the model predictions, i.e. $\mathbf{e}_M = \mathbf{D} - \mathcal{M}(\mathbf{\Theta})$, is always present. Since the measurements are in practice always disturbed also a measurement error $\mathbf{e}_D$ determine a difference between the true system output and the actual observed data $\overline{\mathbf{D}}$, i.e. $\mathbf{e}_D = \overline{\mathbf{D}} - \mathbf{D}$.

Eliminating the unknown true system behavior $\mathbf{D}$ form the error equations, the total prediction error $\mathbf{e}$ can be obtained as the sum of the modeling and measurement error:

$$\mathbf{e} = \mathbf{e}_M + \mathbf{e}_D = \overline{\mathbf{D}} - \mathcal{M}(\mathbf{\Theta}) \tag{2}$$

Equation 2 represents the main starting point for the Bayesian method.

3 BAYESIAN METHOD

In the Bayesian updating framework the model parameters are gathered in the real valued input random vector $\mathbf{\Theta} = \{\Theta_1, \Theta_2, ..., \Theta_N\} \in \mathbb{R}^N$ and modeled as independent RVs defined according to some probability space $\{\Omega, \mathcal{F}, \mathcal{P}\}$ where Ω is the probability space, $\mathcal{F}$ is the σ-Field and $\mathcal{P}$ is the probability measure. If each Θ_i is described by the Probability Density Function (PDF) $\pi_i(\theta_i)$, the joint PDF is given by the product of the N densities.

In the Bayesian approach the updated probabilities of the unknown parameters $\mathbf{\Theta}$ when data $\overline{\mathbf{D}}$ becomes available is quantified by a joint PDF which is known as *posterior distribution* and it is expressed by [17]

$$p(\mathbf{\Theta}|\overline{\mathbf{D}}, M) = c^{-1} p(\overline{\mathbf{D}}|\mathbf{\Theta}, M) p(\mathbf{\Theta}|M) \tag{3}$$

The term $p(\overline{\mathbf{D}}|\mathbf{\Theta})$ - called *likelihood function* - expresses the probability of the data conditional to the unknown/adjustable vector $\mathbf{\Theta}$. The term $p(\mathbf{\Theta}|M)$ is the *prior distribution*, which quantifies the initial plausibility of the vector of parameters $\mathbf{\Theta}$ associated with the model class M. The normalizing constant $c = p(\overline{\mathbf{D}}|M)$ is called the *evidence of model class M*. This normalization makes the integration over the parameter space of the posterior PDF in (3) equal to one. The c constant is given by the multidimensional integration over the parameter space

$$c = p(\overline{\mathbf{D}}|M) = \int p(\overline{\mathbf{D}}|\mathbf{\Theta}) p(\mathbf{\Theta}|M) d\mathbf{\Theta} \tag{4}$$

When a single set of incomplete modal data are used as target, the vector $\overline{\mathbf{D}}$ consists of the extracted modal data from measured acceleration time histories, namely

$$\overline{\mathbf{D}} = \{\hat{f}_{1,j}, ..., \hat{f}_{M,j}, \hat{\mathbf{\Phi}}_{1,j}, ..., \hat{\mathbf{\Phi}}_{M,j}\} \tag{5}$$

where $\hat{f}_{i,j}$ and $\hat{\mathbf{\Phi}}_{1,j}$ are respectively the ith natural frequency and the ith mode shape vector in the jth data set; M is the total number of observed modes.

3.1 Likelihood function

The likelihood function can be interpreted as a measure of the accuracy of the model in describing the measurements. The likelihood function can be obtained according to the Total Probability Theorem as the convolution of the measurement and modeling errors, $\mathbf{e}_D$ and $\mathbf{e}_M$.

In this study no information is available on the individual errors and the effects of both modeling and measurement errors are considered by using the the total prediction error in Equation 2. The error of the i-th natural frequency e_i^f is defined as:

$$e_i^f = f_i(\mathbf{\Theta}) - \hat{f}_i \tag{6}$$

The error of the i-th mode shape vector $e_i^M$is defined by means of Modal Assurance Criterion (MAC) [18]. The MAC coefficient is used in order to measure the correlation between the measured ($\hat{\mathbf{\Phi}}_i$) and the numerically computed ($\mathbf{\Phi}_i(\mathbf{\Theta})$) mode shape vectors. Taking into account that MAC coefficient assumes values between 1 and 0 respectively for perfect match and no correlation its complement $1 - MAC$ can be considered as the residual error for mode shape

$$e_i^{MS} = 1 - \frac{|\hat{\mathbf{\Phi}}_i \mathbf{\Phi}_i(\mathbf{\Theta})|}{(\hat{\mathbf{\Phi}}_i \hat{\mathbf{\Phi}}_i^T)(\mathbf{\Phi}_i(\mathbf{\Theta})\mathbf{\Phi}_i(\mathbf{\Theta})^T)} \tag{7}$$

The uncertainty in e_i^f and e_i^{MS} are modeled as Gaussian vector with zero mean and unknown variance σ^2 therefore the likelihood function is formulated basing on the PDFs of the errors in 6 and 7

$$p(\overline{\mathbf{D}}|\mathbf{\Theta}) \propto exp\left(-\frac{1}{2}\mathbf{e}^T\mathbf{\Sigma}^{-1}\mathbf{e}\right) \tag{8}$$

where $\mathbf{e}$ is a $[2M \times 1]$ vector of the total error

$$\mathbf{e} = \begin{bmatrix} \mathbf{e}_f \\ \mathbf{e}_{MS} \end{bmatrix} \tag{9}$$

and Σ is a $[2M \times 2M]$ total error covariance matrix.

When both natural frequencies and mode shape vectors are considered in the reference data set $\overline{\mathbf{D}}$ mode pairing should be properly carried out ensuring that the comparison of modal properties obtained from the measured data and FE model should be made only when they correspond to the same dynamic mode.

3.2 Computational aspects of posterior distribution

When the prior PDF and the likelihood function are determined, Equations 3 and 4 allow for the updating of the PDFs of the model parameters Θ_i based on experimental observations of the structural system. If the number of parameters and data space dimension is large, the multidimensional integration in Equation 4 cannot be solved analytically and sampling methods such as the Markov Chain Monte Carlo (MCMC) and its derivatives are used. The term MCMC refers to all procedure based on stationary chains of samples to approximate the parameter distributions.

In particular the Metropolis Hastings (MH) algorithm, as an MCMC simulation method, is used in this study [12]. This algorithm is based on generating samples from any target distribution of the uncertain parameters Θ_i. The proposed parameter sample $\mathbf{\Theta}^*$ are generated by a proposal density $q(\mathbf{\Theta}^t|\mathbf{\Theta}^*)$ depending on the current state of the chain. The candidate sample $\mathbf{\Theta}^*$ has a probability of $\rho(\mathbf{\Theta}^t|\mathbf{\Theta}^*)$ to be accepted as next state of the chain $\mathbf{\Theta}^{t+1} = \mathbf{\Theta}^*$; therefore the probability for the candidate sample to be rejected is $1 - \rho(\mathbf{\Theta}^t|\mathbf{\Theta}^*)$. If the candidate is rejected, the current sample is treated as the next sample. The specification of the acceptance probability ρ allows generating a Markov chain with desired target density.

This approach can be computationally prohibitive since it requires the computation of the FE model deterministic solution at each step of the chain and usually it requires about 10^5 samples generations to have solution convergency. In order to obtain a significant reduction of the computational burden an effective method based on the functional approximation of the forward model response in Equation 1 is used. To this end the Polynomial Chaos (PC) representation method [19] is used to a obtain an analytical representation of the model itself as a function of the main random input random parameters leading directly to a surrogate model in the form of response surface. This means that the posterior sampling via MCMC can be carried out directly from the response surface without the need to solve the analytical model for all the samples.

4 POLYNOMIAL CHAOS REPRESENTATION

Let $\mathbf{\Theta}$ be a non Gaussian $\mathbb{R}^N$-valued random vector with N independent components defined by

$$\mathbf{\Theta} = \mathbf{g}(\xi) \tag{10}$$

where g is a deterministic nonlinear function, $\mathbf{g} : \mathbb{R}^K \rightarrow \mathbb{R}^N$, $\xi \sim N(0, \mathbf{I})$ is a $\mathbb{R}^k$-valued vector of k independent and identically distributed, zero mean, unit variance Gaussian RVs and $\mathbf{I}$ denotes the identity matrix having dimension $(k \times k)$.

The solution of the physical model in (1) becomes

$$\mathbf{u} = \mathcal{G}(\xi) \tag{11}$$

where $\mathcal{G} : \mathbb{R}^K \rightarrow \mathbb{R}^M$. Considering a N-variate input and a univariate output, i.e. $M = 1$, and assuming that the model response is a finite variance RV, the structural response can be approximated as

$$\tilde{u} = \tilde{\mathcal{G}}(\xi) = \sum_{\alpha \geq 0}^{N_P - 1} \hat{\mathbf{u}}_\alpha \mathbf{\Psi}_\alpha(\xi) \tag{12}$$

where $\Psi_\alpha(\xi)$ represents the multivariate orthogonal polynomials with finite multi-index set and $\hat{\mathbf{u}}_\alpha$ are the polynomial coefficients. If p indicates the maximum polynomial order, then N_P is given by

$$N_P = \binom{K + p}{p} = \frac{(K + p)!}{K!p!} \tag{13}$$

The polynomial order have to be chosen to guarantee results accuracy. Several different approaches are available for the estimation of the polynomial deterministic coefficients $\hat{\mathbf{u}}_\alpha$ [20, 13].

Using this approach a model sensitivity analysis can be performed in a straight forward manner basing on the orthogonality condition at the base of the mathematical setting of the PC representation since all the statistics of the QoIs can be estimated from the deterministic coefficients statistics. In this paper a Global Sensitivity Analysis (GSA) based on Sobol' coefficients [21] is carried out to determine the influence of each input random model parameter on the final results, assessing the importance of using a proper reference data set in the Bayesian updating framework.

5 NUMERICAL EXAMPLE: A CABLE STAYED FOOTBRIDGE

In order to test the performance of the proposed algorithm for the probabilistic Bayesian updating of a FE model parameters using incomplete modal data, a cable stayed footbridge in Terni (Umbria Region, central Italy) is taken as case study. The footbridge has a total length of 180 m and has two main parts: a curved shape one with a total length of 120 m, which is supported by an asymmetric array of cables connected to a 60 m tall inverted tripod tower through a pair of circular rings; a straight 60 m span with two bowstring arches.

The initial three dimensional FE model was built using the commercial code SAP2000 [22]. Different mechanical characteristics (Table 1) have been selected for the structural components and each stay is modeled with a nonlinear element describing bot tension - stiffening and large deflections.

A pre stress modal analysis was carried out starting from the equilibrium condition under dead load and cable pre tension in order to consider the nonlinear behavior mainly due to cable sag and large deflection. Natural frequencies calculated from the initial FE model are shown in Table 2: seven mode shapes are identified in the range of frequency of interest.

5.1 Dynamic system identification

The footbridge dynamic characterization in terms of natural frequencies and corresponding vibration mode shapes has been obtained from full scale measurements in operating conditions using fourteen uniaxial accelerometers. The obtained acceleration time histories have been used to identify vertical, horizontal and torsional vibration modes with Enhanced Frequency Domain Decomposition (EFDD) method [23].

A single data set with 400 Hz sampling rate was recorded with time lengths 926 s. Reliability of results was investigated using different order of decimation and different type of filters. Seven modes have been clearly identified in the range of frequency of interest. Table 3 summarizes the minimum, f_{min}, and the the maximum, f_{max}, values of the identified natural frequencies considering different signal sampling parameters (e.g. decimation order, filters,

Table 1: Mechanical properties used in the initial FEM.

Material	E	Mass density
	GPa	KN/m^3
Steel S355	210	78
Cables	160	77
Concrete C32/40	33.345	25

Mode	f^{FEM}	Mode's type
	Hz	
1	1.030	Vertical
2	1.514	Lateral
3	1.774	Torsional
4	2.184	Vertical
5	2.365	Lateral
6	2.982	Vertical
7	3.153	Vertical

Table 2: Modal features obtained from initial FE model.

Mode	f^{EXP}_{min}	f^{EXP}_{max}	Mode's type
	Hz	Hz	
1	1.11	1.13	Vertical
2	1.67	1.69	Lateral
3	1.79	1.80	Torsional
4	2.40	2.47	Lateral
5	2.58	2.59	Vertical
6	3.30	3.31	Vertical
7	3.35	3.40	Vertical

Table 3: Range of identified natural frequencies from data sets #1 and #2.

frequency resolution of the output power spectral density spectrum). Initially the natural frequencies of the initial FE model were mostly higher than the measured natural frequencies.

The MAC was used to identify the modal shapes from the experimental data set. In the following, two different MAC matrices will be estimated: the auto-MAC matrix and the MAC matrix. The first is estimated from the measured mode shapes while the second is computed pairing one experimental with one numerical mode shapes. The diagonal terms in the auto-MAC matrix are all equal to one meaning that each mode shape is paired with itself. The MAC matrix is estimated from the experimental mode shapes and the FE analysis mode shapes showing that the matrix diagonal terms are higher than 0.80 indicating a good correlation between the experimental and numerical modal vectors.

5.2 Selection of the updating parameters

The selection of the updating parameters is a key issue in the model updating procedure since they have to be strictly and directly related to the measurement results used as target. A preliminary deterministic sensitivity analysis is thus carried out in order to provide information for an efficient selection.

In particular the sensitivity of the natural frequencies and the mode shapes (in terms of diagonal MAC values) to variation of structural steel and cable Young's moduli, cable tension stiffening, model mass density and stiffnesses of rotational and translational springs used for modeling the soil - structure interaction (Figure 1). It has been found that the variation in the each cable and in the spring stiffness describing the soil - structure interaction has negligible effects on the numerical model eigenfrequencies and eigenvectors. On the contrary, eigenfrequencies and eigenvectors are very sensitive to variations in the steel elastic moduli and the model mass density. It is worth noting that variations in cable elastic moduli provide significant variations in the eigenvectors and small variations in the eigenfrequencies.

Assuming that the model mass density does not vary significantly along the deck only two updating parameters are defined for the Bayesian framework: the deck and cable stiffnesses described by the steel, E_{steel}, and cable, E_{cables}, elastic moduli, respectively. Therefore, the real valued random vector $\mathbf{\Theta} \in \mathbb{R}^2$ has independent components: $\Theta_1 = E_{steel}$ and $\Theta_2 = E_{cables}$.

5.3 Surrogate Model

In this case study, the six experimental natural frequencies and the six corresponding mode shape vectors $\bar{f}_i^{EXP}$ and $\bar{M}_i^{EXP}$ with $i = 1, ..., 6$ are used as reference while the corresponding

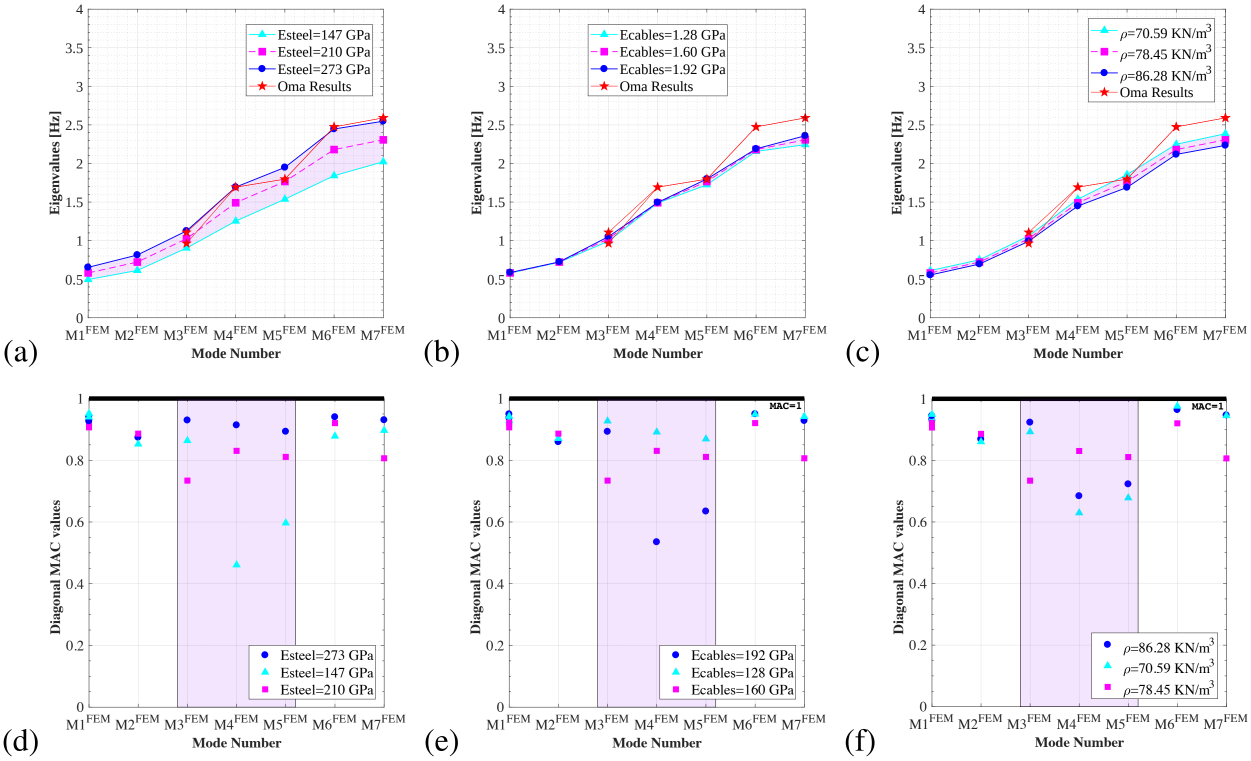


Figure 1: Eigenfrequencies and diagonal MAC values variations with changes in the mechanical parameters (upper panels and lower panels respectively): (a,d) steel modulus of elasticity; (b,e) cables modulus of elasticity; (c,f) model mass density.

six numerical model frequencies and mode shape vectors f_i^{FEM} and M_i^{FEM} with $i = 1, ..., 6$ are set as QoIs. Since the MAC coefficient complement $1 - MAC$ is used in order to evaluate the prediction error e_{MS} as in Equation 7 and considering that the MAC coefficient is very sensitive to small variation of the single eigenvector component, each of the fourteen component of the considered six mode shape vectors are set as QoIs.

The PC expansion in Equation 12 is thus used in order to build a surrogate model for each of the selected QoIs (Figure 2). Two normal distribution has been assumed for the two component of the input random vector Θ_1 and Θ_2 to build the 90 different response surfaces. The initial mechanical characteristics of the two different material used for the deck and cables in Table 1 are used as PDFs mean values; the coefficient of variation (c.o.v.) is assumed in order to avoid unfeasible samples in the simulation procedure. The resulting two PDFs are used also as prior distribution in the Bayesian updating framework (Figure 4).

The maximum polynomial degree p has been set equal to 5 and a complete basis has been built requiring $(p + 1)^N = 36$ analyses. The deterministic coefficients in Equation 12 are evaluated using least square minimization method [24] and a full tensor grid scheme. Once that accurate surrogate models have been built, the variance of the N_P polynomial coefficients $\mathbf{u}_\alpha$ is estimated for each QoI and used to evaluate the first order Sobol' indices, which give information on the influence of the uncertain parameters Θ_1 and Θ_2 on each QoI, e.g. the first six natural frequencies and the 84 eigenvector components of the first six mode shape vectors (Figure 3).

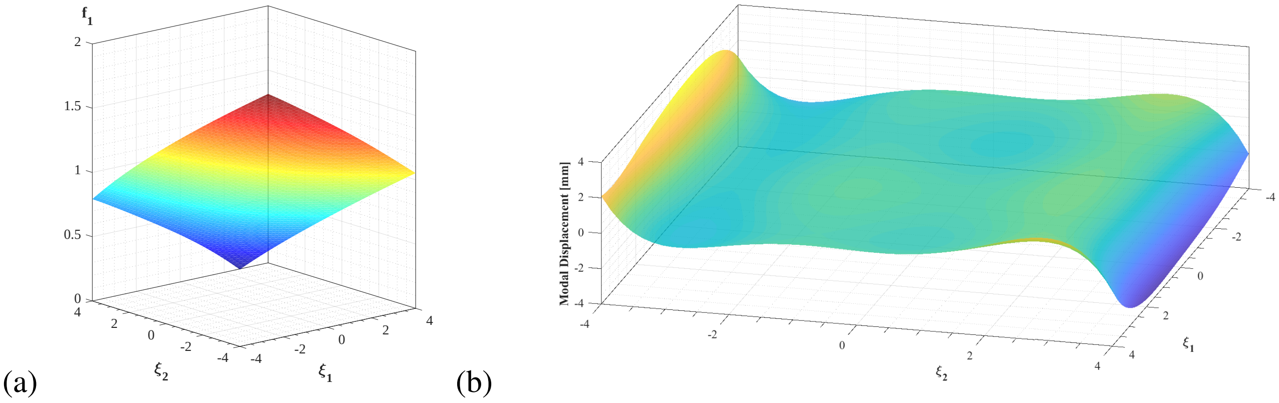


Figure 2: Example of a surrogate model: natural frequency (a) and eigenvector component (b) .

5.4 Bayesian inverse problem solution

Setting $\bar{\mathbf{D}}_1 = \{f_1, ..., f_6\}$ and $\bar{\mathbf{D}}_2 = \{f_1, ..., f_6, \mathbf{\Phi}_1, ..., \mathbf{\Phi}_6\}$ as two different reference vector and replacing the numerical model in Equation 1 with the surrogate model in Equation (12), the posterior marginal PDF of the two dimensional random vector $\mathbf{\Theta} = \{\Theta_1, \Theta_2\}$ can be estimated. In particular, the MCMC MH algorithm is applied requiring the evaluation of the deterministic solution 150,000 times in both cases in order to ensure convergency. It is important to point out that when $\bar{\mathbf{D}}_1$ is used as reference data set, the MCMC MH algorithm is modified using the diagonal MAC coefficients as constraints in order to guarantee the natural frequency/mode shape matching at each step of the chain.

The results of the Bayesian updating procedure are shown in Figure 4. The posterior distribution of Θ_1 has mean values equal to $266GPa$ and $273GPa$ - about 1.25 and 1.30 times the mean value of the prior PDF - when $\bar{\mathbf{D}}_1$ and $\bar{\mathbf{D}}_2$ are used as reference vector respectively.

The posterior distribution of Θ_2 is very similar to the prior PDF when $\bar{\mathbf{D}}_1$ is used as reference, indicating that $\bar{\mathbf{D}}_1$ is non informative with respect to this random parameter. This result was expected since the natural frequencies are mainly influenced by the stiffness of the deck, Θ_1, as shown by the Sobol' indices in Figure 3. On the contrary the posterior distribution of Θ_2 is characterized by an evident maximum at the posterior mean value equal to $184MPa$, about

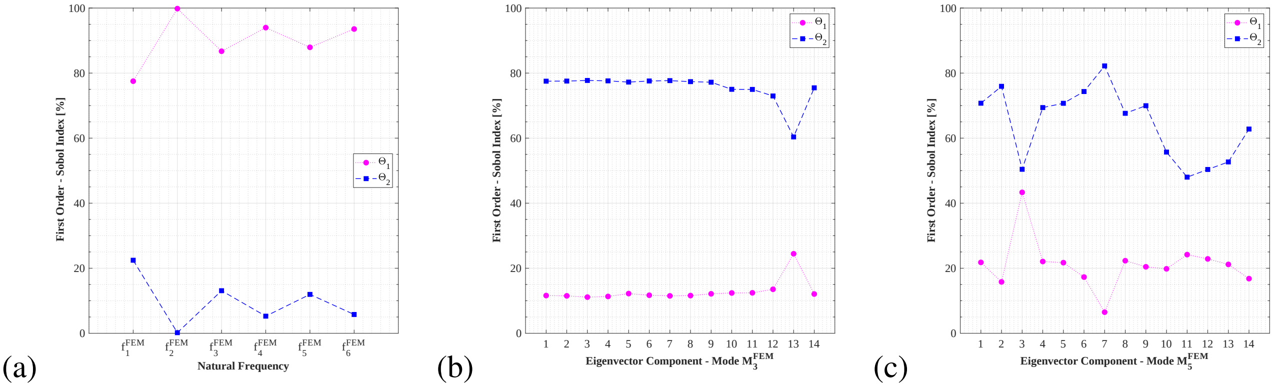


Figure 3: First order Sobol indices: natural frequency (a) and eigenvector component of the 3^{rd} and 5^{th} numerical mode shape (b and c, respectively).

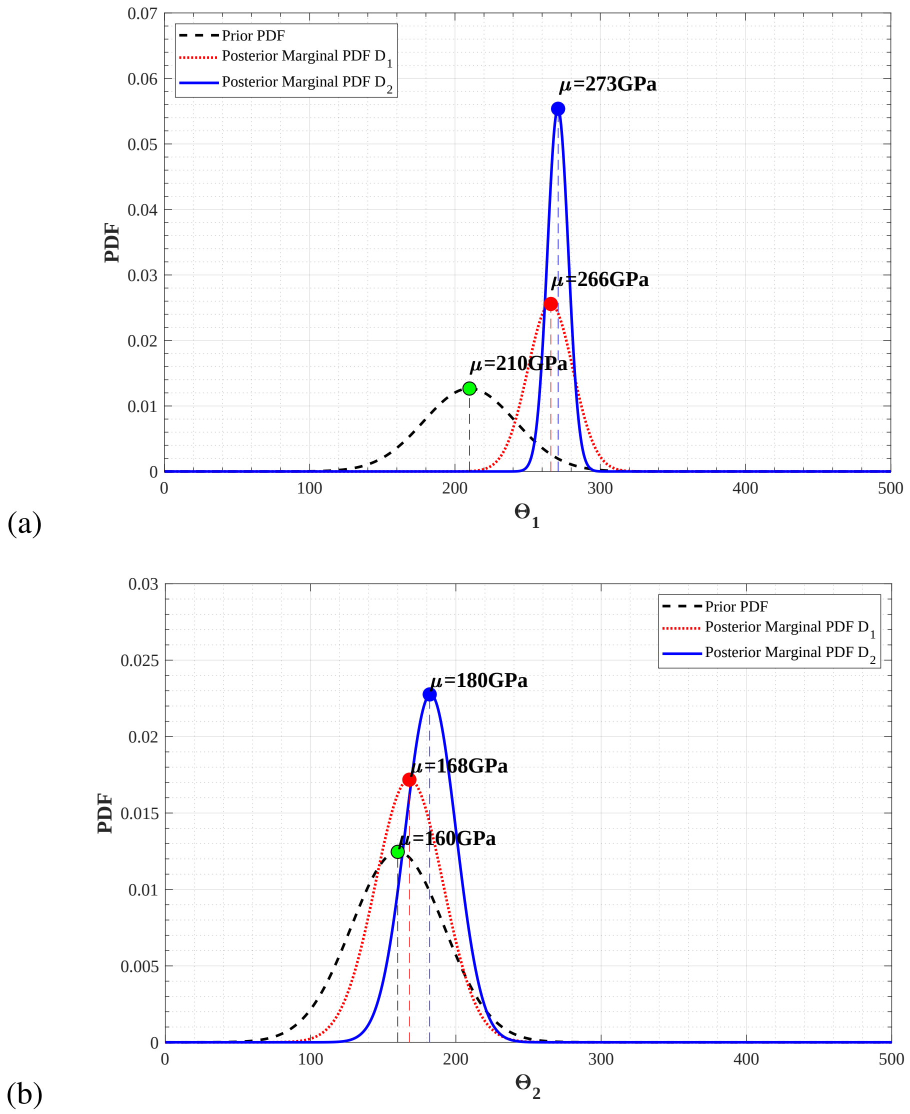


Figure 4: Prior and posterior marginal distributions: (a) deck stiffness; (b) cables stiffness.

1.15 times the mean value of the prior PDF.

Finally, Figure 5 (a) compares the natural frequencies estimated from the experimental data to those obtained with the initial numerical model and the updated model using the posterior mean value of $\mathbf{\Theta}$ when $\bar{\mathbf{D}}_\mathbf{1}$ and $\bar{\mathbf{D}}_\mathbf{2}$ are used as reference data set. Before the Bayesian updating procedure the differences between the experimental and the numerical eigenfrequencies were greater than 8%, with the only exception of the 3^{rd} numerical mode shape for which the error was lower than 1%. After the update carried out using the two considered reference data sets these errors are reduced to values lower than 1% with the exception of the 3^{rd} mode shape for which the error is equal to 8%.

Figure 5 (b) compares the numerical and experimental mode shapes before and after the updating procedure in terms of diagonal MAC values. The initial experimental and numerical mode shapes are characterized by high values of the MAC number. After the update carried out using $\bar{\mathbf{D}}_\mathbf{1}$ as reference, the most significant increase of the MAC values, from 73% to 92%, occurs for the 3^{rd} mode shape (torsional). On the contrary the diagonal MAC value decreases for the 5^{th} and 6^{th} mode shape. After the update carried out using $\bar{\mathbf{D}}_\mathbf{2}$ as reference the diagonal

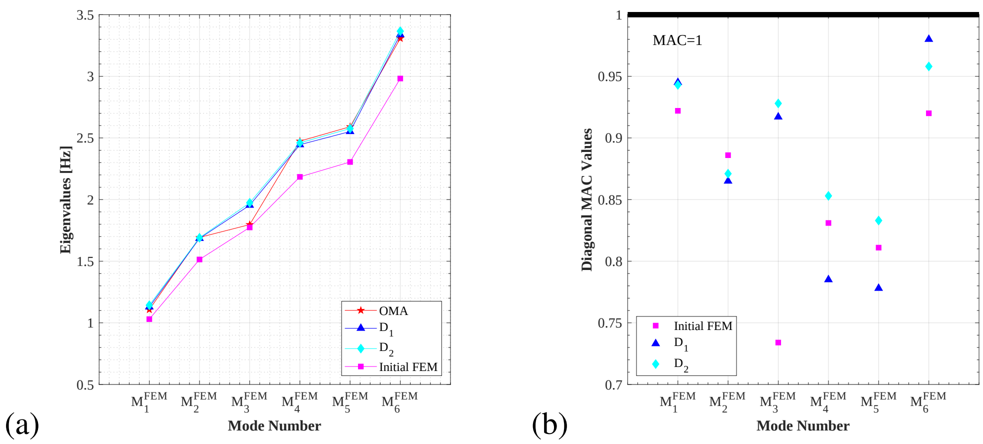


Figure 5: FEM responses before and after the updating Bayesian procedure:(a) natural frequencies; (b) diagonal MAC values.

MAC values increase for each considered mode shape, especially for the 3^{rd}, the 4^{th} and the 5^{th} mode shape, the most influenced by the stiffness of cables, Θ_2.

6 CONCLUSION

In the present work, a robust updating procedure for the calibration of a FE numerical model has been set up in a probabilistic Bayesian framework. The proposed approach is based on dynamic incomplete modal data (natural frequencies and vibration modes) obtained via AVTs and on a functional approximation of the system random response.

First, the initial three dimensional FE model of a cable-stayed footbridge was set up and a sensitivity analysis was carried out both in a probabilistic and deterministic setting in order to select in an efficient manner the most significant parameters to be used in the Bayesian updating procedure targeting the measured natural frequency and mode shape vectors. Second, surrogate models based on the PC representation of the structural system dynamic response were built in order to significantly reduce the computation cost related to the posterior densities estimates by means of MCMC MH procedure. Finally, the updating procedure was carried out using two different reference data set: the first one consists in the experimental natural frequencies and the second one consists in both natural frequencies and corresponding vibration modes. When mode shape are used as target the modal vector prediction error is quantified by means of MAC as the distance between actual correlation and perfect correlation.

The proposed approach overcome the main drawback of the whole Bayesian updating framework related to the unfeasible computational costs making it suitable for real time Structural Health Monitoring (SHM) applications. Furthermore, results demonstrated the importance of using a proper informative data set.

REFERENCES

[1] Bernardini, E. and Spence, S.M.J. and Gioffré, M., Dynamic response estimation of tall buildings with 3D modes: A probabilistic approach to the high frequency force balance method. *Journal of Wind Engineering and Industrial Aerodynamics*, **104-106**, 56–64, 2012.

[2] Matthies, H. G., Uncertainty Quantification with Stochastic Finite Elements. *Encyclopedia of Computational Mechanics*, **27**, 2007.

[3] Rosic, B., Litvinenko, A. and Matthies, H.G., Sampling free linear Bayesian update of polynomial chaos representations. *Journal of Computational Physics*, **231**, 5761–5787, 2012.

[4] Benedettini, F. and Gentile, C., Operational modal testing and FE model tuning of a cable-stayed bridge. *Engineering Structures*, **33**, 2063–2073, 2011.

[5] Daniell, W. E. and Macdonald, J.H.G., Improved finite element modelling of a cable-stayed bridge through systematic manual tuning. *Engineering Structures*, **29**, 358–371, 2007.

[6] Pepi, C. and Gioffré, M. and Comanducci, G. and Cavalagli, N. and Bonaca, A. and Ubertini, F., Dynamic characterization of a severely damaged historic masonry bridge. *Procedia Engineering*, **199**, 3398–3403, 2017.

[7] Marwala, *Finite-element-model Updating Using Computational Intelligence Techniques.* Springer-Verlag, London, UK, 2017.

[8] L. S. Katafygiotis and J. L. Beck, Updating Models and Their Uncertainties: Part II. *Journal of Engineering Mechanics*, **124**, 463–467, 1998.

[9] Yuen K. V., and Beck James L., and Katafygiotis Lambros S, Efficient model updating and health monitoring methodology using incomplete modal data without mode matching. *Structural Control and Health Monitoring*, **13**, 91–107, 2001.

[10] Kucerova, A. and Rosic, B. and Matthies, H.G., Acceleration of uncertainty updating in the description of transport processes in heterogeneous materials. *Journal of Computational and Applied Mathematics*, **236**, 4862–4872, 2012.

[11] Gamerman, D. and Lopes, H.F., *Markov Chain Monte Carlo: Stochastic simulation for bayesian inference.* Chapmann & Hall, 2006.

[12] Hastings, W.K., Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Oxford University Press, Biometrika Trust*, **57**, 97–10, 1970.

[13] Field, R.V. and Grigoriu, M., On the accuracy of the polynomial chaos approximation. *Probabilistic Engineering Mechanics*, **19**, 65–80, 2004.

[14] Soize, C. and Ghanem, R., Physical systems with random uncertainties: Chaos representations with arbitrary probability measure. *SIAM Journal on Scientific Computing*, **26**, 395–410, 2004.

[15] Matthies, H.G. and Brenner, C.E. and Bucher, C. and Soares,C.G., Uncertainties in probabilistic numerical analysis of structures and solids-stochastic finite elements. *Structural Safety*, **19**, 283–336, 1997.

[16] Simoen, E. and De Roeck, G. and Lombaert, G., Dealing with uncertainty in model updating for damage assessment: A review. *Mechanical Systems and Signal Processing*, **56-57**, 123–149, 2015.

[17] Bayes, T., *An essay towards solving a problem in the doctrine of chances.* Philosophical Transactions of the Royal Society, 1763.

[18] All J. Allemang, The Modal Assurance Criterion (MAC): Twenty Years of Use and Abuse. *Journal of Sound and Vibrations*, 14–21, 2003.

[19] Ghanem, R.G. and Spanos, P.D., Stochastic Finite Elements: A Spectral Approach. *Am. J. Math*, **60**, 897–936, 1991.

[20] B. Sudret and S. Marelli and J. Wiart, Surrogate models for uncertainty quantification: An overview. *2017 11th European Conference on Antennas and Propagation (EUCAP)*, 793–797, 2017.

[21] Sudret,B., Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering and System Safety*, **93**, 964–979, 2008.

[22] SAP2000. Static and dynamic finite element of structures. *Computers and structures, Inc Berkeley CA USA*, Inc Berkeley CA USA, 2018.

[23] Brincker, R. and Ventura, C. and Andersen, P., Damping estimation by Frequency Domain Decomposition. *Proceedings of the International Modal Analysis Conference - IMAC*, **01**, 964–979, 2001.

[24] Choi, S.-K. and Canfield, R. and Grandhi, R. and Pettit, C., Polynomial Chaos Expansion with Latin Hypercube Sampling for Estimating Response Variability. *AIAA Journal*, **42**, 1191–1198, 2004.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

ON THE USE OF ENSEMBLES OF METAMODELS FOR ESTIMATION OF THE FAILURE PROBABILITY

C. Amrane, C. Mattrand, P. Beaurepaire, J-M. Bourinet and N. Gayton

Université Clermont Auvergne, CNRS, SIGMA Clermont, Institut Pascal
F-63000 Clermont-Ferrand, France
e-mail: {chahrazed.amrane, cecile.mattrand, pierre.beaurepaire, jean-marc.bourinet, nicolas.gayton}@sigma-clermont.fr

Keywords: Failure probability, Metamodel, Uncertainty Quantification, AK-MCS, Ensemble of Metamodels.

Abstract. *Performing a reliability analysis on engineering applications usually result in a huge number of calls to numerical models especially in the context of low failure probabilities. Combining reliability methods with metamodeling has gained interest in order to reduce the computational burden. Several kinds of metamodels exist, each based on some mathematical assumptions and prior choices regarding their parameters. However, no type and no tuning is optimal in all conditions. It has been shown that combining individual metamodels in the form of a weighted average metamodel can sometimes enhance the accuracy of predictions. This approach is known as Ensemble of Metamodels (EM). The existing EM strategies can be split into two groups, namely local EM and global EM. The later has unchanged weight factors in the design space unlike local EMs. In this paper, the relevance of using ensembles as a substitution for the performance function to estimate the failure probability is investigated. In a first attempt, ordinary Kriging metamodels are solely considered as EM individuals. The focus is rather put on the choice of the kernel. The contribution therefore consists in using EMs, composed of Kriging metamodels with different kernels, in order to study their efficiency for the estimation of failure probabilities. The Active learning reliability method combining Kriging and Monte Carlo Simulation, namely AK-MCS, is here considered for the estimation of failure probabilities. A new learning function is proposed in this work. Two academic examples are studied in order to investigate the potential benefits of such an approach. An analysis of the results is performed in terms of computational cost and prediction accuracy of failure probability.*

1 INTRODUCTION

In a reliability approach, a failure mode is usually expressed by means of a performance function g. The structure fails when g is negative or equal to zero, i.e. $g(\mathbf{x}) \leq 0$, where $\mathbf{x} = (x_1, ..., x_n)$ denotes the vector of random inputs variables. $g(\mathbf{x}) > 0$ means that the structure is safe for the input vector **x**. The limit between the two configurations, i.e. $g(\mathbf{x}) = 0$, is called the Limit State Function (LSF). The failure probability is defined as the integral of the joint density function (pdf) $f_{\mathbf{X}}(\mathbf{x})$ over the failure domain Ω_f for any **x** such as $g(\mathbf{x}) \leq 0$:

$$P_f = \int_{\Omega_f} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \tag{1}$$

where $d\mathbf{x} = dx_1...dx_2$. Analytical and numerical solutions based on usual integration schemes became unfordable, especially in high dimensions, when low failure probabilities are expected and when the performance function is nonlinear and costly-to-evaluate. Several methods exist in the literature to approximate this integral. The reference method relies on Monte Carlo simulation (MCS). The corresponding failure probability estimator is given by:

$$\hat{P}_f = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} I\{g(\mathbf{x}^{(i)}) \leq 0\} \tag{2}$$

where N_{MC} is the sample size and $I\{.\}$ stands for the indicator function. The MCS method is simple and easy to implement, however it requires a huge computational effort for small failure probabilities. An accurate estimate $\hat{P}_f$ of the unknown failure probability is obtained only if the sample size is sufficiently large. The coefficient of variation (C.O.V) of $\hat{P}_f$ can be estimated as follows:

$$C.O.V_{\hat{P}_f} \simeq \sqrt{\frac{1-\hat{P}_f}{N_{MC}\hat{P}_f}} \tag{3}$$

More advanced methods have been developed in order to reduce the number of calls to the performance function. Metamodeling-based simulation methods are the most common and relevant ones. Metamodels are used in reliability analysis to mimic the input-output relationship of the performance function. They consist in simplified mathematical models much less expensive to evaluate. Among metamodels, Polynomial Response Surfaces (PRS), Polynomial Chaos Expansion (PCE), Artificial Neural Networks (ANN) and Support Vector Regression (SVR) have been used in the framework of reliability analysis, see [1, 2, 3, 4, 5, 6, 7, 8] and references therein. Kriging has also appeared appealing for solving reliability problems because of its usual interpolating nature and the straightforward estimation of the local variance of the prediction [9, 10, 11]. Even though metamodeling based reliability approaches have proven their ability to adress complex problems, some issues regarding their tuning remain and may affect their efficiency. In Kriging for example there is no consensus or even criteria for the upstream choice of the kernel and trend functions. An inappropriate decision could result in undesired properties as shown in [12]. So, picking the most appropriate metamodel for a given problem is still a challenging task for users. In order to prevent such a risk, an alternative consists in mixing metamodels in an approach known as Ensemble of Metamodels (EM). It has been shown that the accuracy of EM predictions can sometimes be better then those obtained from individual metamodels [13, 14, 15, 16].

While most researchers have primarily been interested in the use of ensembles of metamodels in optimization , there has been relatively very little work about their use in reliability analysis [17, 12]. In this work, the relevance of using ensembles as a surrogate for the performance function to estimate the failure probability is investigated. The paper is organized as follows. Section 2 introduces strategies available in the literature for mixing metamodels. The Active learning reliability method combining Kriging and Monte Carlo Simulation (AK-MCS) [10] is briefly recalled at the beginning of Section 3. In the same section, the proposed approach which combines AK-MCS and EMs is presented. A modified learning function is introduced for such a purpose. Then, two academic examples are performed in Section 4 to highlight the benefits of this approach.

2 ENSEMBLE OF METAMODELS

Surrogate predictions may differ significantly from the true responses, whose properties are unknown, depending on the assumptions made or in case of unrepresentative design of experiments (DoE) used for their calibration. For the purpose of model tunning, the usual strategy consists in calibrating a set of metamodels and then electing the one with the best metric, e.g. with the minimal cross validation error [18] or root mean square error [19]. Their major drawback is the waste of effort spent on the discarded metamodels calibration. To take advantage of the prediction ability of more than one metamodel, a weighted average metamodel has been proposed in [20]

$$\hat{y}_{ens}(\mathbf{x}) = \sum_{i=1}^{m} w_i(\mathbf{x})\hat{y}_i(\mathbf{x}) \tag{4}$$

where $\hat{y}_{ens}(\mathbf{x})$ is the EM prediction at any given input vector $\mathbf{x}$, m is the number of metamodels used in the EM, w_i is the weight factor of the i^{th} metamodel reflecting the model relative predictive contribution in the ensemble and $\hat{y}_i(\mathbf{x})$ is the i^{th} metamodel prediction. It should be noted that the weight factors sum must be equal to one ($\sum_{i=1}^{m} w_i = 1$) in order to have an unbiased response prediction. In the literature, there are two kinds of strategies for determining weights, namely the global EM and the local EM.

2.1 Global EMs

The weight factors in the global EM approaches are constant over the entire design space. Their values do not depend on the prediction point location, i.e. $w_i(x) = w_i, \forall x$. In this work three global EMs have been investigated. The Bayesian model averaging proposed in [21], here named BMA. The heuristic formulation proposed by Goel et al in [13], here named EG. The optimized weight factor of Acar and Rais-Rohani in [14], here named EME.

Bayesian model averaging (BMA)

BMA combines different model predictions in a Bayesian framework. It could also be used for metamodels, which is the case here. Let M be a set of metamodels, where $M = \{M_i; i = 1, ..., m\}$. For a given data set D, the posterior distribution of the EM prediction using BMA is given by:

$$P(\hat{y}_{ens}(\mathbf{x})|D) = \sum_{i=1}^{m} P(M_i|D)P(\hat{y}_i(\mathbf{x})|M_i, D) \tag{5}$$

where $P(\hat{y}_i|M_i, D)$ is the pdf of the i^{th} metamodel prediction. $P(M_i|D)$ represents the posterior probability masses of metamodels M_i, and therefore sum to one. They can be viewed as weights [22] and by analogy with Eq. (4), $P(M_i|D) = w_i$. They are given, in [21], by:

$$P(M_i|D) = \frac{P(D|M_i)P(M_i)}{\sum_{l=1}^{m} P(D|M_l)P(M_l)} \tag{6}$$

where

$$P(D|M_i) = \int P(D|\theta_i, M_i)P(\theta_i|M_i)d\theta_i \tag{7}$$

is the integrated likelihood of the metamodel M_i, θ_i denotes the model parameter vector, $P(\theta_i|M_i)$ is the a priori PDF of θ_i conditionally to M_i, $P(D|\theta_i, M_i)$ is the likelihood of the data given the metamodel M_i and its parameters θ_i and $P(M_i)$ corresponds to the prior probability mass of metamodel M_i which usually equals $1/m$.

Heuristic proposed by Goel et al. (EG)

The strategy of selecting weights proposed by Goel et al. [13] is based on generalized mean square error GMSE, which is an estimation of the mean square error by leave-one-out. It is formulated as follows:

$$\begin{aligned} w_i &= \frac{w_i^*}{\sum_i w_i^*},\ w_i^* = (E_i + \alpha E_{avg})^\beta \\ E_{avg} &= \frac{\sum_i E_i}{m},\ \alpha < 1 \text{ and } \beta < 0 \\ E_i = \sqrt{GMSE} &= \sqrt{\frac{1}{N}\sum_{k=1}^{N}(y^{(k)} - \hat{y}_i^{(k)})^2} \end{aligned} \tag{8}$$

where $y^{(k)}$ is the true response at a given point $\mathbf{x}^{(k)}$, $\hat{y}_i^{(k)}$ is its prediction from the i^{th} metamodel calibrated from all the DoE except the data pair $(\mathbf{x}^{(k)}, y^{(k)})$ and N is the size of the DoE. α and β are parameters that should be specified beforehand. For instance, $\alpha = 0.05$ and $\beta = -1$ is used in the work of Goel and al [13]. A study of the effect of those parameters have also been performed [13].

Optimization problem proposed by Acar et al. (EME)

The weight factors are here solutions of the minimization of a global error. The influence of the error metric choice is studied in [23]. In their original paper, Acar et al. [14] select the GMSE as a metric defined by:

$$GMSE_{\hat{y}_{ens}} = \frac{1}{N}\sum_{k=1}^{N}(y^{(k)} - \hat{y}_{ens}^{(k)})^2 \tag{9}$$

where $y^{(k)}$ is the true response evaluated at $x^{(k)}$ and $\hat{y}_{ens}^{(k)}$ is its prediction by using EM calibrated from all the DoE points except the data pair $(\mathbf{x}^{(k)}, y^{(k)})$. Then, the weight factors are solutions of the following optimization problem:

$$\begin{aligned} \min_{w} \quad & GMSE_{\hat{y}_{ens}} \\ \text{s.t.} \quad & \sum_{i=1}^{m} w_i = 1 \end{aligned} \tag{10}$$

2.2 Local EMs

Unlike the global EM, weight factors are here function of the prediction point location. They are varied over the design space. According to Acar [15], this strategy may lead to more accurate results. However, an unsuitable weights estimation may lead to an ineffective identification of the locally accurate prediction of an individual metamodel [24].

Variance-based local EM proposed by Zerpa et al. (EV)

Under the assumption of unbiased and uncorrelated predictions, weights in [20] are selected as follows, in order to reduce the variance of the EM:

$$w_i(\mathbf{x}) = \frac{\frac{1}{V_i(\mathbf{x})}}{\sum_{j=1}^{m} \frac{1}{V_j(\mathbf{x})}} \tag{11}$$

where $V_i(\mathbf{x})$ is the prediction variance of the i^{th} metamodel M_i at point **x**.

Spatial local EM proposed by Acar (EA3)

Acar [15] proposes four approaches to compute the weights of Eq. (4) based on the cross validation error and the distance between data points and prediction points. Only the third one is considered here. A weight equal to one is set to the metamodel which minimizes the cross validation error, while other weights are set to zero. The weight at a prediction point is assigned to the weight of its closest data point. This approach is formulated as follows:

$$\begin{aligned} w_i(\mathbf{x}) &= \sum_{k=1}^{N} w_{ik} I_k(\mathbf{x}) \\ I_k(\mathbf{x}) &= \begin{cases} 1 & \text{if } \mathbf{x}^{(k)} \text{ is the closest DoE point to } \mathbf{x} \\ 0 & \text{else} \end{cases} \end{aligned} \tag{12}$$

where w_{ik} is the weight assigned to the metamodel M_i at the DoE point $\mathbf{x}^{(k)}$.

3 KRIGING ENSEMBLE FOR AK-MCS RELIABILITY ANALYSIS

3.1 AK-MCS

Echard et al. [10] has combined adaptative Kriging metamodeling to Monte Carlo simulation. The main idea of this method is to iteratively enrich a Kriging surrogate so that it allows a sufficiently accurate classification of a MC population into a safe or failure domain. A learning function, called U, is evaluated over the whole N_{MC} points in order to determine the next best point to add to the initial DoE. The method is summarized here. Readers may refer to [10] for a more detailed version.

1. Generation of the N_{MC} points in the standard design space, where all the random variables **u** are normally distributed and uncorrelated. This required an appropriate transformation from the physical space of **x**. It should be noted that the performance function g is not evaluated at all those points but only at a few candidates if the adaptive enrichment requires it.

2. Selection of the initial design of experiments (DoE) using one of the techniques described in [25].

3. Calibration of the Kriging model based on the DoE $(D = (\mathbf{u}_i, g_i), i = 1, ..., N)$.

4. Identification of the next best point u^* to enrich that minimizes the U learning function assessed for each point of the MC population.

$$U(\mathbf{u}^*) = \min_{i=1,\dots,N_{MC}} U(\mathbf{u}^{(i)}) \tag{13}$$

   where

$$U(\mathbf{u}^{(i)}) = \frac{|\mu_{\hat{g}}(\mathbf{u}^{(i)})|}{\sigma_{\hat{g}}(\mathbf{u}^{(i)})} \tag{14}$$

   and $\mu_{\hat{g}}(\mathbf{u}^{(i)})$ and $\sigma^2_{\hat{g}}(\mathbf{u}^{(i)})$ respectively denote the Kriging mean prediction and variance.

5. Check of the stopping criterion defined as:

$$U(\mathbf{u}^*) \geq 2 \tag{15}$$

   If the stopping criterion is satisfied, the algorithm stops. The metamodel is considered to be accurate enough. Otherwise, the DoE is updated by evaluating g at point $\mathbf{u}^*$. Then, the algorithm goes back to step 3.

6. Estimation of the failure probability according to Eq. (2), where $\hat{g}$ replaces g.

The simple calculation and implementation of the U learning function makes this method very popular. Improvements have been made by Lelièvre et al. in [11]. In this work, the U learning function is adapted in order to propose an AK-MCS method based on an ensemble of Kriging metamodels.

3.2 Proposed U_{ens} learning function

The natural idea when resorting to AK-MCS based on EM prediction is to use Eq. (14), such that:

$$\mu_{\hat{g}_{ens}}(\mathbf{u}) = \sum_{i=1}^{m} w_i(\mathbf{u})\mu_{\hat{g}_i}(\mathbf{u}) \tag{16}$$

$$\sigma_{\hat{g}_{ens}}(\mathbf{u}) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{m} w_i(\mathbf{u})w_j(\mathbf{u})Cov[\hat{g}_i(\mathbf{u}),\ \hat{g}_j(\mathbf{u})]} \tag{17}$$

where $\hat{g}_i(\mathbf{u})$ and $\hat{g}_j(\mathbf{u})$ are predictions from metamodels M_i and M_j respectively. The variance estimation is here problematic because of the covariance terms $Cov[\hat{g}_i(\mathbf{u}),\ \hat{g}_j(\mathbf{u})]$. According to Ginsbourger et al. [26], handling a sum of Kriging predictors results in a tricky problem for estimating the covariance terms. A tractable solution will require us to make an assumption such as full independence. Hence, they proposed to use mixture of statistical distributions to address the variance estimation problem [26]. By mixing the Kriging distributions, a Kriging EM density function is here estimated and used to define a new learning function named hereafter U_{ens}.

The Kriging conditional distribution of $\hat{g}_{ens}(\mathbf{u})$, noted $f_{\hat{g}_{ens}(\mathbf{u})}$, is expressed as the sum of the Gaussian conditional distributions of $\hat{g}_i(\mathbf{u})$, where i refers to the i^{th} metamodel M_i:

$$f_{\hat{g}_{ens}(\mathbf{u})}(\boldsymbol{.}) = \sum_{i=1}^{m} w_i(\mathbf{u}) f_{\hat{g}_i(\mathbf{u})}(\boldsymbol{.}) \tag{18}$$

Hence, $\hat{g}_{ens}(\mathbf{u})$ is a stochastic process of Gaussian mixtures with mean given in Eq. (16). Let S be the event ″the point **u** of MC population is misclassified by $\hat{g}_{ens}$‵‵. The probability of occurrence of S can be deduced from Eq. (18):

$$P_{ens}(S) = \sum_{i=1}^{m} w_i(\mathbf{u}) \Phi(-U_i(\mathbf{u})) \tag{19}$$

where $\Phi(-U_i(u))$ is the probability of misclassifying the point **u** using the Kriging metamodels. By introducing a conceptual index U_{ens} on the event S, we can write:

$$F_{\hat{g}_{ens}(\mathbf{u})}(-U_{ens}(\mathbf{u})) = \sum_{i=1}^{m} w_i(\mathbf{u}) \Phi(-U_i(\mathbf{u})) \tag{20}$$

where $F_{\hat{g}_{ens}(\mathbf{u})}$ defines the unknown cumulative density function (CDF) of $\hat{g}_{ens}(\mathbf{u})$. By analogy with AK-MCS, we consider here the standard normal CDF Φ. From that, U_{ens} reads:

$$U_{ens} = \Phi^{-1}\left(\sum_{i=1}^{m} w_i(\mathbf{u}) \Phi(U_i(\mathbf{u}))\right) \tag{21}$$

The active learning process is performed in this paper with this new U_{ens} learning function.

4 ACADEMIC VALIDATION

The proposed method is applied to two numerical examples considering ordinary Kriging (OK) with two kernel functions: the Gaussian kernel and the Matern $\nu = \frac{3}{2}$ kernel. Their calibration is carried out with the OPENTURNS toolbox [27]. They are combined through global EMs and local EMs techniques described in Section 2. Crude MCS is used as a reference method. In what follows, M1 corresponds to an OK with a Gaussian kernel and M2 refers to an OK with a Matern $\nu = \frac{3}{2}$ kernel. The AK-MCS method is carried out with the individual metamodels M1 and M2 with the basic U-function. The U_{ens}-function is applied in the learning process of AK-MCS when EM is considered as a surrogate model.

4.1 Example 1: 2D illustrative example

The first example is a symmetric limit state function, with respect to x and y axes, with two standard normal random variables. The performance function is:

$$g(u_1, u_2) = u_1^2 - \frac{u_2^2}{2} + 2 \tag{22}$$

where u_1 and u_2 are standard normal random variables. The example is interesting because it has two disjoint failure regions that may be difficult to identify. AK-MCS is tested with individual metamodels and EMs. The reference failure probability is 0.025, it is estimated using a MCS with 10^4 samples. The proposed U_{ens}-function is applied and Table 1 gives the results, where N_{call} is the number of calls to the performance function.

Method	$\mathrm{N_{call}}$	$\mathrm{P_f}$	Miss-classified points
Monte Carlo	10^4	0.025	/
AK-MCS+M1	13	0.025	0
AK-MCS+M2	41	0.025	0
AK-MCS+BMA	13	0.025	0
AK-MCS+EG	13	0.025	0
AK-MCS+EME	13	0.025	0
AK-MCS+EV	14	0.025	0
AK-MCS+EA3	23	0.025	0

Table 1: Results of failure probability and miss-classified points in example 1.

The results show that the failure probability prediction is the same for all the combinations, whereas the total number of calls to the performance function varies. It is observed that M1 converges faster than M2 and the EMs exhibit an intermediable convergence rate. Most EM strategies perform equally well as the failure probability is always estimated accurately and moderate variations of the numerical efforts are observed. Global EMs converge as fast as the best individual metamodel since they quickly distinguish the best metamodel. Local EMs take a little bit more time to converge compared with the best metamodel. Indeed, M2 may be identified as the best emulator for some points, which reduces the convergence rate. The global LSF approximation might be badly influenced by the worst metamodel in those points. Furthermore, EA3 method has the highest number of calls to the performance function. This might be explained by the fact that M2 is selected as the best metamodel in some points of the DoE, and consequently chosen for the closest samples. But M2 has higher prediction variance and therefore small values of U and slow convergence of the method.

Figure 1 shows the evolution of the highest weights as the method is applied. All EM strategies succeed in identifying M1 as the best metamodel. Figure 2 shows the real LSF and its prediction using individual metamodels and each of EMs mentioned above. There is a perfect fit between the true response and EMs approximations. Metamodels also approaches well the LSF with a slight difference between the true response and M2 prediction. The initial DoE and the enriched points are also plotted. The later are close to the limit state function which indicates the ability of the U_{ens}-function to concentrate the search in the vicinity of the LSF. So the proposed U_{ens}-function can perform an efficient classification since the selected points for the learning process are all close to the LSF with zero miss-classified points for all the EMs.

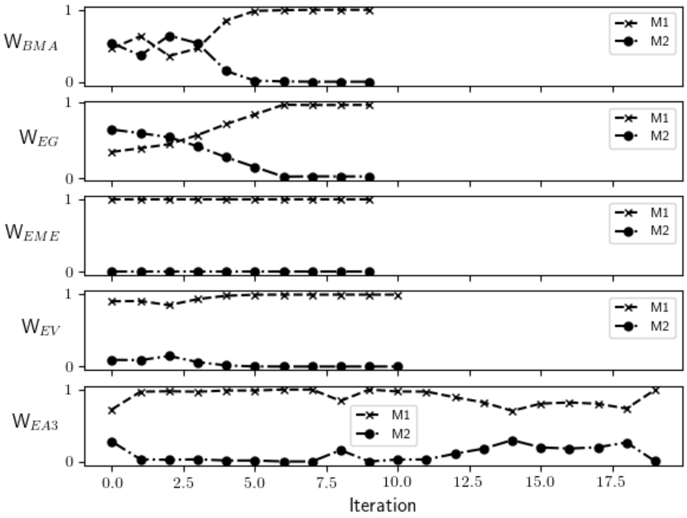


Figure 1: Variation of weights in example 1

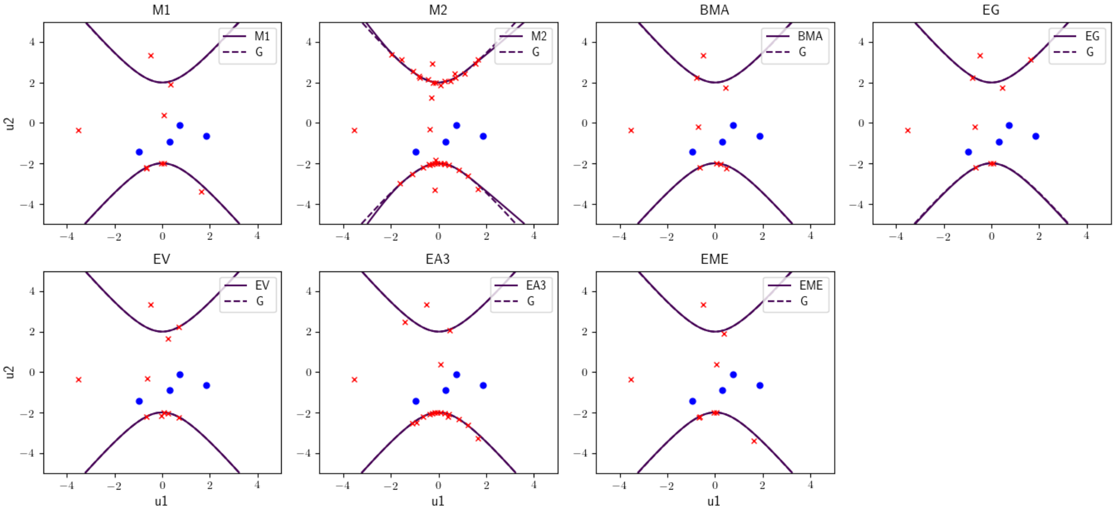


Figure 2: Approximation by AK-MCS combined to individual and EM metamodels in example 1

4.2 Example 2: series system with four branches

The second example is a series system with four branches. Its failure probability was calculated in [10] and is defined as:

$$g(u_1, u_2) = \min \begin{cases} 3 + 0.1(u_1 - u_2)^2 - \frac{(u_1+u_2)}{\sqrt{2}} \\ 3 + 0.1(u_1 - u_2)^2 + \frac{(u_1+u_2)}{\sqrt{2}} \\ (u_1 - u_2) - \frac{7}{\sqrt{2}} \\ (u_2 - u_1) - \frac{7}{\sqrt{2}} \end{cases} \tag{23}$$

where u_1 and u_2 are standard normal random variables. This example is more challenging than the previous one as the LSF function is less regular. The reference failure probability is 2.231×10^{-3} and it is estimated using MCS with 10^6 samples. The results of different combinations of AK-MCS with individual metamodels and with EMs are given in Table 4.2.

Method	$\mathrm{N_{call}}$	$\mathrm{P_f}$	Miss-classified points
Monte Carlo	10^6	2.231×10^{-3}	/
AK-MCS+M1	87	2.230×10^{-3}	1
AK-MCS+M2	59	0.472×10^{-3}	1759
AK-MCS+BMA	82	2.231×10^{-3}	2
AK-MCS+EG	127	2.231×10^{-3}	0
AK-MCS+EME	93	2.231×10^{-3}	0
AK-MCS+EV	70	2.231×10^{-3}	0
AK-MCS+EA3	105	2.231×10^{-3}	0

Table 2: Results of failure probability and miss-classified points in example 2.

The first observation is that M2 fails to converge towards the reference failure probability. This result can also be observed in Figure 3, where M2 approximation is far from the LSF in two branches. In fact, the size of the initial DoE and the corresponding positions affects badly the performance of M2. Despite the failure of M2 to approximate the LSF, EMs are capable to approach it, thus, predicting an accurate probability of failure. It is slightly better than the best individual estimation. The number of calls to the performance function differ from an EM strategy to another since they are based on different approaches. The EG method is the most computationally demanding. This is probably due to the α and β parameters choice since these values are problem-dependent and their tunning is empirical [13]. EA3 method also needs more iterations to converge compared with the best individual. As shown in Figure 4, M2, which has a higher prediction variance, is selected for more than 80% of the prediction points by the end of the learning process, which explains the slow convergence. Though M2 is the most selected in EA3, this EM approaches well the limit state function as we can see in Figure 3, where two corners among four are well captured. In the EME method, the best metamodel is immediately selected, based on GMSE, and adding points to the DoE do not influence the choice. BMA and EV are capable to slightly enhance the failure probability estimation and give a better approximation to the LSF with less iterations than the best individual metamodel. Finally, the modified learning function of AK-MCS has also proven its efficiency in classification in this example. The selected points for the learning process are all in the vicinity of the limit state function as shown in Figure 3. Furthermore, all the samples are well classified with EMs, which is not the case with the individual metamodels.

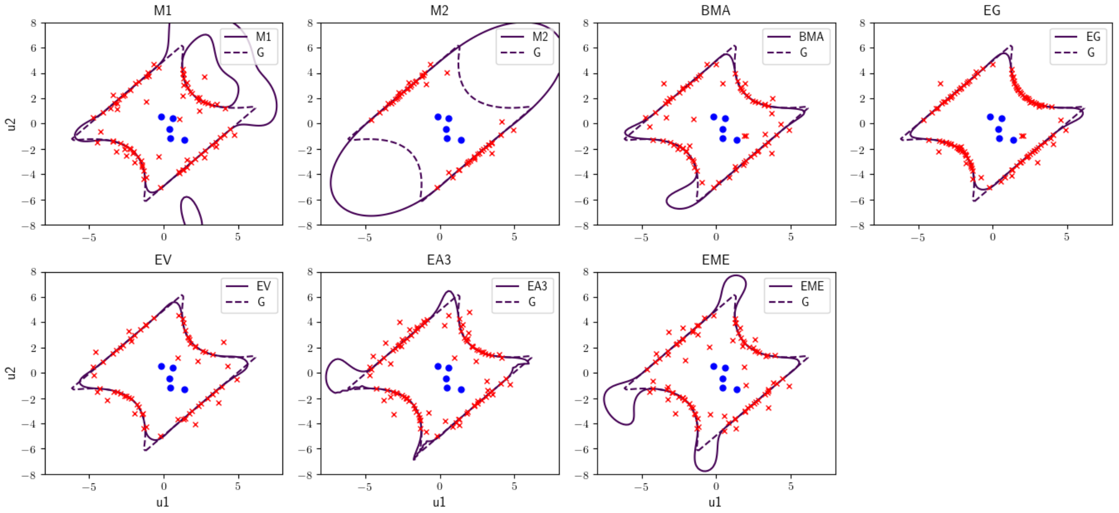


Figure 3: Approximation by AK-MCS combined to individual and EMs metamodels in example 2

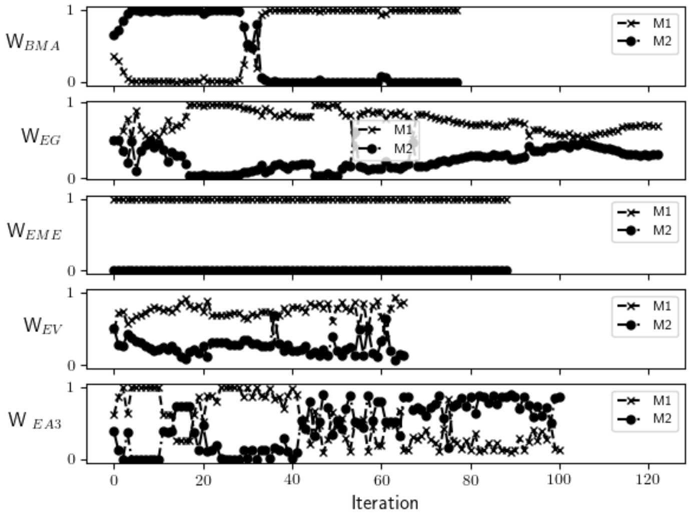


Figure 4: Variation of weights in example 2

5 CONCLUSIONS

In this work, we consider a reliability analysis based on metamodels in order to diminish the computational burden. In fact, advancements in computer science make the performance functions more complicated since complex finite element models are involved. The variety in metamodels types and tunings make choice very difficult to users. Hence, ensembles of metamodels (EM) is an approach to avoid a priori choices of metamodels. This approach is combined to AK-MCS, where a modified learning function of the AK-MCS method is proposed.

In the present paper, two ordinary Kriging metamodels are considered, with different kernel functions (Gaussian and Matern $\nu = \frac{3}{2}$). They are combined to form global and local EMs. The results show that the modified learning function of AK-MCS performs an efficient classification of points with all EM strategies. The failure probability estimated by EMs is equal to the reference MCS failure probability. The local variance based method (EV) and the global Bayesian model averaging (BMA) have reached the MCS failure probability with less number of calls to the performance function than the best metamodel. The heuristic of Goel (EG) and the weights optimization problem (EME) enable an accurate estimation of the failure probability but more iterations were required to reach the reference failure probability. The spatial local EM (EA3) has the same results as the two later methods. We were expecting that this method would yield better results, as it should better account for local behavior of the performance function. Surprisingly, this method does not perform better than the other strategies. This may be explained by the higher variance of the second metamodel

The results obtained so far point out the potential efficiency of ensemble of metamodels in reliability analysis. In fact, EM is not influenced by an inadequate a priori choice of a metamodel or by a poor metamodel that results from a certain choice of design of experiments as is the case in Example 2. However enhancement should be done for the weights calculation especially for EA3 method. Other validation examples should be performed with more than two Kriging models and in higher dimensions. A priori selection of the individual metamodels, based on an error metric, should be performed in order to discard the worst individuals, thus avoiding the slow convergence. Finally, when the best metamodel is not known beforehand, using EM seems to be a suitable strategy to perform metamodel selection and to enhance the performance of the AK method.

REFERENCES

[1] L. Faravelli, Response-surface approach for reliability analysis. *Journal of Engineering Mechanics*, **115**, 2763–2781, 1989.

[2] B. Sudret, A. Der Kiureghian, Comparison of finite element reliability methods. *Probabilistic Engineering Mechanics*, **17(4)**, 337–348, 2002.

[3] M. Papadrakakis, V. Papadopoulos, D. Lagaros, Structural reliability analyis of elastic-plastic structures using neural networks and Monte Carlo simulation. *Computer Methods in Applied Mechanics and Engineering*, **136**, 145-163, 1996.

[4] J-M. Bourinet, Rare-event probability estimation with adaptive support vector regression surrogates. *Reliability Engineering and System Safety*, **150**, 210–221, 2016.

[5] H. M. Gomes, A. M. Awruch, Comparison of response surface and neural network with other methods for structural reliability analysis. *Structural Safety*, **26(1)**, 49–67, 2004.

[6] M. Moustapha, J. M.Bourinet, B. Guillaume, B. Sudret, Comparative study of Kriging and support vector regression for structural engineering applications. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, **4(2)**, 04018005, 2018.

[7] V. Dubourg, B. Sudret, F. Deheeger, Metamodel-based importance sampling for structural reliability analysis. *Probabilistic Engineering Mechanics*, **33**, 47–57, 2013.

[8] B. Sudret, Meta-models for structural reliability and uncertainty quantification. *Proc. Asian-Pacific Symposium on Structural Reliability and its Applications, Singapore, Singapore, May 23-25, 2012.*

[9] B. J. Bichon, M. S. Eldred, L. P. Swiler, S. Mahadevan, J. M. McFarland, Efficient global reliability analysis for nonlinear implicit performance functions. *AIAA journal*, **46(10)**, 2459–2468, 2008.

[10] B. Echard, N. Gayton, M. Lemaire, AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation. *Structural Safety*, **33(2)**, 145–154, 2011.

[11] N. Lelièvre, P. Beaurepaire, C. Mattrand, N. Gayton, AK-MCSi: A Kriging-based method to deal with small failure probabilities and time-consuming models. *Structural Safety*. **73**, 1–11, 2018.

[12] V. S. Sundar, M. D. Shields, Reliability Analysis Using Adaptive Kriging Surrogates with Multimodel Inference. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, **5(2)**, 04019004, 2019.

[13] T. Goel, R. T. Haftka, W. Shyy, N. V. Queipo, Ensemble of surrogates. *Structural and Multidisciplinary Optimization*, **33(3)**, 199–216, 2007.

[14] E. Acar, M. Rais-Rohani, Ensemble of metamodels with optimized weight factors. *Structural and Multidisciplinary Optimization*, **37(3)**, 279–294, 2009.

[15] E. Acar, Various approaches for constructing an ensemble of metamodels using local measures. *Structural and Multidisciplinary Optimization*, **42(6)**, 879–896, 2010.

[16] E. Sanchez, S. Pintos, N. V. Queipo, Toward an optimal ensemble of kernel-based approximations with engineering applications. *Structural and Multidisciplinary Optimization*, **36(3)**, 247–261, 2008.

[17] X. Gu, J. Lu, H. Wang, Reliability-based design optimization for vehicle occupant protection system based on ensemble of metamodels. *Structural and Multidisciplinary Optimization*, **51(2)**, 533–546, 2015.

[18] R.Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection. *14th International Joint Conference on Artificial Intelligence,* **14(2)**, 1137–1145, 1995.

[19] L. Shi, R. J. Yang, P. Zhu, A method for selecting surrogate models in crashworthiness optimization. *Structural and Multidisciplinary Optimization*, **46(2)**, 159–170, 2012.

[20] L. E.Zerpa, N. V. Queipo, S. Pintos, J. L. Salager, An optimization methodology of alkalinesurfactantpolymer flooding processes using field scale numerical simulation and multiple surrogates. *Journal of Petroleum Science and Engineering*, **47(3-4)**, 197–208, 2005.

[21] J. A. Hoeting, D. Madigan, A. E. Raftery, C. T. Volinsky. Bayesian model averaging: a tutorial. *Statistical Science*, 382-401, 1999.

[22] A. E. Raftery, T. Gneiting, F. Balabdaoui, M. Polakowski, Using Bayesian model averaging to calibrate forecast ensembles. *Monthly Weather Review*, **133(5)**, 1155–1174, 2005.

[23] E. Acar, Effect of error metrics on optimum weight factor selection for ensemble of metamodels. *Expert Systems with Applications*, **42(5)**, 2703–2709, 2015.

[24] H. Liu, S. Xu, X. Wang, J. Meng, S. Yang, Optimal weighted pointwise ensemble of radial basis functions with different basis functions. *AIAA Journal*, 3117–3133, 2016.

[25] V. Dubourg, Adaptive surrogate models for reliability analysis and reliability-based design optimization. *PhD thesis, Université Blaise Pascal, Clermont-Ferrand, France*, 2011.

[26] D. Ginsbourger, C. Helbert, L. Carraro, Discrete mixtures of kernels for Kriging–based optimization. *Quality and Reliability Engineering International*, **24(6)**, 681–691, 2008.

[27] M. Baudin, A. Dutfoy, B. Iooss, A. L. Popelin, OpenTURNS: An industrial software for uncertainty quantification in simulation. *Handbook of Uncertainty Quantification*, 2001–2038, 2017.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

A TWO-STAGE SURROGATE MODELLING APPROACH FOR THE APPROXIMATION OF MODELS WITH NON-SMOOTH OUTPUTS

Maliki Moustapha[1], **Bruno Sudret**[1]

[1]Chair of Risk, Safety and Uncertainty Quantification
ETH Zurich
e-mail: {moustapha,sudret}@ibk.baug.ethz.ch

Keywords: Surrogate modelling, Non-smooth outputs, Support vector machines, Kriging.

Abstract. *Surrogate modelling has become an important topic in the field of uncertainty quantification as it allows for the solution of otherwise computationally intractable problems. The basic idea in surrogate modelling consists in replacing an expensive-to-evaluate black-box function by a cheap proxy. Various surrogate modelling techniques have been developed in the past decade. They always assume accommodating properties of the underlying model such as regularity and smoothness. However such assumptions may not hold for some models in civil or mechanical engineering applications, e.g., due to the presence of snap-through instability patterns or bifurcations in the physical behavior of the system under interest. In such cases, building a single surrogate that accounts for all possible model scenarios leads to poor prediction capability. To overcome such a hurdle, this paper investigates an approach where the surrogate model is built in two stages. In the first stage, the different behaviors of the system are identified using either expert knowledge or unsupervised learning, i.e. clustering. Then a classifier of such behaviors is built, using support vector machines. In the second stage, a regression-based surrogate model is built for each of the identified classes of behaviors. For any new point, the prediction is therefore made in two stages: first predicting the class and then estimating the response using an appropriate recombination of the surrogate models. The approach is validated on two examples, showing its effectiveness with respect to using a single surrogate model in the entire space.*

1 INTRODUCTION

The surrogate modelling of computer simulations has become paramount in many engineering applications that rely heavily on high-fidelity models. Surrogate models indeed allow for an inexpensive approximation of the model input-output relationship, thus making computationally intensive analyses, such as design optimization or uncertainty quantification, affordable. In the common setting, the underlying surrogated model is assumed to exhibit accommodating properties such as smoothness and continuity. However, numerous engineering problems involve non-smooth functions, e.g. crash simulation in the automotive industry. Indeed the original model may exhibit some sharp localized features and discontinuities may occur when a bifurcation or an instability appears in the solution path. In general, the functions of interest in this work exhibit different behaviors which can be mapped to certain combinations of the input parameters. The transitions between these domains may be non-smooth, often featuring discontinuities. In mechanical engineering, typical examples are buckling and snap-through characterized by sudden behavior changes (See [3] for instance). Approximating such models with a traditional smooth surrogate model leads to large errors. In this work, we consider a two-stage approach where the different behaviors are first localized and classified and then locally approximated. A similar approach was investigated in [1] and [8], albeit without approximation of the model responses as the latter were used as constraints in an optimization setting where only feasibility of a given design is of interest. In this paper, the general workflow of the proposed methodology is first introduced. This is followed by a brief description of the different blocks of the algorithm. Finally, two application examples are used to show the effectiveness of the proposed approach.

2 PROPOSED APPROACH

2.1 Workflow of the method

The proposed approach for handling non-smooth functions consists of multiple steps as described in the flowchart of Figure 1. Let us consider an experimental design which consists of N uniformly sampled points $\left\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(N)}\right\}$ and their corresponding model evaluations $\left\{y^{(1)} = \mathcal{M}\left(\boldsymbol{x}^{(1)}\right), \dots, y^{(N)} = \mathcal{M}\left(\boldsymbol{x}^{(N)}\right)\right\}$. To build the predictor, the following steps are undertaken:

1. *Clustering*: This is the first step of the approach when the analyst attributes to each observation $\boldsymbol{y}^{(i)}, i = \{1, \dots, N\}$ a class which corresponds to an identified behavior of the system. In the ideal case, this can be done manually using expert knowledge. In the general case though, it is more convenient to rely on an automated approach where the classes are directly learned from the data using *unsupervised learning*.

2. *Classification*: Once the classes are clearly identified, they are mapped to the input space which is then partitioned accordingly. This step is carried out using support vector machines for classification as detailed in the next section.

3. *Regression/Interpolation*: Eventually, the dataset is split into the different groups identified in the previous two steps. For each group, a local surrogate model $\left\{\widehat{\mathcal{M}}_k,\ k = 1, \dots, K\right\}$ is built.

Once the local models are built, it is necessary to recombine them when evaluating a new point. As shown on the right side of Figure 1, this is achieved in three steps:

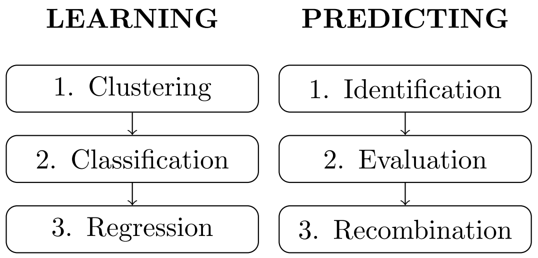


Figure 1: Illustration of the surrogate modelling approach.

1. *Identification*: The very first step is to predict to which class belongs the new point. The previously built support vector classifier can be used in that respect.
2. *Evaluation*: The new point is then evaluated using the different surrogate models.
3. *Recombination*: The final approximation is obtained by combining the different predictions as follows:
$$\widehat{\mathcal{M}}\left(\boldsymbol{x}\right) = \sum_{k=1}^{K} w_k\left(\boldsymbol{x}\right) \widehat{\mathcal{M}}_k\left(\boldsymbol{x}\right), \tag{1}$$
$w_k\left(\boldsymbol{x}\right)$ are weight functions defined such that $\sum_{k=1}^{K} w_k\left(\boldsymbol{x}\right) = 1$. Two different types of weight functions are considered in this work as explained in the next section. In the sequel, we first describe briefly the two surrogate model types used here, namely support vector machines and Kriging.

2.2 Support vector machines for classification basics

Support vector machines are a powerful learning technique developed by Vapnik ([11]) for classification (SVC) and regression (SVR) problems. Let us consider a dataset $\mathcal{C} = \left\{\left(\boldsymbol{x}^{(i)}, \ell^{(i)}\right), i = 1, \dots, N\right\}$, where $\boldsymbol{x}^{(i)}$ are M-dimensional input points and $\ell^{(i)} = \{-1, 1\}$ are the corresponding labels, in the particular case of binary classification considered here.

The support vector classifier is a function of the following form ([9]):

$$\mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right) = \sum_{i=1}^{N} \alpha_i \, \ell^{(i)} \, k\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}\right) + b, \tag{2}$$

where α_i and b are coefficients to calibrate and $k\left(\right)$ is the so-called *kernel* function. The coefficients of the expansion are actually found by solving the following optimization problem ([9]):

$$\begin{array}{ll} \min\limits_{\boldsymbol{\alpha}} & \dfrac{1}{2}\boldsymbol{\alpha}^T\left(\widetilde{\boldsymbol{K}}\boldsymbol{Y}\boldsymbol{Y}^T\right)\boldsymbol{\alpha} + \boldsymbol{c}^T\boldsymbol{\alpha} \\ \text{subject to:} & \boldsymbol{\alpha}^T\boldsymbol{Y} = 0, \qquad \alpha_i \geq 0, \qquad i = \{1, \dots, N\}, \end{array} \tag{3}$$

where $\boldsymbol{c} = \{-1, \dots, -1\}$ is a column vector of size N and $\widetilde{\boldsymbol{K}} = \boldsymbol{K} + 1/C\boldsymbol{I}_N$. In the latter equation, $\boldsymbol{K}$ is the so-called Gram matrix whose components read $K_{ij} = k\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right)$ for $i, j \in \{1, \dots, N\}$, $\boldsymbol{I}_N$ is the identity matrix of size N and C is a penalty coefficient which acts as a regularization term against overfitting.

In this work, we consider the Matérn $5/2$ kernel which reads:

$$k\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \left(1 + \sqrt{5}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\gamma} + \frac{5}{3}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{\gamma^2}\right) \exp\left(-\sqrt{5}\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|}{\gamma}\right), \tag{4}$$

where $\gamma > 0$ is a parameter that needs to be calibrated. Together with the penalty term C, they constitute the set of hyperparameters $\boldsymbol{\theta} = \{C, \gamma\}$ whose proper calibration is crucial for the accuracy of the trained SVC model. Here they are calibrated by minimizing the span estimate of the leave-one-out error ([10]).

After setting up the model, the predicted boundary between the two classes is defined by $\left\{\boldsymbol{x} \in \mathbb{X} : \mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right) = 0\right\}$ while the class of a prediction is given by $\operatorname{sign}\left(\mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right)\right)$.

2.3 Kriging basics

Kriging a.k.a. Gaussian process modelling is a surrogate modelling technique where the function to approximate is considered to be the realization of a stochastic Gaussian process which reads ([6, 7]):

$$\mathcal{M}\left(\boldsymbol{x}\right) = \sum_{j=1}^{p} \beta_j f_j\left(\boldsymbol{x}\right) + Z\left(\boldsymbol{x}\right), \tag{5}$$

where f_j and β_j are a set of p regressors and their corresponding coefficients and $Z\left(\boldsymbol{x}\right)$ is a second-order zero-mean stationary Gaussian process whose covariance reads $\text{Cov}\left[\boldsymbol{x}, \boldsymbol{x}'\right] = \sigma^2 R\left(\boldsymbol{x}, \boldsymbol{x}'; \gamma\right)$. In the latter equation, σ^2 is a constant variance of the process and R is an auto-correlation function with parameters $\boldsymbol{\gamma}$.

The auto-correlation function encodes assumptions made about the function to approximate, *e.g.* smoothness, derivability, etc. In this work, we consider the Matérn $5/2$ auto-correlation as in Eq. (4). The training of the Kriging model is a two-step process. First, the coefficients of the regression together with the process variance are estimates using least-square or equivalently maximum likelihood minimization. Second, the optimal parameters of the auto-correlation function are estimated using cross-validation or maximum likelihood. Once the estimates of the hyperparameters $\left\{\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\sigma}}^2, \widehat{\gamma}\right\}$ are set, the prediction for a new point is assumed to follow a Gaussian distribution whose mean is the actual Kriging predictor and reads:

$$\mu_{\mathcal{M}}\left(\boldsymbol{x}\right) = \boldsymbol{f}^T\left(\boldsymbol{x}\right)\widehat{\boldsymbol{\beta}} + \boldsymbol{r}^T\left(\boldsymbol{x}\right)\boldsymbol{R}^{-1}\left(\boldsymbol{y} - \boldsymbol{F}^T\widehat{\boldsymbol{\beta}}\right), \tag{6}$$

where $\boldsymbol{R}$ is the Gram matrix defined such that $R_{ij} = R\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}; \widehat{\gamma}\right)$, $\boldsymbol{r}\left(\boldsymbol{x}\right) = \left\{R\left(\boldsymbol{x}, \boldsymbol{x}^{(i)}; \widehat{\gamma}\right),\right.$ $\left. i = 1, \dots, N\right\}$ is a cross-correlation vector, $\boldsymbol{F} = \left\{F_{ij} = f_j\left(\boldsymbol{x}^{(i)}\right), i = 1, \dots, N, j = 1, \dots, p\right\}$ and $\boldsymbol{y} = \left\{y_i = \mathcal{M}\left(\boldsymbol{x}^{(i)}\right), i = 1, \dots, N\right\}$ are the observations in the experimental design.

2.4 Models recombination using SVC

In the second step of the approach, an SVC model is used to partition the space. In this paper, only cases with two possible behavior scenarios are considered. Let us assume now that the two sub-regions of the space corresponding to the negative and positive labels of the classifiers are respectively denoted by $\mathcal{R}_1$ and $\mathcal{R}_2$. As explained above, the experimental design $\mathcal{D}$ is split in two subsets $\mathcal{D}_k = \left\{\left(\boldsymbol{x}^{(i)}, y^{(i)}\right) \in \mathcal{D} : \boldsymbol{x}^{(i)} \in \mathcal{R}_k\right\}, k = \{1, 2\}$. Using the subset $\mathcal{D}_1$ (resp. $\mathcal{D}_2$), a Kriging model denoted by $\widehat{\mathcal{M}_1}$ (resp. $\widehat{\mathcal{M}_2}$) is built.

Let us now consider a new point $\boldsymbol{x}$ to evaluate. As described above, we first predict its class, $\operatorname{sign}\left(\mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right)\right)$, using the classifier. This point is then evaluated using the local surrogate models which are eventually recombined following Eq. (1). Two recombination schemes are considered here:

Binary approach

In this case, only the model built over the region in which $\boldsymbol{x}$ is predicted to belong to is used ([2, 4]). The weight function is therefore a simple indicator function, *i.e.* :

$$w_k\left(\boldsymbol{x}\right) = \mathbb{1}_{\mathcal{R}_k}\left(\boldsymbol{x}\right) = \left\{ \begin{array}{ll} 1 & \text{if } \boldsymbol{x} \in \mathcal{R}_k, \\ 0 & \text{otherwise}, \end{array} \right. \tag{7}$$

For the case with only two possible scenarios considered here, Eq. (1) can then be simplified into:

$$\widehat{\mathcal{M}}\left(\boldsymbol{x}\right) = \mathbb{1}_{\mathcal{R}_1}\left(\boldsymbol{x}\right)\widehat{\mathcal{M}_1}\left(\boldsymbol{x}\right) + \mathbb{1}_{\mathcal{R}_2}\left(\boldsymbol{x}\right)\widehat{\mathcal{M}_2}\left(\boldsymbol{x}\right). \tag{8}$$

This is a simple approach but it may yield large errors when the classification of the new point is wrong. The next approach tackles this issue by considering the uncertainty related to the support vector machine classifier.

Weighting approach

In this case, weights associated to each model are computed using the SVC prediction. The more likely a point is to belong to a class, the higher the corresponding weight and vice-versa. To compute the weight, the output of the classifier is post-processed into posterior probabilities using the following parametric sigmoid ([5]):

$$\mathbb{P}\left(\ell\left(\boldsymbol{x}\right) = 1 | \mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right)\right) = \frac{1}{1 + \exp\left(A\, \mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right) + B\right)}, \tag{9}$$

where A and B are parameters that are fit using maximum likelihood estimation on the experimental design. The final prediction is then obtained by setting these probabilities as weights, *i.e.* :

$$w_1\left(\boldsymbol{x}\right) = 1 - \mathbb{P}\left(\ell\left(\boldsymbol{x}\right) = 1 | \mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right)\right) \quad \text{and} \quad w_2\left(\boldsymbol{x}\right) = \mathbb{P}\left(\ell\left(\boldsymbol{x}\right) = 1 | \mathcal{M}^{\text{SVC}}\left(\boldsymbol{x}\right)\right). \tag{10}$$

3 APPLICATIONS

We consider two applications to illustrate the proposed methodology, namely a two-dimensional mathematical function and a snap-though mechanical problem.

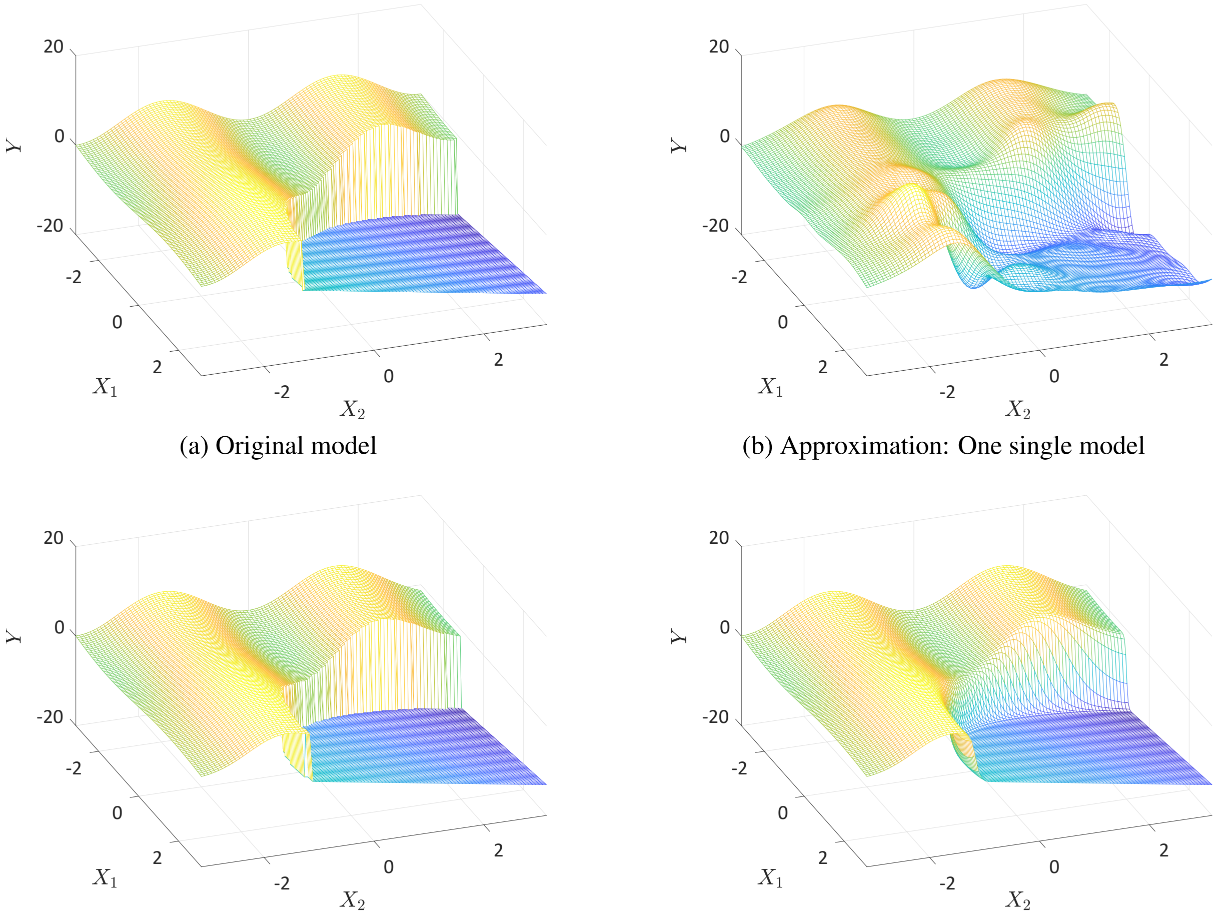
3.1 Two-dimensional mathematical function

Let us consider the two-dimensional mathematical function defined by:

$$\mathcal{M}\left(\boldsymbol{x}\right) = \left\{ \begin{array}{ll} \sin(x_1) + 7\sin(x_2)^2 & \text{if } (x_1 - \pi)^2 + (x_2 - \pi)^2 - 2\pi^2 \geq 0, \\ x_1 - 2x_2 - 10; & \text{otherwise}, \end{array} \right. \tag{11}$$

where $\boldsymbol{x} \in [-\pi, \pi]^2$.

Figure 2a illustrates the function which consists of two distinct regions over which different behaviors of the model can be observed. On one side, the function is highly non-linear whereas on the other, the function is linear and nearly flat. To approximate this function, we use an experimental design of size 100. The two classes are identified using K-means clustering and the input space is partitioned as illustrated in Figure 3 by support vector machines. In this figure, the training points that belong to the flat and highly non-linear regions are shown in blue circles and red squares respectively. With a 100-point training set, the classifier, shown by the black curve, is close enough to the true one, shown by the magenta curve. After building surrogates in



(a) Original model

(b) Approximation: One single model

(c) Approximation: Binary recombination

(d) Approximation: Weighted recombination

Figure 2: Two-dimensional mathematical problem: original *vs.* surrogate models

each region, the two recombination schemes are applied. Figures 2c and 2d show the resulting approximations. The binary case produces a very accurate representation of the model, the only error being the position of the discontinuity. The weighted recombination scheme produces a smooth transition in the margin between the two regions. Finally, using one single surrogate model leads to the approximation shown in Figure 2b where spurious curvatures are added in the vicinity of the discontinuity.

For a quantitative comparison of the different approaches, the following two errors metrics are considered:

$$\begin{aligned} NMSE &= \sum_{i=1}^{N_{val}} \left(\mathcal{Y}_i - \widehat{\mathcal{Y}}_i \right)^2 / \sum_{i=1}^{N_{val}} \left(\mathcal{Y}_i - \bar{\mathcal{Y}} \right)^2, \\ MAE &= \sum_{i=1}^{N_{val}} \left| \mathcal{Y}_i - \widehat{\mathcal{Y}}_i \right| / N, \end{aligned} \tag{12}$$

where $NMSE$ and MAE respectively stand for *normalized mean square error* and *mean absolute error*. In these equations, $\mathcal{Y}$ and $\widehat{\mathcal{Y}}$ are responses of the original and surrogate models on a validation set of size $N_{val} = 10,000$. Table 1 shows the resulting errors where cases $\#1$, $\#2$ and $\#3$ respectively stand for single surrogate, binary recombination and weighted recombina-

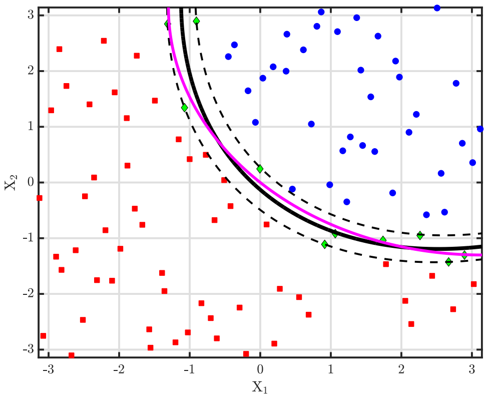


Figure 3: Two-dimensional mathematical problem: classification of the input points using support vector machines

tion. The proposed approach improves the prediction considering any of the two metrics. It is not clear though which of the two recombination schemes is more effective.

	Case #1	Case #2	Case #3
$NMSE$	0.0911	0.0530	0.0346
MAE	1.0124	0.2048	0.2436

Table 1: Two-dimensional mathematical problem: comparison of the resulting errors

3.2 Truss structure subject to snap-through

The second example addresses the problem of a geometrically non-linear two-bar truss structure with a snap-through behavior as illustrated in Figure 4. When loaded, such a structure often behaves linearly with small displacements. However, when a critical limit is reached, the structure becomes unstable and undergoes a sudden large displacement by snapping through another equilibrium point. In this example, we approximate the displacements w of the tip of such a structure considering the random parameters shown in Table 2.

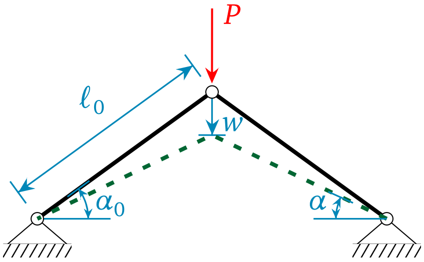


Figure 4: Illustration of the truss structure subject to snap-through

It can be shown that the load at a deformed position follows a relationship given by:

$$P = -2EA\tan(\alpha)(\cos(\alpha_0) - \cos(\alpha)) \tag{13}$$

Parameter	Distribution	Mean	C.o.V.
Load (P in N)	Gumbel	430	0.20
Young's modulus (E in GPa)	Lognormal	210	0.10
Cross sectional area (A in cm^2)	Gaussian	10	0.05

Table 2: Truss snap-through problem: probabilistic input model

where α_0 and α are the inclination angles of the bars at the initial and deformed positions. The corresponding displacement of the tip of the truss then reads:

$$w = l_0 \cos(\alpha_0) \left(\tan(\alpha_0) - \tan(\alpha)\right). \tag{14}$$

In this example, we set $l_0 = 5$ m and $\alpha_0 = 10°$. Using an experimental design of 100 points drawn following the distribution in Table 2, the displacements are computed and shown in Figure 5. We can clearly observe the two behaviors that lead to entirely different displacements.

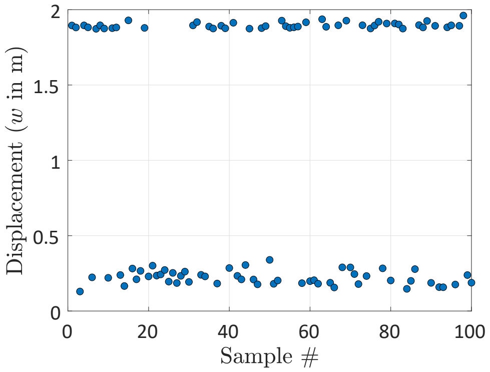
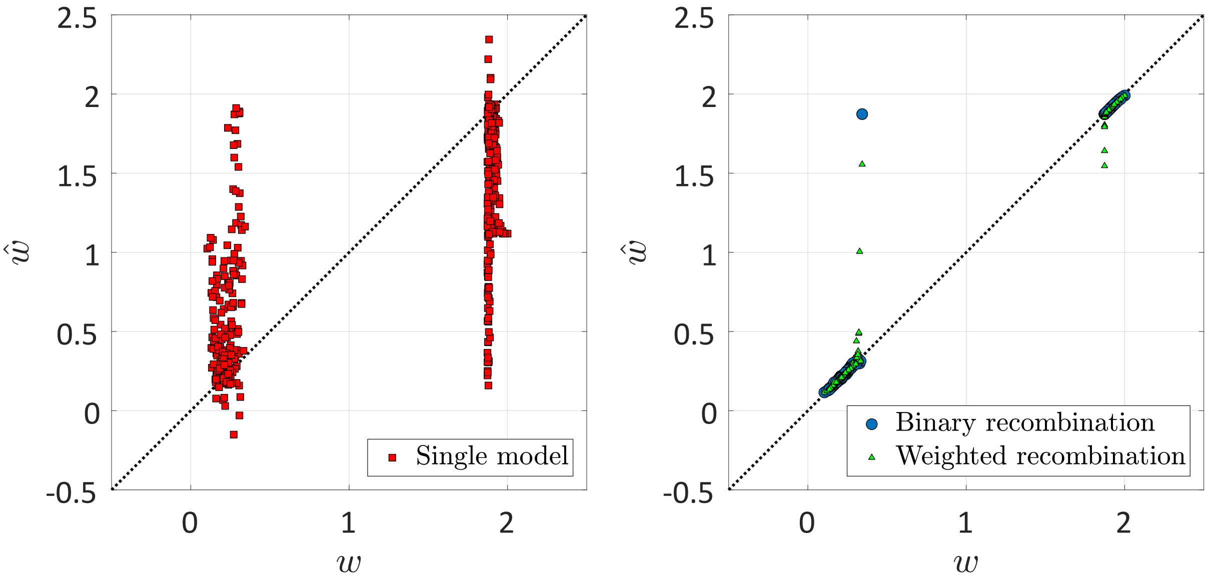


Figure 5: Truss snap-through problem: experimental design model responses

The proposed approach is applied to this experimental design. Figure 6 and Table 3 show the results for comparison. When using a single surrogate model, the instability is not captured and displacements are predicted continuously over the two extreme cases. The proposed approach allows to accurately locate and isolate the input sub-regions that lead to each of the scenarios. The binary approach produces extremely accurate results as long as the class is correctly predicted by the SVM model. The weighted recombination scheme yields locally less accurate results but behaves better than the binary one close to the discontinuity.

	Case $\#1$	Case $\#2$	Case $\#3$
$NMSE$	0.2478	0.0803	0.0670
MAE	0.2714	0.0390	0.0399

Table 3: Truss snap-through problem: comparison of the resulting errors



(a) One single model

(b) Multiple models recombined

Figure 6: Truss snap-through problem: original *vs.* predicted responses

4 CONCLUSION

This paper presents a two-stage approach for the approximation of functions with non-smooth outputs. Focus is given to the particular case when multiple behaviors of the function can be observed. The proposed approach consists in first identifying such behaviors and then classifying them using support vector machines. The resulting prediction is obtained by building local surrogates in each region and then recombining them using two different schemes. Two application examples show the efficiency of the approach with respect to using a unique global surrogate model. The accuracy of the resulting predictions however relies on the accuracy of the classification step. The latter can be increased by using adaptive sampling scheme in order to more accurately define the boundaries between the two regions. Furthermore, the proposed scheme is limited to binary problems and will be extended to the more general case when more than two behaviors of the system can be observed.

References

[1] Basudhar, A., S. Missoum, and A. H. Sanchez (2008). Limit state function identification using Support Vector Machines for discontinuous responses and disjoint failure domains. *Prob. Eng. Mech 23*, 1–11.

[2] Boroson, E. and S. Missoum (2017). Stochastic optimization of nonlinear energy sinks. *Struct. Multidisc. Optim. 55*, 633–646.

[3] Hrinda, G. A. (2010). Snap-through instability patterns in truss structures. In *Proc. 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 12 - 15 April 2010, Orlando, Florida*.

[4] Moustapha, M. (2016). *Adaptive surrogate models for the reliable lightweight design of automotive body structures*. Ph. D. thesis, Université Blaise Pascal, Clermont-Ferrand, France.

[5] Platt, J. C. (1999). Advances in kernel methods. Chapter Fast training of support vector machines using sequential minimal optimization, pp. 185–208. MIT Press.

[6] Rasmussen, C. E. and C. K. I. Williams (2006). *Gaussian processes for machine learning* (Internet ed.). Adaptive computation and machine learning. Cambridge, Massachusetts: MIT Press.

[7] Santner, T. J., B. J. Williams, and W. I. Notz (2003). *The Design and Analysis of Computer Experiments*. Springer, New York.

[8] Serna, A. and C. Bucher (2009). Advanced surrogate models for multidisciplinary design optimization. In *6th Weimar Optimization and Stochastic Days 2009, October 15th-16th, Weimar, Germany*.

[9] Smola, A. J. and B. Schölkopf (2004). A tutorial on support vector regression. *Stat. Comput. 14*, 199–222.

[10] Vapnik, V. and O. Chapelle (2000). Bounds on error expectation for support vector machines. *Neural Comput. 12*(9), 2013–2036.

[11] Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

SURROGATE MODELING CONSIDERING MEASURING DATA AND THEIR MEASUREMENT UNCERTAINTY

Thomas Oberleiter[1], Andreas Michael Müller[2], Tino Hausotte[2] and Kai Willner[1]

[1]Friedrich-Alexander-Universität Erlangen-Nürnberg
Chair of Applied Mechanics
Egerlandstr. 5, 91058 Erlangen, Germany
e-mail: {thomas.oberleiter, kai.willner}@fau.de

[2] Friedrich-Alexander-Universität Erlangen-Nürnberg
Institute of Manufacturing Metrology
e-mail: {andreas.mueller, tino.hausotte}@fmt.fau.de

Keywords: Kriging, Measuring Data, Measuring Uncertainty, Surrogate Modeling

Abstract. *Virtual approaches to manufacturing processes are a common tool in developing components today. Simulations are always containing uncertainties like simplifying assumptions in computer aided modelling, material deviations, fluctuating external loads or other known and unknown influences. To integrate such uncertainties in an early design stage, the input parameters should be defined as intervals, because insufficient data may be available at this stage to provide probability distributions. To consider such epistemic uncertainties, a large number of intervals can be merged into a fuzzy number. For each interval a membership value is assigned which depends on the interval limits and an expert estimation. However, this interval modelling leads to a very high number of expensive evaluations, which is not feasible for a high number of uncertain input parameters. To reduce the calculation time, surrogate models are used. Here, the full model is evaluated only at some grid points and the system response is approximated by mathematical approaches. Design and Analysis of Computer Experiments (DACE) offers a suitable surrogate model based on the Kriging method. The system model substituted in this way can be evaluated in an efficient way, but in addition to the uncertain simulation results, the approximation error dependent on the surrogate model has to be considered. Investigations of first prototypes lead to new knowledge that can be used to improve the surrogate model. Measurements, however, also include errors that are composed of systematic and random errors. The systematic measurement errors are specific errors for each measuring system and task, which are usually corrected during the measurement. However, an estimation of the random measurement error, which represents the precision of the measurement can be taken into account. Two methods are presented. Either an additional constant term is implemented in the standard Kriging or a superposition of two standard Kriging models, which are based on the simulation data and the measurement data, is used. As an application example a cold forging process of a steel gearwheel is employed.*

1 Introduction

The tasks of developing components and defining their tolerances are part of the design process. Due to increased requirements and shorter development times, the manufacturing processes are designed using computer aided simulations These simulations play an important role in manufacturing processes and are therefore often used [1]. While simulations are reproducible, real processes are subject to fluctuations, such that a virtual model can never reproduce a real-world setting and tolerance limits are necessarily needed, to ensure the functions of the developed components.

The origin of the fluctuations are caused by epistemic uncertainties as well as aleatoric uncertainties [2]. Epistemic uncertainties result from insufficient information and can be eliminated by additional effort. Aleatoric uncertainties are systemdependent deviations that cannot be prevented. These uncertainties have to be taken into account in the process design. Varying parameters are a common way of doing this. Therefore, many evaluations are needed to calculate a simulation for different parameter combinations. The system used in this manuscript is a cold forging process to build gearwheels [3].

Due to the complexity of the simulation and the frequently required evaluations, the system is approximated using a surrogate model. This is done in section 2 using Design and Analysis of Computer Experiments (DACE). The surrogate model should reproduce the computer simulation with sufficient accuracy. Subsequent measurements of real parts in section 3 provide verification of the surrogate model. It can be seen that the surrogate model approximates the simulation well, but the simulation does not match with the results of the real measurements. In section 4, the measurement data is then used to discuss two methods to optimize the simulation results. Considering the measurement method and its measurement uncertainty, the possibility of optimization is limited, which is shown in section 4.3 discussing an exclusively measurement based surrogate model. A first approach to integrate the measurement uncertainty into the discussed surrogate models is shown in section 4.4. At the end there is a short summary and an outlook in section 5.

2 Surrogate model

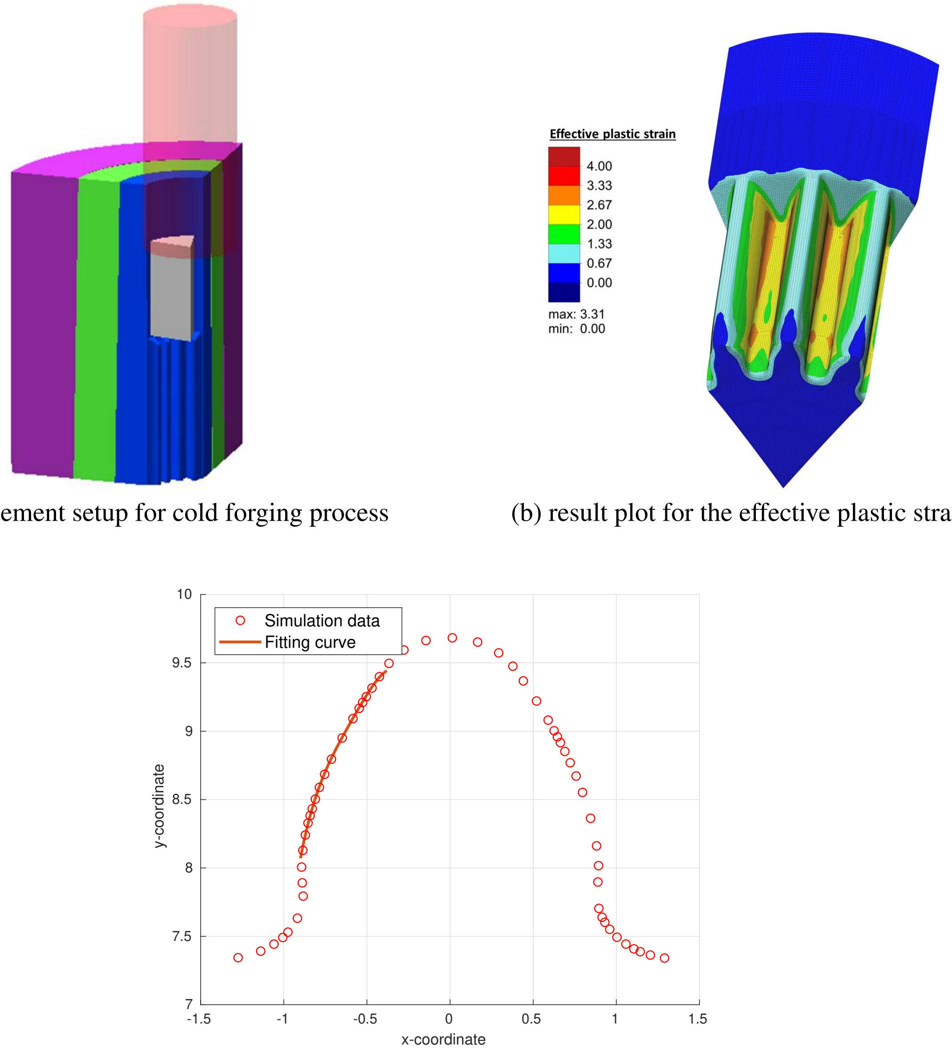
In general, a surrogate model is an approximation to the output function $\hat{f}(\mathbf{p}) \approx f(\mathbf{p})$ at the parameter combination $\mathbf{p}$. Each vector $\mathbf{p} = \{p_1, p_2, ..., p_k\}^T$ consists of k entries. The number of entries k results from the number of input parameters. In order to create a surrogate model, the system is approximated by using a few sampling points

$$\mathbf{P} = \{\mathbf{p_1}, \mathbf{p_2}, ..., \mathbf{p_n}\}^T. \tag{1}$$

Here we use an interpolation, such that the relationship $\hat{f}(\mathbf{p}_i) = f(\mathbf{p}_i)$ applies to the sampling points $\mathbf{p}_i$.

2.1 Cold forging process simulation

In the following, the function f represents a finite element simulation of a cold forging process. In this process, a cylindrical blank with diameter d is extruded forward into a gear die by a punch. The process runs at room temperature (the blank is also not heated, therefore cold forming) and using suitable lubricants, which have a significant influence on the friction force between the blank and the die [4]. Two parameters, the diameter $d = p_1 \in [0.019169\text{ m}, 0.019589\text{ m}]$ and friction coefficient $\mu = p_2 \in [0.08, 0.18]$ are considered as uncertain. This setup is simulated in the commercial software Simufact Forming. This tool is well suited for massive forming processes [5, 6]. Fig. 1 (a) shows the experimental setup in the simulation environment. Because of symmetry it is sufficient to simulate only a quarter of the whole model.



(a) finite element setup for cold forging process

(b) result plot for the effective plastic strain

(c) 2D cut of the result plot for one cog with a fitting curve for the left involute

Figure 1: Simulation data from the cold forging process simulation

The simulation results, as can be seen in Fig. 1 (b) for the effective plastic strain, were explained in [3]. The quantity of interest for a tolerance analysis in gear meshing is the cog involute (see Fig. 1 (c)). To obtain the involute, the STL file of the formed blank is exported from the simulation and a cut is made in the middle of the blank, perpendicular to the flow direction. The nodes located in the immediate vicinity of the cut then describe the 2D shape of the deformed blank, the cogs. The involute can be determined using the standard in [7]. The finite element mesh is designed such that about 16 points are located on the involute.

2.2 DACE model

For the surrogate model $\hat{f}$ exist different approaches, see e.g. [8]. The surrogate model used here is the DACE model. This form of surrogate model is based on the Kriging model developed by D.G. Krige [9]. It contains a random process $\mathbf{Z}(\mathbf{p})$, which influences the surrogate model depending on the distance of the evaluation point to the sampling points [10]. This random process $\mathbf{Z}(\mathbf{p})$ is assumed to have zero mean and covariance between $\mathbf{Z}(\mathbf{p_i})$ and $\mathbf{Z}(\mathbf{p})$, which results in

$$E[\mathbf{Z}(\mathbf{p_i}),\mathbf{Z}(\mathbf{p})] = \sigma^2 R(\theta,\mathbf{p_i},\mathbf{p}), \tag{2}$$

with σ^2 as process variance and $R(\theta,\mathbf{p_i},\mathbf{p})$ as correlation function. For the point correlation

$$R(\theta,\mathbf{p_i},\mathbf{p}) = \prod_{j=1}^{n_c} R_j(\theta, p_j^{(i)} - p_j) \tag{3}$$

holds and a correlation matrix and a covariance matrix can be obtained. For the correlation function the cubic approach,

$$R_j(\theta, p_j^{(i)}, p_j) = 1 - 3\xi^2 + 2\xi^3 \quad \textit{with} \quad \xi = min\{1, \theta|p_j^{(i)} - p_j|\} \tag{4}$$

is used, which contains a weighting factor θ, defining the importance of the parameters. Calculation of optimal values for θ corresponds to a maximum likelihood estimation, for more details see [11].

In addition to the correlation context, the DACE method consists of a regressions model. It is a linear combination of n_c functions $r_1(\mathbf{p})...r_{n_c}(\mathbf{p})$ with regression parameters β_i. Regression model and random process results in the DACE approach

$$\hat{\mathbf{F}}(\mathbf{p}) = \sum_{i=1}^{n_c} \beta_i r_i(\mathbf{p}) + \mathbf{Z}(\mathbf{p}). \tag{5}$$

This model is applied using the MATLAB toolbox following [11, 12]. In the upcoming chapters a first degree polynomial is used as regression model and the weighting factor $\theta \in [0.001, 10]$.

The model is build with a three-level full factorial design for the two parameters (diameter d, friction coefficient μ), resulting in $n = 3^2$ sampling points. The influence of other sampling strategies is not considered in this document, but will be the topic of future research.

2.3 Model for the involute

As already mentioned in section 2.1, the involute is the quantity of interest. More precisely the left involute of the first cog serves as result function $y(x)$. The first cog is defined as the upper cog intersected by the y-axis of the coordinate system (coordinates $y > 0$, $x_{min} < 0 < x_{max}$).

Due to high numerical costs of the finite element simulations, a surrogate model for the involute is needed for parameter studies. If the discrete representation of the involute as given by the finite element result is directly approximated, a large number of parallel surrogate models are required for each individual point. Moreover, a comparison of two involutes is then difficult because the points lie not necessarily in the normal direction of the underlying cog surface and do not allow a determination of the distance between the two involutes. For this reason, the points of the finite element mesh from the simulation are approximated by a polynomial of the form

$$y(x) = c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0 \tag{6}$$

and a surrogate model $\hat{y}(\mathbf{p}, x)$ is build, where the coefficient c_{0-4} of the polynomial are directly approximated by individual DACE models $\hat{c}_i(\mathbf{p})$, as presented in section 2.2. Thus, the involutes from the surrogate model can be compared with the mesh points of the simulation and later also with the measurement points of the real measurement. The root mean squared error (RMSE), which is generally defined by

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \tag{7}$$

is used for that comparison, where $y_i = y(x_i)$.

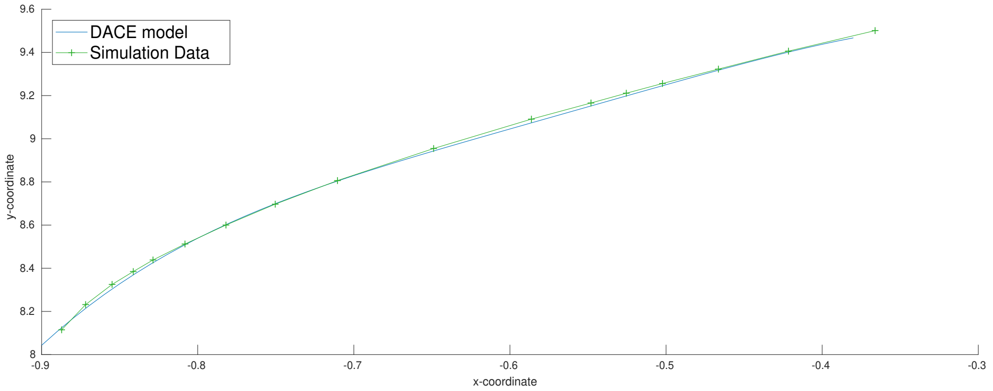


Figure 2: Comparison of involute from surrogate model and simulation

The number of points n in Eq. 7 is for the simulations about 16. Each parameter combination $\mathbf{p}$ has its own involute and thus its own RMSE value. The mean value is then calculated from these RMSE values to determine the global error of the surrogate model over all parameter combinations. Fig. 2 shows one configuration of an involute from the surrogate model and the associated simulation. An evaluation of the RMSE results in a value of 0.004, which is satisfactory. Before we discuss the results of the measurements in comparison to the surrogate model, the next chapter will show how measurement data originate.

3 Measuring Data

Every real measurement is associated with a positive, non-zero measurement error. That means that any measurand can only be determined up to a certain limitation regarding accuracy and precision. Measurement errors are categorized into systematic and random error contributions, which are described as *”component of measurement error that in replicate measurements remains constant or varies in a predicable manner”* and *”component of measurement error that in replicate measurements varies in an unpredictable manner”*, respectively [13].

The measurement uncertainty of a measurement can be calculated by the determination of those mentioned contributions. The application of different measurement methods and measurement objects in combination with numerous environmental influencing factors usually have an effect on the achieved measurement uncertainties. Usually, the measurement uncertainty is determined and associated with a certain standard geometry element (e.g. plane, cylinder). At the Institute of Manufacturing Metrology, the instrument ”single point uncertainty” was introduced and used to locally determine the measurement uncertainty with regard to a geometrically finely resolved reference measurement [14, 15]. The uncertainty contributes are obtained by statistical evaluation of the local distances of repeated measurements to numerous sampling points on the reference geometry.

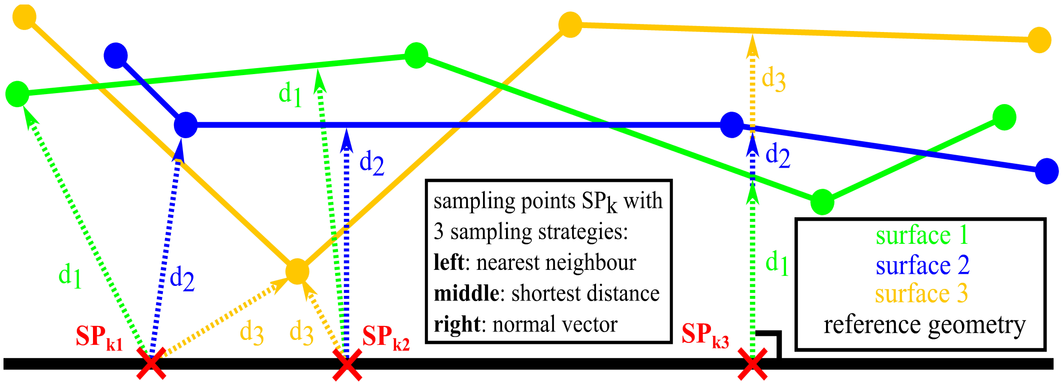


Figure 3: Different sampling strategies to determine the single point uncertainty from repeated measurements [15]

Different sampling strategies are possible to obtain the distances d_n, this contribution uses the ”shortest distance” sampling method, see Fig. 3. The mean intersection distances for each sampling point SP_k represent the systematic measurement error, while the random measurement error is represented by the standard deviation of the intersection distances. If no reference (*”quantity value used as a basis for comparison with values of quantities of the same kind”* [13]) geometry is available, the precision of a measurement can be estimated nonetheless by using the nominal geometry (e.g. CAD) instead. That means that the calculated mean intersection distances then represent the combination of the systematic measurement error and the work piece deviations. The ”single point uncertainty” framework allows the determination of the components of the measurement uncertainty with respect to numerous single points on a geometry, which is well suited for numerous application scenarios [16], including the supply of parameterized descriptions of the local measurement uncertainty [17].

In order to achieve the random measurement error several repeated measurements (n = 20) of a very precisely manufactured steel gear wheel were performed using the tactile coordinate measurement machine (CMM) ”Zeiss UPMC 1200 CARAT S-ACC” in scanning mode in combination with a rotatory stage. The measurements took place in a controlled environment with a constant air temperature of $20\ °C \pm 0.2\ K$ and relative air humidity of $45\ \% \pm 10\ \%$. The operating software ”Zeiss GEAR PRO” was utilized to define and evaluate the measurement task. Because of the repeated measurement of the same measurement object under constant conditions, the scatter of the measurement system traversing the complete measurement chain can be observed in the measurement results.

At the beginning of the measurement, the CMM is defining the work piece coordinate system by the determination of the rotation axis of the gear wheel in combination with a centring operation at a single tooth root, in order to resolve the rotation symmetry. After that, two perpendicular line scans are performed for each tooth flank. The line scans representing a complete gear wheel measurement (2 line scans for each gear wheel flank of each tooth) were geometrically registered against the nominal geometry of the gear wheel (CAD) in order to obtain a stable and convenient coordinate system for the subsequent measurement data evaluations. This step was repeated for each of the repeated measurements (n = 20). For each repeated line scan, the mean starting contour (tactile measurement coordiantes are given in combination with the associated machine probing vector and the distance from the CAD starting contour to the measurement coordiante in the direction of that associated probing vector) was calculated using a least squares procedure, from which the measurements were then sampled. Because of the fact, that the work piece was manufactured very accurately and no superior measurement device was available to determine a reference measurement for the CMM system, the subsequent uncertainty evaluations only consider the random measurement errors, thus the observed ”systematic measurement errors” are regarded as work piece deviations.

4 Measuring Data included in DACE

A comparison of the data determined in section 3 from the measurements with the simulations shows significantly greater deviation than the comparison of surrogate models and simulations in section 2.3, which can be seen exemplary in Fig. 4.

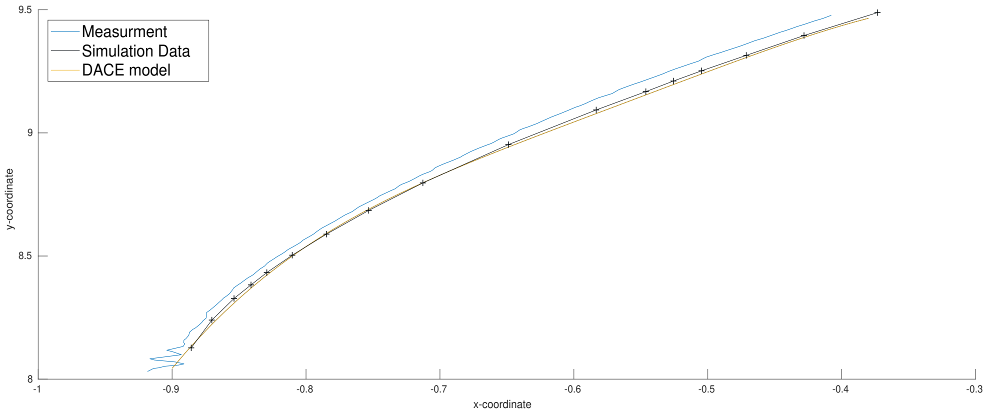


Figure 4: Comparison of involute from surrogate model, simulation and measurement

At first, the evaluation of single gear wheel measurements is discussed without taking the measurement precision (random measurement error) into account. The RMSE between the surrogate model and the simulation is, as we already know, about 0.004, while the RMSE between measurement and simulation is in the order of 0.029. With regard to Fig. 4 it is obvious, that the deviation between measurement and simulation is systematic and not a stochastic error, which leads to the conclusion, that the simulation model is not perfectly fitted. One possibility for optimization would be to improve the simulation, but it would be very time-consuming to further optimize the finite element model. Two alternatives, discussed in the following, are either to integrate the measurement results into the simulation based surrogate model or to construct a separate surrogate model based on measurement data.

4.1 Concept of constant error

The fastest and therefore cheapest approach is extending the DACE model. Eq. 5 is supplemented by a further factor Δ_{Measure}. This factor contains information about the absolute distance of the result variables between surrogate model and measurement values. Assuming that the simulation correctly reproduces the tendency of the parameter variations and thus the surrogate model deviates at each parameter constellation by approximately the same amount, it is sufficient to determine a constant value for Δ_{Measure}. With

$$\Delta_{\text{Measure}} = \mathbf{F}_{\text{Measure}}(\mathbf{p_m}) - \sum_{i=1}^{n_c} \beta_i r_i(\mathbf{p_m}) - \mathbf{Z}(\mathbf{p_m}) \tag{8}$$

the constant error value is calculated, where $\mathbf{F}_{\text{Measure}}(\mathbf{p_m})$ is the measurement result for parameter combination $\mathbf{p_m}$. It is already sufficient to carry out a single measurement on any parameter constellation. The evaluation in Tab. 1 for different constellations shows that the parameter combinations hardly plays a role. However, a parameter constellation that is as central as possible in the parameter space should be preferred, since possible effects at the edges are avoided.

-	const.1	const. 2	const. 3	const.4	const. 5
RMSE	0.0078	0.0079	0.0076	0.0076	0.0078
Distance to space center (in % of the parameter space)	52	36	29	23	2

Table 1: RMSE for different constellations including Δ_{Measure}

A disadvantage of the method is a risk to integrate the error of a single measurement into the whole model. For this reason, this risk can be greatly minimized by forming an average value.

4.2 Concept of superposed surrogate models

If measurements are available or possible, the question arises whether a surrogate model based exclusively on the measurement data makes more sense. However, it has to be considered that measurements are very expensive and they should be avoided as far as possible. Therefore, in the following, a hybrid method is presented, which uses the simulation based surrogate model from section 2 and generates another surrogate model on the measurement data. For the measurement based surrogate model we assume, that the number of sampling points is significantly smaller than the number of sampling points of simulation based model, $|\mathbf{P}_{\text{Measure}}| << |\mathbf{P}_{\text{Sim}}|$. Since the simulation based model for the cold forging example has only nine sampling points, the measurement based model is here limited to only three sampling points. For both surrogate models Eq. 5 is used.

The linear combination of the models

$$\hat{\mathbf{F}}_{\text{hybrid}}(\mathbf{p}) = b_1\hat{\mathbf{F}}_{\text{sim}}(\mathbf{p}) + b_2\hat{\mathbf{F}}_{\text{Measure}}(\mathbf{p}), \tag{9}$$

defines a hybrid method. By the factors b_1 and b_2 with $b_1 + b_2 = 1$ the influence of the respective surrogate model can be adjusted depending on the quality of the simulation and the measurements. In case of a lack of knowledge there should be a 50/50 ratio.

With the first column in Tab. 2 it becomes clear that the measurement based model using only three sampling points already has a better significance than the simulation based model using nine sampling points. Nevertheless, the hybrid model shows an optimum between $\frac{1}{10} < b_1 < \frac{1}{3}$, where the influence of the simulation based surrogate model is nearly 25%. A disadvantage of this method comes up with consideration of the measurement uncertainty in section 4.4.

b_1	0	$\frac{1}{10}$	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{9}{10}$
b_2	1	$\frac{9}{10}$	$\frac{3}{4}$	$\frac{2}{3}$	$\frac{1}{2}$	$\frac{1}{10}$
RMSE	0.0064	0.0058	0.0057	0.0060	0.0073	0.0122

Table 2: RMSE for different constellations including hybrid model

4.3 Comparison of the different methods

Since it was shown in the last section that the accuracy of the measurement based surrogate model is higher than the accuracy of a simulation based model, a surrogate model based only on the measurements is to be generated as a reference, which has as many sampling points as the simulation based model. It must be taken into account that the measurement data cover only a part of the parameter space and therefore no full factorial design is possible. Nine constellations are selected so that the distance between the evaluation points is as large as possible. The RMSE for this surrogate model is 0.0078 and thus also in the order of magnitude of the concept with constant error Δ_{Measure}. Fig. 5 shows the involutes of the five different variants for one parameter constellation.

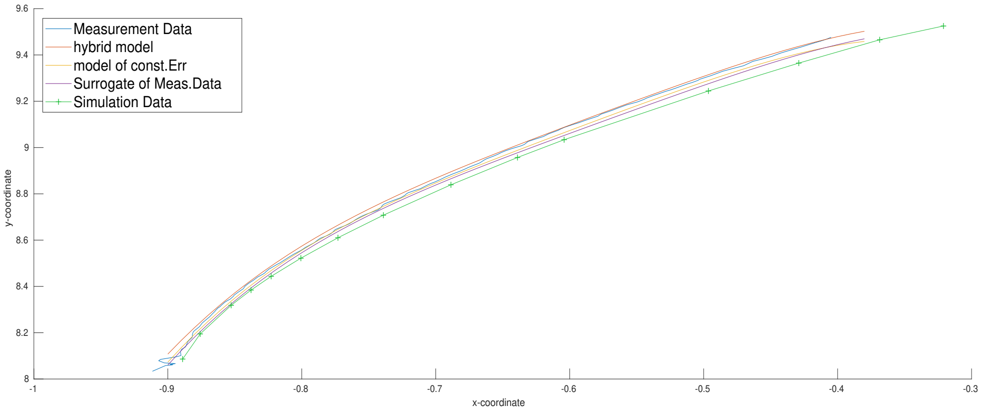


Figure 5: Comparison of all variants

This corresponds approximately to the results of the RMSE evaluation. It is quite surprising that the measurement based surrogate model is not able to convert the much larger information content into accuracy. This circumstance is due to the measurement uncertainty and perhaps the location of the sampling points, which were fixed due to experimental restrictions.

4.4 Integration of measurement uncertainty

As already mentioned in section 3, the locally resolved estimation of the measurement precision results in a distribution (random measurement error), whose standard deviation must be taken into account. This leads to a range of measurement results. Thus, if a measurement is performed only once, the result is subject to uncertainties. A consideration of the measurement uncertainty can therefore be generated by using fuzzy numbers. The fuzzy numbers form intervals, but their values are not necessarily a complete part of the interval. For each value, a membership function is assumed whose values are between zero and one. A membership function of one means that the value is in the interval, while a membership function of zero means that the value is definitely not in the interval. In contrast to classical set theory, where the values only have a membership function of zero or one, values in the fuzzy interval can also have a value in between. Thus, all kinds of uncertainties can be described with fuzzy quantities. Fuzzy arithmetic contains many characteristic features, which are shown e.g. in [18, 19]. Fuzzy numbers can be discretized with α-cuts, where classical intervals are formed with a fixed membership function value.

If the measurement uncertainty is applied to the surrogate model by means of such α-cuts, bands, whose boundaries according to the tolerance for risk, are formed. For the width of the bands in Fig. 6 applies $w(\mu = 0) = 3\sigma$. The other bands for $\mu > 0$ are automatically generated based on the selected fuzzy number. In Fig. 6 a truncated gauss fuzzy number is used. The crisp value $(\mu = 1)$ is the involute of the surrogate model with the concept of constant error.

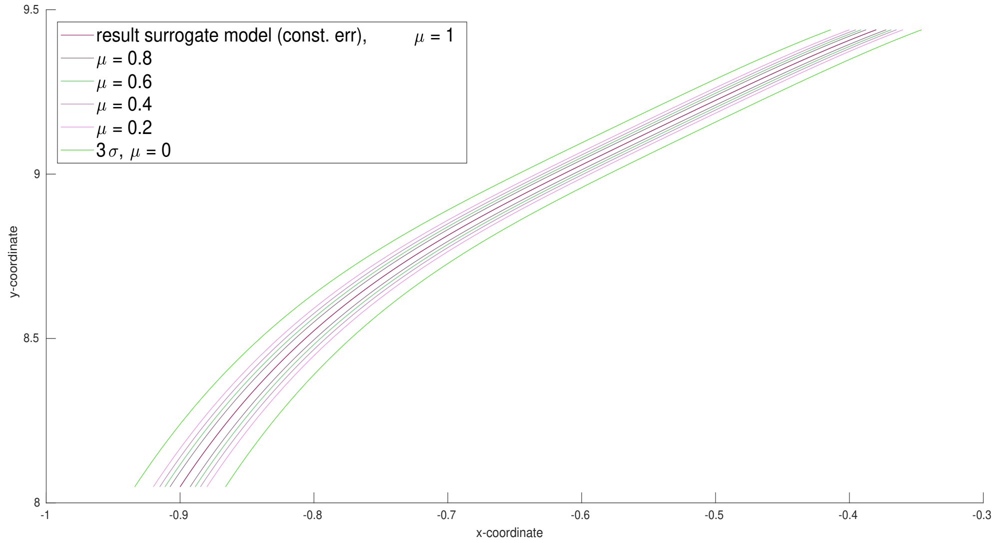


Figure 6: Measurement uncertainty implemented with fuzzy numbers

For the concept of constant error, the result is finally

$$\tilde{\hat{\mathbf{F}}}_{\Delta\,\text{err}}(\mathbf{p}) = \hat{\mathbf{F}}_{\text{Sim}}(\mathbf{p}) + \tilde{\Delta}_{Measure}. \tag{10}$$

The uncertainty in the result is only due to the measurement uncertainty and can therefore be handled very easily. This is different with a uncertainty analysis of the hybrid model. Here

$$\tilde{\hat{\mathbf{F}}}_{\text{hybrid}}(\mathbf{p}) = b_1\hat{\mathbf{F}}_{\text{Sim}}(\mathbf{p}) + b_2\tilde{\hat{\mathbf{F}}}_{\text{Measure}}(\mathbf{p}) \tag{11}$$

shows, that the uncertainty is integrated in the surrogate model itself. A crisp value at $\mu = 1$ does not exist, instead it must already be assumed, that the involute for $\mu = 1$ is uncertain. This cannot be verified exactly.

5 Conclusions

The deviations between simulation and surrogate model, which is based exclusively on these simulations, are significantly smaller than the deviations between simulation and real measurements. An optimization of the simulation based surrogate model has no significant influence and therefore remains unconsidered.

Forming a new surrogate model based exclusively on the measurement data improves the accuracy, but the effort is significantly higher than one of the presented methods in section 4. However, if the simulation is good enough, only a few measurements can be sufficient, since the simulation can correctly map the trend. It is shown that the accuracy does not increase with the number of measurements, since the measurement error only consists of a random contribution, because the systematic contribution is regarded as zero. This uncertainty is reflected in the result functions and is taken into account by means of a band around the surrogate model results. For that case the concept of constant error is easier to handle in contrast to the hybrid concept.

A classification of such surrogate models in tolerance management is a special challenge. In addition to the measurement uncertainty, the tolerance has to be considered. This makes it even more difficult to distinguish between good and bad parts. An estimation under almost 100% coverage of the measurement uncertainty and the tolerance results in a narrow band and can lead to many rejects or tight tolerances. It is at the discretion of the designer to choose suitable tolerances .

The methods presented here make it possible to approximate a process by means of simulation and measurements and to take the measurement uncertainty into account. It was shown, that it is possible to generate a ”good” surrogate model with simulation based models and few measurement data.

Acknowledgments

Gratitude is owed to the German Research Foundation (DFG) for funding the project FOR 2271 ”process-oriented tolerance management based on virtual computer-aided engineering tolls” (project No. 260682773) TP 02 and TP 03 under Grant No. WI 1181/8-1 and HA 5915/9-1. Equally many thanks go to TP 04 ME 2043/55-1 for providing the FE-simulations.

REFERENCES

[1] K. C. Maddux and S. C. Jain, Cae for the manufacturing engineer: The role of process simulation in concurrent engineering, *Advanced Manufacturing Processes, Vol 1*, **3-4**, 365–392, 1986.

[2] J. C. Helton and D. E. Burmaster, Guest editorial: treatment of aleatory and epistemic uncertainty in performance assessments for complex systems, *Reliability Engineering & System Safety, Vol.54*, **2**, 91–94, 1996.

[3] A. Rohrmoßer, B. Heling, B. Schleich, C. Kiener, H. Hagenah, S. Warzack, and M. Merklein, A methodology for the application of virtual evaluation methods within the design process of cold forged steel pinions, *Proceedings of the 22th International Conference on Engineering Design (ICED19), accepted*, 2019.

[4] R. Lorenz, H. Hagenah and M. Merklein, Experimental evaluation of cold forging lubricants using double cup extrusion tests, *Materials Science Forum*, **918**, 65–70.

[5] K. Lange, M. Kammerer, K. Pöhlandt and J. Schöck, *Fließpressen. Wirtschaftliche Fertigung metallischer Präzisionswerkstücke*, 2008.

[6] K. Vollrath and Industrieverband Massivumformung, Simulation of forging processes, *Hagen: Industrieverband Massivumformung*, 2013.

[7] ISO 1328-1:2013, *Cylindrical gears - ISO system of flank tolerance classification - Part 1: Definitions and allowable values of deviations relevant to flanks of gear teeth.*

[8] A. Forrester, A. Sóbester and A. Keane, *Engineering Design via Surrogate Modelling*, 2008.

[9] D. G. Krige, A Statistical Approaches to Some Basic Mine Valuation Problems on the Witwatersrand, *Journal of the Chemical, Metallurgical and Mining Society of South Africa, Vol.52*, 119–132, 1951.

[10] J. Sacks, W. J. Welch, T.J. Mitchell and H.P.Wynn, Design and Analysis of Computer Experiments *Statistical Science, Vol.4*, 409–423, 1989.

[11] S. N. Lophaven, H.B. Nielsen and J. Søndergaard, DACE - A MATLAB Kriging Toolbox, *Technical Report Informatics and Mathematical Modelling, IMM-REP-2002-12*, 2002.

[12] S. N. Lophaven, H.B. Nielsen and J. Søndergaard, Aspects of the matlab toolbox DACE, *Technical Report Informatics and Mathematical Modelling, IMM-REP-2002-13*, 2002.

[13] International vocabulary of metrology - Basic and general concepts and associated terms (VIM); German-English version ISO/IEC Guide 99:2007, Corrected version 2012, ISBN 978-3-410-22472-3

[14] M. Fleßner, A. M. Müller, D. Götz, E. Helmecke, and T. Hausotte, Assessment of the single point uncertainty of dimensional CT measurements, *iCT 2016, Wels, Austria*, 2016.

[15] A. M. Müller und T. Hausotte, Comparison of different measures for the single point uncertainty in industrial X-ray computed tomography, *iCT2019, Padova, Italy*, 2019.

[16] A. M. Müller und T. Hausotte, Utilization of single point uncertainties for geometry element regression analysis in dimensional X-ray computed tomography, *iCT 2019, Padova, Italy*, 2019.

[17] A. M. Müller, T. Oberleiter, K. Willner und T. Hausotte, ”Implementation of parameterized work piece deviations and measurement uncertainties into performant meta-models for an improved tolerance specification, Proceedings of the International Conference on Engineering Design, ICED, *Proceedings of the International Conference on Engineering Design, ICED, accepted*, 2019.

[18] M. Hanss, *Applied Fuzzy Arithmetic - An Indroduction with Engineering Applications*, 2010

[19] D. Dubois, H. Prade, Fuzzy Sets and Systems: Theory and Applications *Mathematics in science and engineering, Vol.144*, 1980

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

MACHINE LEARNING FOR CLOSURE MODELS IN MULTIPHASE FLOW APPLICATIONS

Jurriaan Buist[1,2], Benjamin Sanderse[1], Yous van Halder[1],
Barry Koren[2], and GertJan van Heijst[2]

[1]Centrum Wiskunde & Informatica
Science Park 123, 1098XG Amsterdam
j.f.h.buist@cwi.nl, b.sanderse@cwi.nl, y.van.halder@cwi.nl

[2] Eindhoven University of Technology
PO Box 513, 5600 MB Eindhoven
b.koren@tue.nl, g.j.f.v.heijst@tue.nl

Keywords: Machine Learning, Closure Terms, Multiphase Flow, Two-Fluid Model

Abstract. *Multiphase flows are described by the multiphase Navier-Stokes equations. Numerically solving these equations is computationally expensive, and performing many simulations for the purpose of design, optimization and uncertainty quantification is often prohibitively expensive. A simplified model, the so-called two-fluid model, can be derived from a spatial averaging process. The averaging process introduces a closure problem, which is represented by unknown friction terms in the two-fluid model. Correctly modeling these friction terms is a long-standing problem in two-fluid model development.*

In this work we take a new approach, and learn the closure terms in the two-fluid model from a set of unsteady high-fidelity simulations conducted with the open source code Gerris. These form the training data for a neural network. The neural network provides a functional relation between the two-fluid model's resolved quantities and the closure terms, which are added as source terms to the two-fluid model. With the addition of the locally defined interfacial slope as an input to the closure terms, the trained two-fluid model reproduces the dynamic behavior of high fidelity simulations better than the two-fluid model using a conventional set of closure terms.

1 INTRODUCTION

The simulation of multiphase flow of gas and liquid in a pipeline is a problem of interest in the oil and gas industry. The two fluids can have complex interactions leading to different flow regimes, such as smoothly stratified flow, wavy stratified flow, and slug flow. Predicting the transition from stratified flow to slug flow in dynamic simulations is a difficult problem [21], for which we consider different computational models. We restrict ourselves in this paper to incompressible 2D channel flow, as a simplified representation of 3D circular pipe flow.

A general model which can describe the flow regimes mentioned above is formed by the well-known Navier-Stokes equations. These can be solved numerically, using for example a volume-of-fluid (VOF) method [19] for the treatment of the interface. However, when many model evaluations are needed, such as in uncertainty quantification, solving the full Navier-Stokes equations is too computationally expensive.

We therefore consider a simplified model which is computationally less expensive, the so-called two-fluid model [3]. The 1D two-fluid model is obtained by averaging the Navier-Stokes equations for each fluid, over the respective cross-sections. This spatial averaging process introduces a closure problem; the shear stresses in the flow become unknowns with a priori no direct relation to the averaged quantities present in the two-fluid model. Relations between the averaged quantities and the stresses need to be postulated; these relations are called 'closure terms'.

Conventionally, these closure terms for the two-fluid model are obtained from correlations with experimental data for steady state pipe flow. A pressure difference is applied to a section of the pipe and the resulting volumetric fluxes and liquid holdup (fraction of the total pipe cross-section occupied by the liquid) are measured. These are related to the stresses via the steady state balances for both fluids and via assumptions on the relations between the different stresses, to form the closure terms. On this principle, for example, the widely used Taitel and Dukler [40] closure terms are based.

Alizadehdakhel et al. [1] and Osgouei et al. [32] used physical experimental data to train neural networks to predict pressure drops in two-phase pipe flow. They related the superficial velocities to the spatially and temporally averaged pressure drop, like in the conventional approach, but used a neural network to construct the relation.

For bubbly flow in a vertical channel, Ma et al. [26, 27], introduced a more general approach. They conducted 3D unsteady Navier-Stokes DNS (with front tracking), the results of which can be related to the averaged quantities present in their low-fidelity 1D model, at any point along the 1D model's spatial axis and at any point in time. A neural network was employed to learn the relation between the two. They report satisfactory results, and emphasize the general applicability of their approach: no prior knowledge is needed on the relation between known quantities and the quantities requiring closure. Ma et al. refer to earlier work by Lu et al. [24, 25], who trained a neural network with data from micro-scale DNS simulations of a gas-solid mixture under influence of a shock, to provide closure relations for the particle-particle and gas-particle interactions, for use in coarse macro-scale simulations. Besides these references, in multiphase flow, the literature on machine learning for closure terms is sparse.

However, in the field of turbulence closure modeling, neural networks have already proved their worth, when applied to specific cases. Sargini et al. [38] used a neural network to create a subgrid scale (SGS) model for a Large Eddy Simulation (LES), which reproduces the dynamics of LES using an expensive SGS model (Bardina's scale similar (BFR) SGS model), at a lower computational cost. A similar approach was taken by Tracey et al. [42] for air flow in a data center. Gamahara and Hattori [16] recently used DNS directly to obtain a functional relation for the SGS tensor which shows performance close to that of a Smagorinsky SGS model. Ling et al. [23] learned RANS stress tensors similarly, choosing the inputs and neural network structure such that Galilean invariance is incorporated in the expressions directly.

Motivated by the success of machine learning in the field of turbulence closure modeling, we continue the application of machine learning to multiphase flow, taking inspiration from Ma et al. [26, 27]. We make two new contributions:

- We extend the methodology to more generic neural networks.
- We study a different physical situation (stratified flow versus bubbly flow), with a different low-fidelity model, and different unclosed terms.

Compared to the conventional literature on closure terms for the 1D two-fluid model, the novelties of this work are:

- We base closure terms on the results of fully resolved unsteady 2D Navier-Stokes simulations (for channel flow), which we refer to as our high-fidelity simulations.
- We employ an artificial neural network to find the relation between the two.
- The preceding two points make it straightforward to add non-local, non-instantaneous input variables to the closure relations; in this work we have added the streamwise derivative of the interface height.

The differences between our approach and the conventional approach mean that:

- Closure terms can be constructed for specific cases (specific duct geometries or flow regimes), as long as accurate high-fidelity simulations are available.
- Unsteady behavior may be reproduced more accurately by the low-fidelity model.

With our approach, it is our aim to use high-fidelity simulations to improve the accuracy of low-fidelity simulations, with the promise to reach the accuracy of the high-fidelity model at the cost of the low-fidelity model.

The structure of the paper is as follows. The physics and numerics of the high- and low-fidelity models are discussed in section 2. This leads us to an explanation of the required closure terms, and the fundamental limitations imposed by the model averaging process, which are not fixable by improving closure terms of the considered form.

In section 3 we tune the neural network, and show that closure terms based on steady state flow are unsatisfactory for the case of wavy unsteady flow. We then describe the training of the tuned neural network on wavy unsteady flow data. Finally, section 4 presents the results of applying the trained neural networks as closure terms in the low-fidelity simulations. Here the agreement between the high-fidelity model and the enhanced low-fidelity model is evaluated.

2 HIGH- AND LOW-FIDELITY MODEL DESCRIPTION

Our approach is shown schematically in Figure 1. We have a 2D high-fidelity model for channel flow, with horizontal and vertical velocity components u and w, being functions of the coordinates s and h. The low-fidelity model is 1D and as such only knows velocities u_L and u_G, which are averaged over the portions of the channel containing liquid and gas respectively, so that they are only functions of s (as is the interface height h_{int}). From the 2D high-fidelity field results we calculate these averaged quantities and their corresponding stresses. These represent inputs and desired outputs to a neural network, respectively, between which the neural network is given the task to find a relation. The resulting functions can be fed as closure terms to the low-fidelity model.

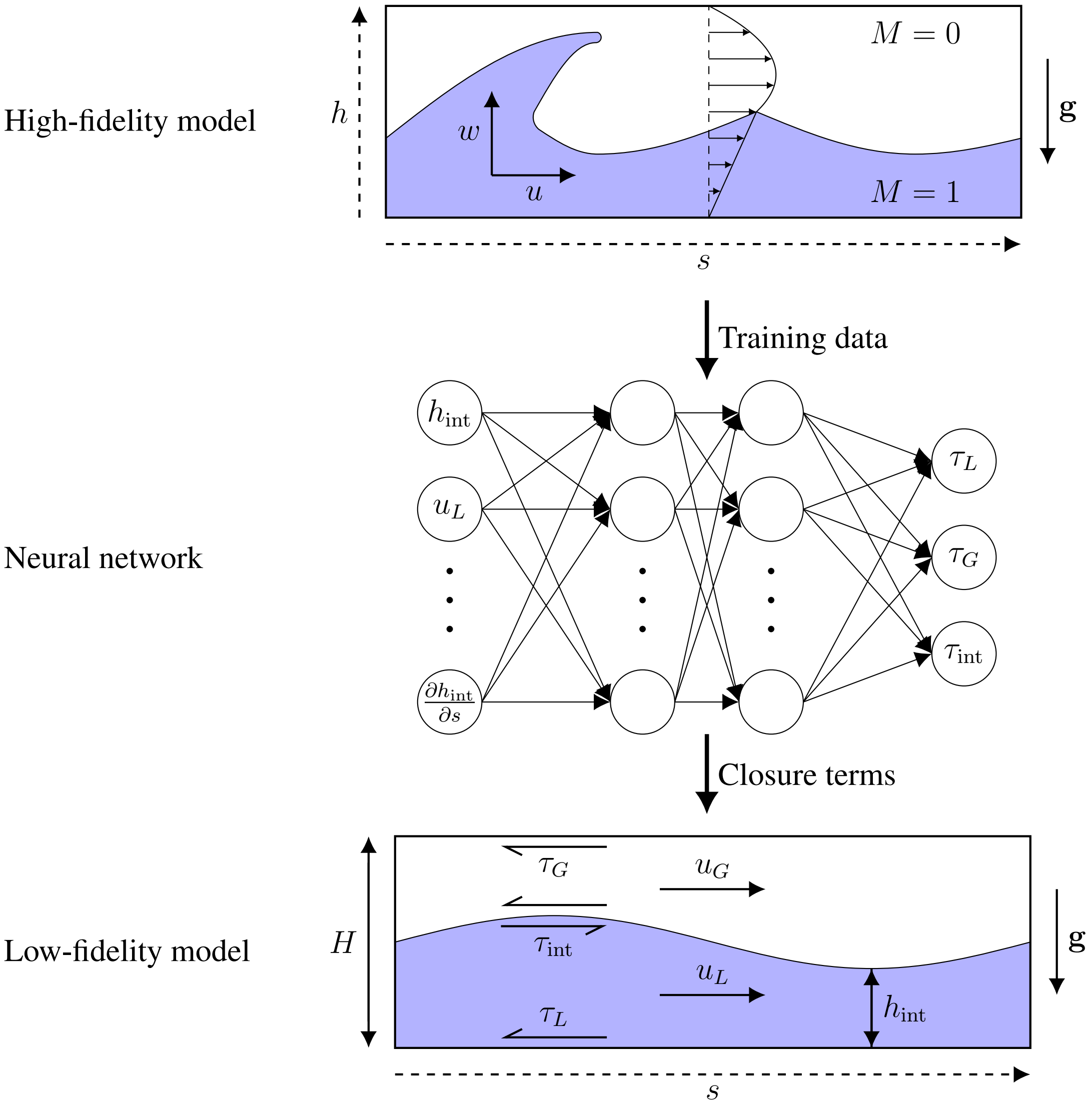


Figure 1: An outline of the approach for learning closure terms from high-fidelity simulations.

2.1 High-fidelity model

The high-fidelity model that we use to generate the training data is the open source code Gerris [33, 34]. It is based on the one-fluid formulation for multiphase flow. This entails the solution of the Navier-Stokes equations for incompressible flow:

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{1}{\rho}\left(-\nabla p + \nabla \cdot \left[\mu \nabla \mathbf{u} + \mu (\nabla \mathbf{u})^{\mathrm{T}}\right]\right) + \mathbf{g}, \tag{2}$$

with velocity field $\mathbf{u} = \mathbf{u}(s,h,t)$ and pressure field $p = p(s,h,t)$ encompassing the entire domain, gravitational acceleration g, density ρ, and viscosity μ (see Figure 1).

Gerris discretizes these equations spatially with a finite volume method on a colocated grid, with central interpolation and the Van Leer generalized minmod limiter with $\theta = 2$ for the face-centered gradient calculation. We do not make use of Gerris' capability to adaptively refine the grid at different levels.

For temporal discretization Gerris uses a second order projection method [11], in which a multilevel Gauss-Seidel iterative method is used to solve the pressure Poisson equation. The velocity advection term is discretized according to the second order unsplit upwind scheme of Bell et al. [4], and for the diffusion term a Crank-Nicolson discretization is employed.

In the one-fluid approach for multiphase flow, the density ρ and viscosity μ are functions of the spatial coordinates, via a marker function $M = M(s,h,t)$:

$$\rho = \rho(M), \quad \mu = \mu(M).$$

This marker function M, typically 1 in the liquid and 0 in the gas, is advected by the velocity field. We make the *assumption of sharp interfaces* [43, p. 22] and disregard phase transition, so that the advection of the marker function can be described by

$$\frac{\mathrm{D}M}{\mathrm{D}t} = \frac{\partial M}{\partial t} + \mathbf{u} \cdot \boldsymbol{\nabla} M = 0. \tag{3}$$

Gerris advects the marker function numerically using the volume-of-fluid (VOF) method. In the VOF method [19], the marker function is averaged over the grid cells to define the color function

$$C_i = \frac{1}{V_i} \int_{V_i} M \, \mathrm{d}V. \tag{4}$$

The color function is a function which gives the volume fraction of the reference fluid in a grid cell. The material properties in grid cells i can then be expressed as functions of this color function. We use the expressions

$$\rho_i = C_i \rho_1 + (1 - C_i)\rho_0, \tag{5}$$

$$\mu_i = \left(\frac{C_i}{\mu_1} + \frac{1 - C_i}{\mu_0} \right)^{-1}. \tag{6}$$

with ρ_1 and μ_1 the density and viscosity of the fluid indicated by $M = 1$ and ρ_0 and μ_0 the fluid indicated by $M = 0$. For the viscosity we do not use an arithmetic mean but rather the harmonic mean [14], which improves the accuracy of the velocities and stresses at a flat, horizontal interface.

Prior to the actual advection step, the interface is reconstructed from the color function using the PLIC method [47]. The color function is then advected geometrically by the velocity field.

We use most of the standard Gerris settings, except that we lower the tolerance of the projection steps from $1 \cdot 10^{-3}$ to $1 \cdot 10^{-6}$. After a convergence study, the grid spacing $\Delta s = \Delta h$ is set to $H/64$, and the time step is set so that the maximum value of

$$\mathtt{CFL} = \frac{|\mathbf{u}|\Delta t}{\Delta s} \tag{7}$$

anywhere in the simulation is 0.8. However, there is an additional constraint that in mixed VOF cells the maximum value should be 0.5. We do not filter the color function (i.e. averaging over multiple cells), to keep the interface relatively sharp.

We choose the one-fluid formulation for multiphase flow with the VOF interface advection method for its conservative properties, its simplicity, and its similarity to our low-fidelity model. Alternative interface advection methods for the one-fluid formulation of multiphase flow include the front tracking [46] and level-set methods [39], but these are not naturally mass conservative.

2.2 Low-fidelity model

Our low-fidelity model is known as the 1D two-fluid model. It is obtained by considering control volumes in a channel, separate for liquid and gas, as pictured in Figure 2. The limit $\delta s \to 0$ is taken, while the control volumes fill the full channel height. Since we do not consider phase change, the velocities are continuous at the interface. The stresses tangential to the interface, the shear stresses, must be continuous, and since we assume hydrostatic balance (without surface tension) the pressure should be continuous along the vertical direction, as well as the stresses along the vertical direction.

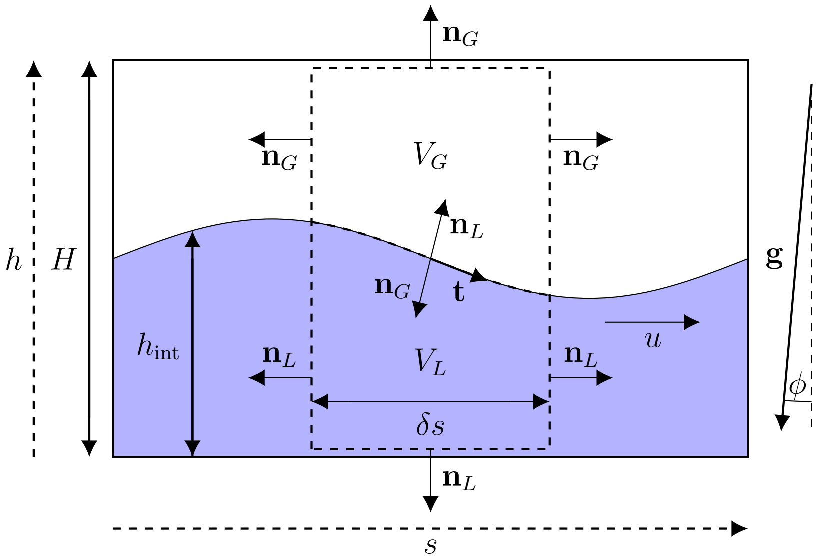


Figure 2: Two small ($\delta s \ll H$) control volumes for two-phase pipe flow. At the top and bottom the control volume is bounded by impenetrable no-slip boundaries. The interface separates the two control volumes.

We obtain one equation for mass balance and one for momentum balance for each fluid. In channel flow these take the following form:

$$\frac{\partial}{\partial t}\left(\rho_L h_{\text{int}}\right) + \frac{\partial}{\partial s}\left(\rho_L u_L h_{\text{int}}\right) = 0, \tag{8a}$$

$$\frac{\partial}{\partial t}\left(\rho_G (H - h_{\text{int}})\right) + \frac{\partial}{\partial s}\left(\rho_G u_G (H - h_{\text{int}})\right) = 0, \tag{8b}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho_L u_L h_{\text{int}}\right) + \frac{\partial}{\partial s}\left(\rho_L u_L^2 h_{\text{int}}\right) = &- \frac{\partial p_{\text{int}}}{\partial s} h_{\text{int}} + L G_L + F_L \\ &- \rho_L h_{\text{int}} g \sin(\phi),\end{aligned} \tag{8c}$$

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho_G u_G (H - h_{\text{int}})\right) + \frac{\partial}{\partial s}\left(\rho_G u_G^2 (H - h_{\text{int}})\right) = &- \frac{\partial p_{\text{int}}}{\partial s}(H - h_{\text{int}}) + L G_G + F_G \\ &- \rho_G (H - h_{\text{int}}) g \sin(\phi),\end{aligned} \tag{8d}$$

with u_L and u_G the averaged velocities of the liquid and gas respectively, ρ_L and ρ_G likewise for the densities, h_{int} the interface height, p_{int} the interfacial pressure, and ϕ the channel inclination. Here the stresses are bundled into closure terms

$$F_L = \tau_L - \tau_{\text{int}}, \quad F_G = \tau_G + \tau_{\text{int}}, \tag{9}$$

and the level gradient terms represent

$$LG_L = -\frac{\partial}{\partial s}\left[\frac{1}{2}\rho_L g \cos(\phi)\, h_{\text{int}}^2\right], \quad LG_G = \frac{\partial}{\partial s}\left[\frac{1}{2}\rho_G g \cos(\phi)\, (H - h_{\text{int}})^2\right]. \tag{10}$$

The equations are of the same form as those for pipe flow in circular cross-sections, but with different relations between the cross-sections, perimeters, and interface height (see e.g. [36]).

In this research the 'Rosa' code developed by Sanderse et al. [35, 36, 37] is employed for solving the incompressible form of (8).

The code discretizes the equations using a finite volume method on a staggered grid. This allows for a strong and straightforward coupling between pressure and velocity. Interpolation is needed for the convective scheme: here we employ a central interpolation, which ensures second order spatial accuracy.

After the system is discretized spatially, the time stepping is considered. We use the constraint-consistent time integration framework for the incompressible two-fluid model presented in [36], with the three-stage, third order strong-stability preserving Runge-Kutta method referenced in [37], which follows Gottlieb et al. [17].

2.3 Closure terms

The liquid wall stress τ_L, gas wall stress τ_G, and interfacial stress τ_{int}, which appear in (9), represent the stresses acting in the streamwise direction:

$$\tau = (\boldsymbol{\tau} \cdot \mathbf{n}) \cdot \hat{\mathrm{s}}, \tag{11}$$

with τ the stress tensor and $\hat{\mathrm{s}}$ the unit vector along the s-axis. Accounting for the no-slip boundary conditions, assuming hydrostatic balance and horizontal length scales far larger than the vertical length scale, they are related to the velocity profile via

$$\tau_L = -\mu_L \frac{\partial u}{\partial h}\bigg|_{h\downarrow 0}, \quad \tau_G = \mu_G \frac{\partial u}{\partial h}\bigg|_{h\uparrow H}, \quad \tau_{\text{int}} = -\mu_G \frac{\partial u}{\partial h}\bigg|_{h\downarrow h_{\text{int}}} = \mu_L \frac{\partial u}{\partial h}\bigg|_{h\uparrow h_{\text{int}}}, \tag{12}$$

in which $x \uparrow y$ and $x \downarrow y$ are limits from below and from above respectively (see [10] for a more detailed discussion).

These stresses are a priori unknown in the 1D two-fluid model, since in this model the velocities are not resolved in the transverse direction, meaning that the stresses cannot be calculated according to (12). Conventionally, steady state experiments are employed to correlate the stresses to the averaged quantities through the steady state balance, essentially implying a streamwise and temporally averaged description of the flow. This yields relations of the form[1]

$$\tau_L, \tau_G, \tau_{\text{int}} = f(h_{\text{int}}, u_L, u_G, \rho_L, \rho_G, \mu_L, \mu_G, H), \tag{13}$$

[1] Expressions for the stresses based on the body forces as opposed to the averaged velocities do not close the steady state equations [13], and cannot generalize to unsteady flow.

in which all the variables on the right-hand side *are* known in the 1D two-fluid model.

Many experiments are needed to obtain good relations, and therefore it may be difficult to find closure terms in the literature which generalize well to the case at hand. Furthermore, when considering wavy flow, with this method of generation of closure terms only the averaged (positive) effect of waves on the interfacial friction can be taken into account; local effects are averaged out.

For the strongly simplified case of laminar, flat interface, fully developed, steady channel flow, Ullmann et al. [45] have derived analytical solutions for the stresses of the form (13). They are given by

$$\tau_L = -\frac{1}{2} f_L \rho_L u_L |u_L| F_L^*, \quad \tau_G = -\frac{1}{2} f_G \rho_G u_G |u_G| F_G^*, \quad \tau_{\text{int}} = -\frac{1}{2} f_G \rho_G \left(u_G - u_L\right) |u_G| F_{\text{int},G}^*, \tag{14}$$

with friction factors

$$f_L = \frac{3}{2}\frac{16}{\text{Re}_L}, \quad f_G = \frac{3}{2}\frac{16}{\text{Re}_G}, \tag{15}$$

depending on Reynolds numbers

$$\text{Re}_L = \frac{\rho_L |u_L| D_L}{\mu_L}, \quad \text{Re}_G = \frac{\rho_G |u_G| D_G}{\mu_G}, \tag{16}$$

based on hydraulic diameters

$$D_L = 2h_{\text{int}}, \quad D_G = 2(H - h_{\text{int}}). \tag{17}$$

F_L^* and F_G^* are the two-phase correction factors for the wall friction:

$$F_L^* = \frac{1 + \frac{1}{2}\frac{u_G}{u_L}\left[\frac{\mu_L}{\mu_G}\frac{u_L}{u_G}\frac{H-h_{\text{int}}}{h_{\text{int}}} - 1\right]}{1 + \frac{\mu_L}{\mu_G}\frac{H-h_{\text{int}}}{h_{\text{int}}}}, \quad F_G^* = \frac{1 + \frac{1}{2}\frac{u_L}{u_G}\left[\frac{\mu_G}{\mu_L}\frac{u_G}{u_L}\frac{h_{\text{int}}}{H-h_{\text{int}}} - 1\right]}{1 + \frac{\mu_G}{\mu_L}\frac{h_{\text{int}}}{H-h_{\text{int}}}}, \tag{18}$$

and $F_{\text{int},G}^*$ that for the interfacial friction:

$$F_{\text{int},G}^* = \frac{1}{1 + \frac{\mu_G}{\mu_L}\frac{h_{\text{int}}}{H-h_{\text{int}}}}. \tag{19}$$

These closure terms form a reference, with which we benchmark our solvers for the case of steady flow, and to which we compare the new neural network closure terms.

2.4 Closure term limitations

By using closure relations of the form (13), we introduce a fundamental limitation. The averaged velocities cannot be translated back uniquely to velocity profiles, the slopes of which determine the stresses; information is lost in the averaging process. The consequence of this uniqueness problem is that for collections of very different velocity profiles, with the same averaged velocities, the closure relations will predict the same stresses, while the actual stresses can in reality be very different. In most of the literature [2, 5, 12, 15, 20, 40, 44] the analysis is therefore limited to fully developed steady state flow, for which the stresses *are* uniquely related to the averaged velocities.

There is a second limitation to the degree to which the low-fidelity model can be made to reproduce results of the high-fidelity model in this framework. Closure terms of the form (13) are introduced as source terms in the low-fidelity model, and cannot be expected to resolve the entire difference in dynamics introduced by the averaging process and the associated assumptions. The difference between the dynamics of the high- and low-fidelity models, in absence of friction (equivalent to taking the homogeneous part of the equations for the two-fluid model at zero inclination), can be illustrated by performing a linear stability analysis of the two models. Results of this are shown in Figure 3, based on the parameters from Table 1.

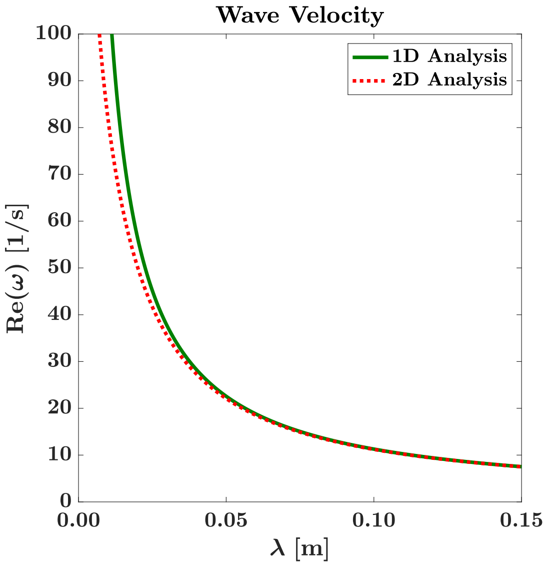


Figure 3: Dispersion relations $\omega(k)$ with $k = 2\pi/\lambda$, for the 1D two-fluid model without friction terms [22], and for inviscid 2D potential flow [29]. The parameters are given in Table 1, with the steady state solution given in Table 2.

Table 1: Test case parameters.

Parameter	Symbol	Value	Units
Background pressure gradient	$\partial p/\partial s$	-1	$\mathrm{kg\,m^{-2}\,s^{-2}}$
Liquid density	ρ_L	998	$\mathrm{kg\,m^{-3}}$
Gas density	ρ_G	1.2	$\mathrm{kg\,m^{-3}}$
Channel height	H	0.01	m
Initial interface height	h_{int}	$0.3H$	m
Liquid viscosity	μ_L	$1.002 \cdot 10^{-3}$	$\mathrm{kg\,m^{-1}\,s^{-1}}$
Gas viscosity	μ_G	$1.82 \cdot 10^{-5}$	$\mathrm{kg\,m^{-1}\,s^{-1}}$
Acceleration of gravity	g	9.81	$\mathrm{m\,s^{-2}}$
Pipe inclination	ϕ	0	degrees

With the analytical solution given by [6] and [45], the corresponding averaged velocities and stresses (for the steady state) can be calculated. These are given in Table 2.

It is observed that the dispersion relation of the 1D model only converges to the dispersion relation of the 2D model at large wavelengths. The cross-sectional averaging of the equations, and the associated assumption of hydrostatic balance, implicitly implies the long wavelength assumption [30].

Table 2: Test case steady state solution, for the parameters given in Table 1.

Parameter	Symbol	Value	Units
Averaged liquid velocity	u_L	0.00818	$\mathrm{m\,s^{-1}}$
Averaged gas velocity	u_G	0.232	$\mathrm{m\,s^{-1}}$
Liquid wall stress	τ_L	-0.00646	$\mathrm{kg\,m^{-1}\,s^{-2}}$
Gas wall stress	τ_G	-0.00354	$\mathrm{kg\,m^{-1}\,s^{-2}}$
Interfacial stress	τ_{int}	-0.00346	$\mathrm{kg\,m^{-1}\,s^{-2}}$
Liquid Reynolds number	Re_L	48.9	-
Gas Reynolds number	Re_G	214	-
Liquid Froude number	Fr_L	0.00114	-
Gas Froude number	Fr_G	0.391	-

2.5 Stress extraction

By extracting stresses from high-fidelity simulations via (12), it *is* possible to consider local and unsteady effects. We can take any position along the s-axis in the simulations and calculate the stresses, and the corresponding averaged variables u_L, u_G, h_{int}, at that point. We can calculate additional, local, quantities, which are not defined in a streamwise averaged description — the streamwise derivatives of the averaged variables — and relate these to the stresses as well. Since we can extract the averaged variables and stresses at any point in time in the unsteady simulations, the same holds for temporal derivatives. Because we use a neural network, such new inputs can easily be added to the closure relations, without prior information on the complex relation between them and the stresses.

The stresses are determined practically by fitting cubic splines to the velocity profiles of the liquid and gas separately and taking their analytical derivatives. For the final determination of the interfacial stress, the stresses at the interface as calculated from the liquid and gas profiles are averaged. For more details, see [10].

3 NEURAL NETWORKS

A neural network is used to construct a relation of the form (13), using the high-fidelity model data. For laminar steady state flow the analytical solution can be used to train the neural network instead of the high-fidelity solution (since they are equal). This simple case, for which conventional closure terms are exact, is used to tune the neural network hyper-parameters (in subsection 3.1 and subsection 3.2). Afterwards (in subsection 3.3 and subsection 3.4), the tuned neural network is applied to the more difficult case of unsteady, wavy flow.

3.1 Neural network settings

The network is a multilayer perceptron network (MLP), implemented in the MATLAB Deep Learning Toolbox [41]. It is tuned mainly by comparison of training data error and validation data error as measured by a mean squared error cost function

$$C = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y_i})^2, \tag{20}$$

where y_i is the data for a set of input variables i and $\widehat{y_i}$ is the model prediction for these inputs. The final value of the cost function is the average of (20) over the three stresses τ_L, τ_G, τ_{int}.

The result is a network with 4 hidden layers with 18 nodes each, each with a hyperbolic tangent activation function, and no regularization term in the cost function (given the size of our training data set). The network is trained using the Levenberg-Marquardt training algorithm [18], an efficient algorithm for smaller networks. A small percentage of the training data (15%) is taken apart and not used for the training; the optimization is stopped if the error on this validation data does not decrease. The training inputs and outputs are mapped to the range $[-1, 1]$ (the same translation and multiplication is later applied to unseen data).

We tested the effect of random initialization via the Nguyen-Widrow algorithm [31] and found that different random initializations yield very similar final values of the validation data error. We also verified the convergence of the training and validation errors with increasing amounts of data. These results are available in [10].

3.2 Performance of networks trained on steady state data

Training the network with steady state data is useful for the network tuning. However, Figure 4 shows that networks trained on steady state data have little predictive capacity for stresses found in wavy unsteady simulations. On the horizontal axis stress values observed in high fidelity simulations are set out, and on the vertical axis the neural network predictions for the same h_{int}, u_L, u_G, ... are given. We show the squared correlation coefficient

$$R^2 = 1 - \frac{\sum_{i=1}^{N} \left(\widehat{y}_i - y_{l,i}\right)^2}{\sum_{i=1}^{N} \left(\widehat{y}_i - (1/N)\sum_{i=1}^{N} \widehat{y}_i\right)^2}, \tag{21}$$

with a range between 0 and 1. In this definition, $\widehat{y}_i$ is the model prediction. We construct a linear fit of the model prediction $\widehat{y}_i$ as a function of the data and call it y_l. The value $y_{l,i}$ is the value of the linear fit at the data point y_i, corresponding to prediction $\widehat{y}_i$.

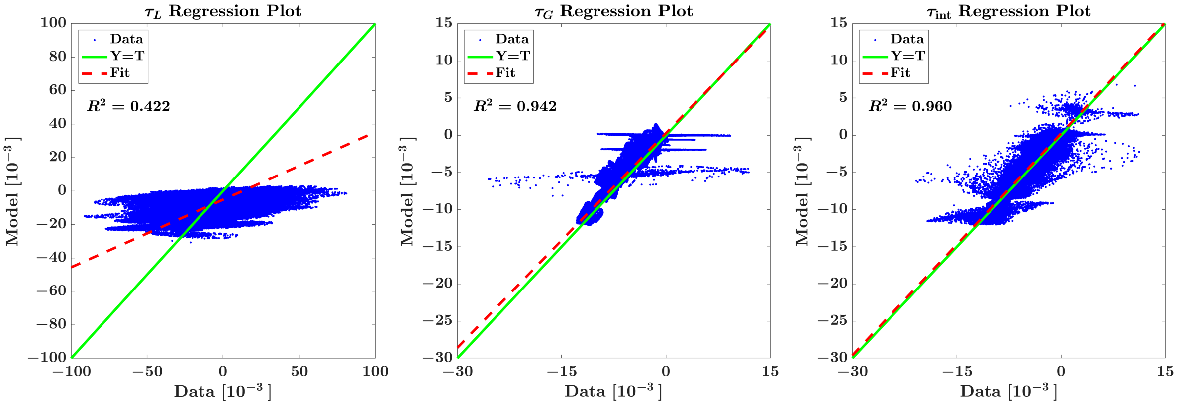


Figure 4: Regression plots for τ_L, τ_G, and τ_{int} with a neural network trained on steady state data as the model, tested on the wavy unsteady high-fidelity simulation data.

The poor performance shown in Figure 4, particularly for the strongly oscillatory liquid stress, could have been expected, as the analytical stresses with which we train the network are derived for steady state flow, while the actual flow is wavy unsteady. Similar poor results are observed when using the analytical stresses as the model directly. This motivates training on unsteady data.

3.3 Generation of wavy unsteady data

We consider 2D channel flow with periodic boundaries left and right under a constant body force in the form of a background pressure gradient $\partial p/\partial s$. No-slip boundary conditions are applied at the top and bottom walls. A sine wave perturbation with wavelength λ and amplitude $\Delta\widehat{h}_{\text{int}}$ is applied to the interface between liquid and gas.

We generate data by running the high fidelity code Gerris 60 times, with varying input parameters randomly selected from the ranges given in Table 3. The parameters are selected from these ranges using Latin Hypercube Sampling [28], ensuring a space-filling sampling, without repetition of parameter values. The material properties, channel height, and the channel inclination are kept at the values given in Table 1. This limits the required amount of (costly) simulations, while allowing practical application to an unsteady flow in a specified pipe and with specified fluids.

The wavelength of the perturbation is fixed at $\lambda = L = 0.12\,\text{m}$, where L is the length of the domain.

Table 3: The ranges of the parameters of the unsteady high fidelity simulations used as training data.

Initialization	$h_{\text{int}}\,[H]$	$\partial p/\partial s\,[\text{Pa/m}]$	$\Delta\widehat{h}_{\text{int}}\,[H]$	Number of simulations
zero wavy	$[0.05, 0.95]$	$[0, -3]$	$[0.00, 0.04]$	30
developed wavy	$[0.05, 0.95]$	$[0, -3]$	$[0.00, 0.04]$	30

The simulations are initialized from two different initial conditions:

- ‘zero wavy’: the velocities in the entire domain are zero,
- ‘developed wavy’: the velocities are initialized at their (flat interface) steady state values (determined analytically [6]),

and then run from $t = 0$ to $t = 10$ seconds. With these initial conditions and the given parameters, we get slowly traveling standing waves (see Figure 9), initially approximated by

$$\Delta h_{\text{int}}(s,t) = 2\Delta\widehat{h}_{\text{int}}\cos(ks - \delta\omega t)\cos(\omega_0 t). \tag{22}$$

These waves are formed as the superposition of two waves traveling in the opposite direction with wave velocities

$$c_1 = \frac{\omega_0 + \delta\omega}{k}, \quad c_2 = \frac{-\omega_0 + \delta\omega}{k}, \tag{23}$$

with $\delta\omega \ll \omega_0$ due to the small Froude numbers (see Table 2). Later, nonlinear and damping effects become important; in most cases the waves are largely damped out at $t = 10$.

Note that the parameter ranges in Table 3 are chosen such that the perturbations damp out in time, avoiding a transition to (near-) slug flow and problems of ill-posedness in the 1D two-fluid model (see e.g. [7]).

3.4 Neural network training

At each time step and at each grid point along s the quantities given in (13) are extracted from the Gerris simulations, yielding many data points, which may not all provide distinct information. The networks are therefore trained on small random subsets of the data.

Neural networks are trained on different portions of the data:

- ‘zero wavy net’: networks trained on data with the ‘zero wavy’ initial condition.
- ‘developed wavy net’: networks trained on data with the ‘developed wavy’ initial condition.
- ‘zero + developed wavy net’: networks trained on a combination of the data with the ‘zero wavy’ initial condition and with the ‘developed wavy’ initial condition.

Per item in the above list, we sample the data with replacement to get five different data sets, each a small percentage of the total data set[2]. We train (randomly initialized) networks on each of these subsets of the data. The final prediction for the stresses is obtained by averaging the predictions of each of the five networks, for a given set of inputs. This averaging procedure is called ‘bagging’ and has been shown to improve accuracy for learning algorithms sensitive to changes in the training data [9]. This technique was also employed by Ma et al. [27].

An extra input is added compared to those given by equation (13): the interface slope $\partial h_{\text{int}}/\partial s$. The interfacial slope can easily be added to the neural network as an input. This input does not fit in conventional closure terms which are calculated for the fully developed steady state, since it is a locally defined variable[3]. If fully developed flow is assumed, or similarly the effect of the wavy interface is averaged out over a length of pipe (as is done by e.g. Andritsos and Hanratty [2]), the average interface slope will be zero (for a flow with a wavy perturbation) and cannot be used to differentiate stresses at different phases of the wavy perturbation.

With this addition, and the given selection of variable parameters, the neural network takes the form shown in Figure 5.

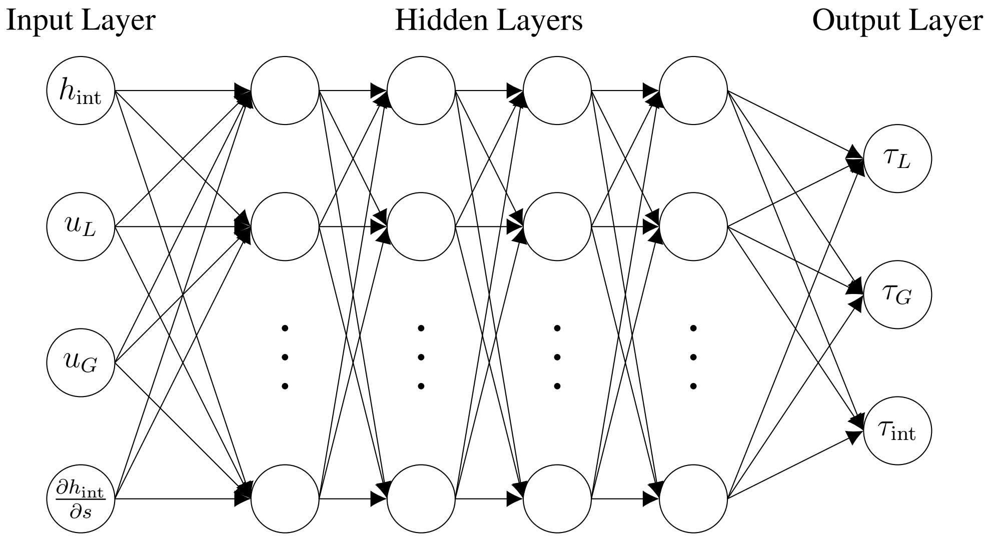


Figure 5: A schematic of the neural network trained in subsection 3.4, and tested in section 4, with four variable inputs, four hidden layers (with 18 nodes per layer), and three outputs.

[2] The percentages of the data sets taken per individual training are 5% for the ‘zero wavy net’ and ‘developed wavy net’, and 10% for the ‘zero + developed wavy net’

[3] An exception is Brauner and Moalem Maron [8], who modified conventional closure terms (based on Taitel and Dukler [40]) to add a dependency of the interfacial stress on the interfacial slope, by matching experimentally observed and theoretically calculated stability boundaries (the latter of which depends on the closure terms). The advantage of our method is that the interfacial slope is included in the correlation from the beginning, in the same straightforward manner as the other inputs.

Resulting regression plots for a ‘zero + developed wavy net’ are shown in Figure 6. The correlation is satisfactory, considering the variation in flow patterns found in the data. The influence of the extra input parameter $\partial h_{\text{int}}/\partial s$ is illustrated by comparison of Figure 6 to Figure 7, where in the latter figure results are shown if this parameter is left out. Apparently the interfacial slope is an important piece of information for the determination of the stresses. This parameter allows the stress prediction to vary based on the wave’s amplitude and local phase, and allows distinction between fully developed and unsteady flow, alleviating the uniqueness issue discussed in subsection 2.4.

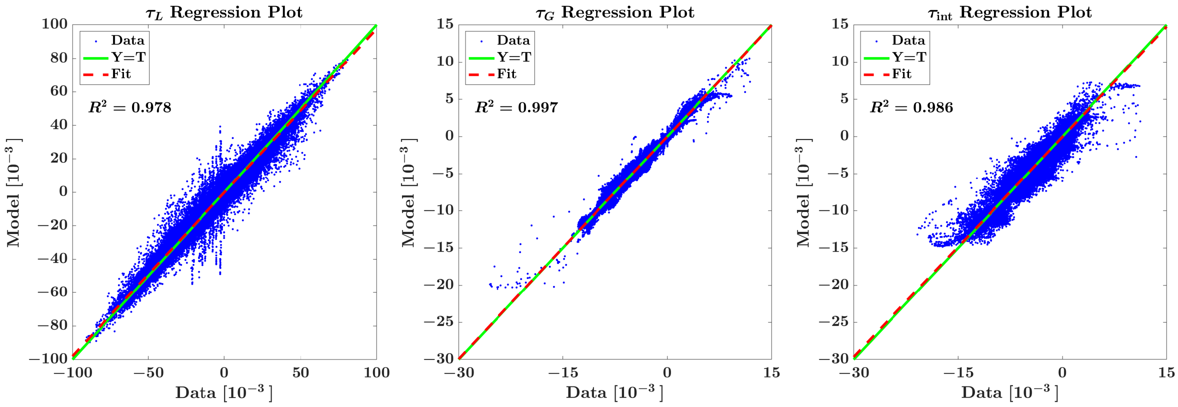


Figure 6: Regression plots for a neural network trained on the data combined for both initial conditions of Table 3 (‘zero + developed wavy net’).

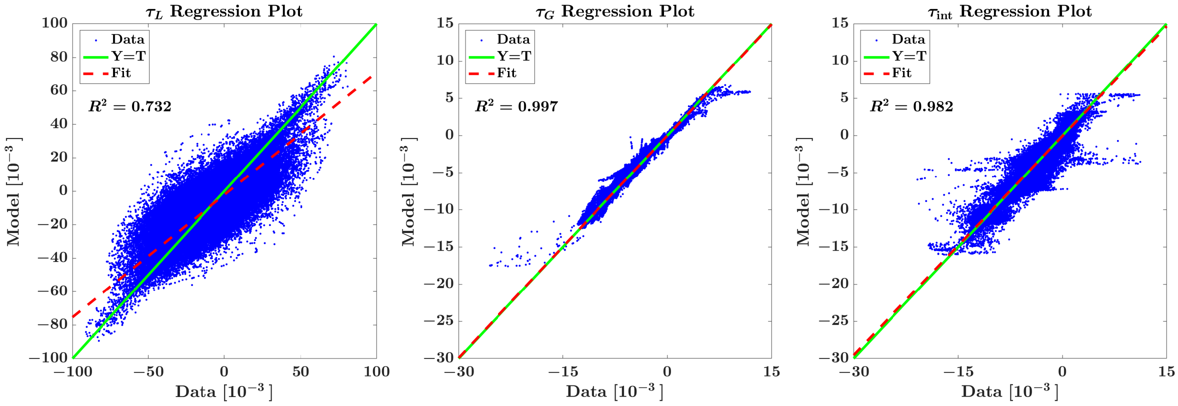


Figure 7: Regression plots for a neural network trained on the data combined for both initial conditions of Table 3 (‘zero + developed wavy net’) excluding the interface slope $\partial h_{\text{int}}/\partial s$ as an input.

4 RESULTS

The true test of the learned closure terms lies in their application to the low-fidelity model Rosa, and comparison of the resulting predictions to high-fidelity model Gerris predictions (presumed to be the truth). At each stage in the Runge-Kutta time integration scheme, the variables as given in (5) are fed to the trained neural networks to arrive at values for the stresses τ_L, τ_G, and τ_{int}. The current MATLAB shallow neural network implementation significantly slows down the Rosa code simulations, compared to when the analytical closure relations (14) are used.

In order to be able to compare Gerris and Rosa results (with different grid resolutions) quantitatively, cubic splines of the variables of interest are constructed, along the horizontal axis. The resulting s-dependent Gerris result at $t = t_i$ is $y_i = y_i(s)$, with $\widehat{y}_i = \widehat{y}_i(s)$ the corresponding Rosa result. We compute characteristic values y_c for each variable of interest, based on analytical solutions for laminar single phase flow. Table 4 shows the following relative error measure for the difference between Gerris and Rosa results, termed the 'normalized averaged error' (NAE):

$$\mathrm{NAE} = \frac{1}{N_T} \sum_{i=1}^{N_T} \sqrt{\frac{1}{L} \int_{s=0}^{L} \left(\frac{y_i - \widehat{y}_i}{y_c} \right)^2 \mathrm{d}s}. \tag{24}$$

The parameter N_T is the total number of time steps and L is the length of the domain.

This error is shown for simulations initialized from different initial conditions, and using different closure terms. Analytical closure terms (14) are tested alongside closure terms learned from the wavy unsteady data of Table 3, using neural networks. Where the neural network assisted error is smaller than the analytical closure error, the error value is highlighted green in Table 4.

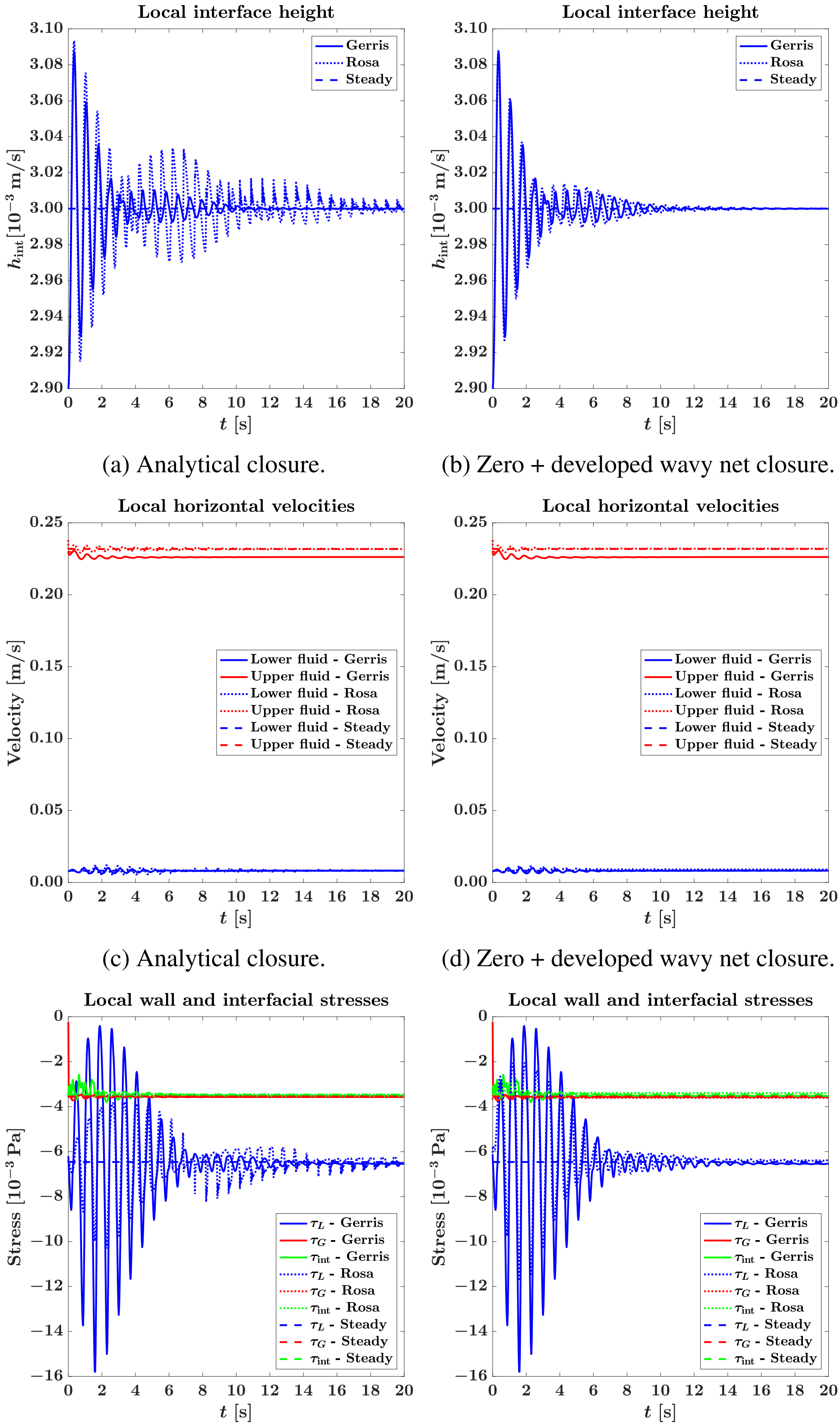
Table 4: Normalized averaged errors (24) between Gerris and Rosa simulations, for different variables of interest. Results are given for Gerris and Rosa simulations starting from different initial conditions, with the Rosa simulations using either analytical or neural network closure terms. Where the neural network closure terms outperform the analytical closure terms (for the same initialization), the result is highlighted in green.

Case		Normalized Averaged Error $[10^{-3}]$					
Initialization	Closure	h_{int}	u_L	u_G	τ_L	τ_G	τ_{int}
zero wavy	analytical	1.05	84.6	13.9	212	7.33	26.4
zero wavy	zero wavy net	0.31	754	4.22	283	26.2	38.5
zero wavy	zero + developed wavy net	0.52	193	15.2	233	10.8	28.5
developed wavy	analytical	1.09	75.5	12.2	215	7.56	18.2
developed wavy	developed wavy net	0.42	385	3.52	215	5.07	18.7
developed wavy	zero + developed wavy net	0.51	112	12.6	173	8.20	26.6

Overall, with error measure (24), the results with neural network closure terms do not show a significant improvement, except perhaps for the interface height. However, this error measure is crude and does not show how well the wave dynamics are reproduced.

We therefore study the values of h_{int}, u_L, u_G, τ_L, τ_G, τ_{int} as a function of time in Rosa simulations for the test case given by Table 1, at a point at the center of the domain ($s = 0.06\,\mathrm{m}$) (see Figure 8). The scale and form of the oscillations are captured better when using the neural network closure terms; the wave damping behavior corresponds better to the high-fidelity simulations. The interface height in the entire domain is shown in Figure 9 for a number of time instants, with the shown Rosa simulations employing the neural network closure terms.

The problem with the analytical closure terms is highlighted in Figure 10, in which the same simulation results are shown for later time instants (with the analytical closure). The waves acquire a sharp wavefront, in the wake of which small spurious waves are formed. These effects are unphysical and are not observed in the Gerris simulations. The neural network closure terms do not suffer from these spurious effects, probably due to their better damping behavior.



(a) Analytical closure.

(b) Zero + developed wavy net closure.

(c) Analytical closure.

(d) Zero + developed wavy net closure.

(e) Analytical closure.

(f) Zero + developed wavy net closure.

Figure 8: Evolution in time of the velocities, stresses and interface height at the center of the domain. Initialized with the ‘developed wavy’ initial condition.

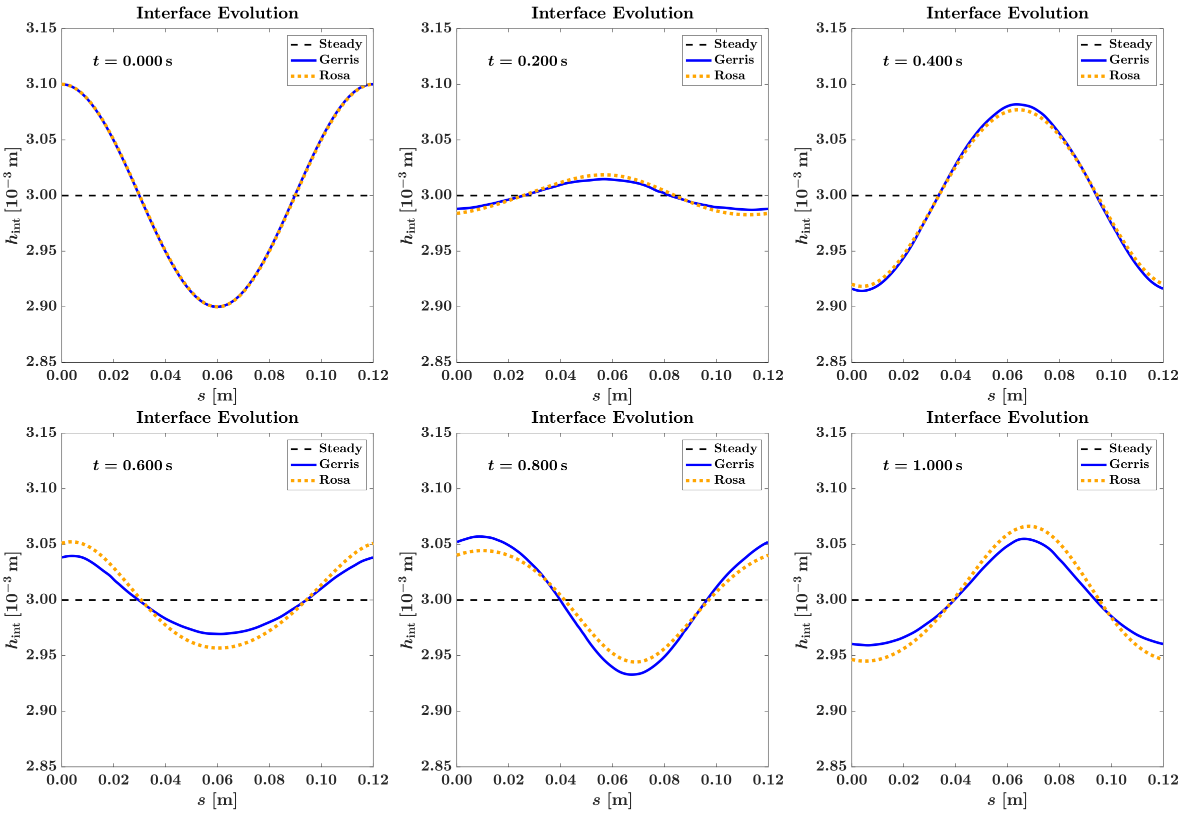


Figure 9: Evolution in time of the interface between liquid and gas throughout the domain, zoomed in at the interface ($H = 0.01\,\mathrm{m}$). Rosa results with a ‘developed wavy’ initialization and ‘zero + developed wavy net’ closure.

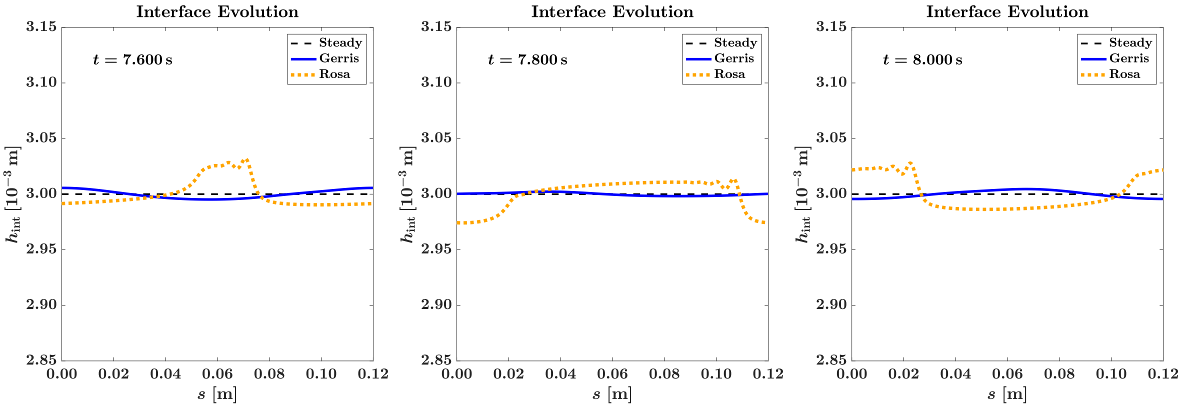


Figure 10: Evolution in time of the interface between liquid and gas throughout the domain, zoomed in at the interface ($H = 0.01\,\mathrm{m}$). Rosa results with a ‘developed wavy’ initialization and analytical closure.

Training the neural network on unsteady simulation data allows the closure terms to capture the unsteady (damping) behavior, differentiating them from conventional steady state closure terms (including those closure terms that consider the streamwise averaged effect of a wavy interface). The addition of the extra closure input parameter $\partial h_{\text{int}}/\partial s$, allows the closure terms to apply the learned differences between steady state and unsteady flow patterns. By providing information on the wave amplitude and local phase, this input parameter enables distinction between steady state flow and increasingly unsteady flow during application in Rosa. This allows the closure terms to provide different results for different phases of the wave damping process. Similarly, the closure terms can produce different stresses at different points along the wave (beyond the distinction made possible by the small differences in interface height and averaged velocities).

One of the problems still visible in Figure 8 is a discrepancy in the steady state gas velocity; this can be solved by further grid refinement of the Gerris simulations.

The remaining main difference between Gerris simulations and Rosa simulations using neural network closure terms is a discrepancy in the wave speed. The wave speed of the Rosa simulations is slightly higher than that of the Gerris simulations, so that the two slowly drift out of phase. This difference in wave speed between Gerris and Rosa simulations can be explained by the fact that a discrepancy between the models remains that cannot be solved via modeling the closure terms (see subsection 2.4). The inviscid dispersion relations for the test case, plotted in Figure 3, indeed show a higher wave speed for the 1D model than for the 2D model.

5 CONCLUSION & OUTLOOK

In this work, we have explored a new approach based on neural networks to solve the longstanding closure problem for stratified multiphase flow in channels. We have trained neural networks on high fidelity simulation data to learn closure terms for the wall and interfacial stresses in a low fidelity model; the 1D two-fluid model for stratified channel flow. An important novelty in our work is the inclusion of the streamwise derivative of the interface height as a feature in the neural network. With this addition, the dynamic wave-damping behavior of high-fidelity simulations was reproduced better than with the conventional (steady state) set of closure terms available in literature [45].

With the proposed framework, closure terms can be constructed for specific flow regimes and duct geometries, as long as high-fidelity simulations are available. The addition of extra inputs to the closure relations, which is straightforward in this framework, alleviates their inherent uniqueness problem. An example of possible extra inputs, besides the interface slope, are the spatial and temporal derivatives of the velocities.

We note that, even with a highly accurate closure model for the stresses, the 1D model will generally not exactly reproduce the 2D results, because the stresses are not the only source of discrepancy between the 1D and 2D model. In principle, it might be possible to eliminate these discrepancies by modeling the difference between high- and low-fidelity model predictions directly, and adjusting the low-fidelity model accordingly. But this approach would be less physical, so that it might not generalize as well.

In the future we aim to improve the framework through closer inspection of the structure of the learned closure terms, and possibly through the inclusion of physical constraints in the network structure. This will open the door to more challenging cases, such as the prediction of slug flow.

REFERENCES

[1] A. Alizadehdakhel, M. Rahimi, J. Sanjari, and A. A. Alsairafi. CFD and artificial neural network modeling of two-phase flow pressure drop. *International Communications in Heat and Mass Transfer*, 36:850–856, 2009.

[2] N. Andritsos and T. J. Hanratty. Influence of interfacial waves in stratified gas-liquid flows. *AIChE Journal*, 33:444–454, 1987.

[3] D. Barnea and Y. Taitel. Interfacial and structural stability of separated flow. *International Journal of Multiphase Flow*, 20:387–414, 1994.

[4] J. B. Bell, P. Colella, and H. M. Glaz. A second-order projection method for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 85:257–283, 1989.

[5] D. Biberg. A mathematical model for two-phase stratified turbulent duct flow. *Multiphase Science and Technology*, 19, 2007.

[6] D. Biberg and G. Halvorsen. Wall and interfacial shear stress in pressure driven two-phase laminar stratified pipe flow. *International Journal of Multiphase Flow*, 26:1645–1673, 2000.

[7] N. Brauner and D. Moalem Maron. Stability analysis of stratified liquid-liquid flow. *International Journal of Multiphase Flow*, 18:103–121, 1992.

[8] N. Brauner and D. Moalem Maron. The role of interfacial shear modelling in predicting the stability of stratified two-phase flow. *Chemical Engineering Science*, 48:2867–2879, 1993.

[9] L. Breiman. Bagging predictors. *Machine Learning*, 24:123–140, 1996.

[10] J. F. H. Buist. *Machine Learning for Closure Models in Multiphase-Flow Applications*. Master's thesis, Eindhoven University of Technology, 2019.

[11] A. J. Chorin. On the convergence of discrete approximations to the Navier-Stokes equations. *Mathematics of Computation*, 23:341–353, 1969.

[12] S. W. Churchill. Friction factor equation spans all fluid flow regimes. *Chemical Engineering*, 84:91–92, 1977.

[13] N. Coutris, J. M. Delhaye, and R. Nakach. Two-phase flow modelling: the closure issue for a two-layer flow. *International Journal of Multiphase Flow*, 15:977–983, 1989.

[14] A. V. Coward, Y. Y. Renardy, M. Renardy, and J. R. Richards. Temporal evolution of periodic disturbances in two-layer Couette flow. *Journal of Computational Physics*, 132:346–361, 1997.

[15] M. Espedal. *An Experimental Investigation of Stratified Two-Phase Pipe Flow at Small Inclinations*. PhD thesis, Norwegian University of Science and Technology, 1998.

[16] M. Gamahara and Y. Hattori. Searching for turbulence models by artificial neural network. *Physical Review Fluids*, 2:054604, 2017.

[17] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM Review*, 43:89–112, 2001.

[18] M. T. Hagan and M. B. Menhaj. Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, 5:989–993, 1994.

[19] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39:201–225, 1981.

[20] J. E. Kowalski. Wall and interfacial shear stress in stratified flow in a horizontal pipe. *AIChE Journal*, 33:274–281, 1987.

[21] B. I. Krasnopolsky and A. A. Lukyanov. A conservative fully implicit algorithm for predicting slug flows. *Journal of Computational Physics*, 355:597–619, 2018.

[22] J. Liao, R. Mei, and J. F. Klausner. A study on the numerical stability of the two-fluid model near ill-posedness. *International Journal of Multiphase Flow*, 34:1067–1087, 2008.

[23] J. Ling, A. Kurzawski, and J. Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, 807:155–166, 2016.

[24] C. Lu, S. Sambasivan, A. Kapahi, and H. S. Udaykumar. Multi-scale modeling of shock interaction with a cloud of particles using an artificial neural network for model representation. *Procedia IUTAM*, 3:25–52, 2012.

[25] C. Lu. *Artificial Neural Network for Behavior Learning from Meso-Scale Simulations, Application to Multi-Scale Multimaterial Flows*. Master's thesis, University of Iowa, 2010.

[26] M. Ma, J. Lu, and G. Tryggvason. Using statistical learning to close two-fluid multiphase flow equations for a simple bubbly system. *Physics of Fluids*, 27:092101, 2015.

[27] M. Ma, J. Lu, and G. Tryggvason. Using statistical learning to close two-fluid multiphase flow equations for bubbly flows in vertical channels. *International Journal of Multiphase Flow*, 85:336–347, 2016.

[28] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21:239–245, 1979.

[29] L. M. Milne-Thomson. *Theoretical Hydrodynamics*. Macmillan, London, 4th edition, 1962.

[30] M. Montini. *Closure Relations of the One-Dimensional Two-Fluid Model for the Simulation of Slug Flows*. PhD thesis, Imperial College London, 2011.

[31] D. Nguyen and B. Widrow. Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In *Proceedings of the 1990 IJCNN International Joint Conference on Neural Networks*, volume 3, pages 21–26, San Diego, CA, USA. IEEE, 1990.

[32] R. E. Osgouei, A. M. Ozbayoglu, E. M. Ozbayoglu, E. Yuksel, and A. Eresen. Pressure drop estimation in horizontal annuli for liquid–gas 2 phase flow: Comparison of mechanistic models and computational intelligence techniques. *Computers & Fluids*, 112:108–115, 2015.

[33] S. Popinet. An accurate adaptive solver for surface-tension-driven interfacial flows. *Journal of Computational Physics*, 228:5838–5866, 2009.

[34] S. Popinet. Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *Journal of Computational Physics*, 190:572–600, 2003.

[35] B. Sanderse, S. Misra, and S. Dubinkina. Numerical simulation of roll waves in pipelines using the two-fluid model. In *Proceedings of the 11th North American Conference on Multiphase Production Technology*, pages 373–386, Banff, Canada, 2018.

[36] B. Sanderse and A. E. P. Veldman. Constraint-consistent Runge–Kutta methods for one-dimensional incompressible multiphase flow. *Journal of Computational Physics*, 384:170–199, 2019.

[37] B. Sanderse, I. E. Smith, and M. H. W. Hendrix. Analysis of time integration methods for the compressible two-fluid model for pipe flow simulations. *International Journal of Multiphase Flow*, 95:155–174, 2017.

[38] F. Sarghini, G. de Felice, and S. Santini. Neural networks based subgrid scale modeling in large eddy simulations. *Computers & Fluids*, 32:97–108, 2003.

[39] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:146–159, 1994.

[40] Y. Taitel and A. E. Dukler. A model for predicting flow regime transitions in horizontal and near horizontal gas-liquid flow. *AIChE Journal*, 22:47–55, 1976.

[41] The MathWorks, Inc. MATLAB Documentation - Deep Learning Toolbox. 2019. URL: `https://nl.mathworks.com/help/deeplearning/index.html` (visited on 02/03/2019).

[42] B. D. Tracey, K. Duraisamy, and J. J. Alonso. A machine learning strategy to assist turbulence model development. In *Proceedings of the 53rd AIAA Aerospace Sciences Meeting*, Kissimmee, Florida. American Institute of Aeronautics and Astronautics, 2015.

[43] G. Tryggvason, R. Scardovelli, and S. Zaleski. *Direct Numerical Simulations of Gas-Liquid Multiphase Flows*. Cambridge University Press, 2011.

[44] A. Ullmann and N. Brauner. Closure relations for two-fluid models for two-phase stratified smooth and stratified wavy flows. *International Journal of Multiphase Flow*, 32:82–105, 2006.

[45] A. Ullmann, A. Goldstein, M. Zamir, and N. Brauner. Closure relations for the shear stresses in two-fluid models for laminar stratified flow. *International Journal of Multiphase Flow*, 30:877–900, 2004.

[46] S. O. Unverdi and G. Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics*, 100:25–37, 1992.

[47] D. L. Youngs. Time-dependent multi-material flow with large fluid distortion. In K. W. Morton and M. J. Baines, editors, *Numerical Methods for Fluid Dynamics*. Academic Press, 1982.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

CONVERGENCE ANALYSIS OF A MULTI-LEVEL KRIGING MODEL: APPLICATION TO UQ IN CFD

Y. Zhang[1,2], R.P. Dwight[2], and Z.-H. Han[1]

[1]School of Aeronautics, Northwestern Polytechnical University
Youyi west road 127, 710072 Xi'an, People's Republic of China
zhangyu91@mail.nwpu.edu.cn, hanzh@nwpu.edu.cn

[2] Aerodynamics Group, Faculty of Engineering, Delft University of Technology
Kluyverweg 1, 2629 HS Delft, The Netherlands
r.p.dwight@tudelft.nl

Keywords: multi-level, gaussian process, uncertainty quantification, computational fluid dynamics, Kriging.

Abstract. *Multi-level surrogate modelling offers the promise of fast approximation to expensive simulation codes for the purposes of uncertainty quantification (UQ). The hope is that a large number of cheap samples from the simulator on coarse grids, can be corrected by a few expensive samples on a fine grid, to build an accurate surrogate. Of the various multi-level approaches, a correction-based method using Gaussian process regression (Kriging) is studied here. In particular, we examine the "additive bridge-function" method, for which – although widely applied – results on theoretical convergence rates and optimal numbers of samples per level are not present in the literature. In this paper, we perform a convergence analysis for the expectation of a quantity of interest (QoI), utilizing convergence results for single-fidelity Kriging, as well as existing multi-level analysis methodology previously applied in context of polynomial-based methods. Rigorous convergence and computational cost analyses are provided. By minimizing the total cost, optimal numbers of sampling points on each grid level are determined. Numerical tests demonstrate the theoretical results for: a 2d Genz function, Darcy flow with random coefficients, and Reynold-Averaged Navier-Stokes (RANS) for the flow over an airfoil with geometric uncertainties. The efficiency and accuracy of this method are compared with standard- and multi-level Monte Carlo. All the test cases show that using our multi-level kriging model significantly reduces cost.*

1 INTRODUCTION

With the rapid growth of computational capacity and improvements in CFD simulation techniques over the past two decades, CFD-based aerodynamic analysis and design have become standard in industry. However, in any analysis of real-world systems there exist uncertainties and errors, e.g.: discretization error, geometry uncertainty and turbulence model-form uncertainty, which can cause the prediction of performance to be poor. Thus, it is necessary to consider the effect of uncertainties on simulation predictions, i.e. which leads inexorably to solving stochastic PDEs. Statistics of the solutions, such as the expectation of a Quantity-of-Interest (QoI), $\mathbb{E}[y]$, can be straightforwardly estimated with Monte-Carlo or Taylor expansions, however, due to the high computational cost of individual CFD simulations, the lack of reliable derivatives, and strong nonlinearities, these methods are often impractical.

Surrogate-based methods on the other hand, can use a modest number of simulations to provide a fast approximation to $\mathbb{E}[y]$, provided the stochastic dimension is moderate. Some common surrogates in the literature use polynomial-chaos expansions (PCE) for interpolation or regression (often with sparsity), radial basis-function (RBF) interpolation, and Kriging. Of these, Kriging is notable for its high flexibility, thanks to its Bayesian roots – and has performed well in practical applications. Multi-fidelity and multi-level variants have been developed, which use additional low-cost simulations to assist in the estimation of $\mathbb{E}[y]$. In *multi-fidelity* methods the low-cost simulation uses a simplified model of the problem (e.g. Euler versus Navier-Stokes); in *multi-level* the low-cost simulation is the same continuous model as the high-cost, but discretized at a coarser grid resolution. In the former case the correlation between high- and low-fidelities is responsible for the cost reduction (if any); in the multi-level case we can use stronger relationships given by the rate of (grid-)convergence of the PDE solver. Although multi-fidelity and multi-level Kriging methods are widely applied in engineering, they are known to be unreliable, and do not consistently reduce costs in practice. This is at least partially a result of the lack of the convergence analysis for these methods – and as a consequence, the lack of rules for optimal sample selection.

There has been enormous work on multi-fidelity or multi-level surrogate modelling. Haftka et al. [1] [2] developed a variable-fidelity kriging model, which uses a multiplicative bridge function to correct the low-fidelity model to approximate the high-fidelity function. Gano et al. [3] developed a hybrid bridge function method, which uses a kriging model to scale the low-fidelity model. Han et. al. [4] improved variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function, to realize a more accurate and robust model. Cokriging was originally proposed in geostatistics community by Journel et al. [5] and then extended to deterministic computer experiments by Kennedy and O'Hagan, called KOH autoregressive model [6]. Han et. al. [7] proposed an improved version of cokriging, which can be built in one step, and a hierarchical kriging model [8], which avoids the cross-variance between low- and high-fidelity model thus is more robust. More recently, the multi-fidelity/level models are introduced into the field of uncertainty quantification. Palar et al. [9] developed a multi-fidelity non-intrusive polynomial chaos method based on regression, which builds two PCEs for both the low-fidelity and correction functions, and then sum it up to provide an estimation for the high-fidelity function. This method has been applied for flow around a NACA0012 airfoil and a Common Research Model wing with flow condition uncertainties, e.g. Mach number, angle of attack. Parussini et al. [10] proposed a recursive multi-fidelity cokriging model and tested it by stochastic Burgers equation and the stochastic Oberbeck-Boussinesq equations. Palar et al. [11] investigated the capability of a Hierarchical Kriging model for uncertainty analysis and further improve it by combining with PCE method. The application to RAE2822 airfoil and CRM wing with flow condition uncertainty shows its high accuracy and robust performance. Narayan et al. [12] proposed a multi-fidelity stochastic collocation method, which leverage inexpensive low-fidelity models to generate surrogates for an expensive high-fidelity model using a parametric collocation (nonintrusive) approach. Zhu et al. [13] present a bi-fidelity algorithm for approximating the statistical moments of stochastic problems and provide a basic error analysis.

In this paper, We choose multi-level rather than multi-fidelity [14], in order to make use of stronger convergence results. The popularity of multi-level methods has increased dramatically in recent years, thanks to the success of Multi-Level Monte-Carlo (MLMC) methods [15, 16, 17]. By correctly choosing the number of Monte-Carlo samples per level, the cost of solving the stochastic PDE can be reduced to a constant multiple of the cost of a single deterministic solution, in the best case. Similar ideas were used by Teckentrup et al. [18] to devise a multi-level stochastic collocation method, dramatically improving upon MLMC for moderate stochastic dimension. This paper addresses the convergence of a particular multi-level method known as Additive Bridge-Function-based multi-level Kriging [19, 20, 21] for estimation of $\mathbb{E}[y]$. We use the Kriging mean as response surface only, the Kriging variance is not used in our analysis. Our work follows closely the outline of [18], but considering Kriging models rather than polynomial models. We employ error bounds derived for RBF interpolation [22] to estimate the interpolation error in the Kriging mean. Finally we provide expressions for optimal number of samples per level to obtain minimum computational cost.

The structure of this paper is as following: in Section 2, the additive-bridge function multi-level Kriging model is described, and in Section 3 it's convergence properties are analysed and computational cost estimates are provided. Section 4 briefly introduces the numerical test cases used in this paper, and numerical results are presented in Section 5, and compared to standard MC and MLMC.

2 METHODOLOGY

A single-fidelity ordinary kriging model is presented in Section 2.1, see also e.g. [23]; and then we describe how to construct a multi-level Gaussian process model from multiple single-level models in Section 2.2.

2.1 Single-fidelity ordinary Kriging

Consider a QoI $y \in \mathbb{R}$, which (possibly via a PDE) is a function of (deterministic) variables $\boldsymbol{\xi} \in \mathbb{R}^M$. Ordinary Kriging represents $y(\boldsymbol{\xi})$ by a Gaussian process Y of the form:

$$Y(\boldsymbol{\xi}) = \rho + Z(\boldsymbol{\xi}), \tag{1}$$

where ρ is an unknown constant and $Z(\boldsymbol{\xi})$ is a stationary Gaussian random process with zero-mean and covariance

$$\mathrm{Cov}[Z(\boldsymbol{\xi}), Z(\boldsymbol{\xi}')] = \sigma^2 R(|\boldsymbol{\xi} - \boldsymbol{\xi}'|). \tag{2}$$

Here $R : \mathbb{R}^+ \to \mathbb{R}$ is a positive-definite covariance kernel, so that the covariance of two points of the process only depends on their Euclidean distance in $\boldsymbol{\xi}$-space, and σ is the standard-deviation.

Given observations $\boldsymbol{y}_\Xi \in \mathbb{R}^N$ at a number of samples $\Xi = (\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_N)$, we can construct the conditional process $Y \mid \boldsymbol{y}_\Xi$. Thanks to Gaussian assumptions, the mean of this process at an unoberserved location $\boldsymbol{\xi}$ can be formulated as a linear combination of the observed responses:

$$\hat{y} = \boldsymbol{\lambda}(\boldsymbol{\xi})^T \boldsymbol{y}_\Xi. \tag{3}$$

In particular by minimizing mean-squared error subject to unbiasedness constraints, the predictor at any unsampled site is given by

$$\hat{y} = \rho + \boldsymbol{r}^T \boldsymbol{R}^{-1}(\boldsymbol{y}_\Xi - \rho \boldsymbol{F}), \tag{4}$$

$$\rho = (\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{F})^{-1}(\boldsymbol{F}^T \boldsymbol{R}^{-1} \boldsymbol{y}_\Xi). \tag{5}$$

Here $\boldsymbol{F} := \mathbf{1} \in \mathbb{R}^N$, the covariance matrix $\boldsymbol{R} := R(\Xi, \Xi) \in \mathbb{R}^{N \times N}$, and finally $\boldsymbol{r} := R(\Xi, \boldsymbol{\xi}) \in \mathbb{R}^N$ is the vector consisting of the covariance of the unobserved sample with respect to all observed sample sites. Using this cheap surrogate, $\mathbb{E}_\xi[y]$ can be evaluated with e.g. Monte-Carlo to any desired accuracy.

2.2 Additive bridge function based multi-level kriging model (AMLK)

If y results from the solution of a PDE, then by varying grid resolution we can have a sequence of numerical approximations to y, denoted $y_0, \ldots, y_L$, of increasing accuracy and increasing computational cost. The heart of the additive bridge function based multi-level Kriging model (AMLK) [19, 20, 21], is then to first write y_L as the telescopic sum

$$y_L = \sum_{l=0}^{L} \delta_l, \qquad \delta_0 := y_0, \quad \delta_l := y_l - y_{l-1},\ l \in \{1, \ldots, L\}, \tag{6}$$

similarly to the MLMC method; and then approximate each δ_l with a single-level Kriging surrogate $\hat{\delta}_l$. An estimate of the expectation of the QoI can then be written:

$$\mathbb{E}_\xi[y] \simeq \mathbb{E}_\xi[\hat{y}_L] := \mathbb{E}_\xi \left[\sum_{l=0}^{L} \hat{\delta}_l \right] = \sum_{l=0}^{L} \mathbb{E}_\xi[\hat{\delta}_l], \tag{7}$$

where the expectations are then evaluated on the surrogate, independently of each other. This decomposition is worthwhile because on the finest level L, the cost of the simulation is high, but the absolute magnitude of δ_L is small, so surrogate modelling errors $(\delta_L - \hat{\delta}_L)$ contribute little to the total error in $\mathbb{E}_\xi[y]$, and therefore sufficient accuracy can be achieved with few samples. In contrast, on the coarsest level many samples are needed to reduce the surrogate modelling error there, but these samples are very cheap to obtain. Potentially then, the total cost of estimating $\mathbb{E}[y]$ at a given accuracy can be reduced compared to the single-level method. Whether or not it is, in fact, reduced is investigated in the next section.

3 CONVERGENCE ANALYSIS OF AMLK

The error of using any surrogate model to estimate $\mathbb{E}[y]$ can be bounded by the discretization error and the surrogate interpolation error separately:

$$|\mathbb{E}[y - \hat{y}_L]| \leq \underbrace{|\mathbb{E}[y - y_L]|}_{\varepsilon_{\Delta x}} + \underbrace{|\mathbb{E}[y_L - \hat{y}_L]|}_{\varepsilon_h}, \tag{8}$$

where the former is a function of grid resolution $\Delta x > 0$, and the latter depends on some sampling parameter denoted h to be specified later.

In the multi-level case, assume that the grid resolution on level l is Δx_l, and further that there exist constants $\alpha, C_d > 0$ (independent of Δx), such that for all fidelity levels $l \in \{0, \ldots, L\}$:

$$\varepsilon_{\Delta x} := |\mathbb{E}[y - y_l]| \leq C_d \Delta x_l^\alpha, \tag{9}$$

i.e. the discrete approximation of $\mathbb{E}[y]$ converges at a fixed rate, where $\alpha = 2$ implies a 2nd-order accurate discretization, etc.

Consider now the interpolation error ε_h. By analogy with radial-basis function (RBF) interpolation [22], for a single-fidelity Gaussian process, the point-wise interpolation error in the process-mean can be expressed in terms of a *fill distance* h, defined as

$$h = h_{\Xi} := \sup_{\boldsymbol{\xi}\in\Omega} \left\{ \min_{\xi_i \in \Xi} \|\boldsymbol{\xi} - \boldsymbol{\xi}_i\| \right\}.$$

Here Ω is the interpolation domain, and Ξ is the sample sites. Then h is the radius of the largest (hyper-)sphere, whose center is contained in Ω, and which contains none of the samples. Given approximation of an infinitely differentiable function, the convergence order is dictated by the continuity of the covariance kernel (or radial basis function in RBF interpolation). For example, when the so-called thin-plate spline $R(r) := (-1)^{k+1} r^{2k} \log r$, with $k \in \mathbb{N}$, $r = |\boldsymbol{\xi} - \boldsymbol{\xi}'| \in \mathbb{R}$, is used, the ℓ^∞-norm of the interpolation error will satisfy $\epsilon \sim O(h^{k+1})$ [22]. For infinitely differentiable covariance kernels, convergence in this norm will be spectral. In this article, all derivations and numerical tests are based on the thin-plate spline with $k = 1$, though the results can be extended to other correlation functions straightforwardly.

The interpolation error of an additive bridge based multi-level model can therefore be written

$$\begin{aligned} \varepsilon_h = |\mathbb{E}[y_L - \hat{y}_L]| &\le \left| \mathbb{E}\left[\sum_{l=0}^{L} \delta_l - \sum_{l=0}^{L} \hat{\delta}_l \right] \right| \qquad (10) \\ &\le \sum_{l=0}^{L} |\mathbb{E}[\delta_l - \hat{\delta}_l]| \\ &\le \sum_{l=0}^{L} C_I \Delta x_l^{\mu} h_l^{\beta}. \end{aligned}$$

where h_l is the fill distance on level l, and β is a constant depending on the correlation function used in the Gaussian process (and the smoothness of the underlying function). Here, h_l^β comes from the approximation properties of the surrogate, and Δx_l^μ describes the magnitude of the interpolation error, which is proportional to size of the function being interpolated.

To limit the total error $|\mathbb{E}[y - \hat{y}_L]|$ to less than ε, we bound both the discretization error and interpolation error by $\varepsilon/2$. First, we choose the finest level L large enough to satisfy $\varepsilon_{\Delta x} = C_d \Delta x_L^\alpha \le \varepsilon/2$. For simplicity, we assume that $\Delta x_l = \eta^{-l} \Delta x_0$, i.e. that grid resolution is increased by a constant factor η on each level. By arbitrarily normalizing Δx_0 to 1, the discretization error constraint becomes

$$C_d \eta^{-L\alpha} \le \varepsilon/2 \implies L = \left\lceil \frac{1}{\alpha} \log_\eta\left(\frac{2C_d}{\varepsilon}\right) \right\rceil. \qquad (11)$$

Similarly, to limit the interpolation error to $\varepsilon/2$, the infill distance of the surrogate must satisfy

$$h_l^\beta \le \frac{C_d \Delta x_L^\alpha \Delta x_l^{-\mu}}{(1+L)C_I} = \frac{C_d \eta^{l\mu - L\alpha}}{(1+L)C_I}. \qquad (12)$$

Given which, the total error is bounded as

$$|\mathbb{E}[y - \hat{y}_L]| \le 2 C_d \Delta x_L^\alpha. \qquad (13)$$

3.1 Cost analysis for AMLK

Having found bounds on h_l in (12), it remains to specify the optimal number of samples per level N_l. Let the cost for a single sample of y_l be T_l. Then the total computational cost is

$$T = \sum_{l=0}^{L} N_l T_l. \tag{14}$$

To choose N_l optimally, we minimize T subject to the constraint on error. Treating N_l as a continuous variables, we solve the optimization problem:

$$\min_{N_l \in \mathbb{R}^+} T, \quad \text{subject to} \quad \sum_{l=0}^{L} C_I \Delta x_l^{\mu} h_l^{\beta} = \varepsilon/2. \tag{15}$$

Further assume that there exist constants $C_c, \gamma \in \mathbb{R}$ (independent of Δx_l), such that the cost of a evaluation is

$$T_l = C_c \Delta x_l^{\gamma}, \tag{16}$$

which is approximately true for typical PDEs solvers. Finally, as it is usually difficult to estimate h (especially in high-dimensional spaces), we choose to treat the estimated error as a function of N (to which we have direct access). So, for a specific sampling method, we assume there exist constants C_s and ν, such that

$$h_l = C_s N_l^{\nu}. \tag{17}$$

Note that ν will vary with the stochastic dimension M, and depends also on the sampling method. For tensor-product samples $\nu = -\frac{1}{M}$ can be seen immediately, i.e. the curse of dimensionality. In terms of N, the convergence rate of the surrogate model method deteriorates with an increasing of number of input variables. In terms of h, it is dimension-independent.

With assumptions (16) and (17), and formulating the constrained minimization problem (15) in terms of a Lagrange multiplier λ, we obtain the equivalent problem, find N_l, λ such that:

$$\frac{\partial f}{\partial N_l} = 0, \quad \frac{\partial f}{\partial \lambda} = 0,$$

where

$$f(N_l, \lambda) = \sum_{l=0}^{L} N_l C_c \Delta x_l^{\gamma} + \lambda \left(\sum_{l=0}^{L} C_I \Delta x_l^{\mu} (C_s N_l^{\nu})^{\beta} - \varepsilon/2 \right). \tag{18}$$

Explicitly

$$\frac{\partial f}{\partial N_l} = C_c \Delta x_l^{\gamma} + \lambda C_I \Delta x_l^{\mu} C_s^{\beta} \nu \beta N_l^{\nu\beta - 1} = 0, \tag{19}$$

$$\frac{\partial f}{\partial \lambda} = \sum_{l=0}^{L} C_I \Delta x_l^{\mu} (C_s N_l^{\nu})^{\beta} - \varepsilon/2 = 0, \tag{20}$$

whereupon solving for N_l gives

$$N_l = \left\lceil \left(\frac{\varepsilon}{2 C_I C_s^{\beta} S(L)} \right)^{\frac{1}{\nu\beta}} (\Delta x_l)^{\frac{\gamma-\mu}{\nu\beta-1}} \right\rceil, \quad S(L) = \sum_{l=0}^{L} \Delta x_l^{\frac{\mu - \gamma\nu\beta}{1-\nu\beta}}. \tag{21}$$

With N_l determined, we now examine the complexity of the multilevel approximation:

$$\begin{aligned}
T_\varepsilon &= \sum_{l=0}^{L} N_l T_l \\
&\le \sum_{l=0}^{L} \left[\left(\frac{\varepsilon}{2C_I C_s^\beta S(L)} \right)^{\frac{1}{\nu\beta}} (\Delta x_l)^{\frac{\mu-\gamma}{1-\nu\beta}} + 1 \right] C_c \Delta x_l^\gamma \\
&= \sum_{l=0}^{L} \left(\frac{\varepsilon}{2C_I C_s^\beta S(L)} \right)^{\frac{1}{\nu\beta}} (\Delta x_l)^{\frac{\mu-\gamma}{1-\nu\beta}} C_c \Delta x_l^\gamma + \sum_{l=0}^{L} C_c \Delta x_l^\gamma \\
&= \sum_{l=0}^{L} \left(\frac{\varepsilon}{2C_I C_s^\beta S(L)} \right)^{\frac{1}{\nu\beta}} C_c (\Delta x_l)^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}} + \sum_{l=0}^{L} C_c \Delta x_l^\gamma \\
&= \left(\frac{\varepsilon}{2C_I C_s^\beta} \right)^{\frac{1}{\nu\beta}} C_c S(L)^{1-\frac{1}{\nu\beta}} + \sum_{l=0}^{L} C_c \Delta x_l^\gamma
\end{aligned} \tag{22}$$

The cost analysis of the AMLK model follows that of multi-level stochastic collocation method in [18]. First, let's bound the second term on the right side. Recall (11), bounding the finest level L by $\frac{1}{\alpha}\log_\eta(\frac{2C_d}{\varepsilon}) + 1$, where η is the scaling parameter of Δx_l ($\Delta x_l = \eta^{-l}$), we have

$$\begin{aligned}
\sum_{l=0}^{L} C_c \Delta x_l^\gamma &\simeq \sum_{l=0}^{L} C_c \eta^{-l\gamma} \\
&= C_c \frac{\eta^{-\gamma L} - 1}{\eta^{-\gamma} - 1} \\
&= C_c \frac{\eta^{-\gamma(L-1)} - \eta^\gamma}{1 - \eta^\gamma} \\
&\le \frac{C_c \eta^{-\gamma(\frac{1}{\alpha}\log_\eta(\frac{2C_d}{\varepsilon}))}}{1 - \eta^\gamma} \\
&= \frac{C_c (2C_d)^{\frac{-\gamma}{\alpha}}}{1 - \eta^\gamma} (\varepsilon)^{\frac{\gamma}{\alpha}}.
\end{aligned} \tag{23}$$

Then, we provide a bound on the geommetric sum $S(L)$. When $\mu \neq \gamma\nu\beta$, we have

$$\begin{aligned}
S(L) = \sum_{l=0}^{L} \Delta x_l^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}} &\simeq \sum_{l=0}^{L} \eta^{-l\frac{\mu-\gamma\nu\beta}{1-\nu\beta}} \\
&= \frac{\eta^{-L\frac{\mu-\gamma\nu\beta}{1-\nu\beta}} - 1}{\eta^{-\frac{\mu-\gamma\nu\beta}{1-\nu\beta}} - 1} \\
&= \frac{\eta^{-(L-1)\frac{\mu-\gamma\nu\beta}{1-\nu\beta}} - \eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}}{1 - \eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}} \\
&= \frac{\eta^{-\frac{\mu-\gamma\nu\beta}{1-\nu\beta}(\frac{1}{\alpha}\log_\eta(\frac{2C_d}{\varepsilon}))} - \eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}}{1 - \eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}} \\
&= \frac{(2C_d)^{-\frac{\mu-\gamma\nu\beta}{\alpha(1-\nu\beta)}}}{1 - \eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}} (\varepsilon)^{\frac{\mu-\gamma\nu\beta}{\alpha(1-\nu\beta)}} - \frac{\eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}}{1 - \eta^{\frac{\mu-\gamma\nu\beta}{1-\nu\beta}}};
\end{aligned} \tag{24}$$

when $\mu = \gamma\nu\beta$, we have

$$S(L) = L + 1 = \frac{1}{\alpha}\log_\eta(\frac{2C_d}{\varepsilon}) + 2. \tag{25}$$

Finally, when $\mu \neq \gamma\nu\beta$, the computation cost versus ε can be bounded by

$$\begin{aligned} T_\varepsilon &\leq \left(\frac{\varepsilon}{2C_I C_s^\beta}\right)^{\frac{1}{\nu\beta}} C_c S(L)^{1-\frac{1}{\nu\beta}} + \sum_{l=0}^{L} C_c \Delta x_l^\gamma \\ &\leq \varepsilon^{\frac{1}{\nu\beta}} \varepsilon^{\frac{\mu-\gamma\nu\beta}{\alpha(1-\nu\beta)}(1-\frac{1}{\nu\beta})} + \varepsilon^{\frac{1}{\nu\beta}} + \varepsilon^{\frac{\gamma}{\alpha}} \\ &= \varepsilon^{\frac{1}{\nu\beta}+\frac{\gamma\nu\beta-\mu}{\alpha\nu\beta}} + \varepsilon^{\frac{1}{\nu\beta}} + \varepsilon^{\frac{\gamma}{\alpha}}; \end{aligned} \tag{26}$$

when $\mu = \gamma\nu\beta$,

$$T_\varepsilon \leq \varepsilon^{\frac{1}{\nu\beta}} |\log_\eta(\varepsilon)|^{(1-\frac{1}{\nu\beta})} + \varepsilon^{\frac{1}{\nu\beta}} + \varepsilon^{\frac{\gamma}{\alpha}}. \tag{27}$$

Consequently, we have

$$T_\varepsilon \leq \begin{cases} \varepsilon^{\frac{1}{\nu\beta}} & \text{if} \quad \mu > \gamma\nu\beta, \\ \varepsilon^{\frac{1}{\nu\beta}} |\log_\eta(\varepsilon)|^{(1-\frac{1}{\nu\beta})} & \text{if} \quad \mu = \gamma\nu\beta, \\ \varepsilon^{\frac{1}{\nu\beta}+\frac{\gamma\nu\beta-\mu}{\alpha\nu\beta}} & \text{if} \quad \mu < \gamma\nu\beta. \end{cases} \tag{28}$$

Usually, in terms of Δx, the size of the difference between two consecutive level has the same convergence rate with the discretization error, thus $\mu = \alpha$, which is also showed in the following numerical test cases. When $\mu < \gamma\nu\beta$, we have

$$\begin{aligned} T_\varepsilon &\leq \varepsilon^{\frac{1}{\nu\beta}+\frac{\gamma\nu\beta-\mu}{\alpha\nu\beta}} \\ &= \varepsilon^{\frac{1}{\nu\beta}(1-\frac{\mu}{\alpha})+\frac{\gamma}{\alpha}} \\ &\leq \varepsilon^{\frac{\gamma}{\alpha}}. \end{aligned} \tag{29}$$

3.2 Parameter estimation and practical details

A practical algorithm is given below. In the first step, a grid convergence study is needed to provide an estimation for discretization error and determine the finest level L. The second step is to estimate the constants assumed in the model of interpolation error.

Recalling (10), the interpolation error of AMLK model is assumed to be $C_I \Delta x_l^\mu h_l^\beta$. Again, h_l^β represents the convergence properties of the surrogate model, and Δx_l^μ comes from the logic that the magnitude of the interpolation error should be proportional to the size of the function being interpolated, which is the difference between two consecutive levels. Here, β and μ are both constants - which are assumed to be independent of level, and can be estimated separately. The assumption of interpolatin error should ideally be numerically verified.

As already mentioned, it is difficult to estimate h in high-dimensional spaces, so we assume that $h_l = C_s N_l^\nu$, so that

$$C_I \Delta x_l^\mu h_l^\beta = C_I \Delta x_l^\mu (C_s N_l^\nu)^\beta = C_I C_s^\beta \Delta x_l^\mu N_l^{\nu\beta},$$

and instead of estimating β directly, we treat the error as a function of N and estimate $\nu\beta$ together, as well as estimating the combined constant $C_I C_s^\beta$ together. Remember that ν varies with the stochastic dimension M and also depends on the sampling method, so must be re-estimated for each new problem.

To estimate these parameters, firstly, by quantifying the interpolation error for the cheapest three levels using standard kriging model with a fixed, small number of sample points, an estimation for μ can be obtained. At the same time, the computational cost per sample on each level is collected, which gives us an estimation of γ. Then, using the coarsest level only, $\nu\beta$ and $C_I C_s^{\beta}$ can be estimated by varying the number of sampling points. With these estimated constants, the optimal number on each level is determined. Through the algorithm below, UQ with the multi-level kriging model can be conducted.

Algorithm: AMLK to estimate $\mathbb{E}[y]$ with error $< \epsilon$

```
Determine the finest level $L$ from (11) interpolation error is $\varepsilon/2$.
Use the cheapest three levels $l = 0, 1, 2$ to estimate $C_I C_s^{\beta}$ and $\gamma, \nu\beta, \mu$.
for $l = 0 : L$ do
    Calculate the optimal number of samples $N_l$ using (21);
    Generate $N_l$ sample points $\Xi_l$, with e.g. Latin hypercube sampling;
    Evaluate $\boldsymbol{y}_l(\Xi_l)$ and $\boldsymbol{y}_{l-1}(\Xi_l)$ with the PDE solver (note $y_{-1} \equiv 0$);
    Evaluate $\boldsymbol{\delta}_l(\Xi_l) := \boldsymbol{y}_l(\Xi_l) - \boldsymbol{y}_{l-1}(\Xi_l)$;
    Construct a kriging model for $\delta_l$;
    Evaluate $\mathbb{E}[\hat{\delta}_l]$ on surrogate model response surface using e.g. Monte Carlo;
end
Evaluate result $\mathbb{E}[\hat{y}_L] = \sum_{l=0}^{L} \mathbb{E}[\hat{\delta}_l]$ ;
```

4 TEST CASES

4.1 Oscillatory Genz function (M=2)

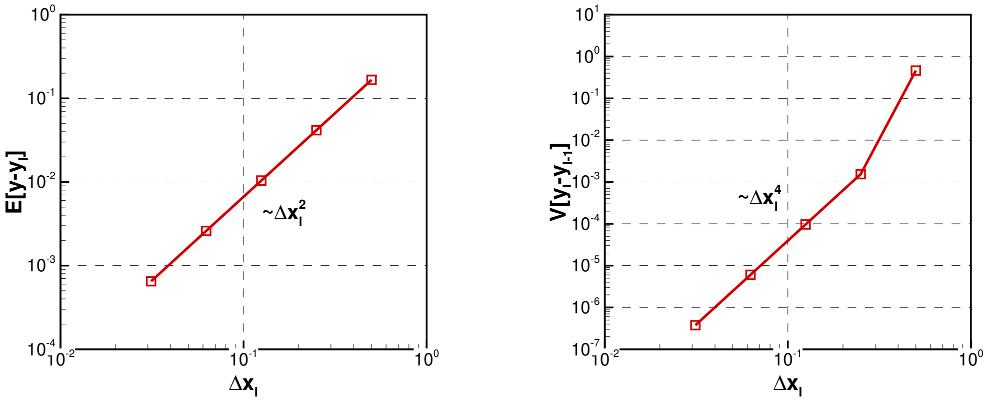
To quickly verify basic properties of AMLK, we consider an almost-trivial analytic test-case based on the osciallatory Genz function in 2d [24]. The basic function is

$$y(\boldsymbol{\xi}) := \cos\left(\pi + 5\xi_1 + 5\xi_2\right), \tag{30}$$

where $\xi_1, \xi_2 \sim \mathcal{U}(0, 1)$ i.i.d. To simulate multi-level analyses we introduce an artificial mesh-dependent term:

$$y_l(\boldsymbol{\xi}) := y(\boldsymbol{\xi}) + \sin(|\boldsymbol{\xi}|)\Delta x_l^2, \tag{31}$$

where $\Delta x_l = 2^{-l}\Delta x_0$, $\Delta x_0 = \frac{1}{2}$, and $L = 4$. Note that $\delta_l(\boldsymbol{\xi})$ has the same form on every level, with only a scale difference. Two hundred random uncertainty samples are used to estimate the discretization error, shown in Figure 1a, which shows quadratic convergence as we expected, $\varepsilon_{\Delta x} = |\mathbb{E}[y - y_l]| \leq C_d \Delta x_l^{\alpha} \sim \Delta x_l^2$. To make a comparison with MLMC method, the variance for the difference function is also estimated, and the rate of convergence is 4, double the rate of convergence of the expectation, as expected in this case, shown in Figure 1b.



(a) Discretization error of multi-level analyses (b) Variance of difference function

Figure 1: Performance plots of multi-level analyses for Genz case

4.2 Darcy flow with random coefficients (M=21)

The first PDE-based test-case is Darcy on $D = (0,1)^d, d = 2$, with both Dirichlet and Neumann boundary conditions [17]:

$$-\nabla \cdot (k(\mathbf{x},\omega)\nabla p(\mathbf{x},\omega)) = 1, \quad \mathbf{x} \in D, \tag{32}$$

$$p|_{x_1=0} = 1, \quad p|_{x_1=1} = 0, \tag{33}$$

$$\frac{\partial p}{\partial \boldsymbol{n}}|_{x_2=0} = 0, \quad \frac{\partial p}{\partial \boldsymbol{n}}|_{x_2=1} = 0, \tag{34}$$

where k is a scalar-valued random field with

$$\log k(\mathbf{x},\omega) = Z(\mathbf{x},\omega) = \mathbb{E}[Z(\mathbf{x},\cdot)] + \sum_{n=0}^{\infty} \sqrt{\theta_n}\xi_n(\omega)b_n(\mathbf{x}), \tag{35}$$

where the Karhunen-Loeve expansion orginates from the covariance function

$$C(\mathbf{x},\mathbf{y}) := \sigma^2 \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|_p}{\lambda}\right), \quad \lambda = 0.3, \quad \sigma^2 = 1, \quad p = 1. \tag{36}$$

where $\{\theta_n\}_{n\in\mathbb{N}}$ and $\{b_n\}_{n\in\mathbb{N}}$ are the eigenvalues and normalised eigenvectors of the covariance matrix. The uncertain variables $\{\xi_n\}_{n\in\mathbb{N}}$ are a sequence of independent, uniform random variables on $[-1,1]$. In this problem, We use 21 terms in the K-L expansion ($M = 21$), which includes 84% of the total spectral energy.

This PDE is solved with finite-volumes on a uniform Cartesian grid of $m \times m$ cells. Let $k_{i,j}$ and $p_{i,j}$ denote the values of k, and p at a cell centre $\mathbf{x}_{i,j}$, $(i,j = 1,\dots,m)$. The discretization used is

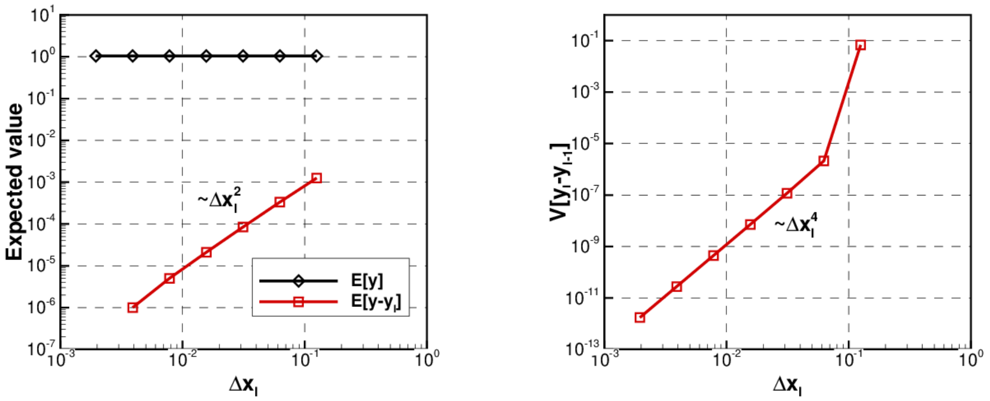
$$-\bar{k}_{i,j-\frac{1}{2}}p_{i,j-1} - \bar{k}_{i-\frac{1}{2},j}p_{i-1,j} - \bar{k}_{i+\frac{1}{2},j}p_{i+1,j} - \bar{k}_{i,j+\frac{1}{2}}p_{i,j+1} + 4p_{i,j}\hat{k}_{i,j} = 0 \tag{37}$$

where $\hat{k}_{i,j} = (\bar{k}_{i,j-\frac{1}{2}} + \bar{k}_{i-\frac{1}{2},j} + \bar{k}_{i+\frac{1}{2},j} + \bar{k}_{i,j+\frac{1}{2}})/4$. Here $\bar{k}_{i,j+\frac{1}{2}}$ is the value at the mid-point of an edge, which is approximated by the arithmetic average of $k_{i,j+1}$ and $k_{i,j}$. At Dirichlet boundaries, the derivative is approximated by a one-sided difference. At Neumann boundaries, the derivative is known explicitly, and k is approximated by $k_{i,j}$. The quantity of interest is

$$y := -\int_0^1 k\frac{\partial p}{\partial x_1}\bigg|_{x_1=1} \mathrm{d}x_2 \simeq \sum_{j=1}^{m} k\frac{\partial p}{\partial x_1}\bigg|_{m+\frac{1}{2},j}, \tag{38}$$

given all of which the discretization error is $O(\Delta x^2)$.

We choose a sequence of spatial grids with the cell size $\Delta x_l = 2^{-l}\Delta x_0$ and $\Delta x_0 = \frac{1}{8}$. Six levels are used, i.e. $L = 5$. The computational cost per sample is measured to be $C_l = C_c \Delta x_l^{\gamma} \propto \Delta x_l^{-2}$, thanks to an efficient multi-grid solver. Taking the grid with $\Delta x = 1/512$ as a reference, the discretization error is estimated using 200 samples per level and shown in Figure 2a. Clean 2nd-order convergence is observed, so $\alpha = 2$ in this case. The variance for the difference function is estimated as an 4th-order convergence rate in terms of Δx, shown in Figure 2b.



(a) Discretization error of multi-level analyses

(b) Variance of difference function

Figure 2: Performance plots of multi-level analyses for Darcy flow case

4.3 RANS flow over an airfoil with geometric uncertainties (M=10)

Finally, a more challenging test-case is considered: Reynolds Averaged Naiver Stokes (RANS) for the RAE2822 airfoil at $M_\infty = 0.734$, $\alpha = 2.79°$, and $\mathrm{Re} = 6.5 \times 10^6$. Manufacturing variability of the aerofoil surface of approximately 0.2% of the chord length is modeled by a Gaussian random fields with the correlation

$$C(\mathbf{s_1}, \mathbf{s_2}) := \sigma^2 \exp\left(-\frac{\|\mathbf{s_1} - \mathbf{s_2}\|_2}{\lambda}\right), \quad \lambda = 0.1, \tag{39}$$

where $\mathbf{s_1}$ and $\mathbf{s_1}$ are two surface nodes, distance is the standard Euclidean norm, and the standard deviation σ is assumed to be

$$\sigma = \begin{cases} (10 - 10x)^{0.7} \times 0.001 & \text{if} \quad 0.9 \le x \le 1.0, \\ 0.001 & \text{if} \quad 0.1 \le x \le 0.9, \\ (10x)^{0.7} \times 0.001 & \text{if} \quad 0 \le x \le 0.1. \end{cases} \tag{40}$$

Similar to Darcy, the Gaussian-process is parametrized by independent standard Gaussian random variables using Karhunen-Loeve:

$$G(\mathbf{s},\omega) = \sum_{k=0}^{\infty} \sqrt{\theta_k}\xi_k(\omega)b_k(\mathbf{s}). \tag{41}$$

The eigenvalues are shown in Figure 3, and the first 10 K-L modes are used to parametrize the perturbation. Figure 4 shows three realizations of the perturbation and corresponding pressure distribution of the perturbed airfoils. The CFD solver used is finite-volume and nominally 2nd-order accurate, though this case exhibits a shock reducing the ℓ^2-norm of the solution to 1st-order. The QoI used in this case the lift coefficient, and is observed to converge at slightly higher-order in practice.

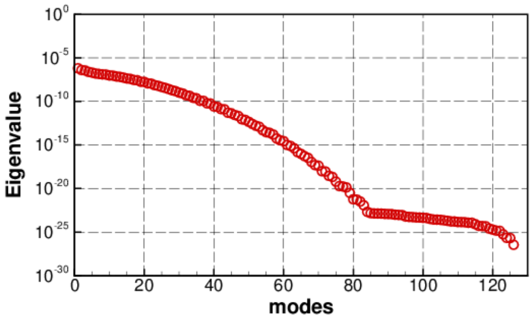
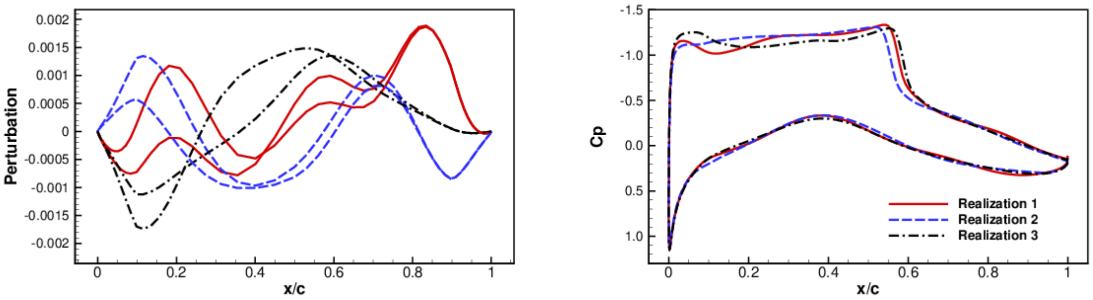


Figure 3: Eigenvalues of K-L expansion for geometric uncertainty



(a) Three realizations of K-L expansion

(b) Pressure distributions

Figure 4: Visualization of the geometric uncertainty in the airfoil surface and corresponding pressure distribution of perturbed airfoils

In this final case, we define 5 grid levels, the parameters of which are given in Table 1. The parameter of the reference computational grid is also shown in the last line. Discretization error is estimated using 100 random samples, shown in Figure 6a. Note that in the Darcy case the discretization error was estimated with respect to grid-cell size, but number of cells is used in this case for simplicity. We have the discretization error $|\mathbb{E}[y - y_l]| \leq C_d K^{\alpha} \sim K^{-1.35}$. The variance for the difference function decreases at the rate of $K^{-3.5}$ in terms of K, shown in Figure 6b.

Table 1: Grid parameters for RANS flow case

Level	# cells	I-cells	J-cells	# surface cells
0	18432	192	96	128
1	41472	288	144	192
2	73728	384	192	256
3	112896	504	224	336
4	165888	576	288	384
Ref	209952	648	324	432

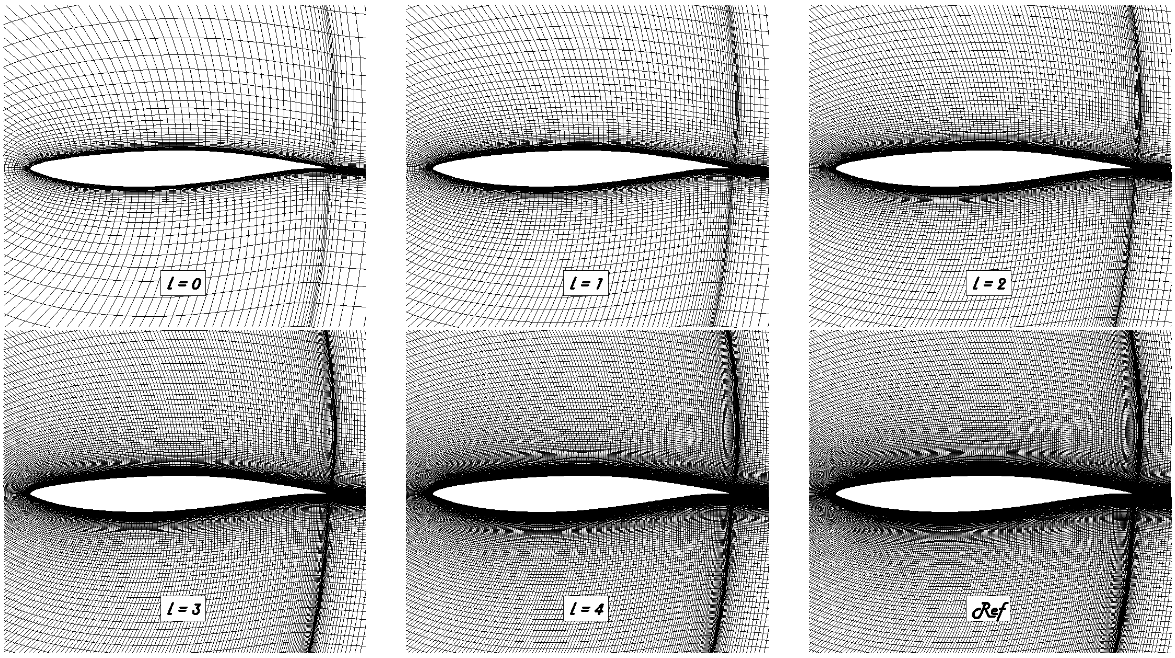
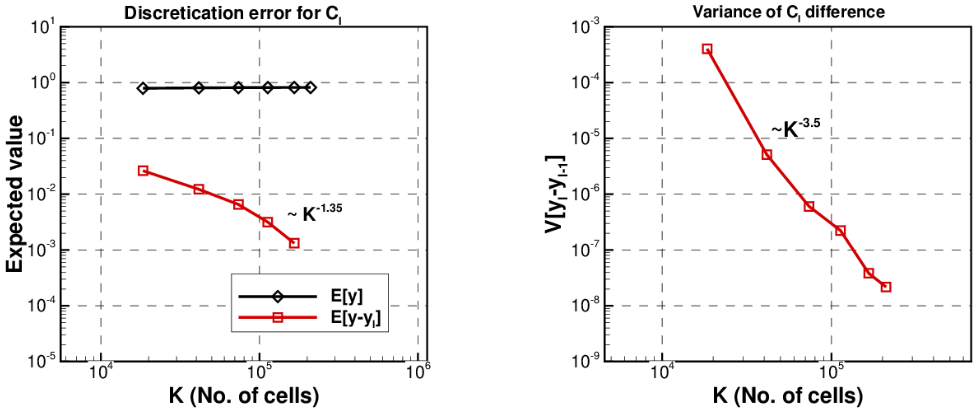


Figure 5: Multi-level computational grids for RANS flow case



(a) Discretization error of multi-level analyses

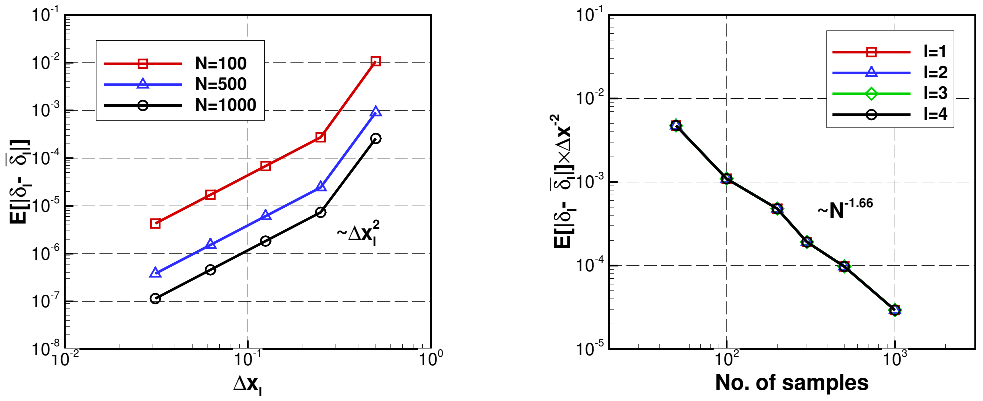
(b) Variance of difference function

Figure 6: Performance plots of multi-level analyses for RANS flow case

5 RESULTS

5.1 2D ”Oscillatory” Genz funcion

First, the unknown constants in the model of interpolation error are estimated. Halton sample sequence is used to generate uniformly-distributed random points in the parameter space. The estimated interpolation error is shown in Figure 7. The left figure shows us that the magnitude of the error decreases at a second order ($\mu = 2$), and in the right figure, it can be seen that the interpolation error from different levels collapses well to a single line when scaled by Δx^2. The parameters estimated on the basis of these plots are given in Table 2. These figures confirms our assumptions in the convergence properties of interpolation error. Following Algorithm 1: given accuracy requirement ε, L is determined by limiting the discretization error to $\varepsilon/2$. The computational cost per sample is assumed as $T_l = C_c \Delta x_l^{\gamma} \sim \Delta x_l^{-1}$. Then the optimal number of sampling points on each level are determined using (21), the result of which is shown in Figure 8a. For comparison, Monte-Carlo (MC) and Multilevel Monte-Carlo (MLMC) methods are also applied to this case. For MC the number of points required for a certain accuracy is simply $N = \sigma^2/\varepsilon^2$. For MLMC, the optimal points per level [17] is given by



(a) Interpolation error versus grid size (b) Interpolation error versus No. of samples

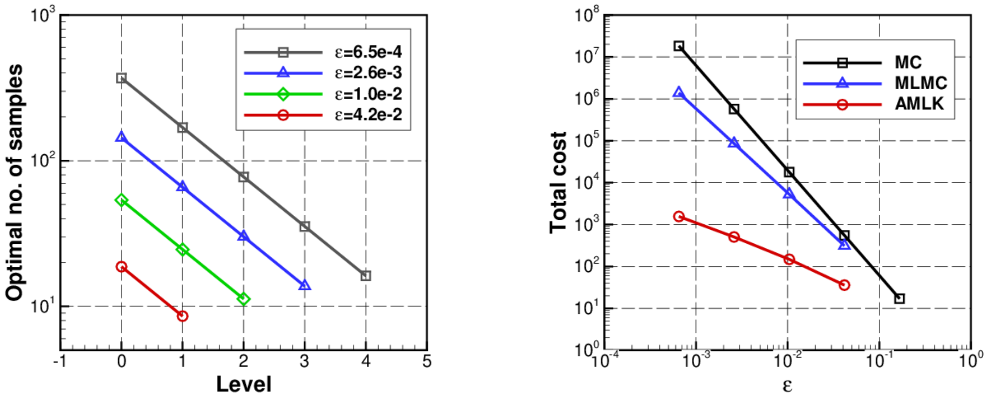
Figure 7: Approximation for interpolation error of AMLK for Genz case

Table 2: Estimated parameters for Genz case

parameters	μ	$\nu\beta$	$C_I C_s^{\beta}$
Estimated value	2.0	-1.66	11.261

$$N_l = \varepsilon^{-2}(\sum_{l=0}^{L} \sqrt{V_l C_l}) \sqrt{\frac{V_l}{C_l}},$$

and a comparison of total cost against MLMC and MC for this problem is shown in Figure 8b. In this paper, standardized costs are presented always, which is scaled by the cost oper sample on the coarsest level. From literature [17], the total computational costs of MC and MLMC should be proportional to $\varepsilon^{-2+\gamma/\alpha}$ and ε^{-2}, if the variance $V[y_l - y_{l-1}]$ decays faster than the increase of T_l. We observe that the computational costs of MC and MLMC method grow along with the improvement of accuracy at the rate of $\varepsilon^{-2.5}$ and ε^{-2}, respectively, which agrees with the theoretical result. In this case, we find μ is larger than $\gamma\nu\beta$, so that the limit of the convergence of cost versus ε should be $\varepsilon^{-1/1.66}$, according to (28). In Figure 8b, the cost of AMLK model increases as $\varepsilon^{-1/1.1}$, which is much slower than other methods, but faster than the theoretical value. The reason is that the lower-order error terms in (26) is also influential. With the increase of the required accuracy, more benefit can be gained by AMLK model.



(a) Optimal No. of points of AMLK

(b) Computational cost versus error

Figure 8: Performance plots of AMLK for Genz case

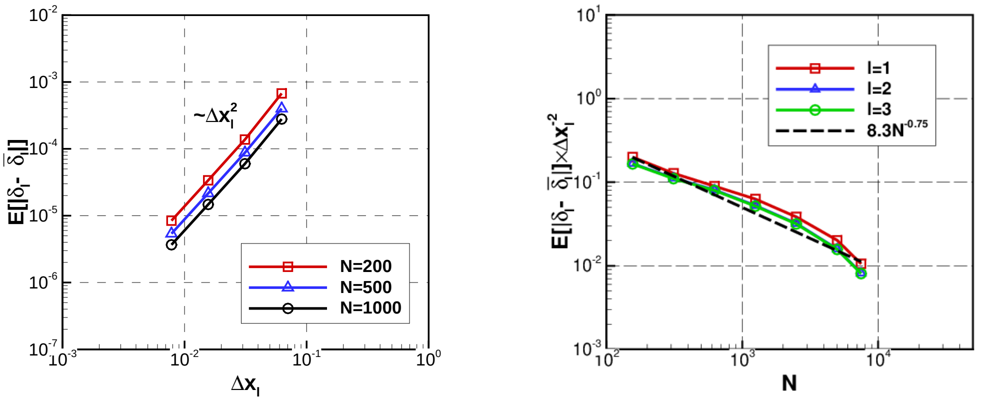
5.2 Darcy flow with random coefficients

Halton sequences are also used in this case. The estimated interpolation error is shown in Figure 9. These results confirms our assumptions again, but with different estimated parameters, shown in Table 3. Based on the estimated parameters, the optimal number of sampling points on each level are shown in Figure 10a, and the total cost are shown in Figure 10b, as well as that of MLMC and MC. The results show that the total costs of MC and MLMC achieve an error of $O(\varepsilon)$ is ε^{-3} and ε^{-2}. Same as the first case, Figure 10b also indicates the cost of AMLK method grows as $\varepsilon^{-1/0.61}$, which is a bit faster than theoretical value $\varepsilon^{-1/0.75}$, but is slower than MC and MLMC methods.

MC is seen to be completely impractical for very low ε but highly comparative for high errors (especially given its simplicity and use of a single grid). Despite MLMC achieving optimal rates, at low ε it is soundly beaten by AMLK. This result is not surprising, as similar performance was observed for polynomial surrogates in [18]. It does however rely on the regularity of the underlying response $y(\boldsymbol{\xi})$, which the MC-based techniques do not. It is therefore instructive to proceed to a case where regularity is not guaranteed, see next section.

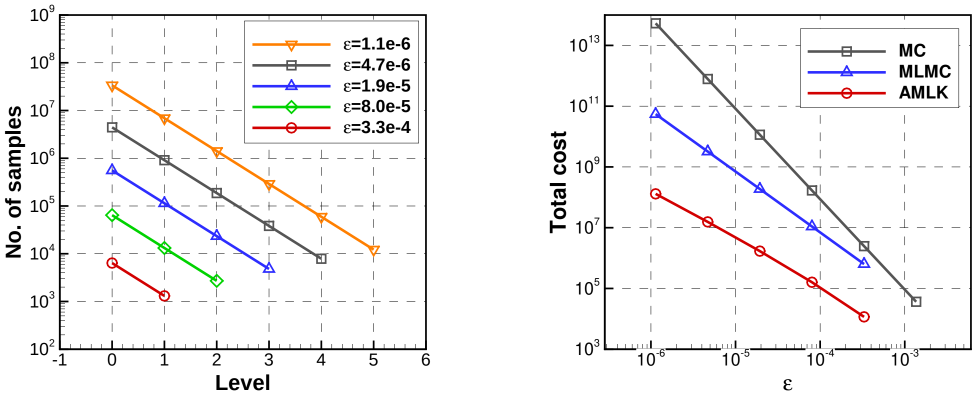
Table 3: Estimated parameters for the Darcy case

Parameter	μ	$\nu\beta$	$C_I C_s^\beta$
Estimated value	2.0	-0.75	8.3



(a) Interpolation error versus grid size (b) Interpolation error versus No. of samples

Figure 9: Approximation for interpolation error of AMLK for Darcy flow case



(a) Optimal No. of points of AMLK (b) Computational cost versus error

Figure 10: Performance plots of AMLK for Darcy flow case

5.3 RANS flow over an airfoil with geometry uncertainties

In this final case, the interpolation error of the AMLK is assumed to be $C_I K^{\mu} h_l^{\beta}$, again independent of level. Similarly to the first case, actually we estimate

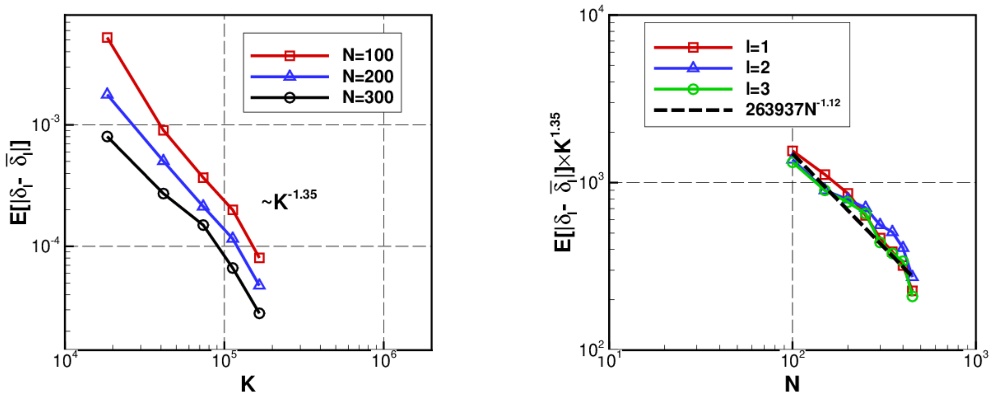
$$C_I M_l^{\mu} h_l^{\beta} = C_I K_l^{\mu} (C_s N_l^{\nu})^{\beta} = C_I C_s^{\beta} K_l^{\mu} N_l^{\nu\beta}.$$

The normally-distributed random samples are obtained by transferring the Halton samples based on the probability integral property. The estimated interpolation error with respect to the number of grid-cells and sampling points are shown in Figure 11 and the estimated parameters is present in Table 4. Once more, the convergence of the interpolation is seen to be independent of the grid level (under appropriate scaling), justifying the choice of level-independent parameters in (10). The computational cost per sample on each level is estimated, shown in Figure 12, which shows that $T_l = C_c K_l^{\gamma} \sim K^{0.93}$.

From Figure 6a, we found that the discretization error on finest level is 0.00132, which indicates that the grid is sufficiently accurate to resolve the lift coefficient to around 0.1 count, which already meets the engineering requirement. However, the observed interpolation errors, whose magnitude ranges from $10^{-4} - 10^{-5}$, are much smaller than the discretization error. It is

Table 4: Estimated parameters for RANS flow case

parameters	μ	$\nu\beta$	$C_I C_s^\beta$
Estimated value	-1.35	-1.12	2.639×10^5



(a) Interpolation error versus grid size (b) Interpolation error versus No. of samples

Figure 11: Approximation for interpolation error of AMLK for RANS flow case

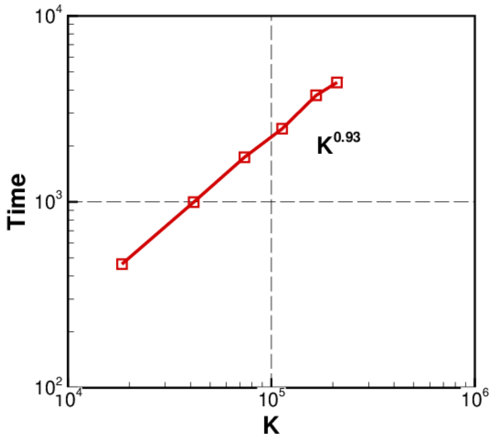
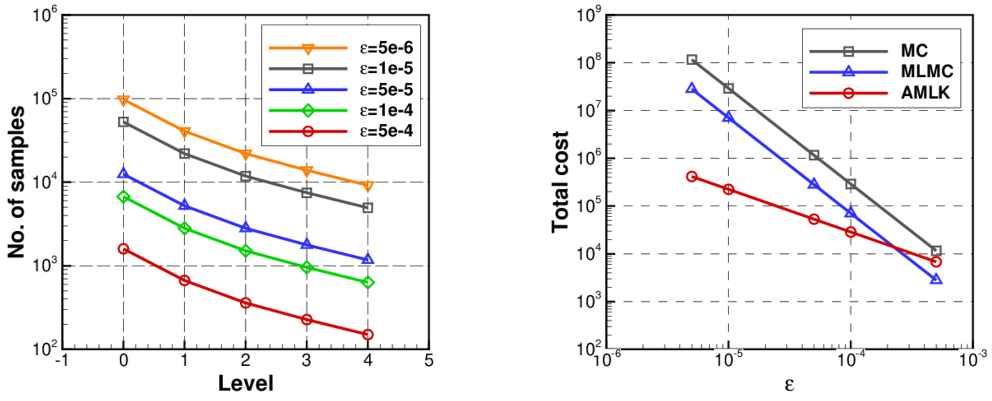


Figure 12: Computational time versus number of grid cells for RANS flow case

impractical to bound the discretization error and interpolation error equally. Therefore, in this case, we fix the discretization error on finest level - 5 levels involved, and estimate the optimal computational cost required to achieving a certain interpolation accuracy.

Based on the estimated parameters, the optimal number of sampling points on each level are shown in Figure 13a, and the total cost are shown in Figure 13b. Both the total costs of MC and MLMC grows at the rate of ε^{-2}, which is consistent with the theoretical result when the involved multi-level analyses are fixed. Meanwhile, as the multi-level analyses are fixed, the cost of AMLK method grows as $\varepsilon^{-1/1.12}$ exactly. In this practical application case, the same convergence property of AMLK model is observed, which can save much computational cost than MC and MLMC methods.



(a) No. of sampling points of AMLK

(b) Computational cost versus error

Figure 13: Performance plots of AMLK for RANS case

6 CONCLUSIONS

In this work, we performed theoretical convergence and cost analyses on the AMLK model, utilizing convergence results for single-fidelity Kriging, as well as existing multi-level analysis of stochastic collocation method. Three numerical test cases with different number of uncertainty variables were utilized to demonstrate the effectiveness of proposed method. All the numerical results verified the assumptions for the mathematical form of discretization error and interpolation error. The comparisons of total computational cost showed that using multi-level kriging model for UQ can significantly reduce the cost, compared with the MLMC and standard MC method.

In this study, only the thin-plate spline was considered as the covariance kernel used in kriging model. As we mentioned before, the convergence property of a kriging model is only dependent on the smoothness of the covariance kernel. For finitely differentiable kernels, the convergence and cost analyses results are analogous to this study. However, for infinitely differentiable kernels, as it shows spectral convergence, the form of interpolation error can be assumed as $C_I \Delta x_l^{\mu} e^{-\tilde{c}/h_l}$ and the convergence study could be conducted accordingly.

On the other hand, the theoretical convergence rate of kriging model was given with respect to the fill distance h. However, it is very difficult to estimate the h, especially for high-dimensional stochastic space. We defined $h = C_s N^{\nu}$ for a specific sampling method and transferred the interpolation error in terms of N instead. In this way, the sampling method is also essential for the convergence study of a surrogate model. In this study, we used the Halton pseudo-random samples for convergence study, which are deterministic, of low discrepancy but appear randomly. To estimate the interpolation error, we generated a series of sample data set with increasing number of points and ensured that a smaller-size data set is always a subset of a larger-size data set, such that the convergence study of interpolation error is consistent and smooth. Nevertheless, it was still difficult to obtain a smooth estimation for the interpolation error in $\mathbb{E}[y]$. In fact, the error in $\mathbb{E}[y]$ is not equivalent to any norm of the point-wise error. We used the mean of the ℓ^1-norm of the point-wise error to bound the error in $\mathbb{E}[y]$. In Figure 7, 9 and 11, we can find that smooth estimations for interpolation error are obtained.

For the RANS flow case with geometric uncertainty, we chose the lift coefficient as quantity of interest and gained good estimation of discretization error. As for the drag coefficient, because of the existences of a strong shock, it was difficult to get a linear convergence of discretization error. One of the largest difficulties we met in this case is that the magnitude of the disretization error was much larger than that of the interpolation error. Thus, we could not bound the two error terms equally, as we did in other two cases. However, even the accuracy on the finest level meets the requirement of engineering application, so it is not necessary to further improve the resolution of computation grid. Therefore, we fixed the finest level, and estimated the minimal total computational cost needed in order to achieve a certain interpolation error.

Besides the additive-bridge function based multi-level kriging model, there are potential models, such as multi-level hierarchical kriging model and cokriging. In future work, the convergence analysis of these two models will be studied comparatively.

REFERENCES

[1] R. T. Haftka, Combining global and local approximations. *AIAA Journal*, **29**(9), 1523–1525, 1991.

[2] K. J. Chang, R. T. Haftka, G. L. Giles, P. J. Kao, Sensitivity-based scaling for approximation structural response. *Journal of aircraft*, **30**(2), 283–288, 1993.

[3] S. E. Gano, J. E. Renaud, B. Sanders, Hybrid variable fidelity optimization by using a kriging-based scaling function. *AIAA Journal*, **43**(11), 2422–2430, 2005.

[4] Z. H. Han, S. Göertz, R. Zimmermann, Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerospace Science and Technology*, **25**, 177–189, 2013.

[5] A. G. Journel, J. C. Huijbregts, Mining geostatistics. *Academic Press*, New York, 1978.

[6] M. C. Kennedy, A. OHagan, Predicting the output from a complex computer code when fast approximations are available. *Biometrika*, **87**(1), 1–13, 2010.

[7] Z. H. Han, R. Zimmermann, S. Göertz, An alternative cokriging model for variable-fidelity surrogate modeling. *AIAA Journal*, **50**(5), 1205–1210, 2012.

[8] Z. H. Han, S. Göertz, Hierarchical kriging model for variable-fidelity surrogate modeling. *AIAA Journal*, **50**(5), 1285–1296, 2012.

[9] P. S. Palar, T. Tsuchiya, G. T. Parks, Multi-fidelity non-intrusive polynomial chaos based on regression. *Computer Methods in Applied Mechanics and Engineering*, **305**, 579–606, 2016.

[10] L. Parussini, D. Venturi, P. Perdikaris, G. E. Karniadakis, Multi-fidelity Gaussian process regression for prediction of random fields. *Journal of Computational Physics*, **336**, 36–50, 2017.

[11] P. S. Palar, K. Shimoyama, Multi-Fidelity Uncertainty Analysis in CFD Using Hierarchical Kriging. *35th AIAA Applied Aerodynamics Conference*, Denver, Colorado, June 5-9, 2017.

[12] A. Narayan, C. Gittelson, D. Xiu, A stochastic collocation algorithm with multi-fidelity models. *SIAM Journal On Scientific Computing*, **36**(2), 495–521, 2014.

[13] X. Zhu, E. M.Linebarger, D. Xiu, Multi-fidelity stochastic collocation method for prediction of statistical moments. *Journal of Computational Physics*, **341**, 386–396, 2017.

[14] G. Geraci, M. S. Eldred, G. Iaccarino, A multifidelity multilevel Monte Carlo method for uncertainty propagation in aerospace applications. *19th AIAA Non-Deterministic Approaches Conference*, Grapevine, Texas, 2017.

[15] M. B. Giles, Multi-level Monte Carlo path simulation. *Operations Research*, **56**(3), 607–617, 2008.

[16] S. Heinrich, Multi-level Monte Carlo Methods. *Lecture Notes in Computer Science*, **2179**, 3624–3651, Springer, Berlin, Heidelberg, 2001.

[17] K. A. Cliffe, M. B. Giles, R. Scheichl, A. L. Teckentrup, Multi-level Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, **14**, 3–15, 2011.

[18] A. L. Teckentrup, P. Jantsch, C. G. Webster, M. Gunzburger, A multilevel stochastic collocation method for partial differential equations with random input data. *SIAM/ASA Journal on Uncertainty Quantification*, **3**, 1046–1074, 2015.

[19] S. Choi, J. J. Alonso, I. M. Kroo, M. Wintzer, Multi-fidelity design optimization of low-boom supersonic business jets. *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, 2004.

[20] P. Wang ; Y. Li ; C. Li An Optimization Framework Based on Kriging Method with Additive Bridge Function for Variable-Fidelity Problem. *14th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, 388–392, IEEE, 2015.

[21] C. Tang, K. Gee, S. Lawrence, Generation of Aerodynamic Data using a Design Of Experiment and Data Fusion Approach. *43rd AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, 2005.

[22] H. Wendland, *Scattered data approximation*, **17**, Cambridge university press, 2004.

[23] D. Zimmerman, C. Pavlik, A. Ruggles, M. P. Armstrong, An experimental comparison of ordinary and universal kriging and inverse distance weighting. *Mathematical Geology*, **31**, 375–390, 1999.

[24] A. Genz, Testing multidimensional integration routines. *Proc. of international conference on Tools, methods and languages for scientific and engineering computation*, B. Ford, J. C. Rault, and F. Thomasset (Eds.). Elsevier North-Holland, Inc., New York, 81–94, 1984.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

MULTI-FIDELITY UNCERTAINTY QUANTIFICATION OF THE FLOW AROUND A RECTANGULAR 5:1 CYLINDER

M. Sakuma[1], N. Pepper[2], A. Kodakkal[1], R. Wüchner[1], K-U Bletzinger[1] and F. Montomoli[2]

[1]Technical University of Munich
Arcistraße 21, 80333 München, Germany
{mayu.sakuma, anoop.kodakkal, wuechner, kub}@tum.de

[2] Imperial College London
London, England SW7 2AZ, United Kingdom
{nicholas.pepper16, f.montomoli}@imperial.ac.uk

Keywords: Computational Fluid Dynamics, Non-Intrusive Polynomial Chaos, Wind Engineering, Multi-fidelity

Abstract. *This work shows the application of Multi-fidelity Uncertainty Quantification to Wind Engineering problems. As test case a rectangular shape is used, with a fillet radius, in order to represent the geometrical variations that can affect buildings or other bluff bodies. The rectangular cylinder used has a chord-to-thickness ratio 5:1. This rectangular shape is an important basic shape for wind engineering tasks, e.g. in case of buildings or other bluff bodies exposed to the flow. Moreover it is well investigated and documented.*

Coarse and fine meshes are used as low and high fidelity models respectively. To perform CFD simulations, the stabilized finite element methods are used in both the high and low fidelity model with a CFD code developed by TUM and the International Center for Numerical Methods in Engineering. The underlying UQ framework is based on a Sparse Arbitrary Moment Based Algorithm (SAMBA) developed at ICL. In the formulation the number of simulations is reduced using a Smolyak sparsity model.

The multi-fidelity extension, with application to wind engineering problems is discussed and presented in this work. The overall goal of such formulation is to gain an accuracy of mixed low-high fidelity simulations comparable to the ones obtained with only high fidelity simulations, at a fraction of the computational cost.

1 Introduction

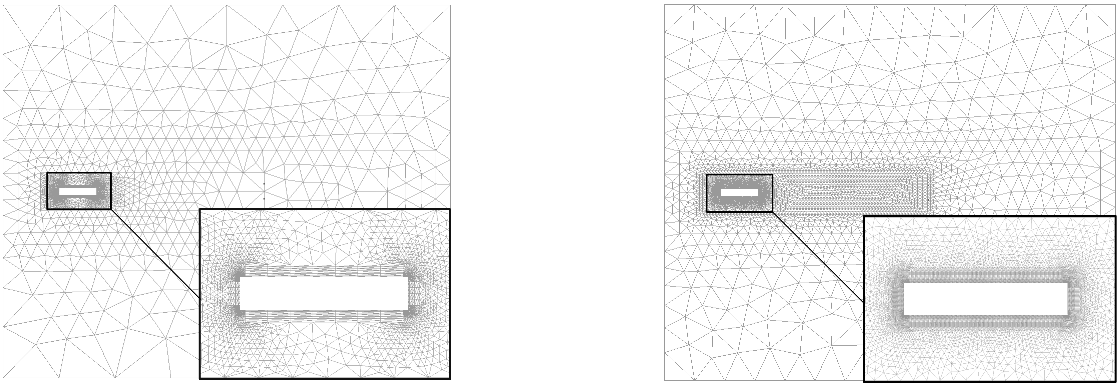
For wind engineering problems, carrying out wind tunnel experiments is expensive. Therefore it has been sought to use Computational Fluid Dynamics (CFD) simulations instead of wind tunnel tests ([12]) for design process and in this purpose it is required to confirm reliability of simulations. Verification and Validation are known as procedures to confirm the reliability of simulations. UQ can be used for the validation, especially for decision making whether results of physical experiment (or observed field data) and computational models are matching or not ([10]). A difficulty of UQ for CFD simulations is that UQ requires running a deterministic simulation several times, while a deterministic simulation of CFD is already computationally expensive. For example, Monte Carlo methods require running thousands of deterministic simulations but one CFD simulation often takes several hours or even days. To overcome this problem, the spectral methods have been used for UQ of CFD simulations ([9]). There are several methodologies of the spectral methods([6]) and in this contribution the Non-Intrusive Polynomial Chaos (NIPC) method is used. NIPC does not require rewriting existing deterministic simulation codes. In order to further reduce the total computational cost of NIPC, a hierarchy of model fidelity is considered ([5]). For computational simulations, there are often several possible model selections and each model has a different accuracy and computational cost. Let us focus on model sets, which have clear accuracy and computational cost hierarchy. For example, considering a CFD model with a fine mesh and a coarse mesh. The fine mesh gives more accurate result than the coarse mesh but the computational cost is higher with the fine mesh than with the coarse mesh. In this case the model with the fine mesh is the high fidelity model and the model with the coarse mesh is the low fidelity model. In this contribution, it is sought to get stochastic results which are as accurate as the results computed only by the high fidelity models using a combination of the high and low fidelity models. By using not only the high fidelity model but also the low fidelity model, the total computational time can be reduced.

As a target CFD simulation, flow around a rectangular cylinder is investigated. The bluff body shape is in high interest in civil engineering (e.g. bridge decks and high-rise buildings) and in other engineering fields and well studied and documented.[2]

2 Flow around a rectangular cylinder

In this contribution flow around a rectangular cylinder with ratio of the chord length (Breadth) to the thickness (Depth) $B/D = 5$ at Reynolds number 400 is investigated. The Reynolds number is computed based on the thickness of the rectangular cylinder. The CFD simulations are solved by the software KratosMultiphysics which is developed by the Technical University of Munich and the International Center for Numerical Methods in Engineering. It solves CFD by the stabilized Finite Element Method and for stabilization, the variational multiscale method (VMS) is used.[3] The CFD is solved in 2D, since it is known from [13] that the flow is two dimensional at low Reynolds number as 400.

For performing multifidelity NIPC, a coarse mesh and a fine mesh is created. The meshes are shown in Fig. 1. The coarse mesh has 7956 cells and 4133 nodes, and the fine mesh has 22313 cells and 11451 nodes. As time discretization, the Bossak time integration method, which is one of the generalized α Newmark methods, is used. Time steps are determined by performing convergence study, $dt = 3.0$[s] for the coarse mesh and $dt = 1.5$[s]. The simulations are carried out until $30,000$[s] and time statistics are calculated by results after $15,000$[s]. About boundary conditions, the inflow velocity is applied to left wall, no-slip condition at the cylinder surface


(a) The coarse mesh

(b) The fine mesh

Figure 1: Details of the fine and coarse meshes used

CFD model	$t-avr(C_d)$	$t-std(C_l)$	S_t
Hourigan et al., 2001	1.12	0.216	0.55
Coarse mesh	1.03	0.0724	0.75
Fine mesh	1.00	0.206	0.56

Table 1: Comparison of time statistics of the aerodynamics coefficient and Strouhal number

and slip condition at far-field are imposed.

The resulted time statistics of aerodynamic coefficient C_d, C_l and the Strouhal number S_t are compared with ones in [8]. In [8], the flow at $R_e = 400$ is solved by the finite element method. C_d, C_l are calculated as:

$$C_d = \frac{2F_d}{\rho {U_{inlet}}^2 D} \qquad\qquad C_l = \frac{2F_l}{\rho {U_{inlet}}^2 B}$$

where F_d, F_l are the drag and lift force subjected to the structure, $\rho = 1.225kg/m^3$ is the density of the air, U_{inlet} is the applied inlet velocity, B is the chord length and D is the thickness of the structure. The Strouhal number is calculated as $S_t = fB/U_{inlet}$, where f is the frequency of the lift force coefficient. In Table 1 the time averaged C_d (: $t-avr(C_d)$) and time standard deviation of C_l (: $t-std(C_l)$) are compared. It can be seen that the fine mesh gets very close values as [8]. The coarse mesh also got very close value of C_d with the one from the literature and the fine mesh, on the other hand there is more difference in C_l and S_t value between results of the fine- and coarse mesh.

Angle of attack and the curvature at every corner of the structure are considered as uncertain input parameters since they are often uncertain in wind tunnel tests[2]. The curvature is illustrated in Fig. 2 and a same value is applied to every corner. It is assumed that the angle of attack has a normal distribution with mean $0.0°$ and standard deviation $3.0°$, and the curvature has a half normal distribution with the location parameter 0.0 and the scale parameter 0.05.

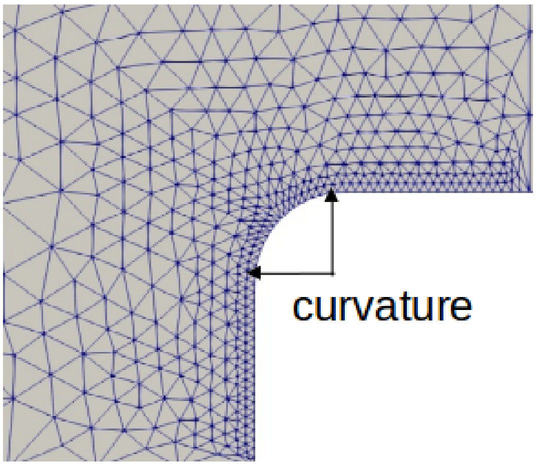


Figure 2: Definition of the curvature

3 The Non-Intrusive Polynomial Chaos methods(NIPC)

Let us consider a stochastic CFD problem. The unknown Quantity of Interest (QoI) Y is expressed as follows.

$$Y = g(\mathbf{X}) \tag{1}$$

where $\mathbf{X} = [X_1, X_2, ..., X_d]$ is the input random variables and g is the CFD simulation. Here the time averaged drag coefficient $t - avr(C_d)$ and the time standard deviation of lift coefficient $t - std(C_l)$ are considered as QoI, and the angle of attack α and the curvature R are considered as input random variables, that is $\mathbf{X} = [X_1, X_2]$.

3.1 The polynomial chaos expansion

Assuming that Y is second order ($Y \in L^2(s)$) i.e. $E[Y^2] < \infty$, Y can be expressed with orthogonal polynomials as follows.

$$Y = \sum_{k \in \mathbb{N}^d} a_k \Psi_k(\mathbf{X}) \tag{2}$$

where $\Psi_k(\mathbf{X})$ is linear combination of multivariate orthogonal polynomials. Let us think about $L(S_i)_p$, the p-th order finite dimensional subspace of $L(S)$. The p-th order polynomial chaos approximation of Y is expressed as:

$$Y \approx Y_p = \sum_{k=0}^{P-1} a_k \Psi_k(\mathbf{X}) \tag{3}$$

The total number of term P in the expansion is calculated as:

$$P = \sum_{q=0}^{p} \frac{(q+d-1)!}{q!(d-1)!} = \frac{(p+d)!}{p!d!} \tag{4}$$

By applying the projection theorem, the deterministic coefficient a_k in Eq.3 is computed as:

$$a_k = E[g(\mathbf{X})\Psi_k(\mathbf{X})] \tag{5}$$

Assuming that the Probabilistic Density Function(PDF) of $\mathbf{X}$ is $f(\mathbf{X})$, Eq.5 is computed by:

$$a_k = \int_{\mathbb{R}^d} g(\mathbf{X})\Psi_k(\mathbf{X})f(\mathbf{X})d\mathbf{X} \tag{6}$$

Applying the Gaussian quadrature rule to Eq.6, a_k is approximated as:

$$a_k \approx \sum_{i_1=1}^{q_1} ... \sum_{i_d=1}^{q_d} g(X_{i_1}, ..., X_{i_d})\Psi(X_{i_1}, ..., X_{i_d})w_{i_1}...w_{i_d} \tag{7}$$

where $q_k, k \in d$ is a number of quadrature points for each univariate quadrature rule, $w_{i_k}, k \in d$ is the weight of each univariate quadrature. Here the number of input random variable $d = 2$, therefore the Eq. 7 is written as:

$$a_k \approx \sum_{i_1=1}^{q_1} \sum_{i_2=1}^{q_2} g(X_{1,i_1}, X_{2,i_2})\Psi(X_{1,i_1}, X_{2,i_2})w_{i_1}w_{i_2} \tag{8}$$

CFD simulations are calculated at each quadrature points and the quadrature points are called as collocation points.

3.2 Sparse Approximation of Moment-Based Arbitrary(SAMBA) Polynomial Chaos

In this contribution, the Sparse Approximation of Moment-Based Arbitrary(SAMBA) Polynomial Chaos is used to compute the orthogonal polynomials and coefficients of the polynomial chaos expansion in Eq. 3. The orthogonal polynomials are known for some classical distribution type of PDFs, for example, the Legendre polynomials for the uniform distribution and the Hermite polynomials for the normal distribution. SAMBA [1] can compute the orthogonal polynomials and Gaussian quadrature points and weights from a moment matrices of input random variables using the theory described in [7], while often the orthogonal polynomials are determined by Askey scheme. SAMBA makes it possible to perform NIPC for any kind of PDFs of input random distributions. In addition, as you can see in Eq. 7, computational cost increases with increase of the dimension of random variables, which is called as *curse of dimensionality*. To overcome the curse of dimensionality, the Smolyak formula is adapted for multiple Gaussian quadrature rules in SAMBA. Fig. 3 shows collocation points for sparse grid level 1 to 3 calculated by SAMBA.

3.3 The multifidelity extension of NIPC

The multifidelity extension of NIPC is introduced in [4] and the additive correction is used here. The idea is that by calculating the high fidelity model with lower sparse grid level and the low fidelity model with higher sparse grid level, the total calculation cost is reduced to get as good accuracy as stochastic results evaluated only by high fidelity model. Let $g_{high}(\mathbf{X})$, $g_{low}(\mathbf{X})$ as system responses obtained by evaluating high- and low fidelity model respectively and an additive correction as $\delta(\mathbf{X}) = g_{high}(\mathbf{X}) - g_{low}(\mathbf{X})$. $g_{high}(\mathbf{X})$ is approximated as:

$$g_{high}(\mathbf{X}) \approx S_{q,d}[g_{high}](\mathbf{X}) \tag{9}$$

where $S_{q,d}[g]$ is the non-intrusive polynomial chaos expansion with the sparse grid level q and the dimension of input random variables d. Then approximation of $S_{q,d}[g_{high}]$ by multifidelity expansion can be written as:

$$S_{q,d}[g_{high}](\mathbf{X}) \approx \tilde{g}_{high}(\mathbf{X}) = S_{q,d}[g_{low}](\mathbf{X}) + S_{q-r,d}[\delta](\mathbf{X}) \tag{10}$$

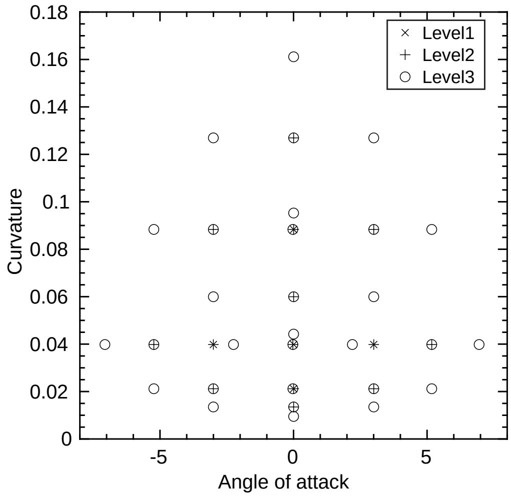


Figure 3: The collocation points for the sparse grid level 1 to 3

where r is a sparse level offset and $r \leq q$. Substituting Eq.3 to Eq. 10, we get:

$$\tilde{g}_{high}(\mathbf{X}) = \sum_{\mathbf{i} \in J_{q,d}} a_{low\mathbf{i}} \Psi_i(\mathbf{X}) + \sum_{\mathbf{i} \in J_{q-r,d}} a_{\delta \mathbf{i}} \Psi_i(\mathbf{X}) \tag{11}$$

where $J_{q,d}$ is a set of multi-indices of the d-dimensional polynomial chaos expansion bases at the level q. The polynomial chaos coefficients of Eq. 11 is calculated as follows.

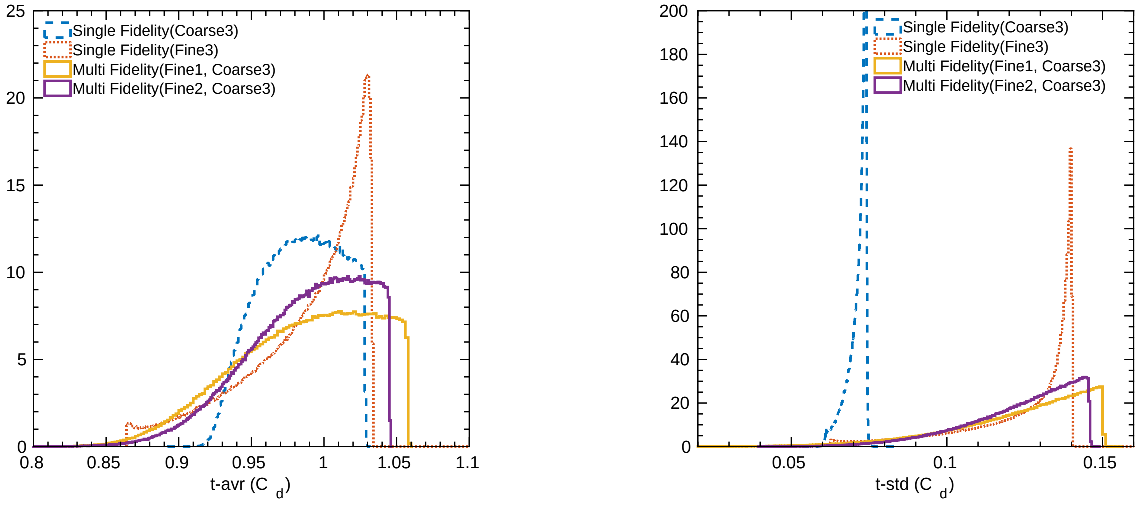
$$a_{low\mathbf{i}} = \int_{\mathbb{R}^D} g_{low}(\mathbf{X}) \Psi_k(\mathbf{X}) f(\mathbf{X}) d\mathbf{X} \tag{12}$$

$$a_{\delta \mathbf{i}} = \int_{\mathbb{R}^D} \{g_{high}(\mathbf{X}) - g_{low}(\mathbf{X})\} \Psi_k(\mathbf{X}) f(\mathbf{X}) d\mathbf{X} \tag{13}$$

Considering that CFD simulations contribute most of computational time and these integrals are estimated by the Gauss quadrature rule with the Smolyak sparse grid method, it is important for saving total computational time that the collocation points are nested. In this contribution, the coarse mesh model and fine mesh model are considered as the low and high fidelity model respectively.

4 Results

Fig.4 shows PDF outlines of $t - avr(C_d)$ and $t - std(C_l)$ computed by single fidelity model of the coarse mesh with the level 3 and the fine mesh with the level 3, and the multifidelity model of the coarse mesh with the level 3 and the additive correction with the level 1 and level 2 using Eq.11. As can be seen in Fig.4(a) the shape of the PDF of $t - avr(C_d)$ computed by the coarse mesh model is different from the one of the fine mesh model. Even with the additive correction, the shape of PDF is not exactly same as the one of the fine mesh model but by using additive correction, the shape of PDF gets closer to the one computed by the fine mesh model



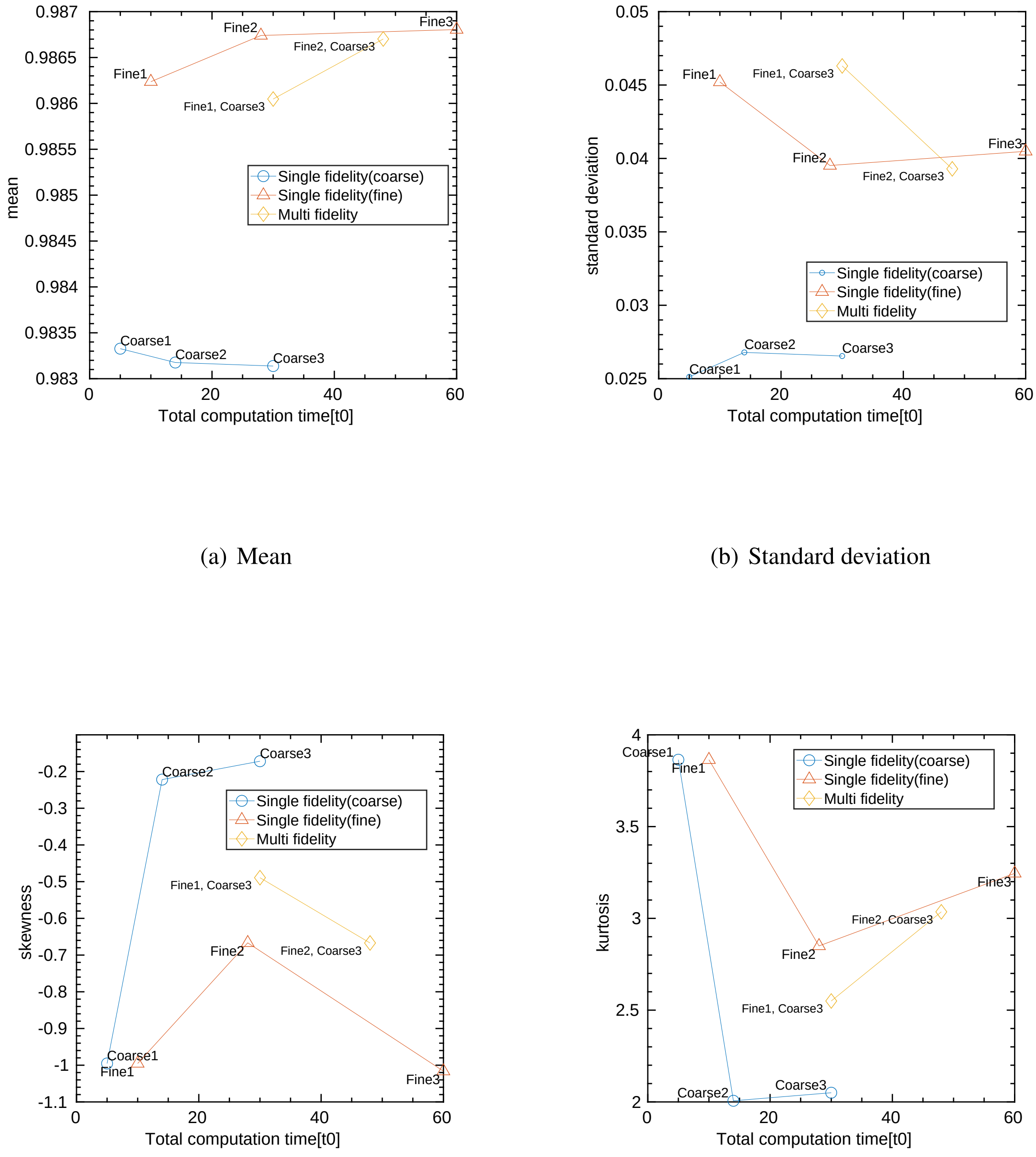
(a) $t - avr(C_d)$

(b) $t - std(C_l)$

Figure 4: The PDF outlines of (a) $t - avr(C_d)$ and (b) $t - std(C_l)$ computed by the single fidelity model (the coarse mesh level 3, the fine mesh level3) and the Multi-fidelity model (the coarse mesh level, the fine mesh level) = (3,1),(3,2))

only. In Fig.4(b) the locations of the PDFs of $t - std(C_l)$ are different between the ones of the coarse mesh with level 3 and the fine mesh with level 3. By applying the additive correction, location and shape of PDF gets similar as the PDF evaluated by the fine mesh only. In the case of both $t - avr(C_d)$ and $t - std(C_l)$, it can be confirmed that by increasing the level of the additive correction, the shape of PDF tends to converge to the shape of the PDF computed by the fine mesh model only. On the other hand, it can be also seen that the multifidelity model is not accurate especially at the upper end of the domain.
Figure 5 and 6 show the statistic moments, which are mean, standard deviation, skewness and kurtosis, computed by the single fidelity model of the coarse mesh with the level 1 to level 3 and the fine mesh with the level 1 to level 3 respectively, and the multifidelity model of the coarse mesh with level 3 and the additive correction level 1 and level 2. The horizontal axis is the total calculation time and the unit t_0 is the calculation time of a coarse mesh deterministic simulation. The ratio of calculation time of the fine mesh deterministic simulation to the one of the coarse mesh deterministic simulation is 2, which is determined by averaging actual calculation time of deterministic simulations. It can be seen that except skewness and kurtosis of $t - std(C_l)$, the statistic moments computed by the fine mesh model and the multifidelity model converge to almost same value, while the ones of the coarse mesh converges to different value. Comparing total calculation time of the single fidelity model of the fine mesh and the multifidelity model, the convergence speed is faster by the single fidelity model of the fine mesh than by the multifidelity model. It happens because in this case the computational time ratio of the fine mesh to the coarse mesh is not very high.

For comparing deterministic results of the coarse mesh and the fine mesh, Table 2 shows the correlation and the mean absolute relative error, which are calculated from 35 deterministic



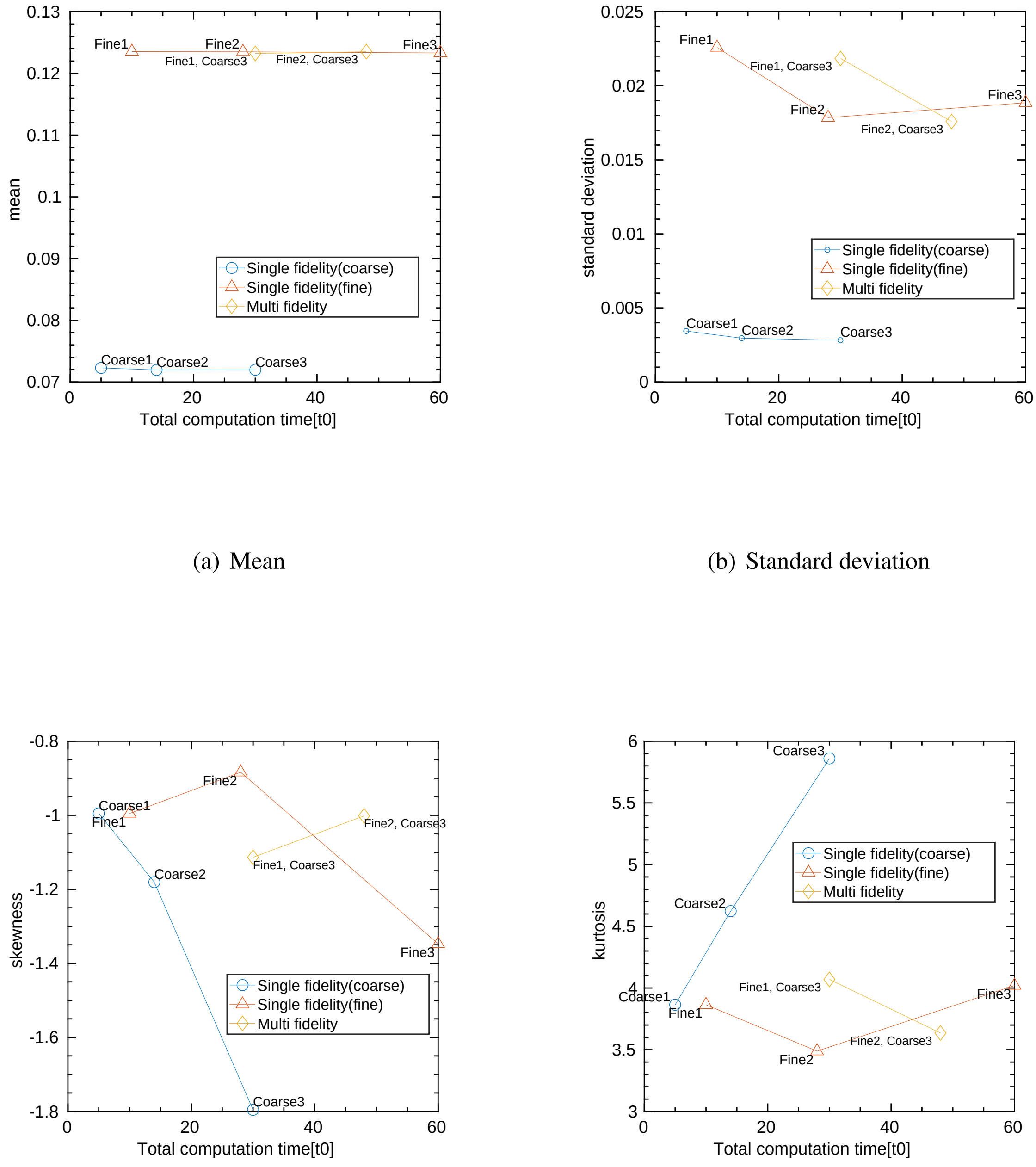
(a) Mean

(b) Standard deviation

(c) Skewness

(d) Kurtosis

Figure 5: The moment convergence of $t - mean(C_d)$



(a) Mean

(b) Standard deviation

(c) Skewness

(d) Kurtosis

Figure 6: The moment convergence of $t - std(C_l)$

	$t-avr(C_d)$	$t-std(C_l)$
Correlation	0.984	0.988
Mean Absolute Relative Error	0.0207	0.360

Table 2: The correlation and the mean absolute relative error between the coarse- and the fine mesh models

	mean	mean ± standard deviation	Deterministic result
$t-avr(C_d)$	0.986	(0.940, 1.032)	1.03
$t-std(C_l)$	0.123	(0.101, 0.145)	0.206

Table 3: The comparison between the stochastic result and the deterministic result

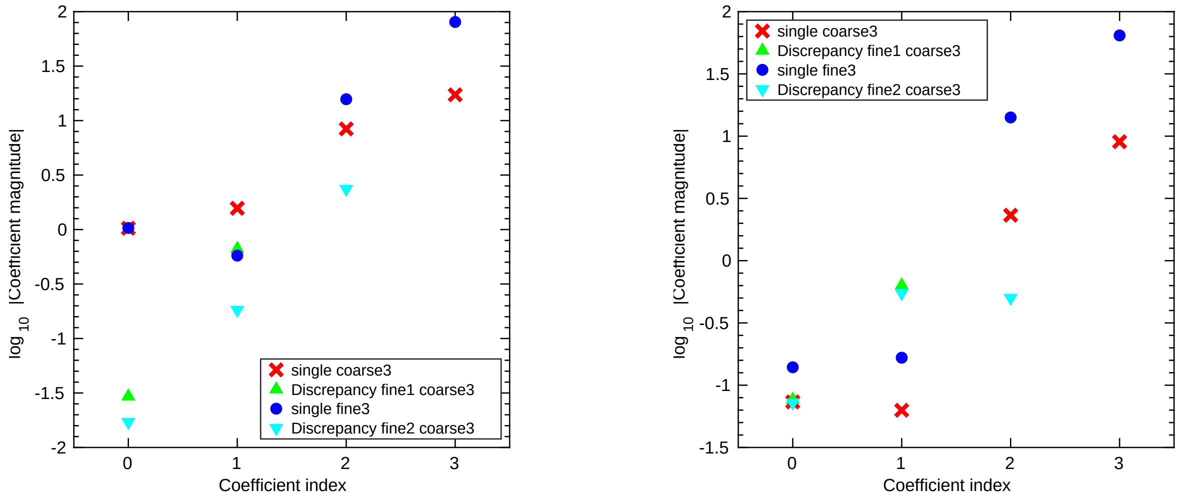
results used for NIPC of the level 1 to 3. The correlation and the mean absolute relative error are calculated as[11]:

$$\mathrm{Correlation} = \left(\frac{\sum_{i=1}^{N} (g_{high_i} - \overline{g}_{high})(g_{low_i} - \overline{g}_{low})}{\sqrt{\sum_{i=1}^{N} (g_{high_i} - \overline{g}_{high})^2} \sqrt{\sum_{i=1}^{N} (g_{low_i} - \overline{g}_{low})^2}} \right)^2 \tag{14}$$

$$\mathrm{MeanAbsoluteRelativeError} = \frac{1}{N} \sum_{i=1}^{N} \left| \frac{g_{low_i} - g_{high_i}}{g_{high_i}} \right| \tag{15}$$

where N is the number of deterministic simulations and $\overline{g}_{low}$ and $\overline{g}_{high}$ is the mean of the N observation data sets. From Table 2 it can be seen that $t-avr(C_d)$ is predicted well by the coarse mesh, while $t-std(C_l)$ has 36.6% error. The correlation values are similar in $t-avr(C_d)$ and $t-std(C_l)$. Even though the coarse mesh cannot predict $t-std(C_l)$ well, the multifidelity NIPC is able to get the similar PDF to the one computed by fine mesh only as discussed before. Fig. 7 shows the spectrum of polynomial chaos expansion coefficients of single fidelity model of the coarse mesh and the fine mesh respectively, and the multifidelity model of the coarse mesh level 3 and the additive correction level 1 and level 2. From Fig. 7, it is found that for (a) $t-avr(C_d)$ the coefficients of the discrepancy is always smaller than the ones of the coarse mesh model, while in (b) $t-std(C_l)$ the coefficients of the discrepancy is smaller but the coefficient index 1. That is, about $t-std(C_l)$ the discrepancy model is less complicated than the coarse mesh model, since it has larger value at the low coefficient index 1.

Let us think about the results of the multifidelity model of the coarse mesh with level 3 and the discrepancy model with level 1. Table 3 compares the stochastic result of the multifidelity model and the deterministic result of the fine mesh with angle of attack $0.0°$ and the curvature 0.0. For $t-avr(C_d)$, the result of deterministic simulation is close to the mean value of the stochastic results. On the other hand, $t-std(C_l)$ has different feature. The deterministic result and the stochastic result are very different. In addition the standard deviation of the $t-std(C_l)$ is 17.7% of the mean value, that is $t-std(C_l)$ has unignorable variation due to input uncertain parameters, the angle of attack and the curvature. $t-std(C_l)$ is caused by flow separation at the leading- and trailing edges and it is physically understandable that the geometry of the edges influences results of $t-std(C_l)$. The stochastic results confirmed that uncertainty of the curvature causes large variation of $t-std(C_l)$ and it is important to take into account the uncertainties in designing procedure. A problem of consideration of the curvature is that, the flow phenomenon has large difference between without curvature and with a curvature even as



(a) $t - avr(C_d)$

(b) $t - std(C_l)$

Figure 7: The spectrum of coefficients of the polynomial chaos expansion

small as 0.01 which is the smallest curvature in the calculated collocation points. This results in that $t - std(C_l)$ calculated by the deterministic simulation with the curvature 0.0 is not included in the stochastic result of $t - std(C_l)$.

5 Conclusion

- The multifidelity NIPC of additive correction is applied to the flow around a rectangular cylinder problem. By the additive correction, the shape of PDF of $t - mean(C_d)$ and the shape and position of PDF of $t - std(C_l)$ are improved.
- By applying the additive correction, the statistic moments of QoIs converge to the similar value as the ones calculated by the single fidelity model of the fine mesh, while the statistic moments calculated by the single fidelity model with the coarse mesh does not converge.
- The uncertainty of the angle of attack and the curvature causes unignorable variation to time statistics of the lift- and drag coefficients, especially $t - std(C_l)$.

6 Acknowledgement

TUM would like to thank the Deutscher Akademischer Austausch Dienst for funding. ICL would like to acknowledge the financial support of EPSRC and Airbus to UQLab.

References

[1] R. Ahlfeld, B. Belkouchi, and F. Montomoli. “SAMBA: Sparse Approximation of Moment-Based Arbitrary Polynomial Chaos”. In: *Journal of Computational Physics* 320 (Sept. 2016), pp. 1–16.

[2] Luca Bruno, Maria Vittoria Salvetti, and Francesco Ricciardelli. “Benchmark on the Aerodynamics of a Rectangular 5:1 Cylinder: An overview after the first four years of activity”. In: *Journal of Wind Engineering and Industrial Aerodynamics* 126 (Mar. 2014), pp. 87–106.

[3] Jordi Cotela Dalmau. “Applications of turbulence modeling in civil engineering”. PhD thesis. Universitat Polit‘ecnica de Catalunya, 2016.

[4] Michael S. Eldred et al. “Multifidelity Uncertainty Quantification Using Spectral Stochastic Discrepancy Models”. In: *Handbook of Uncertainty Quantification*. Ed. by Roger Ghanem, David Higdon, and Houman Owhadi. Cham: Springer International Publishing, 2015, pp. 991–1040.

[5] Roger Ghanem. *Handbook of uncertainty quantification*. New York, NY: Springer Berlin Heidelberg, 2017.

[6] Roger G. Ghanem and Pol D. Spanos. *Stochastic finite elements: a spectral approach*. New York, NY: Springer, 1991.

[7] Gene H Golub and John H Welsch. “Calculation of Gauss Quadrature Rules”. In: *Math. Comp.* 23 (1969), pp. 221–230.

[8] K. Hourigan, M.C. Thompson, and B.T. Tan. “SELF-SUSTAINED OSCILLATIONS IN FLOWS AROUND LONG BLUNT PLATES”. In: *Journal of Fluids and Structures* 15.3 (Apr. 2001), pp. 387–398.

[9] Gerhardus Joseph Alex Loeven. “Efficient uncertainty quantification in computational fluid dynamics.” PhD thesis. 2010.

[10] William L Oberkampf and Timothy G Trucano. “Verification, Validation, and Predictive Capability in Computational Engineering and Physics”. In: *SAND REPORT* (2003), p. 92.

[11] Pramudita Satria Palar et al. “Global Sensitivity Analysis via Multi-Fidelity Polynomial Chaos Expansion”. In: *Reliability Engineering and System Safety* (Oct. 22, 2017).

[12] Tetsuro Tamura, Kojiro Nozawa, and Koji Kondo. “AIJ guide for numerical prediction of wind loads on buildings”. In: *The Fourth International Symposium on Computational Wind Engineering, Yokohama* (2006).

[13] B. T. Tan, M. C. Thompson, and K. Hourigan. “Flow past rectangular cylinders: receptivity to transverse forcing”. In: *Journal of Fluid Mechanics* 515 (Sept. 25, 2004), pp. 33–62.

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

REDUCED MODEL-ERROR SOURCE TERMS FOR FLUID FLOW

Wouter Edeling[1] **and Daan Crommelin** [1,2]

[1] Centrum Wiskunde & Informatica, Scientific Computing Group
Science Park 123, 1098 XG Amsterdam, The Netherlands
e-mail: {Wouter.Edeling, Daan.Crommelin}@CWI.nl

[2] Korteweg-de Vries Institute for Mathematics, University of Amsterdam
Science Park 105-107, 1098 XG Amsterdam, The Netherlands
e-mail: D.T.Crommelin@uva.nl

Keywords: Model error, data-driven surrogate models, ocean flow

Abstract. *It is well known that the wide range of spatial and temporal scales present in geophysical flow problems represents a (currently) insurmountable computational bottleneck, which must be circumvented by a coarse-graining procedure. The effect of the unresolved fluid motions enters the coarse-grained equations as an unclosed forcing term, denoted as the 'eddy forcing'. Traditionally, the system is closed by approximate deterministic closure models, i.e. so-called parameterizations. Instead of creating a deterministic parameterization, some recent efforts have focused on creating a stochastic, data-driven surrogate model for the eddy forcing from a (limited) set of reference data, with the goal of accurately capturing the long-term flow statistics. Since the eddy forcing is a dynamically evolving field, a surrogate should be able to mimic the complex spatial patterns displayed by the eddy forcing. Rather than creating such a (fully data-driven) surrogate, we propose to precede the surrogate construction step by a procedure that replaces the eddy forcing with a new model-error source term which: i) is tailor-made to capture spatially-integrated statistics of interest, ii) strikes a balance between physical insight and data-driven modelling , and iii) significantly reduces the amount of training data that is needed. Instead of creating a surrogate for an evolving field, we now only require a surrogate model for one scalar time series per statistical quantity-of-interest. Our current surrogate modelling approach builds on a resampling strategy, where we create a probability density function of the reduced training data that is conditional on (time-lagged) resolved-scale variables. We derive the model-error source terms, and construct the reduced surrogate using an ocean model of two-dimensional turbulence in a doubly periodic square domain.*

1 INTRODUCTION

In the numerical simulation of coarse-grained turbulent flow problems one has to cope with small-scale processes which cannot be resolved directly on the numerical grid. The effect of the unresolved eddy field enters the resolved-scale equations as an unclosed forcing term, denoted as the eddy forcing, which is highly complex, dynamic, and shows intricate spatio-temporal correlations. Traditionally, the eddy forcing is approximated by deterministic closure models, i.e. so-called parameterizations. In the context of geophysical flows, such parameterizations are based on e.g. the work of Gent-McWilliams [6], or through the inclusion of a tunable (hyper) viscosity term meant to damp the smallest resolved scales of the model [11].

It is well known that no parameterization scheme is perfect, and attempts have been made to improve their performance. For instance, the authors of [15] analysed the transfer of energy and enstrophy in spectral space for a number of parameterizations, and compared their performance to a high-fidelity reference solution of a two-dimensional turbulent flow case. They proposed a deterministic 'energy fixer' scheme, based on adding a weighted vorticity pattern to the computed vorticity field. Recently, data-driven techniques have been applied as well. For instance the recent work of [10] used artificial neural networks to learn the eddy forcing from a set of high-fidelity snapshots.

However, a general limitation of such deterministic approaches is their inability to represent the strong non-uniqueness of the unresolved scales with respect to the resolved scales [1, 16, 12]. Since the resolved scales are generally defined as the convolution of the full-scale solution with some filter, multiple unresolved states can correspond to the same resolved solution. Thus, in general there is no one-to-one correspondence between the resolved-scale state and the unresolved-scale state, and yet deterministic parameterizations do assume such correspondence. As a result, stochastic methods for representing the unresolved scales have received an increasing amount of attention. Early contributions to this topic in the context of ocean modelling includes the work of [1], where the eddy-forcing is replaced by a space-time correlated random-forcing process. Other notable examples include the work of [9, 20, 7], who construct probability density functions (pdfs) of the eddy forcing using a reference solution.

In this study, we also consider a stochastic surrogate method [17, 16], and as a performance indicator we use the degree by which it is able to capture energy and enstrophy statistics. However, we refrain from an approach that is purely data-driven, i.e. one which attempts to learn the eddy forcing directly from reference data. Instead, we replace the eddy forcing with a simpler 'model-error' source term, which we parameterize based on physical arguments. Specifically, we use the energy and enstrophy transport equations to derive a source term which tracks our chosen target statistics. The only remaining unclosed part of our model-error term is representative of the magnitude of these target statistics, i.e. scalars. As a result, the corresponding surrogate model needs to represent only one (or a few) scalar quantities rather than the full eddy forcing field. This amounts to a large dimension reduction (in this study, a reduction by four orders of magnitude), and as a consequence a large reduction in the amount of required training data, while retaining accuracy in the statistics.

The article is organised as follows. In Section 2 we describe the governing equations and multiscale decomposition. The model-error source term derivation and the surrogate method are outlined in Section 3. Initial results are shown in Section 4, and finally the conclusion and outlook are given in Section 5.

2 GOVERNING EQUATIONS

We study the same model as in [18], i.e. the forced-dissipative vorticity equations for two-dimensional incompressible flow. The governing equations read

$$\begin{aligned} \frac{\partial \omega}{\partial t} + J\left(\Psi, \omega\right) &= \nu\nabla^2\omega + \mu\left(F - \omega\right), \\ \nabla^2\Psi &= \omega. \end{aligned} \tag{1}$$

Here, ω is the vertical component of the vorticity, defined from the curl of the velocity field $\mathbf{V}$ as $\omega := \mathbf{e}_3 \cdot \nabla \times \mathbf{V}$, where $\mathbf{e}_3 := (0, 0, 1)^T$. The stream function Ψ relates to the horizontal velocity components by the well-known relations $u = -\partial\Psi/\partial y$ and $v = \partial\Psi/\partial x$. As in [18], the forcing term is chosen as the single Fourier mode $F = 2^{3/2}\cos(5x)\cos(5y)$. The system is fully periodic in x and y directions over a period of $2\pi L$, where L is a user-specified length scale, chosen as the earth's radius ($L = 6.371 \times 10^6 [m]$). The inverse of the earth's angular velocity Ω^{-1} is chosen as a time scale, where $\Omega = 7.292 \times 10^{-5} [s^{-1}]$. Thus, a simulation time period of a single 'day' can now be expressed as $24 \times 60^2 \times \Omega \approx 6.3$ non-dimensional time units. Given these choices, (1) is non-dimensionalized, and solved using values of ν and μ chosen such that a Fourier mode at the smallest retained spatial scale is exponentially damped with an e-folding time scale of 5 and 90 days respectively. For more details on the numerical setup we refer to [18]. Furthermore, our Python source code for (1) can be downloaded from [4].

Finally, the key term in (1) is the Jacobian, i.e. the nonlinear advection term defined as

$$J\left(\Psi, \omega\right) := \frac{\partial \Psi}{\partial x}\frac{\partial \omega}{\partial y} - \frac{\partial \Psi}{\partial y}\frac{\partial \omega}{\partial x}. \tag{2}$$

It is this term that leads to the need for a closure model when (1) is discretized on a relatively coarse grid which lacks the resolution to capture all turbulent eddies.

2.1 Discretization

We solve (1) by means of a spectral method, where we apply a truncated Fourier expansion:

$$\begin{aligned} \omega_{\mathbf{k}}(x, y, t) &= \sum_{\mathbf{k}} \hat{\omega}_{\mathbf{k}}(t) e^{\underline{i}(k_1 x + k_2 y)}, \\ \Psi_{\mathbf{k}}(x, y, t) &= \sum_{\mathbf{k}} \hat{\Psi}_{\mathbf{k}}(t) e^{\underline{i}(k_1 x + k_2 y)}. \end{aligned} \tag{3}$$

The sum is taken over the components k_1 and k_2 of the wave number vector $\mathbf{k} := (k_1, k_2)^T$, and $-K' \leq k_j \leq K'$, $j = 1, 2$. These decompositions are inserted in (1), and solved for the Fourier coefficients $\hat{\omega}_{\mathbf{k}}$, $\hat{\Psi}_{\mathbf{k}}$ by means of the real Fast Fourier Transform. To avoid the aliasing problem in the nonlinear term (2), we use the pseudo spectral method, such that in practice the maximum resolved wave number is K, where $K \leq 2K'/3$ [14]. [1]

To advance the solution in time we use the second-order accurate AB/BDI2 scheme, which results in the following discrete system of equations [14]

$$\begin{aligned} \frac{3\hat{\omega}_{\mathbf{k}}^{i+1} - 4\hat{\omega}_{\mathbf{k}}^{i} + \hat{\omega}_{\mathbf{k}}^{i-1}}{2\Delta t} + 2\hat{J}_{\mathbf{k}}^{i} - \hat{J}_{\mathbf{k}}^{i-1} &= -\nu k^2 \hat{\omega}_{\mathbf{k}}^{i+1} + \mu\left(\hat{F}_{\mathbf{k}} - \hat{\omega}_{\mathbf{k}}^{i+1}\right), \\ -k^2\hat{\Psi}_{\mathbf{k}}^{i+1} - \hat{\omega}_{\mathbf{k}}^{i+1} &= 0. \end{aligned} \tag{4}$$

[1] We use $N \times N$ grids, with an even $N = 2^p$ (e.g. $p = 7$), such that $N = 2K'$ [14].

Here, $\Delta t = 0.01$ and $\hat{J}^i_{\mathbf{k}}$ is the Fourier coefficient of the Jacobian at time level i, computed with the pseudo spectral technique, and $k^2 := k_1^2 + k_2^2$.

2.2 Multiscale decomposition

As in [18], we apply a spectral filter in order to decompose the full reference solution into a resolved ($\mathcal{R}$) and an unresolved component ($\mathcal{U}$), i.e. we use

$$\hat{\omega}^{\mathcal{R}}_{\mathbf{k}} = P^{\mathcal{R}}\hat{\omega}_{\mathbf{k}}, \qquad \hat{\omega}^{\mathcal{U}}_{\mathbf{k}} = P^{\mathcal{U}}\hat{\omega}_{\mathbf{k}}, \tag{5}$$

where the projection operators $P^{\mathcal{R}}$ and $P^{\mathcal{U}}$ are depicted in Figure 1. Note that the full projection operator $P := \mathcal{P}^{\mathcal{R}} + \mathcal{P}^{\mathcal{U}}$ also removes wave numbers due to the use of the pseudo spectral method.

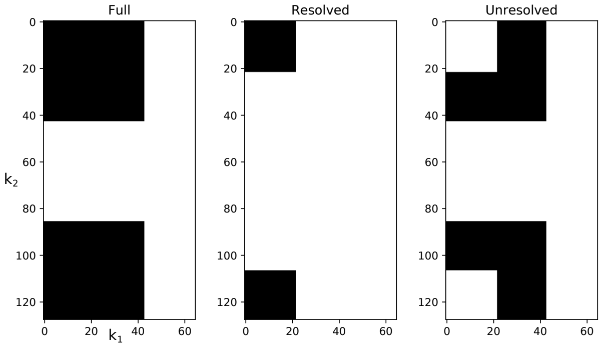


Figure 1: The spectral filter (black=1, white=0) of the full, resolved and unresolved solutions. Due to the fact that we use the real FFT algorithm, only part of the spectrum is computed, as Fourier coefficients with opposite values of $\mathbf{k}$ are complex conjugates in order to enforce real ω and Ψ fields [14].

Applying the resolved projection operator to the governing equations (1) results in the following resolved-scale transport equation

$$\frac{\partial \omega^{\mathcal{R}}}{\partial t} + \mathcal{P}^{\mathcal{R}} J\left(\Psi, \omega\right) = \nu\nabla^2\omega^{\mathcal{R}} + \mu\left(F^{\mathcal{R}} - \omega^{\mathcal{R}}\right) \tag{6}$$

As mentioned, the key term is the Jacobian (2), since due to its non linearity, $\mathcal{P}^{\mathcal{R}} J\left(\Psi,\ \omega\right) \neq \mathcal{P}^{\mathcal{R}} J\left(\Psi^{\mathcal{R}},\ \omega^{\mathcal{R}}\right)$. We therefore write

$$J\left(\Psi, \omega\right) - J\left(\Psi^{\mathcal{R}},\ \omega^{\mathcal{R}}\right) =: r, \tag{7}$$

such that r is the exact subgrid-scale term, commonly referred to as the 'eddy forcing' [1]. The resolved-scale equation (6) can now be written as

$$\frac{\partial \omega^{\mathcal{R}}}{\partial t} + \mathcal{P}^{\mathcal{R}} J\left(\Psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right) = \nu\nabla^2\omega^{\mathcal{R}} + \mu\left(F^{\mathcal{R}} - \omega^{\mathcal{R}}\right) - \overline{r}. \tag{8}$$

We use the notation $\overline{r} := \mathcal{P}^{\mathcal{R}} r$ for the sake of brevity. A snapshot of the resolved vorticity $\omega^{\mathcal{R}}$ and corresponding resolved eddy forcing $\overline{r}$ is depicted in Figure 2. Notice the fine-grained character of the eddy forcing compared to the vorticity field.

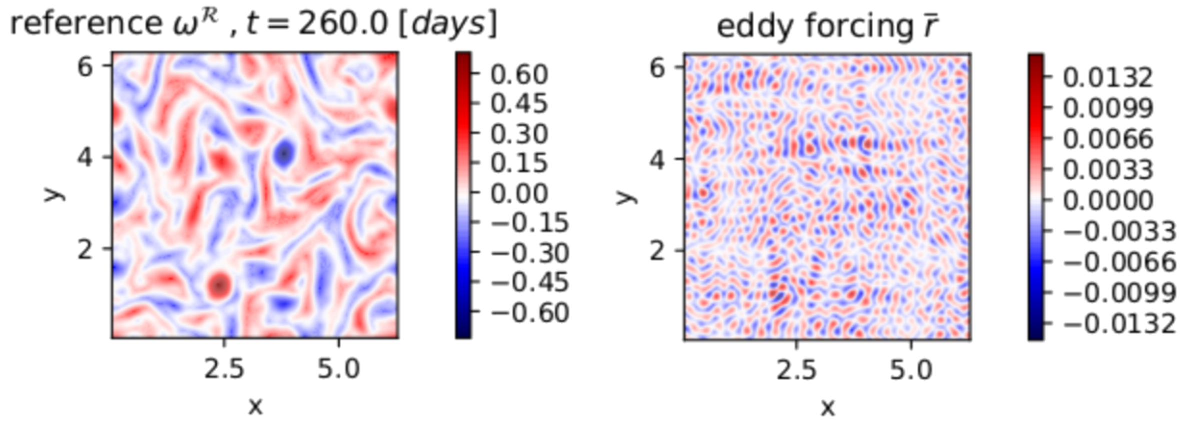


Figure 2: A snapshot of the exact, reference vorticity field $\omega^{\mathcal{R}}$ and the corresponding eddy forcing.

2.3 Prediction of climate statistics

Ultimately, our goal is to integrate (8) in time, such that we can compute the long-term climate statistics of the energy $E^{\mathcal{R}}$ and enstrophy $Z^{\mathcal{R}}$ densities, defined as

$$E^{\mathcal{R}} := \frac{1}{2}\left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}} \mathrm{d}x\mathrm{d}y = -\frac{1}{2}\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right), \tag{9}$$

$$Z^{\mathcal{R}} := \frac{1}{2}\left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \left(\omega^{\mathcal{R}}\right)^2 \mathrm{d}x\mathrm{d}y = \frac{1}{2}\left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right). \tag{10}$$

Here $\mathbf{V}^{\mathcal{R}}$ is the two-dimensional vector of the resolved velocity components in x and y direction. For conciseness, we use the short-hand notation

$$(\alpha, \beta) = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \alpha\beta \; \mathrm{d}x\mathrm{d}y, \tag{11}$$

to denote the integral of the product $\alpha\beta$ normalized by the area of the flow domain. The derivation of the last equality of (9) can be found in Appendix A.

3 EDDY-FORCING SURROGATE

We cannot integrate (8) since it is still unclosed (due to the ω and Ψ dependence of (7)), a problem which we aim to solve by creating a data-driven surrogate of $\bar{r}$, denoted by $\widetilde{r}$. For our present purpose, we define an 'ideal' surrogate $\widetilde{r}$ for the eddy forcing as one which satisfies the following set of requirements:

1. **Data-driven**: In absence of a single 'best' deterministic parameterization of r, we opt for a model inferred from a pre-computed database of high-fidelity reference data.

2. **Stochastic**: In general, the resolved scales are defined as a convolution of the full solution with some (spatial/spectral) filter. As a result there is no longer just a single unresolved-scale field that is consistent with the resolved-scale solution. This ambiguity provides us with the motivation for a stochastic model for the unresolved, small-scale fields.

3. **Correlated in space and time**: As demonstrated by Figure 2, the reference eddy forcing shows complex spatial structures. A surrogate of the full eddy forcing would ideally reflect these as well.

4. **Conditional on the resolved variables**: The resolved and unresolved scales are in reality two-way coupled. Hence, the eddy-forcing surrogate should not be independent from the resolved solution.

5. **Pre-computed & cheap**: While the reference database can be computationally expensive to compute, the resulting data-driven surrogate must be cheap.

6. **Extrapolates well**: To justify the cost of creating the reference database in the first place, the data-driven model must be able to predict the chosen quantity of interest well, *substantially beyond* the (time) domain of the data.

As mentioned, we will measure the performance of a surrogate model by its ability to accurately represent the statistics of (9)-(10). Thus, we do not expect from the resolved-scale model forced by the surrogate the ability to produce individual flow fields which are in absolute lock-step with the high-fidelity data, especially considering the stochastic nature of the surrogate.

One possible course of action, explored in e.g. [17, 10], is to directly create a *full-field* surrogate $\widetilde{r}\left(x,y;t\right) \in \mathbb{R}^{N\times N}$, using a database reference snapshots in time of the exact eddy forcing (7). Here, N is the number of grid points in one spatial direction, typically 2^7, 2^8 or higher. Constructing a full-field, dynamic surrogate of a quantity as complex as the eddy forcing is a challenging task, and storing a potentially large amount of reference snapshots can lead to high memory requirements [17]. We therefore propose to precede the surrogate construction step with a procedure that significantly compresses the training data.

3.1 Reduced surrogate

Note that our statistical quantities of interest (9) and (10) are scalars. Instead of creating a full-field $N \times N$ surrogate $\widetilde{r}\left(x,y;t\right)$, we will first replace the exact $\overline{r}$ in (8) with a simpler alternative, where the unclosed component is reflective of the size of the statistical quantities we aim to approximate in the first place. A simple option is to specify

$$-\overline{r}\left(x,y;t\right) = \tau\left(t\right)\omega^{\mathcal{R}}\left(x,y;t\right), \tag{12}$$

where $\tau\left(t\right)$ is an unknown, time-varying scalar. Clearly, this choice is arbitrary, and (12) will not match the eddy forcing (7). Instead, we think of (12) as an example of a 'model-error term', meant to correct the unparameterized ($\overline{r} = 0$) model in some sense. In our case, a deviation from the exact eddy forcing does not pose a problem because of the freedom that integrated quantities-of-interest give us, such that we only need our $\omega^{\mathcal{R}}$ and $\Psi^{\mathcal{R}}$ fields to approximate the truth in the weak sense of (9) and (10). We can examine the effect of (12) on the evolution equations of $E^{\mathcal{R}}$ and $Z^{\mathcal{R}}$, and subsequently combine physical insight with a data-driven approach to find the time series of τ that constrains their evolution to the reference values. A *reduced surrogate* now only needs to be constructed from this scalar time series, instead of from the full-field evolution of (7).

The evolution equation of $E^{\mathcal{R}}$ (see Appendix A) satisfies

$$\frac{\mathrm{d}E^{\mathcal{R}}}{\mathrm{d}t} = -\left(\psi^{\mathcal{R}}, \frac{\partial\omega^{\mathcal{R}}}{\partial t}\right) = -2\nu Z^{\mathcal{R}} - 2\mu U^{\mathcal{R}} - 2\mu E^{\mathcal{R}} + \left(\psi^{\mathcal{R}}, \overline{r}\right), \tag{13}$$

where we denote the integral $\left(\Psi^{R}, F\right)/2$ as $U^{\mathcal{R}}$. If we insert (12) into (13), the last term on the right-hand side becomes

$$\left(\psi^{\mathcal{R}}, \overline{r}\right) = -\tau\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right) = 2\tau E^{\mathcal{R}}. \tag{14}$$

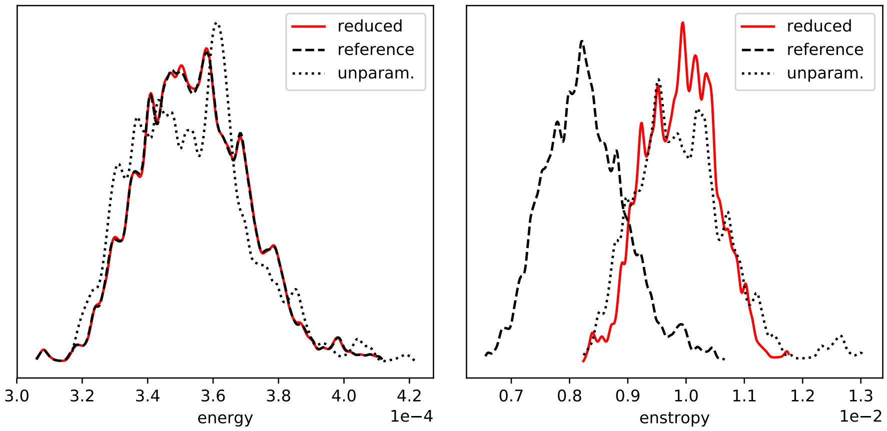


Figure 3: The pdfs of the energy (left) and enstrophy (right), of the reduced ($\overline{r} = \tau\omega^{\mathcal{R}}$), reference ($\overline{r}$ given by (7)) and unparameterised ($\overline{r} = 0$) solution.

The last equality follows from the definition (9). Thus, the physical insight is that (12) leads to the additional term $2\tau E^{\mathcal{R}}$, which either acts to produce or dissipate $E^{\mathcal{R}}$ depending on the sign of τ. Let us denote the difference between the projected reference energy and $E^{\mathcal{R}}$ as $\Delta E := E - E^{\mathcal{R}}$, where $E := -\left(\mathcal{P}^{\mathcal{R}}\Psi, \omega\right)/2$. Any quantity without superscript, e.g. E or ω, is a reference quantity computed from (1). Now, for the data-driven determination of the τ time series, we require τ to be positive when $\Delta E > 0$, i.e. to increase production when $E^{\mathcal{R}}$ is too low, and to dissipate energy when $\Delta E < 0$. We parameterize τ via an analytic relationship which reflects this property:

$$\tau := \tau_{max} \tanh\left(\frac{\Delta E}{E^{\mathcal{R}}}\right). \tag{15}$$

Here, τ_{max} is a user-specified constant, which we set to one for now. During the training period, we can compute (15) every Δt, building up a reference time series.

To test the validity of our approach, we run the system (8) for a simulation period of 8 years. Besides τ, at every Δt we also sample the energy and enstrophy of the reference, reduced and unparameterised solution, i.e. using $\overline{r}$ given by (7), (12) and zero respectively[2]. The energy and enstrophy probability density functions (pdfs) generated from those samples can be found in Figure 3. By virtue of (15), the energy pdfs of the reference and the reduced solution practically overlap. This demonstrates that it is possible to obtain statistically-equivalent energy solutions using training data reduced by a factor of N^2 compared to the full-field surrogate case[3].

However, we have two quantities of interest, and (12) also has an effect on the $Z^{\mathcal{R}}$ equation (a term $2\tau Z^{\mathcal{R}}$ appears). Since we train τ to track $\mathcal{P}^{\mathcal{R}}E$, we cannot expect a perfect $Z^{\mathcal{R}}$ pdf, and in fact, Figure 3 shows that the situation does not improve upon the unparameterised model, which displays a large bias in $Z^{\mathcal{R}}$ values. Rather than trying to construct a different τ which is some compromise between accuracy in $E^{\mathcal{R}}$ and $Z^{\mathcal{R}}$, we opt for two separate time series, each of which acts on either the energy or enstrophy evolution equation alone.

[2] Note that no surrogate is used yet, we are generating a large set of training data.

[3] In the example of Figure 3, $N^2 = 128^2 = 16384$.

3.2 Orthogonal patterns

We replace our initial simple choice (12) with

$$-\overline{r} = \tau_E \Psi' + \tau_Z \omega', \tag{16}$$

where Ψ' and ω' are patterns of the resolved vorticity and stream function. We choose Ψ' such that $\tau_E \Psi'$ only acts on the $E^{\mathcal{R}}$ equation, and produces no additional source term in the enstrophy equation. The converse must be true for the $\tau_Z \omega'$ term. This will allow us to train τ_E on ΔE alone, and τ_Z only on $\Delta Z := Z - Z^R$. Since the $E^{\mathcal{R}}$ and $Z^{\mathcal{R}}$ evolution equations are forced by $-\left(\Psi^R, \partial\omega^{\mathcal{R}}/\partial t\right)$ and $\left(\omega^{\mathcal{R}}, \partial\omega^{\mathcal{R}}/\partial t\right)$ respectively (see (13) and appendix A), this suggests a Gram-Schmidt type of approach to make Ψ' orthogonal to $\left(\omega^{\mathcal{R}}, \cdot\right)$ and likewise for ω' and $\left(\Psi^R, \cdot\right)$. Setting:

$$\Psi' = \psi^{\mathcal{R}} - \frac{\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)}{\left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right)}\omega^{\mathcal{R}} \ \text{ and } \ \omega' = \omega^{\mathcal{R}} - \frac{\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)}{\left(\psi^{\mathcal{R}}, \psi^{\mathcal{R}}\right)}\psi^{\mathcal{R}}, \tag{17}$$

yields

$$\left(\omega^{\mathcal{R}}, \tau_E \Psi'\right) = 0 \ \text{ and } \ \left(\psi^{\mathcal{R}}, \tau_Z \omega'\right) = 0. \tag{18}$$

The additional source term in the $E^{\mathcal{R}}$ equation now becomes

$$-\left(\psi^{\mathcal{R}}, \tau_E \Psi'\right) = -\tau_E\left(\psi^{\mathcal{R}}, \psi^{\mathcal{R}}\right) + \tau_E \frac{\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)^2}{\left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right)} = 2\tau_E\left[\frac{\left(E^{\mathcal{R}}\right)^2}{Z^{\mathcal{R}}} - S^{\mathcal{R}}\right] := 2\tau_E S' \tag{19}$$

Here, we defined the integrated square stream function as $S^{\mathcal{R}} := \left(\psi^{\mathcal{R}}, \psi^{\mathcal{R}}\right)/2$. Since $\left(E^{\mathcal{R}}\right)^2/Z^R - S^{\mathcal{R}}$ has the dimension of the squared stream function, we introduce the final shorthand notation $S' := \left(E^{\mathcal{R}}\right)^2/Z^{\mathcal{R}} - S^{\mathcal{R}}$ in (19). In a similar vein, (16) produces the following source term in the $Z^{\mathcal{R}}$ equation:

$$2\tau_Z Z' \ \text{ with } \ Z' := Z^{\mathcal{R}} - \frac{\left(E^{\mathcal{R}}\right)^2}{S^{\mathcal{R}}}. \tag{20}$$

We parameterise τ_E and τ_Z using the same procedure as in Section 3.1, only now we need to incorporate the sign of S' and Z' to correctly activate either the production or dissipation of $E^{\mathcal{R}}$ and $Z^{\mathcal{R}}$, i.e.

$$\tau_E := \tau_{E,max}\tanh\left(\frac{\Delta E}{E^{\mathcal{R}}}\right)\cdot \mathrm{sgn}(S') \ \text{ and } \ \tau_Z := \tau_{Z,max}\tanh\left(\frac{\Delta Z}{Z^{\mathcal{R}}}\right)\cdot \mathrm{sgn}(Z'). \tag{21}$$

Again, we leave the proper estimation of parameters for a later study, and simply set $\tau_{E,max} = \tau_{Z,max} = 1$. Furthermore, $\mathrm{sgn}(X) = 1$ when $X \geq 1$ and -1 otherwise. Repeating the simulation of Section 3.1, inserting (16) in (8) yields the results depicted in Figure 4. Now, both pdfs match the reference well. Only a very small discrepancy in the $E^{\mathcal{R}}$ pdf can be observed, which might fixed by tuning $\tau_{E,max}$. The corresponding τ_E, τ_Z reference time series are shown in Figure 5.

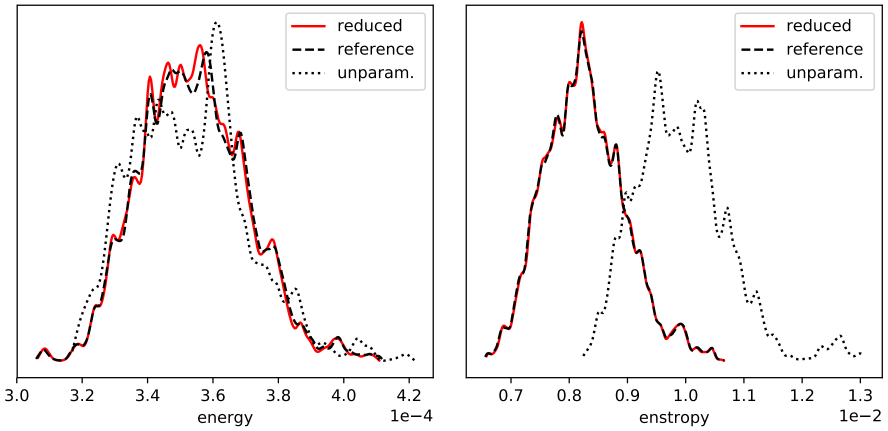


Figure 4: The pdfs of the energy (left) and enstrophy (right), of the reduced ($\overline{r} = \tau_E \Psi' + \tau_Z \omega'$), reference ($\overline{r}$ given by (7)) and unparameterised ($\overline{r} = 0$) solution.

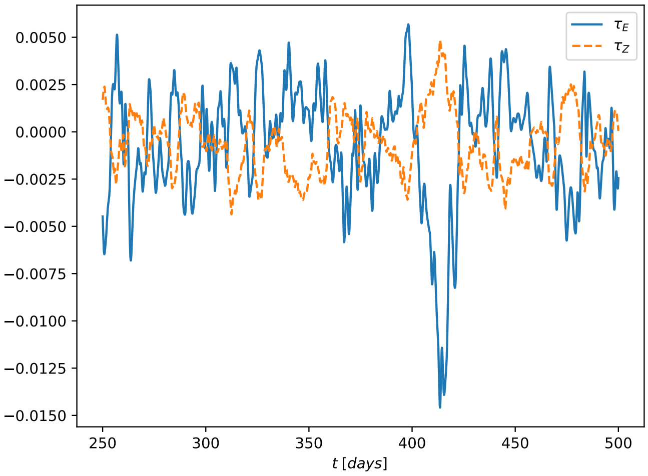


Figure 5: Training time series of τ_E and τ_Z over 500 days. Note that there seems to be a negative correlation between the two time series.

3.3 Surrogate construction

We will build on the resampling stategies as developed by [16, 2]. In general, these methods model the unresolved term at time t_{i+1} by sampling from the conditional probability distribution of the reference data. In our case, we keep the functional forms of (21), such that ΔE and ΔZ can be chosen as the unresolved terms in need of a surrogate model:

$$\begin{aligned}\widetilde{\Delta E}_{i+1} &\sim \Delta E_{i+1} \mid \mathcal{E}_i,\ \mathcal{E}_{i-1}, \cdots \\ \widetilde{\Delta Z}_{i+1} &\sim \Delta Z_{i+1} \mid \mathcal{Z}_i,\ \mathcal{Z}_{i-1}, \cdots\end{aligned} \tag{22}$$

Here, $\widetilde{\Delta E}_{i+1}$ denotes the data-driven resampling surrogate at time t_{i+1}, whereas as ΔE_{i+1} represent actual reference data from the training run, and likewise for $\widetilde{\Delta Z}_{i+1}$. The set of 'conditioning variables' $\mathcal{E}_i, \mathcal{Z}_i$ etc contain variables from the resolved model. They can be (functions of) $E^{\mathcal{R}}$, S' or any other (scalar) quantity, as long as we also have access to it outside the training period. Examples of these conditional distributions are $\Delta E_{i+1} \mid E_i^{\mathcal{R}}$ and $\Delta Z_{i+1} \mid \widetilde{\Delta Z}_i, Z_i^{\mathcal{R}}$. We could assume a Markov property ($\Delta E_{i+1} \mid \mathcal{E}_i$), or build in a larger memory. Note that by design, (22) already satisfies many of the properties listed in Section 3, e.g. it is data-driven, stochastic and conditioned on resolved variables.

The main challenges with this approach are twofold. Clearly, the first challenge concerns the actual formation of the conditional distribution, i.e. how to map the observed conditioning variables to plausible subsets of ΔE_{i+1} and ΔZ_{i+1} samples from which $\widetilde{\Delta E}_{i+1}$ and $\widetilde{\Delta Z}_{i+1}$ can be randomly sampled. The second challenge concerns the proper choice of conditioning variables, which is somewhat reminiscent of the choice of 'features' in a machine-learning context.

3.4 Building the distribution

We will illustrate the approach using ΔE, the same procedure applies for ΔZ. To map $\mathcal{E}_i$ to some subset of plausible ΔE_{i+1} values we use the so-called 'binning' approach of [16]. First, consider a snapshot sequence of ΔE

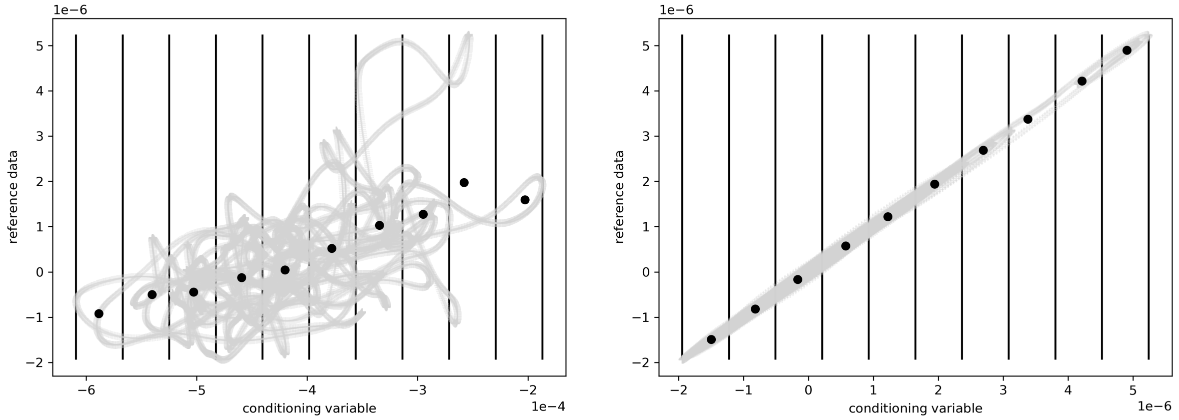
$$\mathbf{\Delta E}_1^S = \{\Delta E_1, \Delta E_2, \cdots, \Delta E_i, \cdots, \Delta E_S\}, \tag{23}$$

where i is the time index. In addition, we also have snapshots of corresponding conditioning variables

$$\mathbf{E}_1^S = \{\mathcal{E}_1, \mathcal{E}_2, \cdots, \mathcal{E}_S\}. \tag{24}$$

Let C be the total number of time-lagged conditioning variables used in (22). We then proceed by creating C-dimensional disjoint bins[4], each bin spanning a unique conditioning variable range, and containing a number of associated ΔE values, see Figure 6. Note that not all bins may contain samples, especially if two or more conditioning variables are used. If during prediction an empty bin is sampled, the data of the nearest bin (in Euclidean sense) is used instead. Once a bin is selected by $\mathcal{E}_i$, the resulting subset of ΔE values can be sampled randomly, or one might sample from the local bin average instead, leading to a deterministic prediction.

[4] We used equidistant bins, but this is not a hard requirement.



(a) Low correlation between ΔE_{i+1} and $\mathcal{E}_i$.

(b) High correlation between ΔE_{i+1} and $\mathcal{E}_i$.

Figure 6: Two binning objects, with the reference ΔE_{i+1} data on the vertical axis and the conditioning variable $\mathcal{E}_i$ on the horizontal axis. Vertical lines separate the different bins, and the black dots represent the local bin means.

3.5 Choice of conditioning variables

Ideally we would like the conditioning variables of (22) to display some correlation with ΔE_{i+1} and ΔZ_{i+1}. In this case, the range of plausible reference values in the selected subset is smaller. Consider the two bins depicted in Figure 6, each with 1 conditioning variable ($\Delta E_{i+1} \mid \mathcal{E}_i$). The binning object of Figure 6(a) shows considerable less correlation between $\mathcal{E}_i$ and ΔE_{i+1} than its counterpart in Figure 6(b). As a result, each bin contains a larger spread in possible ΔE values, leading to more noisy $\widetilde{\Delta E}_{i+1}$ predictions.

We continue by drawing up a list of candidate conditioning variables, and computing the temporal correlation coefficients

$$\rho\left(\Delta E_{i+1}, \mathcal{E}_i\right) = \frac{Cov\left[\Delta E_{i+1}, \mathcal{E}_i\right]}{\sigma\left(\Delta E_{i+1}\right)\sigma\left(\mathcal{E}_i\right)} \quad \text{and} \quad \rho\left(\Delta Z_{i+1}, \mathcal{Z}_i\right) = \frac{Cov\left[\Delta Z_{i+1}, \mathcal{Z}_i\right]}{\sigma\left(\Delta Z_{i+1}\right)\sigma\left(\mathcal{Z}_i\right)} \tag{25}$$

from a reference time series of 500 days. Here $Cov\left(\cdot,\cdot\right)$ is the covariance operator and $\sigma\left(\cdot\right)$ is the standard deviation. Specifically, we will select individual source terms from the $E^{\mathcal{R}}$ and $Z^{\mathcal{R}}$ equations as candidate $\mathcal{E}_i$ and $\mathcal{Z}_i$, the rationale being that these will also (in part) drive the evolution equations of ΔE and ΔZ. The complete list, including the correlation coefficient values, is shown in Table 1. Previously undefined conditioning variables (occurring in the $Z^{\mathcal{R}}$ equation), are $V^{\mathcal{R}} := \left(\omega^{\mathcal{R}}, F\right)/2$ and $O^{\mathcal{R}} := \left(\nabla^2\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$. This strategy for selecting candidate conditioning variables is reasonable, as many show substantial correlation with the reference data, hovering around the ± 0.5 mark. Clear exceptions are $E^{\mathcal{R}}$ (which correlates much less), and $\tau_E S'$, $\tau_Z Z'$, which show very high correlation.

4 RESULTS

This section contains the initial exploratory results of the methodology outlined in the preceding sections. For validation and training purposes we ran the reference model (1) for a simulation period of 8 years, storing reference data and conditioning variables every Δt. Here, is amounts to roughly 1.8×10^6 snapshots per variable. When predicting, the training data must be stored in memory to allow for fast resampling. If the reference snapshots are full field, this can lead to high memory requirements [17]. Subsampling the reference data reduces the memory constraints, although this leads to a surrogate with an intrinsic time step that is larger

	$\mathcal{E}_i,\ \mathcal{Z}_i$	$\rho\left(\Delta E_{i+1}, \mathcal{E}_i\right)$	$\rho\left(\Delta Z_{i+1}, \mathcal{Z}_i\right)$
$Z^{\mathcal{R}}$:	$\left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$	0.4017	0.336
$E^{\mathcal{R}}$:	$-\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$	0.1401	0.0951
$U^{\mathcal{R}}$:	$\left(\psi^{\mathcal{R}}, F\right)/2$	0.5497	0.598
$S^{\mathcal{R}}$:	$\left(\psi^{\mathcal{R}}, \psi^{\mathcal{R}}\right)/2$	-0.5091	-0.4857
$V^{\mathcal{R}}$:	$\left(\omega^{\mathcal{R}}, F\right)/2$	-0.5467	-0.5965
$O^{\mathcal{R}}$:	$\left(\nabla^2\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$	-0.4993	-0.4394
$\tau_E S'$:	$\tau_E\left(\left(E^{\mathcal{R}}\right)^2/Z^{\mathcal{R}} - S^{R}\right)$	0.9484	0.8876
$\tau_Z Z'$:	$\tau_Z\left(Z^{R} - \left(E^{\mathcal{R}}\right)^2/S^{\mathcal{R}}\right)$	0.8915	0.999

Table 1: Correlation coefficients.

than the Δt of (4), and thus can only be updated after a certain number of Δt time cycles [2]. A clear advantage of our current surrogate approach, is that we can store the full 8 year reduced training set in memory, without the need for subsampling.

We subdivide the results into tests of increasing complexity:

T1: A one-way coupled simulation where the resolved equation (8) provides the conditioning variables, without replacing $\overline{r} = \tau_E\left(\Delta E\right)\Psi' + \tau_Z\left(\Delta Z\right)\omega'$ in (8) with the surrogate $\widetilde{r} = \tau_E(\widetilde{\Delta E})\Psi' + \tau_Z(\widetilde{\Delta Z})\omega'$. The surrogates $\widetilde{\Delta E}$ and $\widetilde{\Delta Z}$ are not extrapolated, i.e. they are constructed using the full 8 year reference data set, so no simulation outside the time period of the training data takes place.

T2: A two-way coupled simulation, still without surrogate extrapolation.

T3: A two-way coupled simulation with surrogate extrapolation.

4.1 Results T1

T1 serves as a verification of our code, as in this case the exact ΔE and ΔZ are still used in (21) to compute τ_E and τ_Z. Now, if implemented correctly, surrogates such as $\widetilde{\Delta E}_{i+1} \sim \Delta E_{i+1} \mid \left(\tau_E S'\right)_i$ and $\widetilde{\Delta Z}_{i+1} \sim \Delta Z_{i+1} \mid \left(\tau_Z Z'\right)_i$, must follow the reference data closely, given the high correlations displayed in Table 1. This is confirmed by the results of Figure 7.

4.2 Results T2

T2 is the first real test of the surrogate method due to its two-way coupled nature. As a result, trajectories of $\widetilde{\Delta E}$ and $\widetilde{\Delta Z}$ can no longer be expected to follow the reference data. Discrepancies between the exact (reduced) eddy forcing (16) and its surrogate will cause the model forced by the surrogate to develop its own dynamics. We reiterate here that our goal is to predict the time-averaged flow statistics, which might still be feasible if we are not in absolute lockstep with ΔE and ΔZ. Even two full-scale simulations with slightly different initial conditions will diverge from each other (due to their turbulent nature), yet can converge in a statistical sense.

We tested a variety of surrogates, which differed through the set of selected conditioning variables. All were Markovian in character, conditioned on variables from the previous time step alone. Thus far, almost all considered surrogates improved upon the $Z^{\mathcal{R}}$ bias of the unparameterized model, although they showed some varying performance amongst each other.

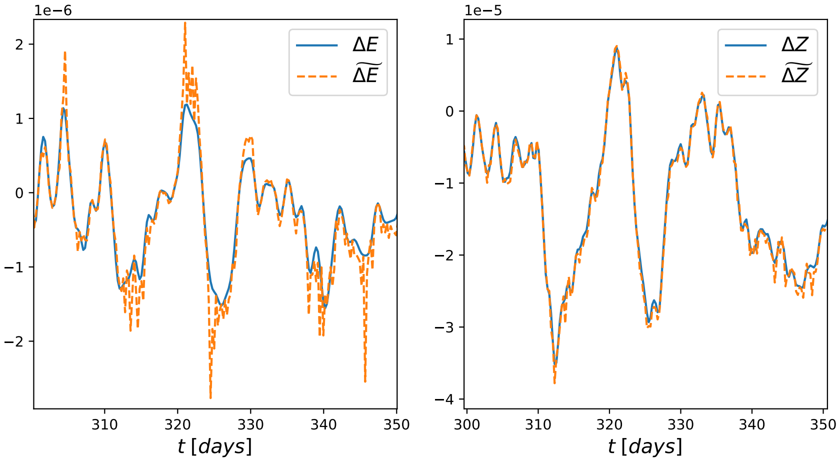


Figure 7: T1 time series for ΔE and ΔZ and their corresponding surrogates over a 50 day period. The $\widetilde{\Delta E}$ surrogate is noisier due to the lower correlation with its conditioning variable (see Table 1).

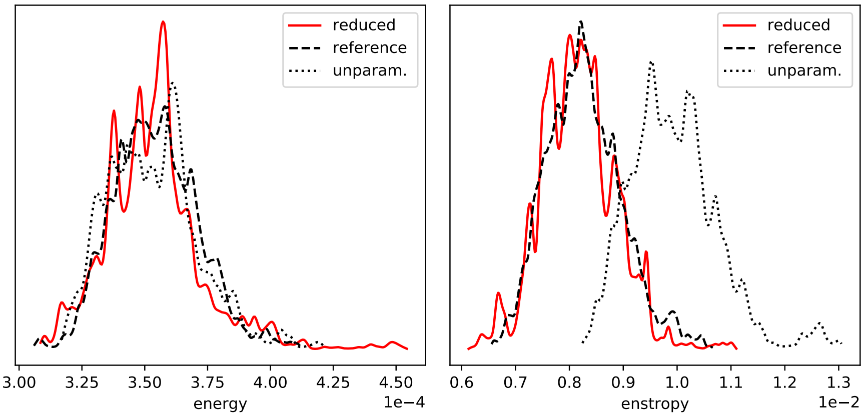


Figure 8: The pdfs of the energy (left) and enstrophy (right), of the reduced surrogate ($\widetilde{r} = \tau_E \left(\widetilde{\Delta E}\right) \Psi' + \tau_Z \left(\widetilde{\Delta Z}\right) \omega'$), reference ($\overline{r}$ given by (7)) and unparameterised ($\overline{r} = 0$) solution. The surrogates were both conditioned on $Z^{\mathcal{R}}, E^{\mathcal{R}}, U^{\mathcal{R}}, S^{\mathcal{R}}$ of the previous time step.

For brevity, we only show a representative sample of results. Consider the results of Figure 8, which shows the pdfs obtained using the surrogates $\Delta E_{i+1} \mid Z_i^{\mathcal{R}}, E_i^{\mathcal{R}}, U_i^{\mathcal{R}}, S_i^{\mathcal{R}}$ and $\Delta Z_{i+1} \mid Z_i^{\mathcal{R}}, E_i^{\mathcal{R}}, U_i^{\mathcal{R}}, S_i^{\mathcal{R}}$, with 10 bins per conditioning variable. As expected, the pdfs do not show the same (near) perfect overlap with the reference compared to the training case of Figure 4, but the match is still accurate. Surrogates conditioned on e.g. $\left(Z^{\mathcal{R}}, E^{\mathcal{R}}, U^{\mathcal{R}}\right)$ or $\left(Z^{\mathcal{R}}, U^{\mathcal{R}}, S^{\mathcal{R}}\right)$ showed fairly similar results. Somewhat degraded performance (although overall still better than $\overline{r} = 0$), is obtained when conditioning on $\left(E^{R}, U^{\mathcal{R}}, S^{\mathcal{R}}\right)$, see Figure 9. While the $Z^{\mathcal{R}}$ bias is still corrected for, the pdfs of the surrogate underestimate the variance. The only exception, which did not improve upon the unparameterized model, was when conditioning on $\tau_E S'$ and $\tau_Z Z'$, despite the high correlations of Table 1. A possible cause is that, when predicting, we are forced to condition on $\tau_E(\widetilde{\Delta E})S'$ instead of $\tau_E\left(\Delta E\right)S'$, as the latter is not available outside the training period. Perhaps using conditioning variables such as $\tau_E S'$ and $\tau_Z Z'$ should be viewed as some form of overfitting, leading to a surrogate which is unlikely to generalize well beyond the training set. A possible remedy might be to increase the time lag [17].

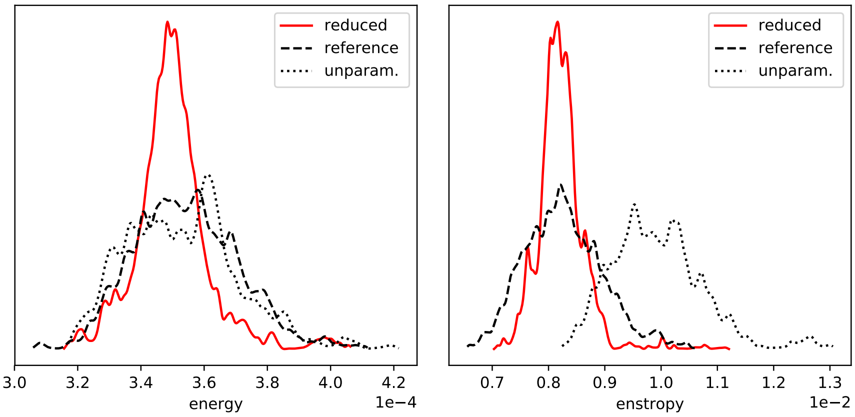


Figure 9: The pdfs of the energy (left) and enstrophy (right), of the reduced surrogate ($\widetilde{r} = \tau_E \left(\widetilde{\Delta E}\right) \Psi' + \tau_Z \left(\widetilde{\Delta Z}\right) \omega'$), reference ($\overline{r}$ given by (7)) and unparameterised ($\overline{r} = 0$) solution. The surrogates were both conditioned on $E^{\mathcal{R}}, U^{\mathcal{R}}, S^{\mathcal{R}}$ of the previous time step.

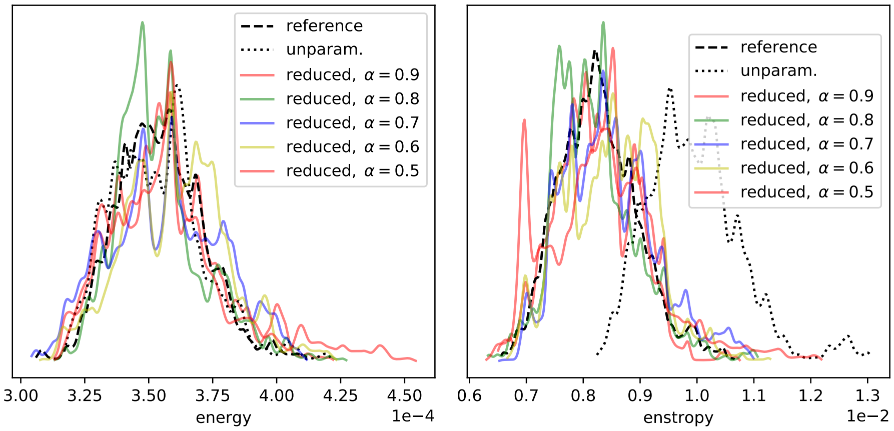


Figure 10: The pdfs of the energy (left) and enstrophy (right), of several extrapolated reduced surrogates ($\widetilde{r} = \tau_E \left(\widetilde{\Delta E}\right) \Psi' + \tau_Z \left(\widetilde{\Delta Z}\right) \omega'$), reference ($\overline{r}$ given by (7)) and unparameterised ($\overline{r} = 0$) solution. The surrogates were both conditioned on $Z^{\mathcal{R}}, E^{\mathcal{R}}, U^{\mathcal{R}}, S^{\mathcal{R}}$ of the previous time step.

4.3 Results T3

Predictive capability outside the training set should be the goal of any data-informed numerical simulation tool. In our case, this goal concerns prediction outside the time interval covered by the training set. We take tentative steps in this direction by incrementally reducing the time interval of the training set for the $\Delta E_{i+1} \mid Z_i^{\mathcal{R}}, E_i^{\mathcal{R}}, U_i^{\mathcal{R}}, S_i^{\mathcal{R}}$ and $\Delta Z_{i+1} \mid Z_i^{\mathcal{R}}, E_i^{\mathcal{R}}, U_i^{\mathcal{R}}, S_i^{\mathcal{R}}$ surrogates, while keeping the simulation time T_{sim} fixed to 8 years. Figure 10 shows the resulting pdfs, obtained using a training set spanning the first $T_{train} = \alpha T_{sim}$ years, with $\alpha \in \{0.9, 0.8, 0.7, 0.6, 0.5\}$. No significant deviation from the unextrapolated T2 test case is observed, which demonstrates the predictive capability of the surrogate method.

Finally, we note that all results can replicated via the source code and corresponding input files, available for download at [3].

5 CONCLUSION & OUTLOOK

We presented a method to create a stochastic surrogate model, conditioned on time-lagged observable variables, from a set of training data of a multiscale dynamical system. The novelty of our approach is found in the derivation of model-error source terms designed to track chosen spatially-integrated statistics of interest. We denote these as 'reduced' model error terms, as they lead to a significant reduction in the amount of training data. Although using less data might seem counter productive, we argue that this leads to an easier surrogate construction. Furthermore, our reduced framework allows us to step away from a fully-data driven, physics-blind, surrogate, and inform part of our model-error term based on the transport equations of the target statistics.

Future work includes further testing the extrapolative capability of the method. Another interesting research option would be to contrast the performance of our conditional time-lagged surrogate with machine-learning alternatives, such as random forests or neural nets. Recent relevant work also considered a combination of both approaches[13]. Finally, a further interesting avenue of future research is the *a-priori* incorporation of constraints from mathematical physics. For instance, when rewriting the eddy forcing in tensor format, certain constraints on the tensor shape can be found [19]. Such an approach opens up the possibility for efficient, physics-constrained uncertainty quantification, see e.g. [5] for examples in steady flow problems or [8] for large-eddy simulations.

ACKNOWLEDGEMENTS

This research is funded by the Netherlands Organization for Scientific Research (NWO) through the Vidi project ”Stochastic models for unresolved scales in geophysical flows”, and from the European Union Horizon 2020 research and innovation programme under grant agreement #800925 (VECMA project).

We also thank W.T.M. Verkley for making his vorticity equation source code available to us.

REFERENCES

[1] P.S. Berloff. Random-forcing model of the mesoscale oceanic eddies. *Journal of Fluid Mechanics*, 529:71–95, 2005.

[2] D. Crommelin and E. Vanden-Eijnden. Subgrid-scale parameterization with conditional markov chains. *Journal of the Atmospheric Sciences*, 65(8):2661–2675, 2008.

[3] W.N. Edeling. Tau ez - uncecomp branch (github repository). `https://github.com/wedeling/TAU_EZ/tree/uncecomp`, 2019.

[4] W.N. Edeling. vorticity-solver (github repository). `https://github.com/wedeling/vorticity-solver`, 2019.

[5] W.N. Edeling, G. Iaccarino, and P. Cinnella. Data-free and data-driven rans predictions with quantified uncertainty. *Flow, Turbulence and Combustion*, 100(3):593–616, 2018.

[6] P.R. Gent and J.C. Mcwilliams. Isopycnal mixing in ocean circulation models. *Journal of Physical Oceanography*, 20(1):150–155, 1990.

[7] I. Grooms and L. Zanna. A note on 'toward a stochastic parameterization of ocean mesoscale eddies'. *Ocean Modelling*, 113:30–33, 2017.

[8] L. Jofre, S.P. Domino, and G. Iaccarino. A framework for characterizing structural uncertainty in large-eddy simulation closures. *Flow, Turbulence and Combustion*, 100(2):341–363, 2018.

[9] P. Mana and L. Zanna. Toward a stochastic parameterization of ocean mesoscale eddies. *Ocean Modelling*, 79:1–20, 2014.

[10] R. Maulik, O. San, A. Rasheed, and P. Vedula. Subgrid modelling for two-dimensional turbulence using neural networks. *Journal of Fluid Mechanics*, 858:122–144, 2019.

[11] J.C. McWilliams. The emergence of isolated coherent vortices in turbulent flow. *Journal of Fluid Mechanics*, 146:21–43, 1984.

[12] T. Palmer and P. Williams. *Stochastic physics and climate modelling*. Cambridge University Press Cambridge, UK, 2010.

[13] S. Pan and K. Duraisamy. Data-driven discovery of closure models. *SIAM Journal on Applied Dynamical Systems*, 17(4):2381–2413, 2018.

[14] R. Peyret. *Spectral methods for incompressible viscous flow*, volume 148. Springer Science & Business Media, 2013.

[15] J. Thuburn, J. Kent, and N. Wood. Cascades, backscatter and conservation in numerical models of two-dimensional turbulence. *Quarterly Journal of the Royal Meteorological Society*, 679(140):626–638, 2014.

[16] N. Verheul and D. Crommelin. Data-driven stochastic representations of unresolved features in multiscale models. *Commun. Math. Sci*, 14(5):1213–1236, 2016.

[17] N. Verheul, J. Viebahn, and D. Crommelin. Covariate-based stochastic parameterization of baroclinic ocean eddies. *Mathematics of Climate and Weather Forecasting*, 3(1):90–117, 2017.

[18] W.T.M. Verkley, P.C. Kalverla, and C.A. Severijns. A maximum entropy approach to the parametrization of subgrid processes in two-dimensional flow. *Quarterly Journal of the Royal Meteorological Society*, 142(699):2273–2283, 2016.

[19] S. Waterman and J.M. Lilly. Geometric decomposition of eddy feedbacks in barotropic systems. *Journal of Physical Oceanography*, 45(4):1009–1024, 2015.

[20] L. Zanna, P. Mana, J. Anstey, T. David, and T. Bolton. Scale-aware deterministic and stochastic parametrizations of eddy-mean flow interaction. *Ocean Modelling*, 111:66–80, 2017.

A ENERGY AND ENSTROPHY EQUATIONS

For convenience, we reproduce certain relevant derivations regarding the $E^{\mathcal{R}}$ and $Z^{\mathcal{R}}$ transport equations from [18]. The energy (density) is defined as

$$E^{\mathcal{R}} := \frac{1}{2}\left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}} \mathrm{d}x\mathrm{d}y, \tag{26}$$

where $\mathbf{V}^{\mathcal{R}}$ is the vector containing the velocity components in x and y direction. It can be rewritten as $E^{\mathcal{R}} = -\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$ via

$$\mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}} = \nabla\psi^{\mathcal{R}} \cdot \nabla\psi^{\mathcal{R}} = \nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right) - \psi^{\mathcal{R}}\nabla^2\psi^{\mathcal{R}} = \nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right) - \psi^{\mathcal{R}}\omega^{\mathcal{R}} \tag{27}$$

The first equality follows from the definition $\mathbf{V}^{\mathcal{R}} := \left(-\partial\psi^{\mathcal{R}}/\partial y, \partial\psi^{\mathcal{R}}/\partial x\right)^T$, while the second stems from the product rule of a scalar ($\psi^{\mathcal{R}}$) and a vector ($\nabla\psi^{\mathcal{R}}$):

$$\nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right) = \nabla\psi^{\mathcal{R}} \cdot \nabla\psi^{\mathcal{R}} + \psi^{\mathcal{R}}\nabla^2\psi^{\mathcal{R}}. \tag{28}$$

Finally, the last equality of (27) simply follows from the governing equations (1). The term $\nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}\right)$ disappears when integrated over the spatial domain, after application of the divergence theorem in combination with the doubly periodic boundary conditions. This leaves $E^{\mathcal{R}} = -\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)/2$. To obtain the energy equation, start with

$$\frac{\mathrm{d}E^{\mathcal{R}}}{\mathrm{d}t} = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \frac{\partial}{\partial t}\left[\frac{1}{2}\mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}}\right]\mathrm{d}x\mathrm{d}y = \left(\frac{1}{2\pi}\right)^2 \int_0^{2\pi}\int_0^{2\pi} \mathbf{V}^{\mathcal{R}} \cdot \frac{\partial\mathbf{V}^{\mathcal{R}}}{\partial t}\mathrm{d}x\mathrm{d}y. \tag{29}$$

Similar to the analysis above, we use the relation $\mathbf{V}^{\mathcal{R}} \cdot \mathbf{V}^{\mathcal{R}}_t = \nabla \cdot \left(\psi^{\mathcal{R}}\nabla\psi^{\mathcal{R}}_t\right) - \psi^{\mathcal{R}}\omega^{\mathcal{R}}_t$ (where the subscript t denotes $\partial/\partial t$) to obtain

$$\frac{\mathrm{d}E^{\mathcal{R}}}{\mathrm{d}t} = -\left(\Psi^{R}, \frac{\partial\omega^{\mathcal{R}}}{\partial t}\right) = \left(\psi^{\mathcal{R}}, P^{\mathcal{R}}J\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)\right) - \nu\left(\psi^{\mathcal{R}}, \nabla^2\omega^{\mathcal{R}}\right) - \mu\left(\psi^{\mathcal{R}}, F - \omega^{\mathcal{R}}\right) + \left(\psi^{\mathcal{R}}, \overline{r}\right) \tag{30}$$

Using integration by parts and the periodic boundary conditions it can be shown that the first term on the right-hand side satisfies $\left(\psi^{\mathcal{R}}, P^{\mathcal{R}}J\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right)\right) = \left(J\left(\psi^{\mathcal{R}}, \psi^{\mathcal{R}}\right), \omega^{\mathcal{R}}\right) = 0$, since the Jacobian of two equal arguments is zero [18]. Furthermore, using the self-adjoint nature of the Laplace operator, we have $\left(\psi^{\mathcal{R}}, \nabla^2\omega^{\mathcal{R}}\right) = \left(\nabla^2\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right) = \left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right)$. This leads to

$$\frac{\mathrm{d}E^{\mathcal{R}}}{\mathrm{d}t} = -\nu\left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right) - \mu\left(\psi^{\mathcal{R}}, F\right) + \mu\left(\psi^{\mathcal{R}}, \omega^{\mathcal{R}}\right) + \left(\psi^{\mathcal{R}}, \overline{r}\right), \tag{31}$$

which equals (13). Using the same procedure, the evolution equation for the enstrophy reads

$$\frac{\mathrm{d}Z^{\mathcal{R}}}{\mathrm{d}t} = \left(\omega^{\mathcal{R}}, \frac{\partial\omega^{\mathcal{R}}}{\partial t}\right) = \nu\left(\omega^{\mathcal{R}}, \nabla^2\omega^{\mathcal{R}}\right) + \mu\left(\omega^{\mathcal{R}}, F\right) - \mu\left(\omega^{\mathcal{R}}, \omega^{\mathcal{R}}\right) - \left(\omega^{\mathcal{R}}, \overline{r}\right). \tag{32}$$

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

EXPLORATION OF MULTIFIDELITY APPROACHES FOR UNCERTAINTY QUANTIFICATION IN NETWORK APPLICATIONS

Gianluca Geraci[1], Laura P. Swiler[1], Jonathan Crussell[2] and Bert J. Debusschere[3]

[1]Sandia National Laboratories
Optimization and Uncertainty Quantification, Albuquerque, NM
e-mail: {ggeraci,lpswile}@sandia.gov

[2]Sandia National Laboratories
Data Science & Cyber Analytics, Livermore, CA
e-mail: jcrusse@sandia.gov

[3] Sandia National Laboratories
Reacting Flow Research, Livermore, CA
e-mail: bjdebus@sandia.gov

Keywords: Cyber Analytics, Emulytics, Uncertainty Quantification, Monte Carlo, Multifidelity

Abstract. *Communication networks have evolved to a level of sophistication that requires computer models and numerical simulations to understand and predict their behavior. A network simulator is a software that enables the network designer to model several components of a computer network such as nodes, routers, switches and links and events such as data transmissions and packet errors in order to obtain device and network level metrics. Network simulations, as many other numerical approximations that model complex systems, are subject to the specification of parameters and operative conditions of the system. Very often the full characterization of the system and their input is not possible, therefore Uncertainty Quantification (UQ) strategies need to be deployed to evaluate the statistics of its response and behavior. UQ techniques, despite the advancements in the last two decades, still suffer in the presence of a large number of uncertain variables and when the regularity of the systems response cannot be guaranteed. In this context, multifidelity approaches have gained popularity in the UQ community recently due to their flexibility and robustness with respect to these challenges. The main idea behind these techniques is to extract information from a limited number of high-fidelity model realizations and complement them with a much larger number of a set of lower fidelity evaluations. The final result is an estimator with a much lower variance, i.e. a more accurate and reliable estimator can be obtained. In this contribution we investigate the possibility to deploy multifidelity UQ strategies to computer network analysis. Two numerical configurations are studied based on a simplified network with one client and one server. Preliminary results for these tests suggest that multifidelity sampling techniques might be used as effective tools for UQ tools in network applications.*

1 INTRODUCTION

Uncertainty quantification (UQ) is a field of study drawing from statistics, mathematics, and computational science [37] [29] [6]. Simulation models for engineering and physics applications are often developed to help assess a design or performance requirement. The past few decades have seen an unprecedented increase in the complexity and sophistication of computational simulation models due to improvements in computer architectures/processors as well as in advanced software frameworks. Typically, one does not run a simulation model just once but multiple times, to explore the effects of different parameters and scenarios. The capability to quantify the effects of uncertainty when using a model to inform a scientific or regulatory decision is critical. There have been a number of large-scale regulatory assessments performed using uncertainty quantification on computational models. Notable examples include the performance of geologic repositories for the disposal of nuclear waste [20], computational fluid dynamics for aircraft design, and climate model predictions [38].

The basic framework for uncertainty quantification is identifying and characterizing uncertain input parameters, representing the input uncertainty (typically in the form of probability distributions), propagating uncertainties in the inputs through the model (typically by drawing samples of the uncertain parameters from their respective distributions and running the model at those sample values to create an ensemble of model runs), and analyzing the output to determine statistics on the output quantities of interest. A number of activities related to UQ that can inform the UQ process include sensitivity analysis [36], verification and validation [32], and dimension reduction [4]. There are many related issues and research directions in UQ which include sample design (e.g. how does one choose the input samples at which to run the model), inclusion of other uncertainties (e.g. numerical uncertainties, uncertainties in observational data used to calibrate models, model form uncertainty), and types of uncertainties (e.g. aleatory, epistemic, interval uncertainties). The scientific computing community has endeavored to develop methods which are as efficient as possible to perform UQ on computationally expensive simulation models. In this paper, we present one particular class of UQ methods called multifidelity methods that we feel is well suited for the analysis of network and cyber modeling. Multifidelity UQ techniques have gained popularity in the last decade or so when the need for UQ of high-fidelity numerical simulations led to the design of techniques capable of containing the overall computational burden. In this contribution, the focus in on multifidelity sampling strategies given the features of the network applications. In a broad sense, it is possible to include the so-called multi-level and multi-index approaches [15, 16, 19], multifidelity MC [33, 34], multilevel-multifidelity techniques [11, 7, 14] and approximate control variates [17] in this class of approaches. Multifidelity UQ strategies have been successfully used in a variety of context ranging from turbulent-laden flows in a radiative environment [22], aerospace applications [14] and cardiovascular flows [9]. Our goal in this work is to explore these methods in the context of UQ on computer network applications.

Network models can aid network operators and designers when making decisions. For instance, network operators can use models to understand the potential impacts of changes to their network before affecting the operation system. Network designers can use models to understand design trade-offs before network creation. These models can drastically reduce the cost and risk of deployment. The terminology of network modeling generally designates two distinct choices: simulations and emulations. Simulations are similar to their physics-modeling counterparts: they use a deep understanding of the underlying processes to simulate network components and interactions in software. Emulations, on the other hand, run the real software on virtualized hardware which allows them to capture unknown or poorly understood behaviors. This realism comes at the cost of increased computational cost.

The reasons for performing uncertainty quantification on network models is similar to that of engineering models: to understand how uncertainty in inputs (such as device and network configuration, threats, and network topology) propagates to network outputs (such as network availability, traffic loads, etc.) In this exploratory study, the focus is on multifidelity sampling UQ strategies which has the potential to naturally treat system responses with noise, bifurcations, or discontinuities in the presence of a large number of uncertain parameters. This scenario is expected to be particularly relevant for network simulations and emulations.

The remainder of the manuscript is organized as it follows. In Section 2 the network modeling approach is described and, in particular, two network softwares are described, namely a simulator `ns`-3 and an emulator `minimega`. Section 3 introduces some generalities on the multifidelity sampling approaches. Numerical examples are presented in Section 4. Conclusions close the paper in Section 5.

2 NETWORK MODELS

As stated earlier, there are generally two types of network modeling: network simulation and network emulation. Network simulators rely on careful implementations of how “real systems” respond to inputs and the processes that drive them which makes them useful to study well-understood behaviors of systems but not necessarily emergent behaviors. Depending on what the model is being used for, this could require a very in-depth understanding of the system that we wish to model. Simulations can even aid designers that wish to understand the trade-offs in the underlying processes when the real software has not been created yet. On the other hand, network emulation runs the real software on virtualized hardware which decreases the semantic gap between the model and the operational system.

Comparing simulations and emulations, we find that they have different strengths. Simulations can be fast to develop and capture the core behavior of well-understood system. Since they control the clock, simulations can run faster than real time. Additionally, multiple network simulations can run in parallel because they are not timing dependent or reliant on virtualized hardware which may be limited. This means that we can run many instances of our network simulation for every emulation. Emulations, which run the real software, should more closely match the real systems. In our multifidelity UQ, we aim to leverage the strengths of both forms of modeling. We can use the inexpensive network simulation as our low-fidelity model and the emulation as the high-fidelity model.

In addition to network modeling, network operators and designers may also use physical testbeds in order to understand their systems. Physical testbeds are costly to build and maintain and may not be suitable for all types of tests. Related work has compared network emulation to physical testbeds to discover where and how they differ [5]. In future work, we could expand upon our levels of multifidelity to include results from a physical testbed (or even an operational network) as the highest fidelity.

2.1 The ns-3 network simulator

ns-3 [21] is a discrete network simulator for Internet Protocol (IP) and non-IP networks. It has been widely used by the academic community to understand existing and emerging network designs and protocols [8, 31, 35, 39]. ns-3 allows users to construct simulations from reusable components to configure nodes, topologies, and applications.

Interestingly, ns-3 supports leveraging code from real applications or kernels in the simulator. For example, there are tests to incorporate the entire Linux kernel networking stack. This hybridization of ns-3 likely increases its fidelity which benefits our multifidelity UQ approach since the more correlated our low- and high-fidelity models are, the faster the convergence.

2.2 The minimega network emulator

`minimega` [28] is a toolset developed by Sandia National Laboratories to launch and manage virtual machines (VMs) to emulate networks. It wraps QEMU [3] and KVM [23] to launch the VMs and Open vSwitch [10] to connect the VMs to virtual networks with user-defined topologies. `minimega` includes a scriptable interface that includes many APIs to support the experimentation lifecycle such as capturing data and running services.

3 MULTIFIDELITY UNCERTAINTY QUANTIFICATION

In this section a multifidelity sampling approach is described. For this particular application, it is reasonable to assume that the high-fidelity (HF) model is unbiased and that lower accuracy network representations are generated and added to a limited number of HF evaluations in order to decrease the variance of the sampling estimator, *i.e.* increasing its reliability from an user perspective. This is a slightly different scenario than, for instance, a classical multilevel MC application where usually it is possible to control the accuracy (bias) of the high-fidelity model in order to balance the full mean square error of the estimator [16]. For a generic quantity of interest (QoI) of the system, $Q : \mathbb{R}^d \ni \Xi \to \mathbb{R}$, *e.g.* the number of requests per second processed by a server, the goal is to compute some statistics. In this work, the expected value $\mathbb{E}\left[Q\right]$ of the QoI is considered, but an extension to higher-order moments it is also possible. The Monte Carlo (MC) estimator for $\mathbb{E}\left[Q\right]$ can be written as

$$\mathbb{E}\left[Q\right] = \int_{\Xi} Q(\boldsymbol{\xi}) p(\boldsymbol{\xi}) \,\mathrm{d}\boldsymbol{\xi} \approx \hat{Q}_N^{\mathrm{MC}} = \frac{1}{N}\sum_{i=1}^{N} Q(\boldsymbol{\xi}^{(i)}) = \frac{1}{N}\sum_{i=1}^{N} Q^{(i)}, \tag{1}$$

where N realizations of the vector of random input $\boldsymbol{\xi} \in \Xi$ are drawn according to the joint probability distribution $p(\boldsymbol{\xi})$. For each realization of the vector of random input $\boldsymbol{\xi}$, the value of the QoI $Q^{(i)} = Q(\boldsymbol{\xi}^{(i)})$ is evaluated by performing a network simulation and extracting the desired quantity. $\hat{Q}^{\mathrm{MC}}$ represents a random variable itself and, if Q has finite variance $\mathbb{V}ar\left[Q\right] < \infty$, it is possible to show that the estimator is unbiased, *i.e.* $\mathbb{E}\left[\hat{Q}^{\mathrm{MC}}\right] = \mathbb{E}\left[Q\right]$ and

$$\mathbb{V}ar\left[\hat{Q}_N^{\mathrm{MC}}\right] = \frac{\mathbb{V}ar\left[Q\right]}{N}. \tag{2}$$

A classical result, that follows from the central limit theorem, states that for $N \to \infty$ the error $\hat{Q}_N^{\mathrm{MC}} - \mathbb{E}\left[Q\right]$ is distributed as a normal distribution with zero mean and variance equal to $\mathbb{V}ar\left[\hat{Q}_N^{\mathrm{MC}}\right]$. It follows that the root mean square error (RMSE) is equal to $\mathbb{V}ar^{\frac{1}{2}}\left(Q\right)/\sqrt{N}$, from which it follows the well known rate of convergence of $\mathcal{O}(N^{-1/2})$ for the MC estimator. The inspection of the RMSE reveals important features of the MC estimator that make it particularly suited for the network applications considered in this work. Albeit the slow rate of convergence corresponds to a limit in obtaining accurate statistics with a limited number of realizations of the QoI (*i.e.* network simulations), it is also possible to note that neither the dimensionality of the system or the smoothness of Q appear in the RMSE. This situation is different from other quadrature rules in which the rate of convergence is ultimately related to the order of continuous derivatives of the integrand and the number of dimensions. The MC estimator is therefore a convenient, and very often the only practical choice, when one deals with both noisy responses and possibly bifurcations/discontinuities of the system response. Both cases are common in network simulations. Moreover, it is reasonable to imagine that for a realistic network topology the number of uncertain parameters, d, might easily reach order hundreds of parameters, thus preventing the efficient use of other UQ techniques like spectral methods, *i.e.* Polynomial Chaos expansions (PC) [27]. Given the prohibitive computational cost required for each network simulation, which limits the maximum affordable number N, in order to decrease the RMSE of the estimator the only viable solution is to change the problem in a way that reduces $\mathbb{V}ar^{\frac{1}{2}}\left(Q\right)$ while keeping the value of $\mathbb{E}\left[Q\right]$ unaltered. It is important to note that, whenever a computationally cheaper evaluation of Q might be obtained without sacrificing the overall numerical accuracy, this possibility should be considered. In this work, every model introduced to alleviate the computational burden is assumed to introduce a non-negligible bias with respect to the target network system (which in this work is considered the truth system).

The pivotal idea of the multifidelity sampling strategies is the following. A small set of evaluations of the high-fidelity system is used to guarantee the convergence of the estimator to its statistics; in addition to this set, a larger number of evaluations from inaccurate but more computationally efficient systems (*e.g.* `ns`-3 network simulations as opposed to high-fidelity `minimega` emulations) is aggregated with the high-fidelity set in order to obtain an estimator with the lowest variance given a prescribed computational budget. The so-called optimal control variate (OCV) method can be used for this scope [24, 26, 25]. In the OCV estimator, a MC estimator based on N high-fidelity evaluation, $\hat{Q}_N^{HF,\mathrm{MC}}$, is extended to include weighted sums of contributions based on M lower-fidelity models for which we consider their expected value to be known *a priori*

$$\hat{Q}^{\mathrm{OCV}} = \hat{Q}_N^{\mathrm{HF,MC}} + \sum_{i=1}^{M} \alpha_i \left(\hat{Q}_i - \mu_i \right), \tag{3}$$

where $\hat{Q}_i$ and μ_i represent a MC estimator and the exact mean of the ith low-fidelity model, respectively and the weights $\alpha = \left[\alpha_1, \dots, \alpha_M\right]^{\mathrm{T}} \in \mathbb{R}^M$ are introduced as optimization parameters. For simplicity and without loss of generality, the number of the N_i evaluations of the ith low-fidelity model is assumed proportional to the number of high-fidelity simulation N through a coefficient r_i, *i.e.* $N_i = \lceil r_i N \rceil$. By means of simple manipulations it is possible to show that such estimator is unbiased, *i.e.* $\mathbb{E}\left[\hat{Q}^{\mathrm{OCV}}\right] = \mathbb{E}\left[\hat{Q}_N^{\mathrm{HF,MC}}\right] = \mathbb{E}\left[Q\right]$ for any choice of the vector α. Under this framework, once the covariance matrix $C \in \mathbb{R}^{M \times M}$ amongst Q_i and the vector of covariances c between Q and each Q_i are defined, the optimal weights $\alpha^\star$ are obtained as

$$\alpha^\star = \underset{\alpha}{\operatorname{argmin}}\, \mathbb{V}ar\left[\hat{Q}^{\mathrm{OCV}}\right] = -C^{-1}c, \tag{4}$$

and the corresponding variance is

$$\mathbb{V}ar\left[\hat{Q}^{\mathrm{OCV}}\right] = \frac{\mathbb{V}ar\left[Q\right]}{N} \left(1 - \frac{c^{\mathrm{T}}C^{-1}c}{\mathbb{V}ar\left[Q\right]}\right) = \frac{\mathbb{V}ar\left[Q\right]}{N} \left(1 - R_{\mathrm{OCV}}^2\right). \tag{5}$$

It is evident that $R_{\mathrm{OCV}}^2 = \frac{c^{\mathrm{T}}C^{-1}c}{\mathbb{V}ar[Q]}$ represents a positive quantity and $0 \leq R_{\mathrm{OCV}}^2 \leq 1$, therefore the variance of the OCV estimator is always lesser or equal than the corresponding MC variance (based on high-fidelity realizations only). It is also important to note that if the OCV estimator is obtained as an extension of a MC estimator based on N high-fidelity simulations by adding N_i low-fidelity simulations for $i = 1, \ldots, M$, its overall cost would naturally be higher than MC. An optimal sample allocation between models is in general needed in order to obtain an efficient OCV estimator given a prescribed computational cost. Although the OCV method provides an elegant mathematical solution to decrease the RMSE of a plain MC estimator, in a practical computational settings it is necessary to estimate the values of μ_i which are unknown at the beginning of the computations, *e.g.* in this work it is not even known a priori the expected value of the `ns`-3 QoI. In order to address this limitation, it is possible to partition the set of low-fidelity evaluation in two (possibly overlapping) subsets and using each of them to compute the term $\hat{Q}_i$ and an approximation of μ_i, $\hat{\mu}_i$, respectively. Interesting properties and analogy between this approach and other multifidelity approaches discussed in literature can be drawn for this framework, called Approximate Control Variate [17], however this is beyond the scope of the present work. We only note here that for particular choices of the low-fidelity simulations partitioning, it is possible to show that these estimators might exhibit an higher variance reduction than an OCV estimator with only one low-fidelity model, OCV-1 (although the final variance of the estimator would ultimately depend on the possibility to approach the theoretical variance reduction without incurring in a overwhelming low-fidelity cost). On the contrary, this possibility is prevented in more classical recursive schemes for which it is possible to demonstrate that the variance reduction is lesser than the one corresponding to OCV-1 [17].

In the present work, the goal is to demonstrate that is indeed possible to use the multifidelity sampling idea in the context of network simulations, therefore the extension to the most efficient partitioning scheme of the low-fidelity evaluations N_i is left for a future work. Given this narrower focus, here the case of a single low-fidelity model is explicitly addressed. For the case of a single low-fidelity model two possible choices of partitioning for the low-fidelity simulations are available. The set can be split in both overlapping or independent sets of simulations (by construction we assume that the cardinality of the set adopted to evaluate $\hat{\mu}_i$ is larger than the one corresponding to the set used for $\hat{Q}_i$). In both cases, the performances of the estimator (in term of its variance) are equivalent (the difference is limited to a dissimilar value for the optimal coefficient α_1), therefore in this work the case of fully overlapping partitioning is considered. Under these assumptions the ACV-1 estimator is equivalent to the multifidelity Monte Carlo (MFMC) estimator adopted in [33, 30, 34], *i.e.* the term $\hat{Q}_i$ is computed by means of N evaluations (shared with the HF model) whereas the approximation $\hat{\mu}_i$ is evaluated by adding another set of $N_1 - N = (r_1 - 1)N$ independent evaluations. The final form of the estimator is

$$\hat{Q}^{\mathrm{ACV}-1} = \frac{1}{N}\sum_{i=1}^{N} Q^{(i)} + \alpha_1 \left(\frac{1}{N}\sum_{i=1}^{N} Q_1^{(i)} - \frac{1}{N_1}\left(\sum_{i=1}^{N} Q_1^{(i)} + \sum_{j=1}^{N_1-N} Q_1^{(j)} \right) \right). \tag{6}$$

Simple manipulations lead to an optimal coefficient selection where

$$\begin{aligned} \alpha_1^{\star} = -C^{-1}c &= -\left(\mathbb{V}ar\left[Q_1\right]\right)^{-1} \left(\rho_1 \mathbb{V}ar^{\frac{1}{2}}\left(Q_1\right) \mathbb{V}ar^{\frac{1}{2}}\left(Q\right) \right) \\ &= -\rho_1 \frac{\mathbb{V}ar^{\frac{1}{2}}\left(Q\right)}{\mathbb{V}ar^{\frac{1}{2}}\left(Q_1\right)}, \end{aligned} \tag{7}$$

where ρ_1 denotes Pearson's correlation coefficient. This coefficient choice corresponds to a minimum variance for the multifidelity estimator (ACV-1) equal to

$$\mathbb{V}ar\left[\hat{Q}^{\mathrm{ACV}-1}\right] = \frac{\mathbb{V}ar\left[Q\right]}{N}\left(1 - \frac{r_1 - 1}{r_1}\rho_1^2\right). \tag{8}$$

It is important to note that, for the case of one single low-fidelity model, the OCV estimator reduces to OCV-1 and its variance reduction term is $R^2_{\text{OCV}-1} = \rho_1^2$, thus the factor $\frac{r_1-1}{r_1} < 1$ stems from the need for estimating $\hat{\mu}_1$ in the ACV setting. The optimal sample allocation for the generic ACV estimator can be obtained in closed form only in the case of a single low-fidelity model, and again corresponds to the solution previously discussed in literature [33, 30, 34]: the optimal number of low-fidelity simulations to obtain a prescribed variance for the estimator, *i.e.* $\mathbb{V}ar\left[\hat{Q}^{\text{ACV}-1}\right] = \varepsilon^2$, corresponds to a value of r_1 equal to

$$r_1^\star = \sqrt{\frac{\mathcal{C}}{\mathcal{C}_1}\frac{\rho_1^2}{1-\rho_1^2}}, \tag{9}$$

where $\mathcal{C}$ and $\mathcal{C}_1$ corresponds to a measure of the computational cost (for instance the runtime of a simulation) for the high-fidelity and low-fidelity model, respectively. For particular choices of the low-fidelity partitioning that are based on imposing a recursive sampling scheme as noted in [17], a solution in closed form can be obtained also for $M > 1$ and this case is the MFMC introduced in [34], however in this latter case the variance reduction would always be $R^2_{\text{MFMC}} < \rho_1^2$. The corresponding number of required high-fidelity simulations to obtain a variance equal to ε^2 is obtained as

$$N^\star = \frac{\mathbb{V}ar\left[Q\right]}{\varepsilon^2}\left(1 - \frac{r_1^\star - 1}{r_1^\star}\rho_1^2\right) = \frac{\mathbb{V}ar\left[Q\right]}{\varepsilon^2}\Lambda(r^\star, \rho_1^2), \tag{10}$$

where the function $\Lambda = \Lambda(r, \rho^2) = \left(1 - \frac{r-1}{r}\rho^2\right)$ is introduced for compactness. The previous equation is also useful to quantify the computational cost reduction that might be obtained through the ACV-1 estimator. A MC estimator based on N_{MC} would have a variance equal to $\mathbb{V}ar\left[Q\right]/N_{MC}$, therefore for obtaining an ACV-1 estimator with equivalent variance the total number of high-fidelity simulations would be equal to $N_{MC}\Lambda(r^\star, \rho_1^2)$ and its total cost

$$\mathcal{C}_{\text{tot}} = N^\star\left(1 + \frac{\mathcal{C}_1}{\mathcal{C}}r^\star\right) = N_{\text{MC}}\Lambda(r^\star, \rho_1^2)\left(1 + \frac{\mathcal{C}_1}{\mathcal{C}}r^\star\right). \tag{11}$$

The ACV-1 estimator would be more efficient as the product $\Lambda(r^\star, \rho_1^2)\left(1 + \frac{\mathcal{C}_1}{\mathcal{C}}r^\star\right)$ decreases. It should be noted that this term depends only on the efficiency of the low-fidelity model, *i.e.* $\frac{\mathcal{C}_1}{\mathcal{C}}$, and its correlation with the high-fidelity model, *i.e.* ρ_1^2. To summarize, in a practical setting a multifidelity estimator as ACV-1 might be used to obtain slightly different objectives. Given a target accuracy for the estimator, the computational burden can be optimally distributed between the high- and low-fidelity model to guarantee that only the minimum possible computational cost is required. Alternatively, given a MC estimator based on high-fidelity simulations, an additional set of low-fidelity evaluations can be added to obtain the most efficient estimator given the computational effort invested in the high-fidelity realizations and the characteristics of the low-fidelity model, *i.e.* computational efficiency and correlation.

4 NUMERICAL EXAMPLES

In this section two numerical tests are conducted. A simple network topology consisting of a server and a client is studied under different operative conditions (see below for details). Two multifidelity test cases are considered in the following. In the first test, both high- and low-fidelity models are defined in the `ns-3` network simulator. This test case has also served

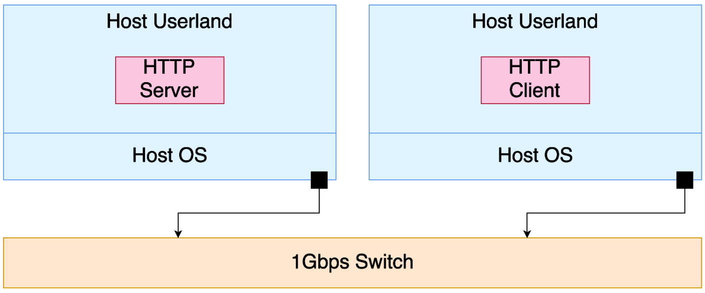


Figure 1: Simple network configuration used in this work for testing the multifidelity UQ approaches.

to test the coupling between `ns`-3 and the Sandia National Laboratories' UQ software Dakota [1, 2]. In contrast, for the second demonstration problem, the high-fidelity model is defined in `minimega` whereas the low-fidelity model is based on `ns`-3. For both test cases, the performance of two possible low-fidelity models are considered to provide a preliminary indication about the achievable trade-off between correlation and computational efficiency. For the `minimega`/`ns`-3 case, the use of both low-fidelity models at the same time is also considered as an exploratory investigation of the efficiency of the OCV strategy (which uses more than one low-fidelity model) compared to OCV-1 (which is based on a single low-fidelity model).

4.1 Experiment workload

For our numerical examples, we study a simple network topology consisting of a server and a client as depicted in Figure 1. The topology consists of two endpoints, one of which runs an HTTP server and the other of which runs an HTTP client. We will attempt to model the interactions between the client and server in this topology using both simulation and emulation.

The primary QoI in our scenario is the number of requests the client completes per second. We use multifidelity UQ to study the affects of several uncertain parameters such as the size of the HTTP response (*ResponseSize*), the delay introduced by the switch (*Delay*), and the speed of the switch (*DataRate*).

For the minimega emulation, we leveraged the models that we constructed during previous work [5]. Since this is an exploratory study, we do not attempt to vary *design* parameters such as the virtual network interface type as done in the previous work. In this work, we use e1000 network drivers and 1 virtual CPU.

For the ns-3 simulations, we created a topology to match Figure 1. We modified the built-in HTTP server and client implementations to better match the behaviors of the HTTP server and client used in the emulation (protonuke, a traffic generation tool in the minimega [28] toolset, and ApacheBench, a server benchmarking tool, respectively). Specifically, we modified the built-in simulated client to close and re-establish TCP connections after each request. We made this modification because the keep-alive behavior has been shown to have significant effects on HTTP performance [18]. In future work, we could explore how the correlation between the high- and low-fidelity models changes based on this modification. ApacheBench also supports keep-alives, creating yet another possible experiment. To match the emulation, we also parameterized the number of requests to perform and the response size in ns-3.

4.2 A simplified network topology in `ns-3`

The first demonstration is based on `ns-3` simulations for both the high- and low-fidelity models. The goal of the UQ analysis is to quantify the expected value for the number of requests per second in a scenario in which a total of 100 requests are exchanged between client and server with a payload of 16MB. The case of two uncertain network parameters is considered: the *DataRate* is considered uniformly distributed between 5 and 500 Megabits per second (Mbps), whereas the *Delay* is uniformly distributed between 1 and 3 milliseconds. The uncertain parameters and their distributions are reported in Table 1.

Uncertain variable	Disribution
DataRate	$\mathcal{U}(5, 500)$Mbps
Delay	$\mathcal{U}(1, 3)$ms

Table 1: Uncertain parameters and their distribution for the first demonstration case.

Two low-fidelity models are considered for this test case. The first low-fidelity model is obtained by reducing the payload from 16MB to 1MB, this model is dubbed simply LF. The second low-fidelity model is generated by both reducing the payload from 16MB to only 500B and the number of requests from 100 to 10 in an attempt to obtain a very fast simulation; this latter model is named LF*. The computational runtime for the three models and their computational cost normalized with respect to the high-fidelity model (HF) are reported in Table 2.

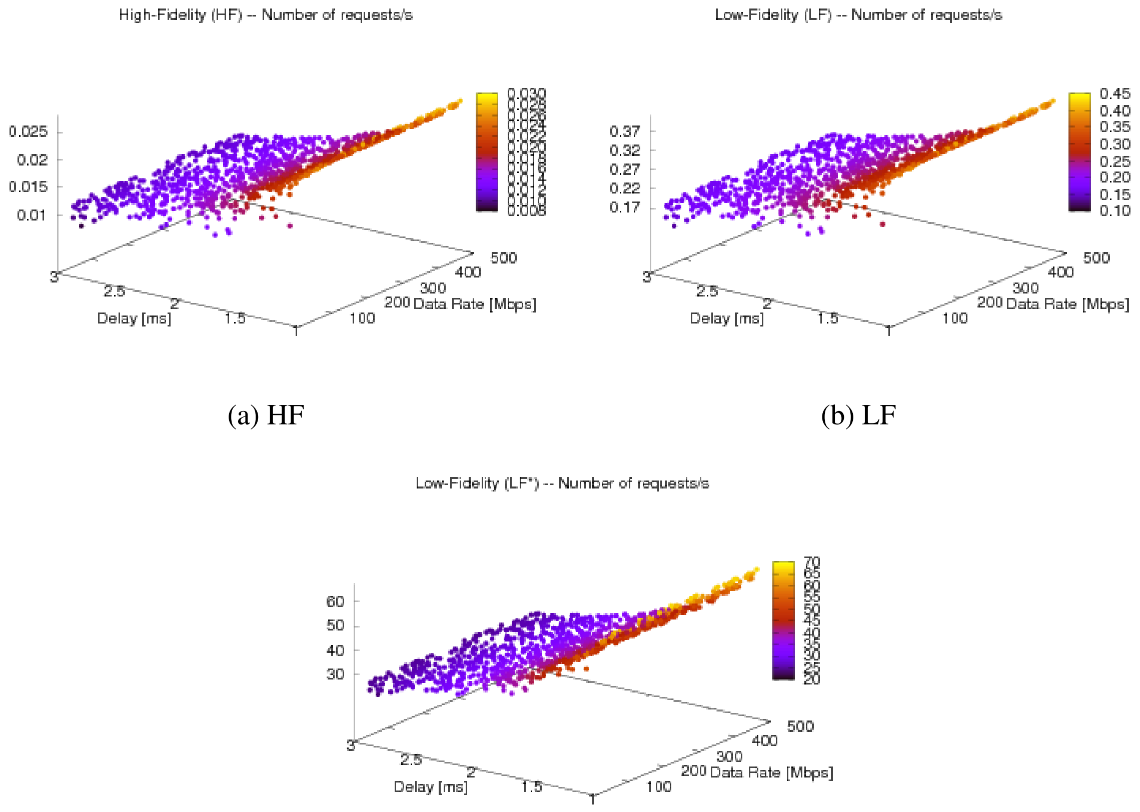
Model	runtime [s]	Normalized Cost
HF	1200	1
LF	50	0.0417
LF*	0.15	0.000125

Table 2: Runtime and computational cost for the models used in the first demonstration.

The responses of the three models are shown in Figure 2 for reference.

As a first result, a total of 700 high-fidelity simulations has been obtained for the high-fidelity model. Afterwards, a subset of the high-fidelity simulations has been extracted and paired with an equivalent number of low-fidelity simulations in order to estimate the correlation. Once the correlation between the high- and low-fidelity model has been evaluated, the optimal number of low-fidelity realizations has been computed by resorting to Eq. (9) and the $(r_1 - 1)N$ additional number of independent low-fidelity evaluations has been obtained. The total cost of the estimator, expressed in term of equivalent HF network simulations, is evaluated by resorting to Eq. (11) and the corresponding variance is computed with Eq. (8). In Figure 3a the value of the standard deviation of the ACV-1 estimator is reported with respect to the equivalent computational cost. Note that the convergence of all the estimators is roughly order $N^{-1/2}$, whereas their constant reflects the reduced variance achieved by introducing the low-fidelity evaluations as control variate.

From a practical standpoint, a reduced variance/standard deviation corresponds to a tighter confidence interval for the estimation of the expected value of the QoI. In order to demonstrate

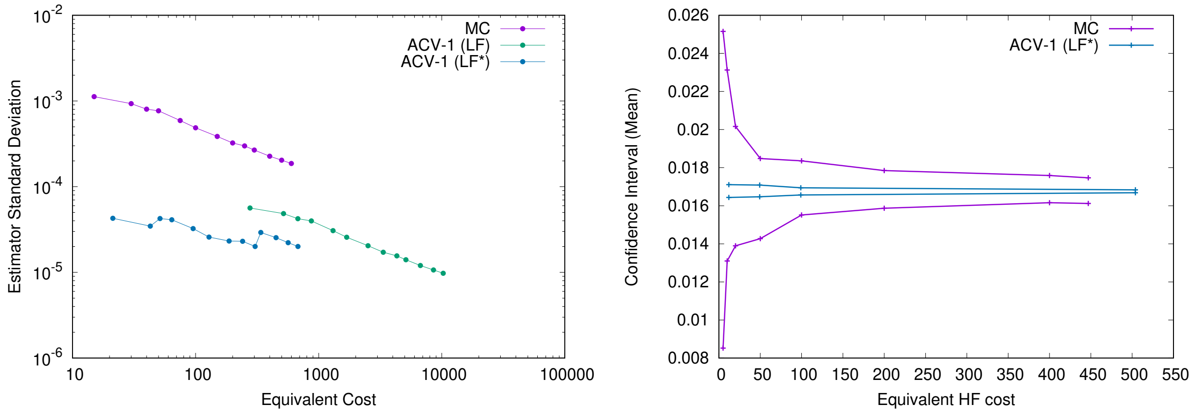


(a) HF

(b) LF

(c) LF*

Figure 2: Responses for the three models of the first demonstration case. The qualitative behavior is similar for the three cases, however the values of requests/s predicted by the two low-fidelity model is much higher than the HF model.



(a) Estimator Standard Deviation for the first test case.

(b) 99.7% confidence interval for the estimator value.

Figure 3: Estimator standard deviation for the simple MC and two variants of the ACV-1 estimator (a). The 99.7% confidence interval for the MC and ACV-1 (LF*) estimator values (b). In both figures, the QoI is the number of requests/s for which the expected value is desired.

this, Dakota has been coupled with ns-3 and several estimator evaluations have been obtained by targeting several estimator variances. The values of the 99.7% confidence intervals for MC and ACV-1 based on LF* are reported in Figure 3b. It is possible to observe that the ACV-1 estimator produces much more reliable estimations for the expected value of the QoI for a very limited computational cost: a very tight confidence interval can be obtained with a computational cost that corresponds to approximately only 100 HF network simulations. A comparable confidence interval cannot be obtained with even 450 HF simulations when using a plain single fidelity MC estimator.

4.3 Extension to minimega/ns-3 multifidelity analysis

The second experiment has a more realistic flavor and consists in the analysis of the same network configuration presented in Figure 1 by means of emulations based on minimega. Therefore, in this scenario minimega represents the unbiased high-fidelity model. The computational cost and resources needed to pursue a UQ study based of a network emulation model is generally prohibitive, thus a ns-3 simulation model is introduced as low-fidelity model. The goal of the UQ analysis is to compute the expected value of the number of requests per seconds for an operative conditions in which 100 requests are exchanged between server and client. For this test case, two uncertain parameters are considered. Consistently with the previous example the *DataRate* has been considered uniformly distributed between 5 and 500 Mbps. The second uncertain parameter has been chosen to be the payload, *i.e. ResponseSize* which is assumed to be log-uniformly distributed between 500B and 16MB. The uncertain parameters are reported in Table 3.

One of the main difference with respect to the previous test case is that the simulations in minimega are intrinsically stochastic, *i.e.* distinct repetitions of the same network configuration are expected to produce slightly different results. This is a product of emulation being subject to real-world timing in the virtual (and underlying physical) hardware and not running off a simulated clock. For this simple configuration it has been observed that a limited number of repetitions, of the order of 10 repetitions, was sufficient to characterize in average the response of a system for a fixed set of uncertain parameters. Complex network configurations might require the adoption of more sophisticated techniques to control the overall error induced on the statistics by the variability in minimega, however this is beyond the scope of the exploratory study conducted here and it is left for subsequent studies.

Uncertain variable	Disribution
DataRate	$\mathcal{U}(5, 500)$Mbps
ResponseSize	ln$\mathcal{U}(500, 16 \times 10^6)$B

Table 3: Uncertain parameters and their distribution for the second demonstration.

Two low-fidelity models are defined by using ns-3. The first low-fidelity model has been obtained by reducing the number of requests from 100 to 10. Additionally, the parameter *Delay*, which does not have a counterpart in minimega, has been chosen as 50ms by observing its impact on the response. In the future, in the presence of more complex network configurations and a large set of parameters, a formal calibration process might be performed. Hereinafter, this low-fidelity model is referred as LF. The second low-fidelity model has been obtained by reducing the number of requests to the extreme case of a single request and the parameter *Delay* has been fixed at the value of 5ms. The runtime and the normalized computational cost for the three models used in this numerical experiment are reported in Table 4.

Model	runtime [s]	Normalized Cost
HF	2680	1
LF	42.88	0.016
LF*	5.36	0.002

Table 4: Runtime and computational cost for the models used in the second demonstration.

It is important to note that in the following numerical experiments the computational cost is measured in terms of equivalent runtime for a serial execution. This is not necessarily the case when the low-fidelity simulations (as ns-3 in this case) might be potentially evaluated in parallel. In this latter case, the LF computational cost normalized by the HF cost would have been smaller (*i.e.* more efficient LF model) than the normalized cost reported in Table 4. Nonetheless, since the serial execution is expected to provide the worst case scenario, this is the chosen metric for the performance comparison in the following. Another advantage stemming from this choice is that the results might be seen as hardware independent, in contrast to the parallel execution scenario in which the results would be only relative to the particular configuration adopted, *i.e.* the number of parallel threads available.

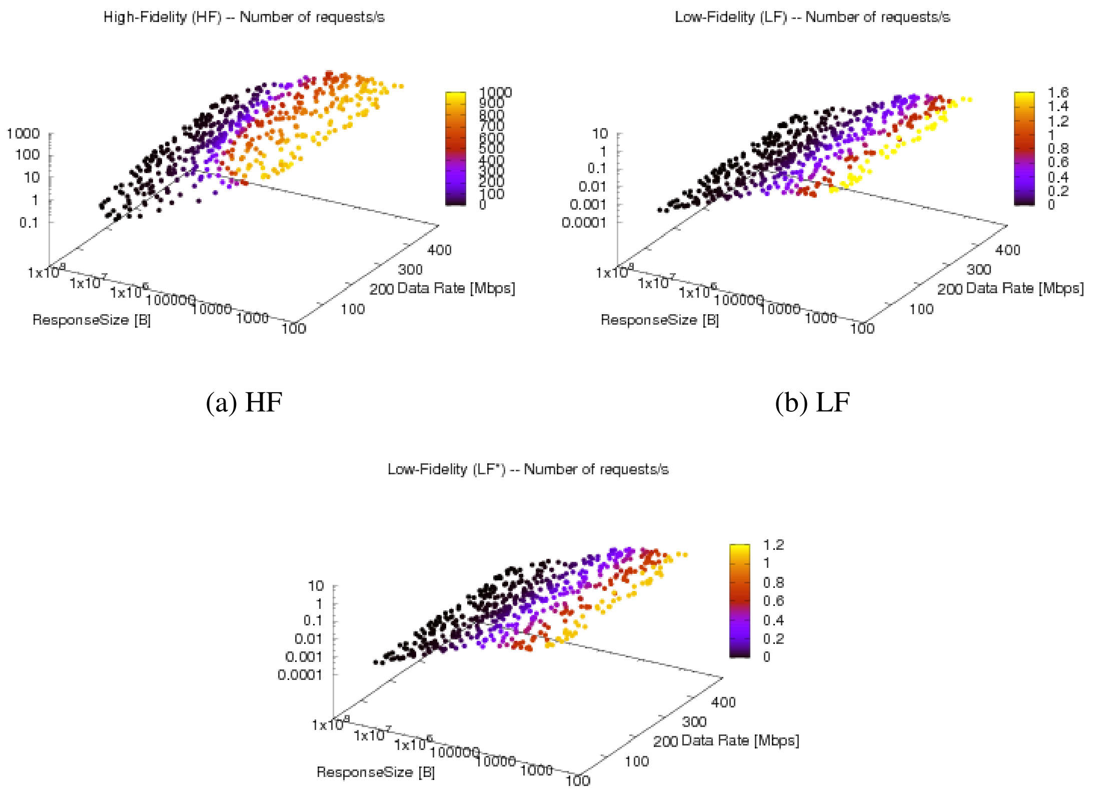
The responses of the three models for the second test case are also reported in Figure 4 for reference. A total of 500 network emulations has been obtained for minimega and several realizations of a MC estimator have been obtained for an increasing number of simulations. From the set of 500 HF runs, a sequence of subsets with increasing number of runs, has been extracted to serve as a basis for the ACV-1 estimators. These subsets are first used to compute corresponding LF simulations and their correlation with the HF. Afterward, the oversampling ratio r_1 is estimated from Eq. (9) and the corresponding set of (additional) LF runs is evaluated in ns-3. Finally, the ACV-1 estimator is evaluated by resorting to Eq. (6). In Figure 5a the performance of the different estimators are reported in term of their standard deviation. The expected rate of convergence for all the sampling estimators, $\mathcal{O}\left(N^{-1/2}\right)$ is also observed.

The 99.7% confidence interval on the expected value for the number of requests per second is also reported in Figure 5b to demonstrate the increased reliability of the MF estimators. The ACV-1 estimator based on the LF* model exhibits the highest performance. However, the ACV-1 estimators based on both LF show similar performance and they can be clearly seen as more efficient than the plain MC estimator based on high-fidelity evaluations only.

4.4 Exploring the potential of including multiple low-fidelity models

In order to explore the possibility to obtain an additional variance reduction by introducing more than one low-fidelity model, the performance of the OCV estimator (which assumes the low-fidelity statistics known) based on the simultaneous use of LF and LF* is compared to the OCV-1 estimator (where one single low-fidelity model is used). These results are meant to serve only as an indication of the potentiality of an ACV estimator (in which multiple LF models are used simultaneously but their expected values are unknown) as described in [17] because the final performance of the algorithm would need to include the cost of the low-fidelity models.

First, the correlation matrix for this test case is reported in Table 5. Both low-fidelity models are very well correlated with the high-fidelity model, which is an indication that the multifidelity

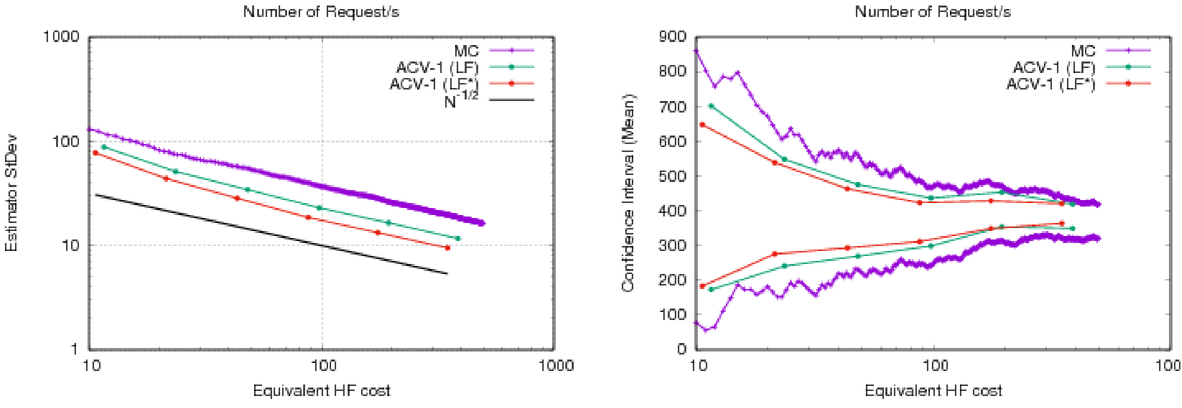


(a) HF

(b) LF

(c) LF*

Figure 4: Responses for the three models of the second demonstration case. For this case the two low-fidelity models (`ns`-3) are very similar between them, whereas the HF model `minimega` exhibits a much higher number of request/s.



(a) Estimator Standard Deviation for the first test case.

(b) 99.7% confidence interval for the estimator value.

Figure 5: Estimator standard deviation for the simple MC and two variants of the ACV-1 estimator (a). The 99.7% confidence interval for MC and ACV-1 based on both the low-fidelity estimators are reported (b). In both figures, the QoI is the number of requests/s for which the expected value is desired.

	HF	LF	LF*
HF	1	0.86	0.90
LF	0.86	1	0.99
LF*	0.90	0.99	1

Table 5: Correlation matrix for the models used in the second test cases.

Estimator (low-fidelity models)	$\mathbb{V}ar\left[\hat{Q}_N^{\mathrm{OCV}}\right]/\mathbb{V}ar\left[\hat{Q}_N^{\mathrm{MC}}\right]$	$\mathbb{V}ar\left[\hat{Q}_N^{\mathrm{ACV}}\right]/\mathbb{V}ar\left[\hat{Q}_N^{\mathrm{MC}}\right]$
Multifidelity (HF-LF)	0.26	0.39
Multifidelity (HF-LF*)	0.19	0.23
Multifidelity (HF-LF-LF*)	0.08	N/A

Table 6: Variance Reduction obtained by several estimators based on the three models HF, LF and LF* for the second test case.

estimator might be very effective. Moreover, the two low-fidelity models are almost perfectly correlated between them.

In Table 6, the three multifidelity estimators are reported in term of their normalized variance, *i.e.* the ratio between their variance and the one for a plain MC estimator with the same number of HF simulations. In the first column the normalized variance for the case of known low-fidelity statistics (OCV) is reported. The use of both LF models simultaneously achieves the greatest variance reduction exhibiting only 8% of the variance of the corresponding MC estimator. The estimation of the LF statistics, as explained in Section 3, reduces the effectiveness of the OCV estimators as can be observed in the second column of Table 6 where for the ACV-1 estimator the normalized variance is reported. In general, the ACV estimator based on multiple LF models requires the specification of the LF partitioning scheme (see [17]) and a numerical optimization to obtain the sample allocation in closed form. The accurate quantification of the performance of this estimator are left for a future study, however it is promising to observe a variance reduction gap between OCV-1 and OCV which might translate to a similar gap between ACV-1 and ACV.

5 CONCLUDING REMARKS

In this work, multifidelity uncertainty quantification has been performed for network applications. Two approaches have been considered for the network computations: a simulation approach based on the network simulator `ns`-3 and the network emulator `minimega`. The UQ tool of choice has been a multifidelity sampling approach based on a control variate which is capable of maximizing the variance reduction whenever multiple low-fidelity models are available. A simple network configuration consisting of a server and a client has been configured and two possible test cases have been addressed. The first case is a simulation only case where both the high- and low-fidelity model are evaluated in `ns`-3. The second case is more realistic and based on `minimega` as high-fidelity model and `ns`-3 as low-fidelity one. For both test cases, the multifidelity sampling approach has been demonstrated to be more efficient than a plain MC estimator. Albeit the results obtained in this work are promising they would need to be verified for more complex network configurations where the topology exhibits a higher degree of complexity and the number of uncertain parameters is much larger. Additional care would also need to be devoted to the representation of discrete variables which are very natural when dealing with networks and strategies to automatically create lower fidelity models given a particular (possibly large) network topology. Future work will also focus on understanding and mitigating the degradation of the correlation amongst network models in the presence of dissimilar input parametrizations following what has been done for computational science models in [13, 12].

6 ACKNOWLEDGEMENTS

This work has been supported by the SECURE project within the Grand Challenge LDRD Program at the Sandia National Laboratories. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energys National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

REFERENCES

[1] B. M. Adams, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, J. R. Frye, G. Geraci, R. W. Hooper, P. D. Hough, K. T. Hu, J. D. Jakeman, M. Khalil, K. A. Maupin, J. A. Monschke, E. M. Ridgway, A. Rushdi, J. A. Stephens, L. P. Swiler, J. G. Winokur, D. M. Vigil, and T. M. Wildey. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.7 theory manual. Technical Report SAND2014-4253, Sandia National Laboratories, Albuquerque, NM, Updated November 2018. Available online from http://dakota.sandia.gov/documentation.html.

[2] B. M. Adams, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Ebeida, M. S. Eldred, J. R. Frye, G. Geraci, R. W. Hooper, P. D. Hough, K. T. Hu, J. D. Jakeman, M. Khalil, K. A. Maupin, J. A. Monschke, E. M. Ridgway, A. Rushdi, J. A. Stephens, L. P. Swiler, J. G. Winokur, D. M. Vigil, and T. M. Wildey. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.7 users manual. Technical Report SAND2014-4633, Sandia National Laboratories, Albuquerque, NM, Updated November 2018. Available online from http://dakota.sandia.gov/documentation.html.

[3] F. Bellard. Qemu, a fast and portable dynamic translator. In *USENIX Annual Technical Conference, FREENIX Track*, volume 41, page 46, 2005.

[4] P. G. Constantine. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*, volume 2. SIAM, 2015.

[5] J. Crussell, T. M. Kroeger, A. Brown, and C. Phillips. Virtually the same: Comparing physical and virtual testbeds. In *2019 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2019.

[6] A. Dienstfrey and R. e. Boisvert. *Uncertainty Quantification in Scientific Computing*. 10th IFIP WG 2.5 Working Conference, WoCoUQ2011. Springer, 2012.

[7] H. Fairbanks, A. Doostan, C. Ketelsen, and G. Iaccarino. A low-rank control variate for multilevel monte carlo simulation of high-dimensional uncertain systems. *Journal of Computational Physics*, 341:121–139, 2017.

[8] J. Farooq and T. Turletti. An ieee 802.16 wimax module for the ns-3 simulator. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, page 8. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009.

[9] C. M. Fleeter, G. Geraci, D. E. Schiavazzi, A. M. Kahn, M. S. Eldred, and A. L. Marsden. Multilevel multifidelity approaches for cardiovascular flow under uncertainty. In *Sandia Center for Computing Research Summer Proceedings 2017, A.D. Baczewski and M.L. Parks, eds.*, volume Technical Report SAND2018-2780O, pages 27–50. Sandia National Laboratories, 2018.

[10] L. Foundation. Open vswitch, 2019.

[11] G. Geraci, G. Iaccarino, and M. S. Eldred. A multi fidelity control variate approach for the multilevel monte carlo technique. *CTR Annual Research Briefs 2015*, pages 169–181, 2015.

[12] G. Geraci and M. S. Eldred. Leveraging intrinsic principal directions for multifidelity uncertainty quantification. In *Technical Report SAND2018-10817*. Sandia National Laboratories, 2018.

[13] G. Geraci, M. S. Eldred, A. A. Gorodetsky, and J. D. Jakeman. Leveraging active directions for efficient multifidelity uq. In *Proceedings of the 7th European Conference on Computational Fluid Dynamics (ECFD 7)*, pages 2735–2746, 2018.

[14] G. Geraci, M. S. Eldred, and G. Iaccarino. A multifidelity multilevel monte carlo method for uncertainty propagation in aerospace applications. In *19th AIAA Non-Deterministic Approaches Conference*, page 1951, 2017.

[15] M. B. Giles. Multilevel Monte Carlo path simulation. *Operations Research*, 56(3):607–617, 2008.

[16] M. B. Giles. Multilevel Monte Carlo methods. *Acta Numerica*, 24:259–328, 2015.

[17] A. A. Gorodetsky, G. Geraci, M. Eldred, and J. D. Jakeman. A generalized framework for approximate control variates. *arXiv preprint arXiv:1811.04988v2*, 2018.

[18] D. Gourley, B. Totty, M. Sayer, A. Aggarwal, and S. Reddy. *HTTP: the definitive guide*. " O'Reilly Media, Inc.", 2002.

[19] A.-L. Haji-Ali, F. Nobile, and R. Tempone. Multi-index Monte Carlo: when sparsity meets sampling. *Numerische Mathematik*, 132(4):767–806, Apr 2016.

[20] J. Helton, C. Hansen, and P. e. Swift. Performance assessment for the proposed high-level radioactive waste repository at yucca mountain, nevada. *Reliability Engineering and System Safety, Special Issue*, pages 1–456, 2014.

[21] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, and J. Kopena. Network simulations with the ns-3 simulator. *SIGCOMM demonstration*, 14(14):527, 2008.

[22] L. Jofre, G. Geraci, H. Fairbanks, A. Doostan, and G. Iaccarino. Multi-fidelity uncertainty quantification of irradiated particle-laden turbulence. In *Center for Turbulence Research Annual Research Briefs*, pages 21–34. Center for Turbulence Research, Stanford University, 2017.

[23] A. Kivity, Y. Kamay, D. Laor, U. Lublin, and A. Liguori. kvm: the linux virtual machine monitor. In *Proceedings of the Linux symposium*, volume 1, pages 225–230. Dttawa, Dntorio, Canada, 2007.

[24] S. Lavenberg, T. Moeller, and P. Welch. *Statistical results on multiple control variables with application to variance reduction in queueing network simulation*. IBM Thomas J. Watson Research Division, 1978.

[25] S. S. Lavenberg, T. L. Moeller, and P. D. Welch. Statistical results on control variables with application to queueing network simulation. *Operations Research*, 30(1):182–202, 1982.

[26] S. S. Lavenberg and P. D. Welch. A perspective on the use of control variables to increase the efficiency of monte carlo simulations. *Management Science*, 27(3):322–335, 1981.

[27] O. Le Maitre and O. M. Knio. *Spectral Methods for Uncertainty Quantification with Applications to Computational Fluid Dynamics*. Springer Netherlands, 2010.

[28] minimega developers. minimega: a distributed vm management tool, 2019.

[29] M. G. Morgan and M. Henrion. *Uncertainty: A Guide to Dealing with Uncertainty in Quantitative Risk and Policy Analysis*. Cambridge University Press, 1990.

[30] L. Ng and K. Willcox. Multifidelity approaches for optimization under uncertainty. *Int. J. Numer. Meth. Engng.*, 100(10):746–772, 2014.

[31] B. Nguyen, A. Banerjee, V. Gopalakrishnan, S. Kasera, S. Lee, A. Shaikh, and J. Van der Merwe. Towards understanding tcp performance on lte/epc mobile networks. In *Proceedings of the 4th workshop on All things cellular: operations, applications, & challenges*, pages 41–46. ACM, 2014.

[32] W. L. Oberkampf and C. J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, 2010.

[33] R. Pasupathy, B. W. Schmeiser, M. R. Taaffe, and J. Wang. Control-variate estimation using estimated control means. *IIE Transactions*, 44(5):381–385, 2012.

[34] B. Peherstorfer, K. Willcox, and M. Gunzburger. Optimal model management for multifidelity Monte Carlo estimation. *SIAM Journal on Scientific Computing*, 38(5):A3163–A3194, 2016.

[35] K. Rabieh, M. M. Mahmoud, K. Akkaya, and S. Tonyali. Scalable certificate revocation schemes for smart grid ami networks using bloom filters. *IEEE Transactions on Dependable and Secure Computing*, 14(4):420–432, 2017.

[36] A. Saltelli, K. Chan, and E. Scott. *Sensitivity Analysis*. John Wiley & Sons, 2000.

[37] R. Smith. *Uncertainty Quantification: Theory, Implementation, and Applications*. Computational Science and Engineering. SIAM, 2013.

[38] T. Stocker, D. Qin, G.-K. Plattner, M. Tignor, S. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex, and P. M. (eds.). IPCC, 2013: Climate change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change. Technical report, 2013.

[39] F. Van den Abeele, J. Haxhibeqiri, I. Moerman, and J. Hoebeke. Scalability analysis of large-scale lorawan networks in ns-3. *IEEE Internet of Things Journal*, 4(6):2186–2198, 2017.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

OPENTURNS AND ITS GRAPHICAL INTERFACE

Michaël Baudin[1], Thibault Delage[1], Anne Dutfoy[1], Anthony Geay[1], Ovidiu Mircescu[1], Aurélie Ladier[2], Julien Schueller[2], and Thierry Yalamas[2]

[1] EDF R&D
6, quai Watier, 78401, Chatou Cedex - France,
michael.baudin@edf.fr

[2] Phimeca Engineering
18/20 boulevard de Reuilly, 75012 Paris - France,
yalamas@phimeca.com

Keywords: Uncertainty Quantification.

Abstract. *OpenTURNS is an open source library for uncertainty propagation by probabilistic methods. Developed in a partnership of five industrial companies (EDF, Airbus, Phimeca, IMACS and ONERA), it benefits from a strong practical feedback. Classical algorithms of UQ are available : central dispersion, probability of exceedance, sensitivity analysis, metamodels and stochastic processes. Developed in C++, OpenTURNS is also available as a Python module and has gained maturity thanks to more than 10 years of development.*

However, there are situations where the engineer in charge of performing an uncertainty study does not want to use a programming language such as C++ or Python. In this context, providing a graphical user interface (GUI) may allow to greatly increase the use of OpenTURNS and, more generally, of the UQ methodology.

In this paper, we present a basic tutorial of OpenTURNS in Python and will review the new features in the library, which include new incremental statistical estimators. In the second part, we review the new features in the open source GUI will be presented.

1 Introduction

OpenTURNS is a C++ library for uncertainty propagation by probabilistic methods. OpenTURNS is also available as a Python module and has gained maturity thanks to more than 10 years of development. However, there are situations where the engineer in charge of performing an uncertainty study does not want to use a programming language such as C++, Python (e.g. OpenTURNS) or Matlab. In this context, providing a graphical user interface (GUI) may allow to increase the use of OpenTURNS and, more generally, of the UQ methodology.

2 OpenTurns

OpenTURNS[2, 3, 5] is an open source software, available as a C++ library and a Python interface. It works under the Linux and Windows environments. The key features of OpenTURNS are the following:

- open source initiative to secure the transparency of the approach,
- generic to the physical or industrial domains for treating of multi-physical problems,
- high performance computing,
- includes a variety of algorithms in order to manage uncertainties in several situations,
- contains complete documentation.

OpenTURNS is available under the LGPL license.

The main features of OpenTURNS are uncertainty quantification, uncertainty propagation, sensitivity analysis and metamodeling.

Moreover generic wrappers allows to link OpenTURNS to any external code G.

OpenTURNS can be downloaded from `www.openturns.org` which offers different precompiled packages specific to several Windows and Linux environments. It is also possible to download the source files from the Github server and to compile them within another environment: the OpenTURNS Developer's Guide provides advices to help compiling the source files. Finally, most Python users use `conda` or `pip` to install OpenTURNS.

3 A tutorial example : the flooding model

3.1 Introduction

In this paper, we illustrate our discussion with a simple application model that simulates the height of a river. The figure 1 presents the dyke that protects industrial facilities. When the river height exceeds the one of the dyke, flooding occurs. This academic model is used as a pedagogical example in [8]. The model is based on a crude simplification of the 1D hydrodynamical equations of SaintVenant under the assumptions of uniform and constant flowrate and large rectangular sections. It consists of an equation that involves the characteristics of the river stretch:

$$Y = Z_v + H \quad \text{with} \quad H = \left(\frac{Q}{BK_s\sqrt{\frac{Z_m - Z_v}{L}}} \right)^{0.6}, \tag{1}$$

where Y is the maximal annual overflow, H is the maximal annual height of the river, B is the river width and L is the length of the river stretch. In this paper, we set the values of L and B

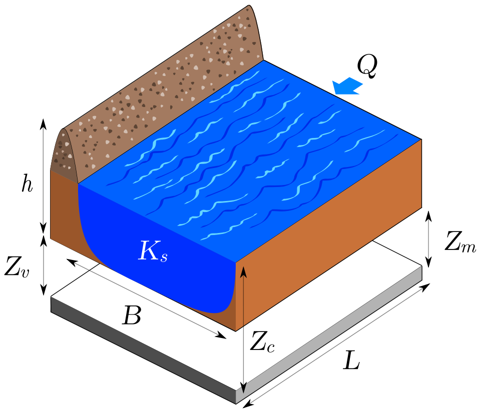


Figure 1: The flood example: simplified model of a river.

Input	Description	Unit	Probability distribution
Q	Maximal annual flowrate	m^3/s	Gumbel $\mathcal{G}(scale = 558, mode = 1013)$
K_s	Strickler coefficient	-	Normal $\mathcal{N}(30, 7.5)$
Z_v	River downstream level	m	Uniform $\mathcal{U}(49, 51)$
Z_m	River upstream level	m	Uniform $\mathcal{U}(54, 56)$

Table 1: Input variables of the flood model and their probability distributions.

parameters :

$$L = 5000, \quad B = 300.$$

The other four input variables Q, K_s, Z_v and Z_m are defined in Table 1 with their probability distribution. The randomness of these variables is due to their spatio-temporal variability, our ignorance of their true value or some inaccuracies of their estimation. We make the hypothesis that the input variables are independent.

The goal of this study is twofold:

- we want to estimate the mean river height $E(Y)$,
- we want to perform the sensitivity analysis of the model, i.e. we want to rank the inputs Q, K_s, Z_v and Z_m with respect to their contributions to the variability of the output Y.

3.2 Define the random vector

In this section, we present the Python script which allows to define the output random vector in OpenTURNS.

We begin by importing the required modules.

```
from openturns.viewer import View
import openturns as ot
from math import sqrt
import pylab as pl
```

We first define the function through which we want to propagate the uncertainties with the `def` operator.

```
def functionFlood(X) :
    Hd = 3.0
    Zb = 55.5
    L = 5.0e3
    B = 300.0
    Zd = Zb + Hd
    Q, Ks, Zv, Zm = X
    alpha = (Zm - Zv)/L
    H = (Q/(Ks*B*sqrt(alpha)))**(3.0/5.0)
    Y = H + Zv
    return [Y]
```

Then we convert this Python function into an OpenTURNS function with the `PythonFunction` class.

```
input_dimension = 4
g = ot.PythonFunction(input_dimension, 1, functionFlood)
```

Now we create the distributions for the input variables.

- There are several ways to set the parameters of the Gumbel distribution for the Q variable. Here the parameters are defined with the scale and mode parameters, which corresponds to the `GumbelAB` class.
- The Q and K_s variables must remain positive (a negative value is not compatible with the physical model). For this reason, we must truncate the distribution with `TruncatedDistribution`.

```
myParam = ot.GumbelAB(1013., 558.)
Q = ot.ParametrizedDistribution(myParam)
otLOW = ot.TruncatedDistribution.LOWER
Q = ot.TruncatedDistribution(Q, 0, otLOW)
Ks = ot.Normal(30.0, 7.5)
Ks = ot.TruncatedDistribution(Ks, 0, otLOW)
Zv = ot.Uniform(49.0, 51.0)
Zm = ot.Uniform(54.0, 56.0)
```

We set the descriptions of the random variables: they are used for the graphics.

```
Q.setDescription(["$Q␣(m^3/s)$"])
Ks.setDescription(["$Ks␣(m^{1/3})/s)$"])
Zv.setDescription(["Zv␣(m)"])
Zm.setDescription(["Zm␣(m)"])
```

The `drawPDF` method plots the the probability distribution function of the variable.

```
Q.drawPDF()
```

The previous session produces the figure 2. When we closely look at the PDF of Q, we see a small increase of the density for $Q = 0$, because of the truncation of the distribution.

Then we create the input random vector `inputvector`: by default, the copula is independent. Finally, we create the output random vector `Y`.

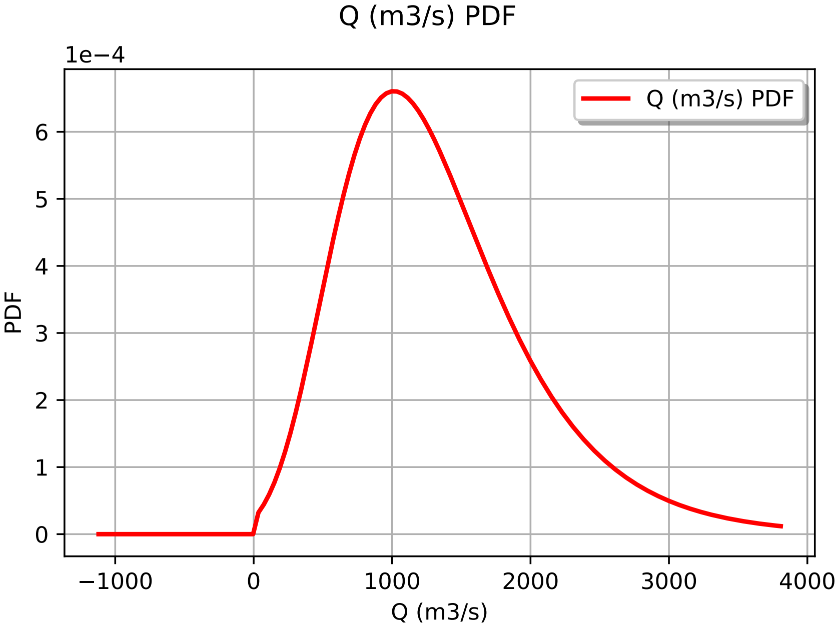


Figure 2: The probability density function of the variable Q.

```
X = ot.ComposedDistribution([Q, Ks, Zv, Zm])
inputRV = ot.RandomVector(X)
Y = ot.RandomVector(g, inputRV)
```

These steps are typical of *probabilistic programming*: we have defined the random variables involved in the problem *without* having generating a sample so far.

4 Estimating the mean with an incremental algorithm

4.1 Theory

In this section, we present the principles that are used in a new incremental algorithm in OpenTURNS 1.12; the goal of this algorithm is to estimate the mean of a random variable. Moreover, we would like to let the user as free as possible from the internal details of the algorithm and get the best possible performance on a supercomputer.

Assume that the output $Y \in \mathbb{R}^{n_Y}$ is a random vector and that we want to estimate the mean $E(Y_i)$ for $i = 1, ..., n_Y$.

The Monte Carlo method is based on the the sample mean:

$$\mu_i = \frac{1}{n} \sum_{j=1}^{n} y_i^{(j)}$$

for $i = 1, ..., n_Y$ where n is the sample size and $Y_j^{(i)}$ are i.i.d. outcomes of the random output.

The algorithm is based on the fact that the sample mean is asymptotically gaussian:

$$\mu_i \to \mathcal{N}\left(E(Y_i), \frac{V(Y_i)}{n}\right).$$

for $i = 1, ..., n_Y$ where $V(Y_i)$ is the variance of the i-th output and n is the sample size.

In general, most users set the sample size n in advance and estimate the precision afterwards. Let s_i be the (unbiased) sample standard deviation of the output Y_i:

$$s_i = \sqrt{\frac{1}{n-1}\sum_{j=1}^{N}\left(y_i^{(j)} - \mu_i\right)^2}$$

for $i = 1, ..., n_Y$. The absolute precision of the estimate μ_i can be evaluated based on the sample standard deviation of the estimator:

$$\sigma_i = \frac{s_i}{\sqrt{n}}$$

for $i = 1, ..., n_Y$. If $\mu_i \neq 0$ and $E(Y_i) \neq 0$, the relative precision can be estimated based on the coefficient of variation σ_i/μ_i for $i = 1, ..., n_Y$.

Instead, suppose that we set the absolute precision in advance and wish to determine the smallest sample size n that achieves this precision. If the variance $V(Y_i)$ is known (which rarely happens in practice), we can set the value of n so that the standard deviation $\sqrt{V(Y_i)}/\sqrt{n}$ is small enough. In the case where we want to set the relative precision, we can consider the coefficient of variation of the estimator $\frac{\sqrt{V(Y_i)}}{E(Y_i)\sqrt{n}}$ as a criterion (if $E(Y_i) \neq 0$). However, we generally do not know the values of neither $E(Y_i)$ nor $V(Y_i)$. This is why setting the sample size n in advance is not an easy task for the user in general.

The purpose of the algorithm is to increase the sample size n incrementally until a stopping criteria is met. At each iteration, we approximate the values of $E(Y_i)$ and $V(Y_i)$ by their empirical estimators, which allows to evaluate the stopping criteria.

In order to get the best possible performance on distributed supercomputers and multi-core workstations, the size of the sample increases by block. For exemple, if the block size is equal to 100, then the sample size is equal to 100, 200, etc... On each block, the evaluation of the outputs can be parallelized, which allows to improve the performance of the algorithm. More details on this topic are presented in the section 6.4.

Since there are in general several outputs, i.e. $n_Y \geq 1$, we use a stopping criteria which is based on a operator. There are three mathematical stopping criteria available:

- through an operator on the coefficient of variation $\frac{\sigma_i}{\mu_i}$ (relative criterion),
- through an operator on the standard deviation σ_i (absolute criterion),
- on the maximum standard deviation per component: $\sigma_i \leq \max_{i=1,...,n_Y} \sigma_i$ (absolute criterion).

By default, the maximum coefficient of variation is used, i.e. the operator is the *maximum* so that the algorithm stops when:

$$\max_{i=1,...,n_Y} \frac{\sigma_i}{\mu_i} \leq max_{COV}.$$

4.2 Tutorial

In this section, we present how to use the `ExpectationSimulationAlgorithm` class in the tutorial flooding example.

We set the maximum number of iterations with the `setMaximumOuterSampling` so that we use at most 1000 iterations. In order to evaluate the function with blocks of size 10, we use the `setBlockSize` method. In this simulation, we use a relative stopping criteria and configure the maximum coefficient of variation to be equal to 0.001.

```
algo = ot.ExpectationSimulationAlgorithm(Y)
algo.setMaximumOuterSampling(1000)
algo.setBlockSize(10)
algo.setMaximumCoefficientOfVariation(0.001)
```

The computationnaly intensive part of the simulation is associated with the `run` method.

```
algo.run()
```

Once the simulation is done, the `getResult` method allows to access the results.

```
result = algo.getResult()
expectation = result.getExpectationEstimate()
cv = result.getCoefficientOfVariation()[0]
print("Mean␣=␣%f␣" % expectation[0])
print("Number␣of␣calls␣to␣G␣=␣%d" % g.getCallsNumber())
print("Coef.␣of␣var.=%.6f" % (cv))
```

The previous session prints the following output.

```
Mean = 52.520729
Number of calls to G = 500
Coef. of var.=0.000994
```

The estimate of the mean has a known asymptotical gaussian distribution, which can be retrieved with the `getExpectationDistribution` method. We emphasize that the output of the `getExpectationDistribution` method is a `Distribution` in the OpenTURNS sense: the whole information is available, not just a part of it, making the output as programmatically meaningful as possible.

```
expectationDistribution = result.getExpectationDistribution()
expectationDistribution.drawPDF()
```

The previous script produces the figure 3. The figure shows that we have an accurate estimate of the mean, up to approximately 2 significant digits.

5 Estimate sensitivity indices with an incremental algorithm

5.1 Theory

In this section, we present the principles that are used in a new incremental algorithm in OpenTURNS 1.12 which computes the Sobol' sensitivity indices.

In [9] the authors derive a method to estimate the Sobol' sensitivity indices ; one of the advantages of the new estimator is that it is associated with an asymptotic distribution, which is derived thanks to the so called "delta"-method [16]. Based on a suggestion by R.Lebrun, A. Dumas [4] used the same theoretical method in order to derive the asymptotic distribution of Sobol' sensitivity indices already available in OpenTURNS.

5.1.1 Overview

Let us denote by $X \in \mathbb{R}^{n_X}$ the input random vector. Suppose that $Y = G(X) \in \mathbb{R}^{n_Y}$ is the corresponding output random vector, where G is the computer code. In this case the algorithm operates on aggregated indices. In order to simplify the discussion, let us make the hypothesis that there is only one output, i.e. $n_Y = 1$.

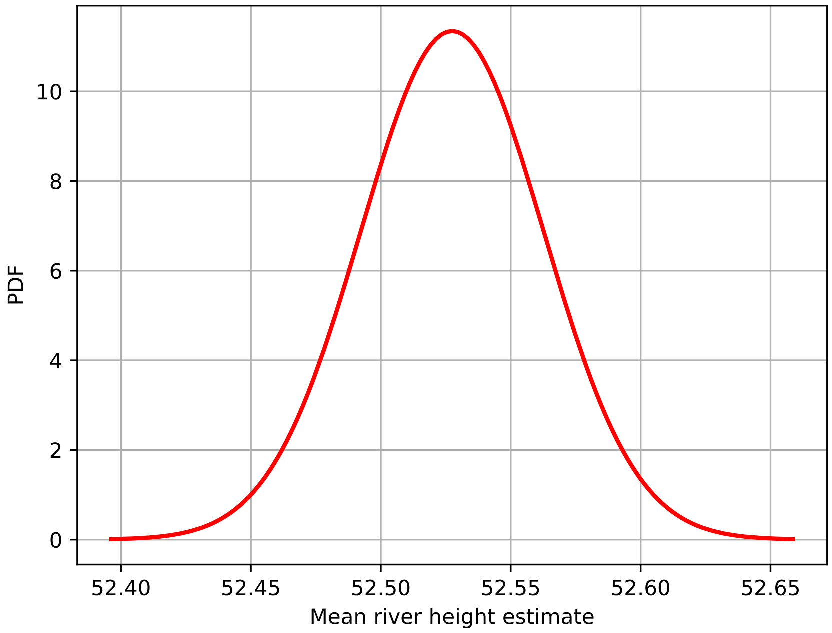


Figure 3: The probability density function of the estimate of the mean of the river height.

The Sobol' first order S_i and the total order sensitivity indices T_i are defined by

$$S_k = \frac{V(E(Y|X_i))}{V(Y)}, \qquad T_k = 1 - \frac{V\left(E\left(Y|X_{-i}\right)\right)}{V(Y)},$$

for $k = 1, ..., n_X$, where $-i$ is the set of indices which are different from i. In the remaining of this section, we focus on the first order sensitivity indice and let the reader consider [4] for the total order indices. Moreover, the derivation is the same for all input variables so that we omit the indice i in order to simplify the notations.

5.1.2 Asymptotic distribution

The algorithm is based on the fact that the estimators of the first and total order Sobol' sensitivity indices asymptotically have the gaussian distribution. This gaussian distribution can be derived from the so called "delta"-method.

Indeed, assume that the Sobol' estimator is

$$\overline{S} = \Psi\left(\overline{U}\right)$$

where Ψ is a multivariate function, U is a multivariate sample and $\overline{U}$ is its sample mean. Each Sobol' estimator can be associated with a specific choice of function Ψ and vector U. Therefore, the multivariate delta method implies:

$$\sqrt{n}\left(\overline{U} - \mu\right) \rightarrow \mathcal{N}\left(0, \nabla\psi(\mu)^T \Gamma \nabla\psi(\mu)\right)$$

where μ is the expected value of the Sobol' indice, $\nabla\psi(\mu)$ is the gradient of the function Ψ and Γ is the covariance matrix of $\overline{U}$. An implementation of the exact gradient $\nabla\psi(\mu)$ was derived for all estimators in OpenTURNS. In the algorithm, the unknown value μ is replaced by its estimator in order to compute the covariance matrix.

Each available estimator in the library provides its own distribution, namely the Saltelli, Mauntz-Kucherenko, Jansen and Martinez estimators.

5.1.3 Stopping criteria

Let us denote by Φ_k^F (resp. Φ_k^T) the cumulated distribution function of the asymptotic gaussian distribution of the first (resp. total) order sensitivity indice of the k-th input variable, for $k = 1, ..., n_X$. We set $\alpha \in [0, 1]$ the level of the confidence interval and $\epsilon \in (0, 1]$ the length of the confidence interval. The algorithms stops when, on all components, one of the two following conditions are satisfied :

- first and total order indices have been estimated with enough precision or
- the first order indices are separable from the total order indices.

The precision is said to be sufficient if the $1 - 2\alpha$ confidence interval is smaller than ϵ :

$$(\Phi_k^F)^{-1}(1-\alpha) - (\Phi_k^F)^{-1}(\alpha) \leq \epsilon$$

and

$$(\Phi_k^T)^{-1}(1-\alpha) - (\Phi_k^T)^{-1}(\alpha) \leq \epsilon$$

for $k = 1, ..., n_X$. The first order indices are *separable* from the total order indices if

$$\Phi_k^F(1-\alpha) \leq \Phi_k^T(\alpha)$$

for $k = 1, ..., n_X$. This criteria allows to stop when the algorithm has detected an interaction between input variables with sufficient precision.

5.2 Tutorial

In this section, we present how to use the `SaltelliSensitivityAlgorithm` classe in the tutorial flooding example.

We first set the parameters of the algorithms. The `alpha` variable is set so that a 90% confidence interval is used. In order to get confidence intervals which are not greater than 0.1, we set the variable `epsilon` variable accordingly. The block size corresponds to the size of the Sobol' design of experiment generated at each iteration. Finally, the `batchsize` variable contains the number of points evaluated simultaneously by the model.

```
alpha = 0.05 # 90% confidence interval
epsilon = 0.1 # Confidence interval length
blocksize = 50 # Size of Sobol experiment at each iteration
batchsize = 16 # Number of points evaluated simultaneously
```

Then we create the algorithm and configure it so that it uses the previous variables. Moreover, we use the `setMaximumOuterSampling` method so that the algorithm uses at most 100 iterations.

```
estimator = ot.SaltelliSensitivityAlgorithm()
estimator.setUseAsymptoticDistribution(True)
algo = ot.SobolSimulationAlgorithm(X, g, estimator)
algo.setMaximumOuterSampling(100) # number of iterations
algo.setBlockSize(blocksize)
algo.setBatchSize(batchsize)
algo.setIndexQuantileLevel(alpha) # alpha
algo.setIndexQuantileEpsilon(epsilon) # epsilon
algo.run()
```

Once that the algorithm has run, the results can be retrieved and estimates of first and total order indices can be printed.

```
result = algo.getResult()
fo = result.getFirstOrderIndicesEstimate()
to = result.getTotalOrderIndicesEstimate()
print("First␣order␣=␣%s" % (str(fo)))
print("Total␣order␣=␣%s" % (str(to)))
```

The previous script produces the following output.

```
First order = [0.575962,0.225763,0.357743,0.0216308]
Total order = [0.495489,0.176668,0.331708,0.00600383]
```

These estimates required 30 000 evaluations of the computer code.

We can obtain the asymptotic distribution of the first and total order indices. For example, the following script extracts the first component of the asymptotic distribution of the first order indice (which corresponds to the variable Q) and plots it.

```
dist_fo = result.getFirstOrderIndicesDistribution()
dist_fo_i = dist_fo.getMarginal(0)
graph = dist_fo_i.drawPDF()
graph.setTitle("S0")
graph.setXTitle("S0")
```

The previous script produces the figure 4.

In order to get a more compact view of the first and total order indices along with their confidence intervals, we often represent the 90% confidence intervals with a vertical bar. The figure 5 presents the Sobol' indices with asymptotic confidence intervals. We observe that the confidence intervals are relatively small, as expected.

6 New features in the graphical user interface

6.1 Introduction

There are situations where the engineer in charge of performing an uncertainty study does not want to use a programming language such as C++ or Python. In this context, providing a graphical user interface (GUI) may allow to greatly increase the use of OpenTURNS and, more generally, of the UQ methodology.

This is why we develop since 2016 a graphical user interface (GUI) of OpenTURNS, which is integrated within SALOME [6]. This GUI is developed with the OpenSource LGPL license, which is the same as OpenTURNS and SALOME. SALOME binaries for the Linux platform are provided at the following URL:

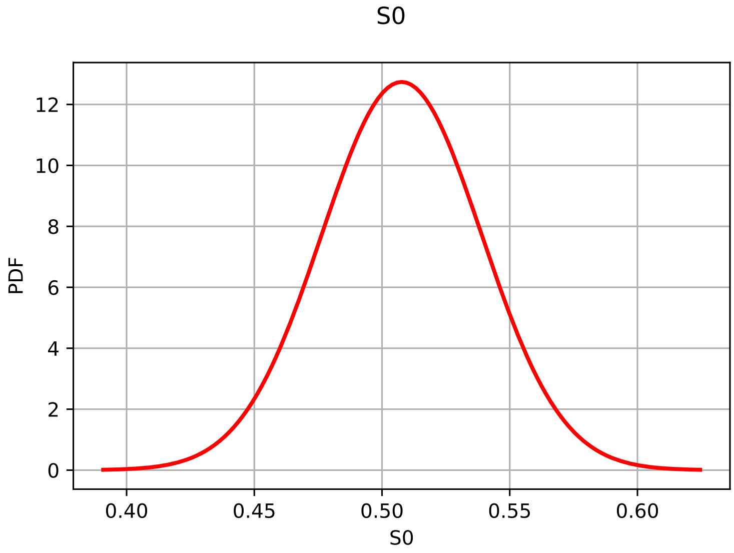


Figure 4: Asymptotic distribution of the first order Sobol' indices for the Q variable.

`https://www.salome-platform.org/contributions/edf_products`

The figure 6 presents the main window of the graphical user interface. The left pane contains the tree view which prints the opened studies and the main objects in each study. The right pane displays the main features of the interface, which makes the whole process easier for new users who might be unfamiliar with the uncertainty quantification methodology. The bottom pane is a Python console which allows to program the interface.

Each box in the right pane represents a single step in the global methodology ; the whole process is presented as a tree. Stop signs represent a method that cannot be used because a step must be fully completed before. When a new study is created, most boxes are greyed out, except the leftmost “Model definition” box, which require to define either a physical model (i.e. a computer code) or a data model (i.e. a CSV data file). Each time a step is completed, the corresponding steps which are then available are activated which allows the user to progress.

Details on the main features and the internal architecture of the GUI were already presented in [3], this is why this paper focuses on the new features.

6.2 Dependency structures

The GUI allows to define advanced dependency structures, based on copulas. The figure 7 presents the dialog box in which the copulas can be selected and configured.

The principle is to create sub-groups within the input variables. Within a given sub-group, we can select the copula and configure its parameters. Seven copulas are available: independent, Gaussian, Ali-Mikhail-Hak, Clayton, Farlie-Gumbel-Morgensten, Frank or an inference result.

For example, the figure 7 considers the situation in which the model has five inputs named X0, X1, X2, X3 and X4. In this particular model, the sub-group [X0,X1] is associated with the Gaussian copula while the sub-group [X3,X4] is associated with the Gumbel copula. The variable X2 remains independent from the others in this model.

Moreover, any multivariate sample can be used to estimate the parameters of a copula. In this case, the results of an inference can be reused in a dependency model.

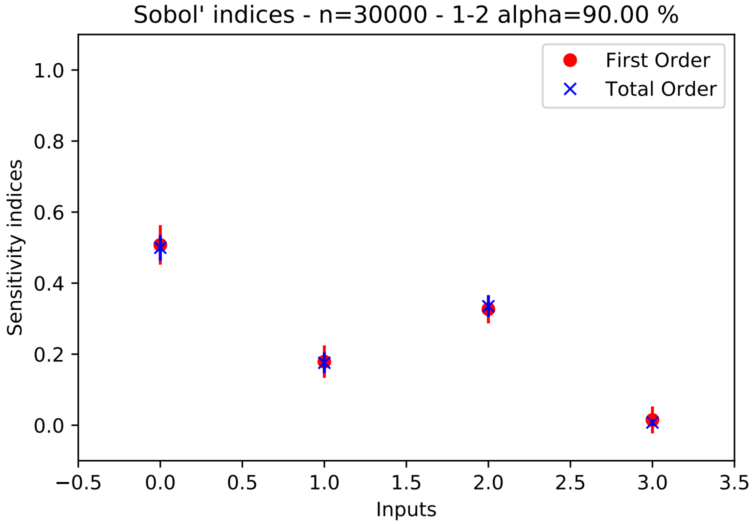


Figure 5: Sobol' indices with asymptotic confidence intervals.

6.3 Screening with the Morris method

The qualitative sensitivity analysis based on Morris's method [11] aims at selecting the significant input variables in a costly computer code which may have a large number of inputs. The GUI performs the screening analysis based on the OpenTURNS `otmorris` module [1].

The method makes use of a number of levels ℓ of levels, which is set by the user. Let $\Delta_i > 0$ be the step for the i-th input variable in the physical space, for $i = 1, ..., n_X$. This increment is computed from the number of levels L and the range of the i-th input variable.

The second parameter of the method is the number r of trajectories used in the design of experiment.

The k-th computed elementary effect associated to the i-th input marginal is the finite difference:

$$e_i^k = \frac{G\left(x_i^{(k')}\right) - G\left(x_i^{(k)}\right)}{\Delta_i}$$

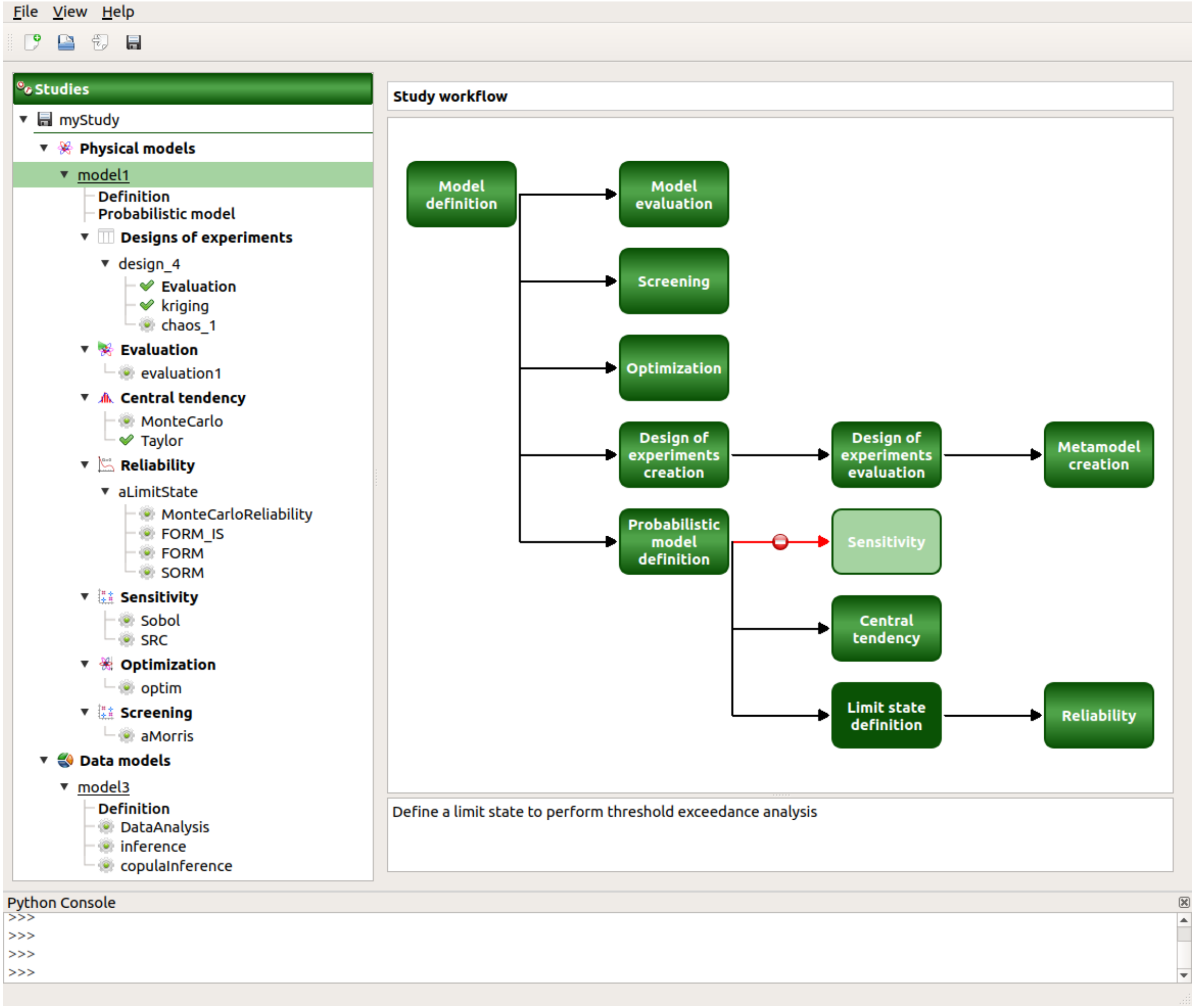
for $i = 1, ..., n_X$ and $k = 1, ..., r$. In the previous equation, the input points $x_i^{(k)}$ and $x_i^{(k')}$ are two points which differ from Δ_i in the physical input space. These points are computed based on a design of experiment which aims at grossly sampling the input space, generally with a rather large value of Δ_i.

The method computes μ_i, μ_i^* and σ_i, respectively the mean, absolute mean and the standard deviation of the elementary effects:

$$\mu_i = \frac{1}{r}\sum_{k=1}^{r} e_i^k, \qquad \mu_i^* = \frac{1}{r}\sum_{k=1}^{r} |e_i^k|, \qquad \sigma_i = \sqrt{\frac{1}{r}\sum_{k=1}^{r}(e_i^k - \mu_i)^2},$$

The goal of this method is to set the inputs variables into three classes, based on μ_i^* and the $\rho_i = \frac{\mu_i^*}{\sigma_i}$ factors:

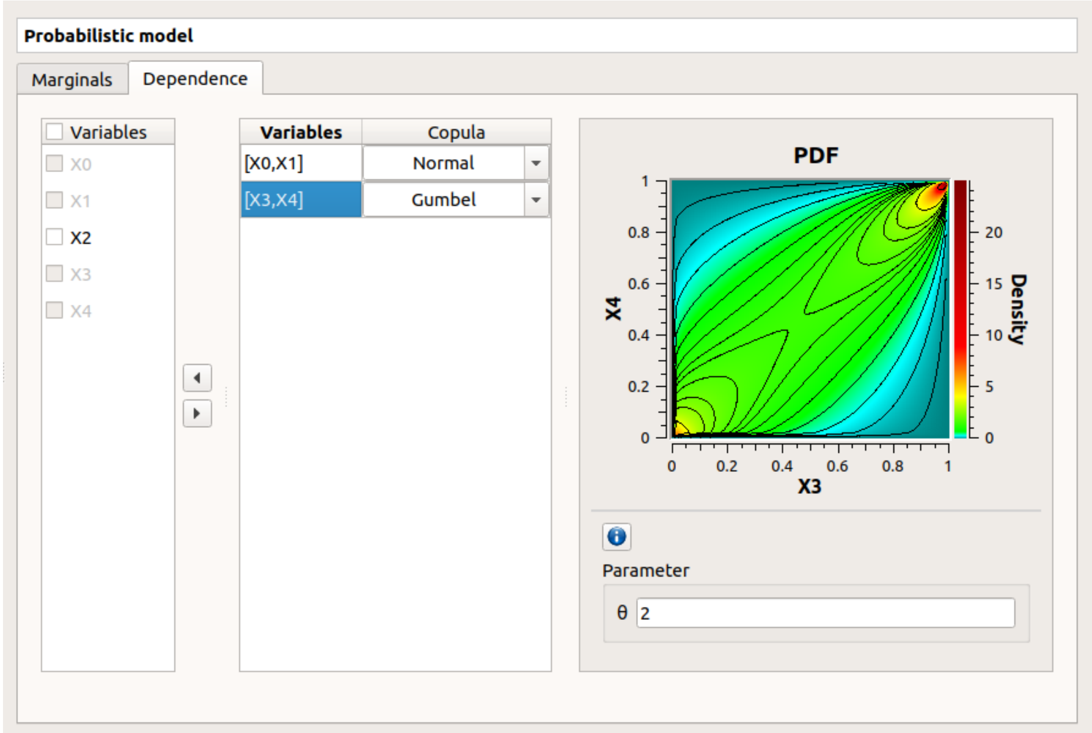
1. if μ_i^* is close to zero, the i-th variable has no effect,



Define a limit state to perform threshold exceedance analysis

```
>>>
>>>
>>>
>>>
```

Figure 6: Main window of OpenTURNS' graphical user interface.



Variables	Copula
[X0,X1]	Normal
[X3,X4]	Gumbel

θ 2

Figure 7: Managing a copula in the OpenTURNS GUI.

2. if $\rho_i \leq 0.5$ the i-th variable has almost linear effects,

3. if $\rho_i \geq 1$ the i-th variable has non-linear and non-monotonic effects

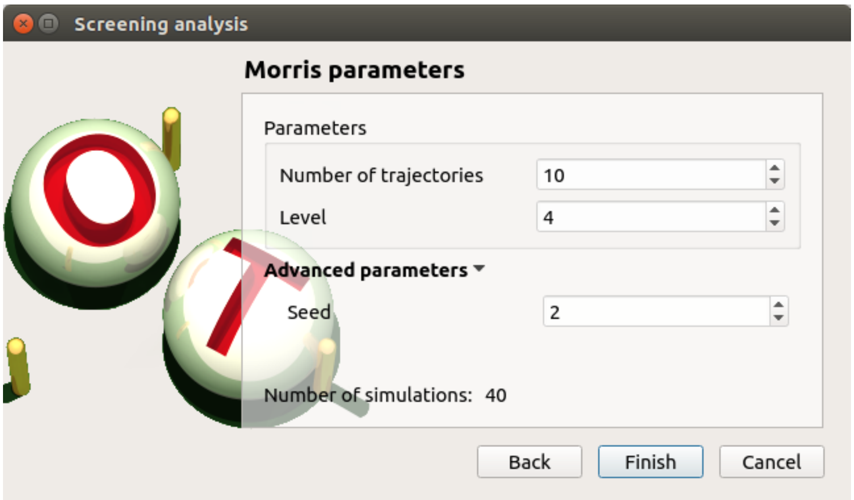
The figure 8 presents the dialog box which contains the parameters of the algorithm. The user can set the number of trajectories and the number of levels for each variable. The dialog box automatically computes the corresponding number of simulations and prints it in the bottom of the dialog box.

Once the simulations are performed, the figure 9 presents the results associated with a physical model which has 20 input variables. The main figure presents the mean and standard deviations of the elementary effects. A table (not shown in the figure) containing the list of input variables allows to see in which category fall each variable. A default classification is done by the GUI, but can be modified by the user.

6.4 Easy high performance computing

Within SALOME, users can access the remote high performance computing resources available at EDF R&D. Based on 16 100 cores, the Porthos supercomputer (2014) for example, can perform as high as 600Tflops (peak) [15]. The latest supercomputer at EDF R&D, Gaïa (2019), can perform as high as 3 052 Tflops (peak) [14] thanks to its 41 000 cores.

Within the GUI, the user can run simulations which are executed on a remote supercomputer with a minimum amount of configuration. The figure 10 presents the dialog box which is displayed in the context of a central tendency study based on a Monte-Carlo simulation. Enabling the *Parallelize status* checkbutton allows to select the computing resource available in the user's environment. The number of processes can be chosen by the user according to the hardware available and the amount of computing required by the simulation. Each job is associated with



Screening analysis

Morris parameters

Parameters

Number of trajectories: 10

Level: 4

Advanced parameters

Seed: 2

Number of simulations: 40

Figure 8: Performing screening with Morris's method in the GUI.

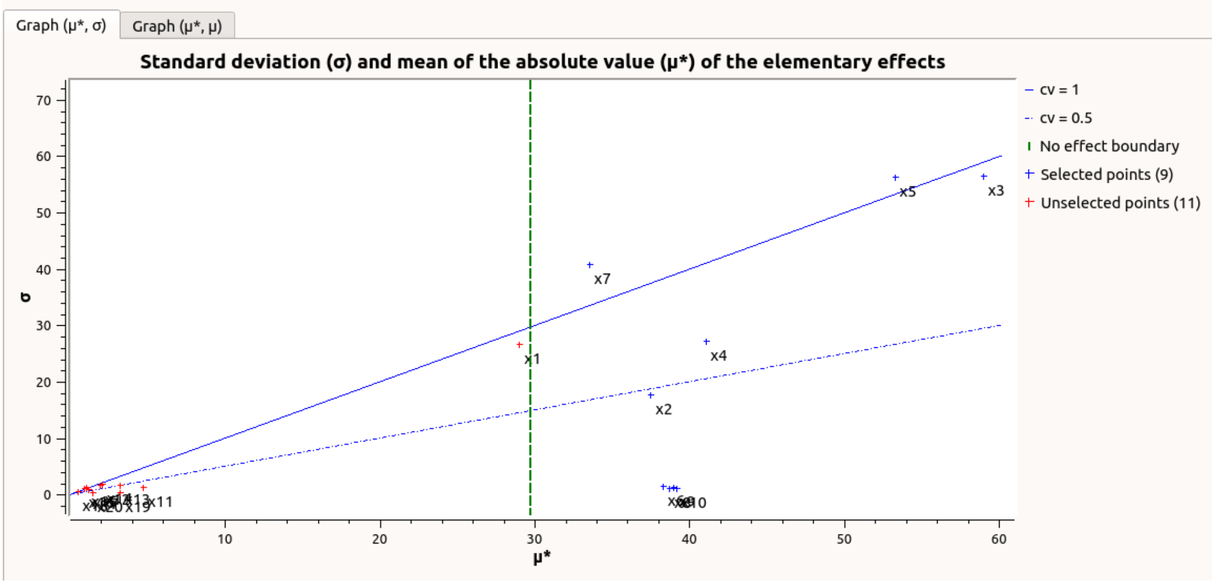


Figure 9: Results of the screening with Morris's method in the GUI.

a time limit which defines the maximum duration of one job. In most practical situations extra input files are used by the computer code (e.g. the mesh), which can be configured in the dialog box as well. The job submission is based on SLURM, but the user does not have to configure these low-level parameters which are handled automatically by the algorithms, with the principles which we now present.

The key point is to exploit the maximum possible amount of parallelism in the computations. However, between the start of the simulation and the end (which might take minutes, hours or days in the longest situations), most users want to regularily have a feedback on the execution of the simulation. For these reasons, the algorithms are performed based on blocks, which define a sub-sample on which the parallel computation can take place. At the end of each block, a progress bar is updated along with statistics which shows the number of executed simulations, the elapsed time and the value of the stopping criteria (e.g. the coefficient of variation of the mean estimator). A *Stop* button allows to interrupt the simulation.

Consider for example the situation presented in the figure 11, where a design of experiments involving 24 points must be evaluated. The parameters configured by the user in this example is the size of the block, which is set to 12, and the number of processors, which is set to 4. In this case, the simulation starts with a first job (e.g. a SLURM job) involving the points with indices from 1 to 12, and ends with a second job involving the points with indices 13 to 24. In both jobs, each processor is in charge of the evaluation of three points.

6.5 Perspectives: one-dimensional stochastic processes

In this section, we present the current developments of the GUI, which focuses on the management on stochastic fields.

Indeed, there are various situations in which the simulator through which we propagate the uncertainties produces a stochastic process. This happens for example in the case where the simulator produces a time series or a one-dimensional spatial field.

The figure 12 presents a sample of trajectories in the GUI. In general, the sample size is large and this graphics does not convey much information, because the trajectories overlap and hide each other.

Obviously, this situation is more complex than the classical output random vector that many engineers are used to and require more advanced probabilistic methods. The most common way of managing such a situation is to use a dimension reduction method such as the functional principal component analysis or the Karhunen-Loève decomposition [12].

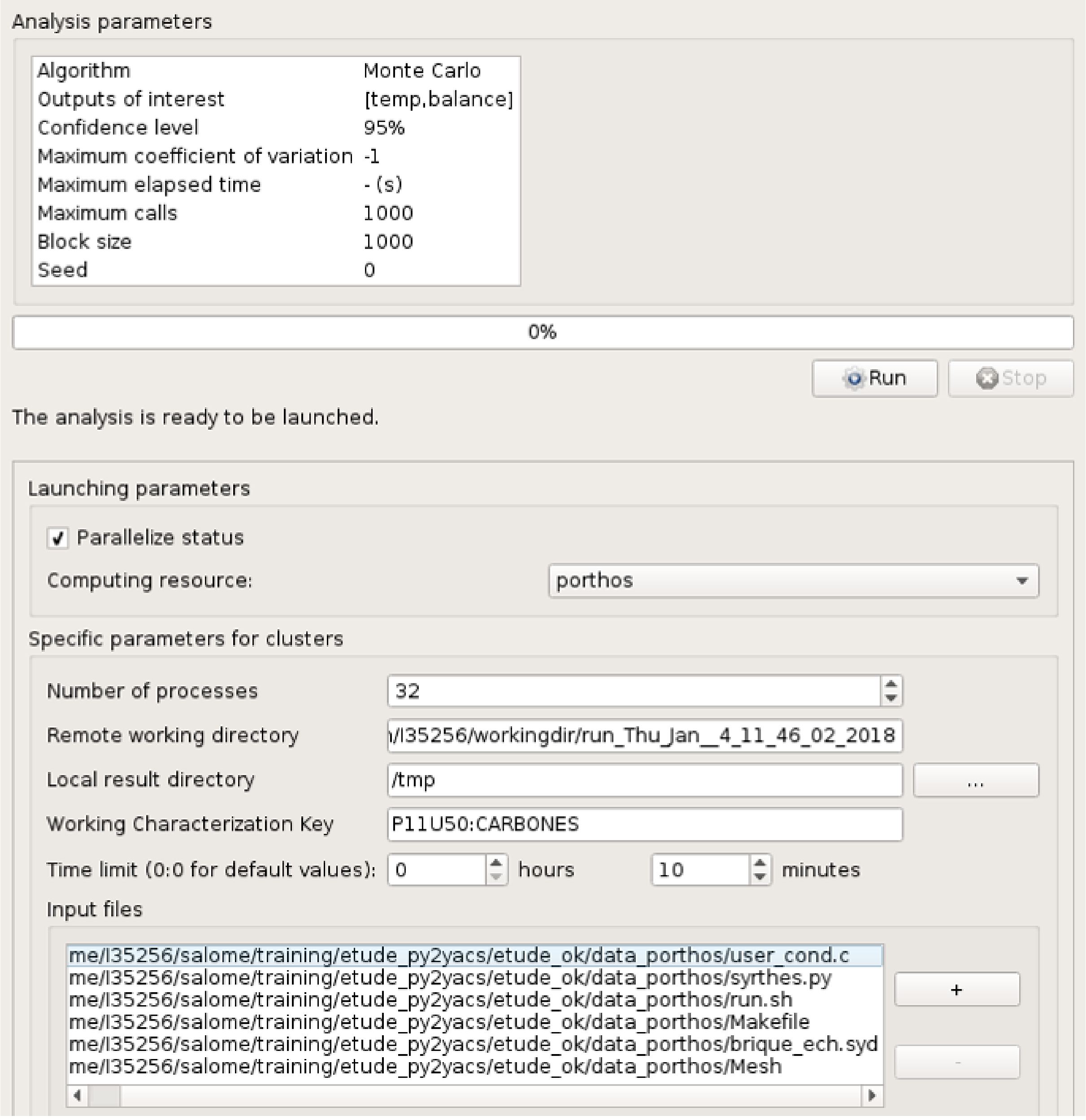
This is why Ribes et al [13] developed a new visualization tool in Paraview [10], based on a work by Kitware funded by EDF. This tool is the *functional bag chart*, also known as the highest density region plot in the bibliography [7]. The figure 13 presents the functional bag chart of a sample set of trajectories. This graphics allows to plot a functional boxplot in the sense that it plots a functional 95% confidence region. The graphics also allows to detect outlier trajectories, i.e. trajectories which achieve a low density in the reduced space.

The future version will extend these functional analyses to higher dimensions, including 2D stochastic fields.

References

[1] Airbus-EDF-IMACS-Phimeca. Otmorris module: Morris screening method module. https://github.com/openturns/otmorris.

[2] Michaël Baudin, Anne Dutfoy, Bertrand Iooss, and Anne-Laure Popelin. *OpenTURNS:*



Analysis parameters

Algorithm	Monte Carlo
Outputs of interest	[temp,balance]
Confidence level	95%
Maximum coefficient of variation	-1
Maximum elapsed time	- (s)
Maximum calls	1000
Block size	1000
Seed	0

0%

The analysis is ready to be launched.

Launching parameters

☑ Parallelize status

Computing resource: porthos

Specific parameters for clusters

Number of processes: 32

Remote working directory: \/I35256/workingdir/run_Thu_Jan__4_11_46_02_2018

Local result directory: /tmp

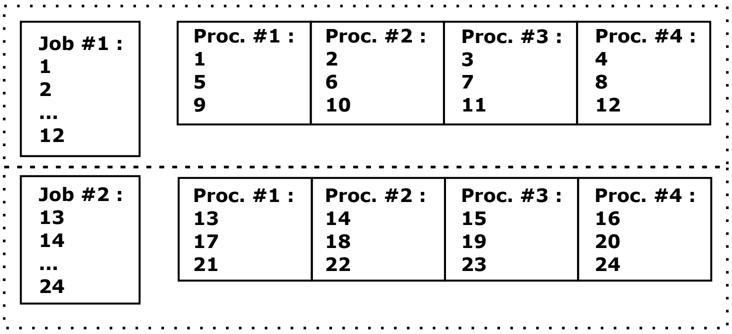
Working Characterization Key: P11U50:CARBONES

Time limit (0:0 for default values): 0 hours 10 minutes

Input files

```
me/I35256/salome/training/etude_py2yacs/etude_ok/data_porthos/user_cond.c
me/I35256/salome/training/etude_py2yacs/etude_ok/data_porthos/syrthes.py
me/I35256/salome/training/etude_py2yacs/etude_ok/data_porthos/run.sh
me/I35256/salome/training/etude_py2yacs/etude_ok/data_porthos/Makefile
me/I35256/salome/training/etude_py2yacs/etude_ok/data_porthos/brique_ech.syd
me/I35256/salome/training/etude_py2yacs/etude_ok/data_porthos/Mesh
```

Figure 10: Launching a parallel computation within the GUI.



	Proc. #1 :	Proc. #2 :	Proc. #3 :	Proc. #4 :
Job #1 : 1 2 ... 12	1 5 9	2 6 10	3 7 11	4 8 12
Job #2 : 13 14 ... 24	13 17 21	14 18 22	15 19 23	16 20 24

Figure 11: A block-based simulation performed on a supercomputer. We assume that we want to evaluate a design of experiments made of 24 points ; these points are numbered 1, 2, ..., 24. We consider a block size of 12 and a number of processors equal to 4. In this case, the block-based simulation uses two jobs.

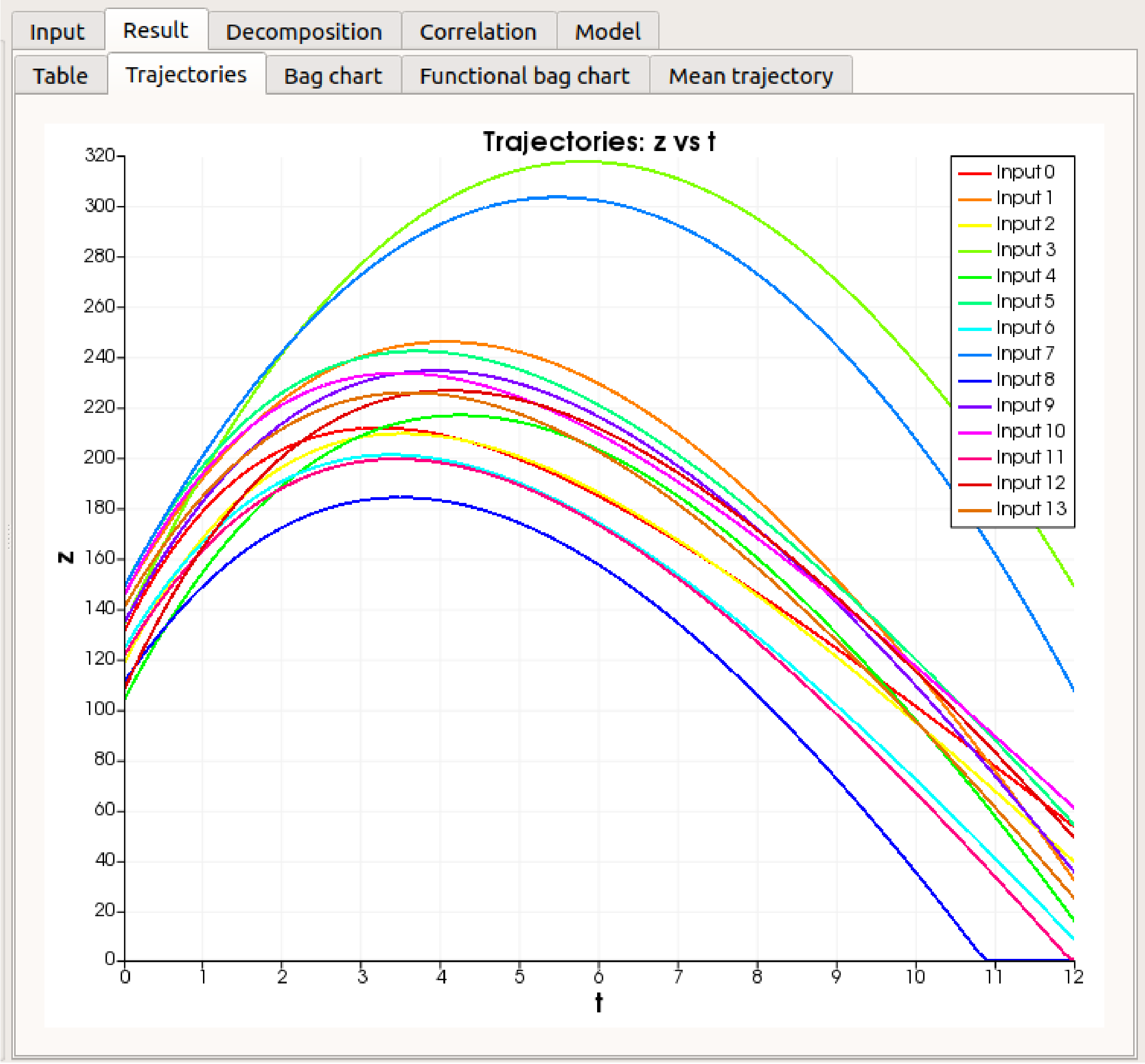


Figure 12: A sample of trajectories in the GUI.

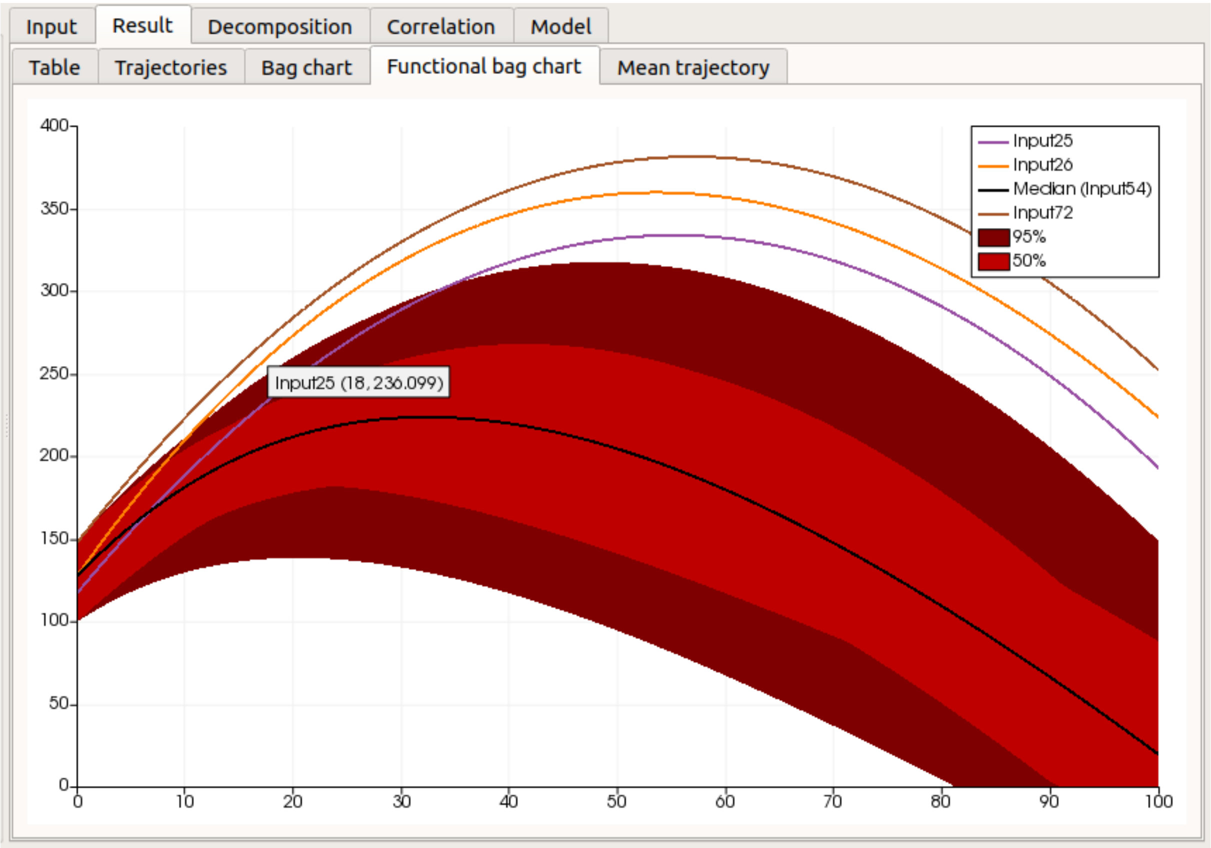


Figure 13: The functional bag chart of Paraview to plot a functional boxplot and detect outlier trajectories.

An Industrial Software for Uncertainty Quantification in Simulation, pages 1–38. Springer International Publishing, Cham, 2016.

[3] Michaël Baudin, Anne Dutfoy, Anthony Geay, Anne-Laure Popelin, Aurélie Ladier, Julien Schueller, and Thierry Yalamas. The graphical user interface of openturns, a uq software in simulation. UNCECOMP 2017, 15 - 17 June 2017 Rhodes Island, Greece. Eccomas Proceedia UNCECOMP (2017) 238-257 `https://www.eccomasproceedia.org/conferences/thematic-conferences/uncecomp-2017/5366`.

[4] Antoine Dumas. Lois asymptotiques des estimateurs des indices de Sobol'. Application de la méthode delta. EDF, by Phiméca Engineering.

[5] EADS, EDF, Phimeca Engineering, and IMACS. Openturns, an open source initiative for the treatment of uncertainties, risks'n statistics. `www.openturns.org`.

[6] EDF, CEA, and Open Cascade. Salome, the open source integration platform for numerical simulation. `www.salome-platform.org`.

[7] R.J. Hyndman. Computing and graphing highest density regions. *American Statistician*, 50:120–126, 1996.

[8] B. Iooss and P. Lemaître. A review on global sensitivity analysis methods. In C. Meloni and G. Dellino, editors, *Uncertainty management in Simulation-Optimization of Complex Systems: Algorithms and Applications*. Springer, 2015.

[9] Alexandre Janon, Thierry Klein, Agnès Lagnoux, Maëlle Nodet, and Clémentine Prieur. Asymptotic normality and efficiency of two sobol index estimators. *ESAIM: Probability and Statistics*, 18:342–364, 2014.

[10] Kitware. Paraview. `www.paraview.org`.

[11] M.D. Morris. Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33:161–174, 1991.

[12] J.O. Ramsay and B.W. Silverman. *Functional data analysis*. Springer, 2005.

[13] Alejandro Ribes, Joachim Pouderoux, Anne-Laure Popelin, and Bertrand Iooss. Visualizing statistical analysis of curve datasets in Paraview. 11 2014.

[14] TOP500.org. Top 500, gaïa - bull intel cluster. `https://www.top500.org/system/179569`.

[15] TOP500.org. Top 500, porthos - ibm nextscale nx360m5. `https://top500.org/system/178462`.

[16] A. W. van der Vaart. *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics, 2000.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

COMPUTING WITH UNCERTAINTY: INTRODUCING PUFFIN THE AUTOMATIC UNCERTAINTY COMPILER

Nick Gray, Marco De Angelis and Scott Ferson

Institute for Risk and Uncertainty, University of Liverpool, United Kingdom
e-mail: nickgray@liverpool.ac.uk

Keywords: Uncertainty Quantification, Uncertainty Compiler,

Abstract. *Although engineers often recognise the advantages of applying uncertainty analysis to their complex simulations, they often lack the time, patience or expertise to undertake that analysis. We describe a software tool, named puffin, that takes existing code and converts in to uncertainty aware code in the same language making use of intrusive uncertainty propagation techniques. It can work either automatically or with user specification of the uncertainties involved in the system.*

1 INTRODUCTION

Modern engineering is all about numerical calculation, with the inexorable growth of computer power more of these calculations are being undertaken with ever more complex computer simulations. These developments means that new technologies, as digital twins [1], have begun to be explored. Engineers need to make calculations even when there is uncertainty about the quantities involved.

There are two types of uncertainty *aleatory* and *epistemic* with in the numerical calculations essential to engineering. Aleatory uncertainty arises from the natural variability in dynamical environments and material properties, errors in manufacturing processes or inconsistencies in the realisation of systems. Aleatory uncertainty cannot be reduced by empirical effort. Epistemic uncertainty is caused by measurement imperfections or lack of perfect knowledge of a system. This could be due to not knowing the full specification of a system in the early phases of engineering design.

Imperfect scientific understanding of the underlying physics or biology involved, would cause uncertainty in the future performance of a system even after the design specifications have been decided. If uncertainties are small they can often be neglected or swept away by looking at the worst-case scenarios. However, in situations where the uncertainty is large, or would affect an engineering decision, this approach is suboptimal or impossible. Instead, a comprehensive strategy of accounting for the two kinds of uncertainty is needed that can propagate imprecise and variable numerical information through calculations.

Many engineers work with legacy computer codes that do not take full account of uncertainties. Because analysts are typically unwilling to rewrite their codes, various simple strategies have been used to remedy the problem, such as elaborate sensitivity studies or wrapping the program in a Monte Carlo loop. These approaches treat the program like a black box because users consider it uneditable. However, whenever it is possible to look inside the source code, it is better characterised as a *crystal box* because the operations involved are clear but fixed and unchangeable in the mind of the current user.

Strategies are needed that automatically translate original source into code with appropriate uncertainty representations and propagation algorithms. We have developed an uncertainty compiler for this purpose, named Puffin[1], along with an associated language. It handles the specifications of input uncertainties and inserts calls to an object-oriented library of intrusive uncertainty quantification (UQ) algorithms. We use ANTLR [2], a parser/lexer generator, and Python to translate *uncertainty näive* code into code with a full account of uncertainty in the same language. In theory, the approach could work with any computer language. We currently support Python and later versions will handle FORTRAN, C, R and MATLAB languages.

2 PUFFIN LANGUAGE

In order to develop Puffin it was first essential to build an uncertainty language. Puffin language enables users to specify the uncertainties involved in their code before compiling it into pre-existing scripts. Currently enables uncertainty analysis with five types:

- Interval (unknown value or values for which sure bounds are known), [3]
- Probability distribution (random values varying according to specified law such as normal, lognormal, Weibull, etc., with known parameters),

[1] In ornithology puffins belong to the family auks, as we are making an automatic uncertainty compiler (auc) puffin seemed like a fitting name

- P-box (random values for which the probability distribution cannot be specified exactly but can be bounded),[4]
- Confidence box (confidence structure that is a representation of inferential uncertainty about a parameter compatible with both Bayesian and frequentist paradigms), [5]
- Natural language expressions (such as *about 7.2* or *at most 12*)

We are also planning other . For each language that the compiler is to support a library of intrusive UQ code is required that will allow these types of numbers to be freely mixed together in mathematical expressions to reflect what is known about each quantity. Such libraries already exist for MATLAB and R and a python equivalent is currently in active development.

In Puffin language, if compiling into languages with immutable values and editable variables, -> will be used for immutable values and = for editable variables, in languages where this isn't the case they can be used interchangeably. # are used for comments. Guillemets surround code snippits from the target language. Both single and double quotation marks can be used in Puffin, although if the target language is pernickety about which one is used then the user will have to be aware of this themselves.

2.1 Intervals

An interval is an uncertain number representing values obeying an unknown distribution over a specified range, or perhaps a single value that is imprecisely known even though it may in fact be fixed and unchanging. Intervals thus embody epistemic uncertainty. Intervals can be specified by a pair of scalars corresponding to the lower and upper bounds of the interval. Interval arithmetic computes with ranges of possible values, as if many separate calculations were made under different scenarios. However, the actual computations the software does are made all at once, so they are very efficient. As shown in Figure 1, there are several different formats for specifying intervals. All types of intervals are defined using square brackets, this simplest definition is for the lower bound and upper bound to be comma separated within the square brackets. Plus minus intervals can be defined with either a positive number or a percentage. They can also be defined by a single number within the brackets in which case the significant digits are used for the bounds of the interval. There may sometimes be uncertainty about the endpoints, this can be specified using nested intervals such as shown in line 5

```
[1] a -> [1,2]
[2] b -> [1±2]          #[-1,3]
[3] c -> [1±2%]         #[0.98,1.02]
[4] d -> [1.0]          #[0.95,1.05]
[5] e -> [[0,1],[2,3]]  #[0,3]
```

Figure 1: Syntax for defining intervals in Puffin language, ± can be substituted with +- or -+. The comments show what the interval is taken to be when compiling.

2.2 Distributions and P-Boxes

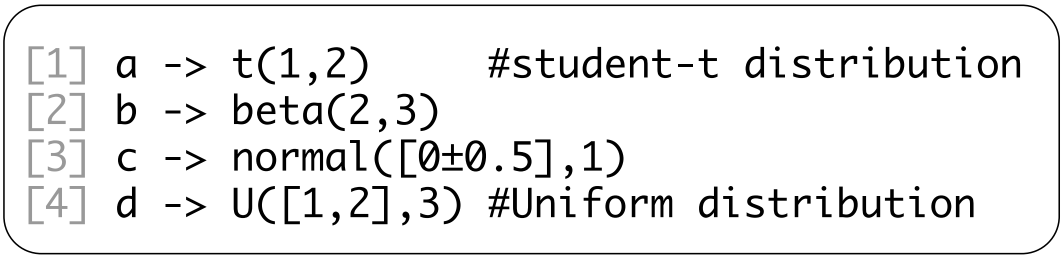
Probability distributions are specified by their shape and parameters, such as gaussian(5,1), uniform(0,9), or weibull(3,6). A non-exhaustive list of distributions available in the langauge is shown in table 1, however we are planning to add more. As with all keywords in Puffin they can be defined in either all caps, all lower or sentence case. For distributions with common short names then these will also be accessible, for example *N* for the normal distribution. P-boxes can be specified as probability distributions with intervals for one or more of their parameters. If the shape of the underlying distribution is not known, but some parameters such as the mean, mode, variance, etc. can be specified (or given as intervals), the software will construct distribution-free p-boxes whose bounds are guaranteed to enclose the unknown distribution subject to constraints specified.

Bernoulli	beta	binomial	Cauchy
chi-squared	delta	empirical distributions	exponential
F distribution	Frechet	gamma	geometric
Gaussian	Gumbel	Laplace	logistic
lognormal	logtriangular	normal	Pareto
Pascal	Poisson	power function	rayleigh
reciprocal	Simpson	Student-t	trapzoidal
triangular	uniform	Wakeby	Weibul

Table 1: Some of the distributions available in the uncertainty language

Probability bounds analysis integrates interval analysis and probabilistic convolutions which are often implemented with Monte Carlo simulations. It uses p-boxes, which are bounds around probability distributions, to simultaneously represent the aleatory uncertainty about a quantity and the epistemic uncertainty about the nature of that variability. Probability distributions are special cases of p-boxes, so one can do a traditional probabilistic analysis with the add-in as well. The calculations the software does are very efficient and do not require Monte Carlo replications.

Figure 2 shows several difference distribution and p-box assignments.



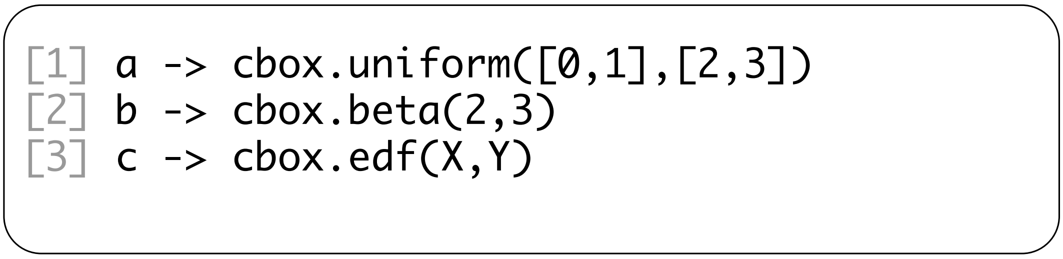
```
a -> t(1,2)     #student-t distribution
b -> beta(2,3)
c -> normal([0±0.5],1)
d -> U([1,2],3) #Uniform distribution
```

Figure 2: Syntax for defining p-boxes in Puffin language

2.3 C-Boxes

Confidence boxes (c-boxes) are imprecise generalisations of traditional confidence distributions, which, like Student's t–distribution, encode frequentist confidence intervals for parameters of interest at every confidence level. They are analogous to Bayesian posterior distributions in that they characterise the inferential uncertainty about distribution parameters estimated from sparse or imprecise sample data, but they have a purely frequentist interpretation that makes them useful in engineering because they offer a guarantee of statistical performance through repeated use. Unlike confidence intervals which cannot usually be used in mathematical calculations, c-boxes can be propagated through mathematical expressions using the ordinary machinery of probability bounds analysis, and this allows analysts to compute with confidence, both figuratively and literally, because the results also have the same confidence interpretation. For instance, they can be used to compute probability boxes for both prediction and tolerance distributions. C–boxes can be computed in a variety of ways directly from random sample data. There are c-boxes both for parametric problems (where the family of the underlying distribution from which the data were randomly generated is known to be normal, lognormal, exponential, binomial, Poisson, etc.), and for nonparametric problems in which the shape of the underlying distribution is unknown. Confidence boxes account for the uncertainty about a parameter that comes from the inference from observations, including the effect of small sample size, but also the effects of imprecision in the data and demographic uncertainty which arises from trying to characterise a continuous parameter from discrete data observations.

In Puffin language, c-boxes can be defined using dot notation shown in Figure 3. All distributions that work with p-boxes are also available in c-box form.



```
[1] a -> cbox.uniform([0,1],[2,3])
[2] b -> cbox.beta(2,3)
[3] c -> cbox.edf(X,Y)
```

Figure 3: Syntax for defining c-boxes in Puffin language. Line 3 shows the definition from an empirical distribution function where X and Y would represent arrays of data

2.4 Hedge Words

In order to make uncertainty analysis as simple as possible for the end user Puffin language allows for users to be able to input their uncertainties using natural language expressions such as *about* or *almost*. Table 2, lists some the allowed hedge words and their possible interpretations.

Hedge words can be interpreted as intervals [6] or or c-boxes. [7]

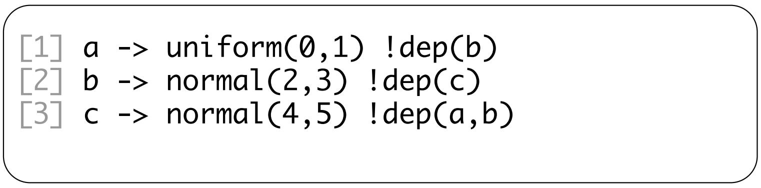
Hedged Numerical Expression	Possible Interpretation
about x	$[x \pm 2 \times 10^{-d}]$
around x	$[x \pm 10 \times 10^{-d}]$
count x	$[x \pm \sqrt{x}]$
almost x	$[x - 0.5 \times 10^{-d}, x]$
over x	$[x, x + 0.5 \times 10^{-d}]$
above x	$[x, x + 2 \times 10^{-d}]$
below x	$[x - 2 \times 10^{-d}, x]$
at most x	$[0, x]$
at least x	$[x, \infty]$
order x	$[x/2, 5x]$
between x and y *or* from x to y	$[x, y]$
x out of y	cbox.beta(a,b)

Table 2: Hedge expressions and their mathematical equivalent. Note: d is the number of significant figures of x

2.5 Dependence assumptions

By default, the language assumes that each newly specified probability distribution or p-box is stochastically independent of every other. Users can change this assumption by specifying nature of the dependence using the syntax shown in Figure 4.

In addition, Puffin language automatically tracks calculations that were used to compute uncertain numbers and will modify the default assumption of independence if appropriate. For instance, an increasing monotone function (such as log, exp, and sqrt) of a distribution creates an uncertain number that is perfectly dependent on the original distribution. Reciprocation creates an uncertain number that is oppositely dependent on the original distribution. When the function that transforms an uncertain number is complex and the relationship between the original distribution and the result cannot be educed, the two are assigned the unknown dependence. If the two later are used in a calculation, Fréchet convolution, which makes no assumption about the dependence between the arguments, is used to combine them. Fréchet convolution must be used because an assumption of independence would be untenable, because one argument is a direct function of the other. Generally, Fréchet convolution creates p-boxes from precise probability distributions, or widens the results from p-boxes relative to convolutions that assume independence or some other precise dependence function. The extra width represents the additional uncertainty arising from not knowing the dependence function. Users can countermand the languages automatic tracking of dependence and specify the assumption to be used in any particular convolution.Âż



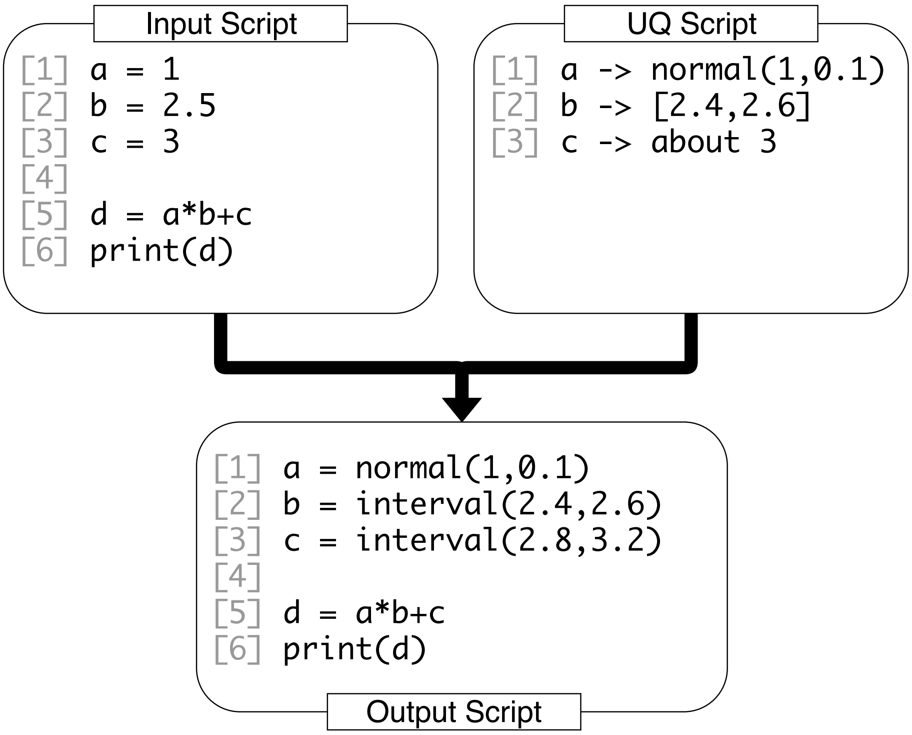
```
[1] a -> uniform(0,1) !dep(b)
[2] b -> normal(2,3) !dep(c)
[3] c -> normal(4,5) !dep(a,b)
```

Figure 4: Syntax for adding dependance between variables in Puffin Language

3 PUFFIN COMPILER

The process for taking the uncertainty naive code and adding in appropriate uncertainty analysis can be done in two different ways: the automatic approach and the second is to specify the uncertainty using the language described above. Currently the compiler can only be used from the terminal or command line. We are planning on developing a user interface for the compiler that is based on the open source *winmerge*[2] software. It will allow the user to be able to see the differences between the original and uncertainty code.

To be perhaps more useful the second method allows the end user to specify the uncertainty manually using the uncertainty language described above. It is possible to generate the Puffin langauge file by running the *–getpuffin* command when using the compiler, this will parse over the file and get all the variable declarations within the script. Figure 5 shows an example of using the compiler.



```
a = 1
b = 2.5
c = 3

d = a*b+c
print(d)
```

```
a -> normal(1,0.1)
b -> [2.4,2.6]
c -> about 3
```

```
a = normal(1,0.1)
b = interval(2.4,2.6)
c = interval(2.8,3.2)

d = a*b+c
print(d)
```

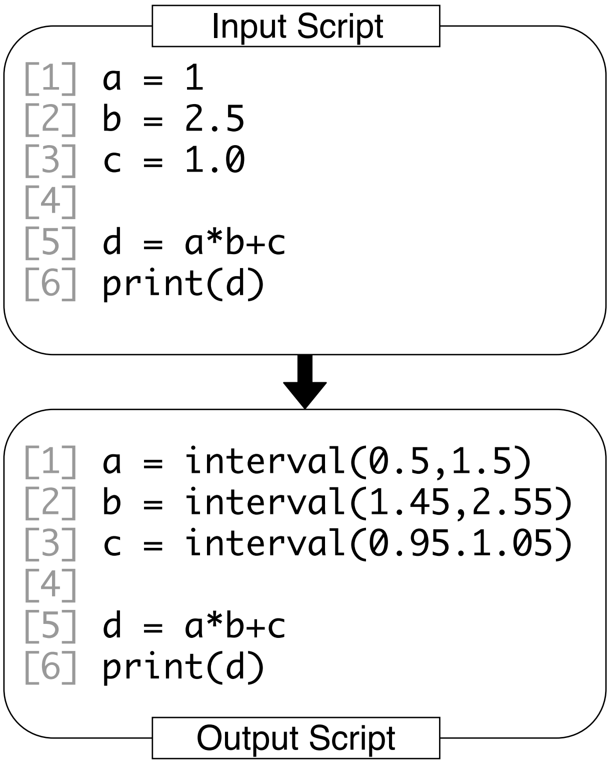
Figure 5: The result of using the compiler whilst defining the uncertainty in Puffin langauge on a simple pseudocode script.

3.1 Automatic Uncertainty Analysis

The automatic approach takes the significant figures of the assignments and uses that information as a proxy for the uncertainty for an example see Figure 3.1. When using this mode the compiler will need to *tread carefully* around mathematical constants such as π or e for which there is no uncertainty. For example if the variable had value equal to 3.14159 then it would be pretty clear that it is referring to the mathematical constant however if the value was 3.1 then it could be ambiguous. 3.141 could also cause problems, the correct rounding of π to 3 decimal places is 3.142 however 3.141 is so ubiquitous as the start of π that it would be a simple error

[2] winmerge.org/

for an analysis to make when creating code.



```
a = 1
b = 2.5
c = 1.0

d = a*b+c
print(d)
```

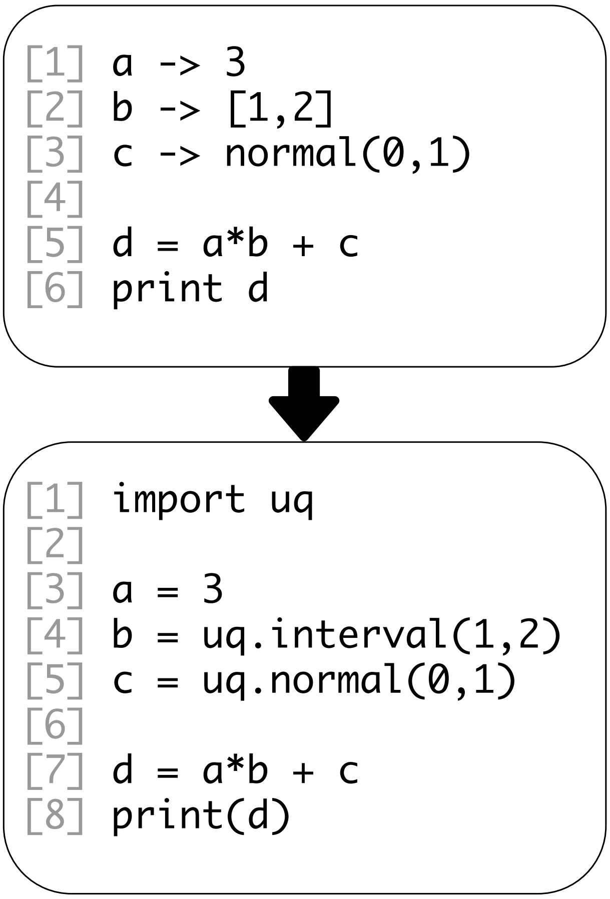
```
a = interval(0.5,1.5)
b = interval(1.45,2.55)
c = interval(0.95.1.05)

d = a*b+c
print(d)
```

Figure 6: The result of using the compiler in automatic mode on a simple pseudocode script.

3.2 Direct Compiler

Once Puffin language has been fully developed we are intending to create a direct compiler that allows creation of scripts in the code. Initially the compiler will turn the Puffin code directly into Python 3 code. Figure 7 shows direct translation from Puffin language to Python.



```
a -> 3
b -> [1,2]
c -> normal(0,1)

d = a*b + c
print d
```

```
import uq

a = 3
b = uq.interval(1,2)
c = uq.normal(0,1)

d = a*b + c
print(d)
```

Figure 7: Direct translation from Puffin language to Python 3

4 REPEATED VARIABLE PROBLEM

A limitation of using the Puffin compiler to incorporate uncertainty analysis into numerical calculations arise from multiple occurrences of an uncertain variable in a mathematical expression. Let $a = [1, 2]$, $b = [-1, 1]$ and $c = [3, 4]$. Applying interval arithmetic naively gives

$$ab + ac = [1, 10] \tag{1}$$

but also

$$a(b + c) = [2, 10] \tag{2}$$

One would expect that the results of equation 1 and equation 2 would be the same as, algebraically, $ab + ac \equiv a(b + c)$ however the distributive law of real numbers does not generally hold for uncertain numbers. In the case of intervals, the expression with repeated uncertain quantities may be wider than the one with no such repetitions, even when they would be equivalent for real values, the uncertain number appearing twice in the first formulation means, in effect, the uncertainty it represents is entered twice into the resulting calculation.

This problem besets most uncertainty quantification methods, although an advantage of Monte Carlo methods is that they can escape this problem. This uncertainty inflation would also occur if a calculation is conducted in multiple steps. For instance, if the first term ab in the example above is calculated on one line and on a new line ac is calculated before the final sum is calculated in a third line, the uncertainty of a will have been introduced into the final result twice leading to the inflated uncertainty shown in equation 1.

If possible, the number of repetitions of uncertain variables should be reduced by algebraic manipulation to avoid possible inflation of the uncertainty. This would apply whether the uncertain parameter is an interval, distribution, p-box or c-box. It should be noted that only the repeated variable matters when reducing the expression because other variables can be as arbitrarily complex, with as many repeats required. For instance if x is the only uncertain in equation 3 then the fact that b is repeated five times is irrelevant.

$$(a+bx)^n(c+dx)^n = \left(\frac{(2bdx+ad+bc)^2-(ad+bc)^2+4abcd}{4bd}\right)^n \tag{3}$$

Unfortunately, it is not always possible to reduce all multiple occurrences. For example, equation 4 cannot be reduced to a single instance of x. In such cases, special or ad hoc strategies must be devised, even partial solutions improve calculations.

$$x^3+x^2+x+1 \tag{4}$$

When computing with intervals, we are guaranteed that the result of a computation with repeated variables will have width no smaller than the correct answer. Therefore, even if multiple occurrences of the variable cannot be reduced, a conservative estimate of the final value can still be calculated. In risk assessment such an estimate may meet the practical needs of an analysis. For probability distributions and p-boxes, this guarantee holds when using FrÃl'chet convolutions, however it does not extend to cases where independence has been assumed or precise dependancies have been specified between the variables. The size of any error cause by repeated variables is dependant on the particulars of the mathematical expression as well as the quantities involved

A useful extension to the compiler would be to automatically detect and simplify mathematical expressions with repeated uncertain variables, even over multiple lines, in a way that reduces the repetitions of parameters containing uncertainty or in situations where they cannot be simplified then display a warning to the user. An example of how this might look when generating the Puffin langauge file form a script is show in Figure 8. Although this problem is known to be NP-hard in general, software strategies can be designed to find expressions with fewer repetitions of the same variable. In instances where no solution could be found then the compiler should issue appropriate warnings to the user. We are exploring a strategy that repeatedly applies mathematical identities that reduce the number of appearances of uncertain parameters. There are many such reducing templates. The approach is to parse an expression into a binary tree, and search for matches with a reducing template in each subtree. The search is iterated over all the templates and over all subtrees, and it is repeated until no further reduction occurs. To shorten the list of reducing templates, the matching algorithms automatically test multiple rearrangements of the subtree that are implied by associativity and commutativity of basic operators.

```
#! Automatically reduced number of repetitions
#! of variable x in line:
x = x*a+x*b -> x = x*(a+b)

#! Automatically reduced number of repetitions
#! of variables x and y in line:
z = (x+y)/(1-xy) -> z = tan(arctan(x)+arctan(y))

#!! Can't find repeated variable reduction for x
#!! in line:
z = (a+x)/(b+x)
#!! May cause artificial uncertainty inflation
```

Figure 8: Example Puffin syntax showing how a generated Puffin file could highlight repeated variables to the user and automatically reduce where possible

Acknowledgement

The authors would like to acknowledge the support of the UK Engineering and Physical Science Research Council through grant number EP/R006768/1

REFERENCES

[1] Mike Shafto, Mike Conroy, Rich Doyle, Ed Glaessgen, Chris Kemp, Jacqueline LeMoigne, and Lui Wang. Modeling, Simulation, Information Technology and Processing Roadmap - Technology Area 11. Technical report, National Aeronautics and Space Administration, 2012.

[2] Terence Parr. *The Definitive ANTLR 4 Reference*. The Pragmatic Bookshelf, Dallas, USA, 2012.

[3] Ramon E Moore, R Baker Kearfott, and Michael J Cloud. *Introduction to Interval Analysis*, volume 110. Society for Industrial and Applied Mathematics, Philadelphia, USA, 2009.

[4] Scott Ferson, Vladik Kreinovich, Lev Ginzburg, Davis S Myers, and Kari Sentz. Constructing Probability Boxes and Dempster-Shafer Structures. Technical Report January, Sandia National Lab.(SNL-NM),, Albuquerque, United States, 2003.

[5] Michael Scott Balch. Mathematical foundations for a theory of confidence structures. *International Journal of Approximate Reasoning*, 53(7):1003–1019, 2012.

[6] Scott Ferson, Jason O'Rawe, Andrei Antonenko, Jack Siegrist, James Mickley, Christian C. Luhmann, Kari Sentz, and Adam M. Finkel. Natural language of uncertainty: numeric hedge words. *International Journal of Approximate Reasoning*, 57:19–39, feb 2015.

[7] Scott Ferson, Michael Balch, Kari Sentz, and Jack Siegrist. Computing with Confidence. In *Proceedings of the Eighth International Symposium on Imprecise Probability: Theory and Applications*, Compiègne, France, 2013.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

MACHINE-LEARNING TOOL FOR HUMAN FACTORS EVALUATION – APPLICATION TO LION AIR BOEING 737-8 MAX ACCIDENT

C. Morais[1], K. Yung[2], and E. Patelli[1]

[1] Institute for Risk and Uncertainty, University of Liverpool
Chadwick Building, Peach Street, Liverpool, L69 7ZF, United Kingdom
e-mail: {caroline.morais, edoardo.patelli}@ liverpool.ac.uk

[2] Faculty of Applied Science & Engineering, University of Toronto
35 St. George Street, Room 157, Toronto, ON M5S 1A4, Canada
e-mail: kalai.yung@mail.utoronto.ca

Abstract

The capability of learning from accidents as quickly as possible allows preventing repeated mistakes to happen. This has been shown by the small time interval between two accidents with the same aircraft model: the Boeing 737-8 MAX. However, learning from major accidents and subsequently update the developed accident models has been proved to be a cumbersome process. This is because safety specialists use to take a long period of time to read and digest the information, as the accident reports are usually very detailed, long and sometimes with a difficult language and structure.

A strategy to automatically extract relevant information from report accidents and update model parameters is investigated. A machine-learning tool has been developed and trained on previous expert opinion on several accident reports. The intention is that for each new accident report that is issued, the machine can quickly identify the more relevant features in seconds – instead of waiting for some days for the expert opinion. This way, the model can be more quickly and dynamically updated. An application to the preliminary accident report of the 2018 Lion Air accident is provided to show the feasibility of the machine-learning proposed approach.

Keywords: Bayesian network updating, accident reports, Uncertainty Quantification, machine-learning, Boeing 737-8 MAX.

1 INTRODUCTION

The industry should learn from past accidents to design and manage safer industrial installations, which is described by the ‘learning from incidents’ concept. There are some industrial recommended practices on how companies should use this concept [1] and research on how they are actually using it [2] or how it could be used [3]-[4]. The most acknowledged practice is the risk assessment, where a multi-disciplinary team revise the design according to information about past accidents, components reliability and human reliability.

Comprehensive risk assessments include human, organizational and technological factors [5], where human error probability is the likelihood of an individual to initiate or trigger a sequence of events that can lead to an accident. However, human behaviour is highly variable and depends not only on the individual but also on the organizational and technological factors – all of them sources of aleatoric and epistemic uncertainties. To obtain these probabilities optimizing the ‘learning from incidents’ concept, Morais et al. have developed an approach that uses human factors data from major accidents and a probabilistic tool that accommodates those uncertainties [6]. Bayesian networks were chosen to model human errors due to the possibility of updating the model and its outputs with new evidence for each new accident report that is issued [7]. Reading an accident report and extracting the necessary information required to update the probabilities of the human errors require significant efforts and the availability of a risk specialist [8], resources that are not always available.

In the paper, a machine learning tool based on text recognition and supporting vector machine is proposed to automatically extract relevant information from accident reports. Previous works have used machine-learning to classify textual narratives for aviation and railway into defined (taxonomy) or dynamic (ontology) categories [9]-[10]. The main differences is that they have used a tanomy/ontology not entirely relevant for the human error model, and they have used voluntarily submitted reports, where the model needed inputs from investigation reports.

The proposed procedure also allows creating a “virtual risk expert” trained on using predefined taxometry. The “virtual expert” is than able to process accident reports and extract relevant information in real-time. The proposed methodology is applied to analyse the accident report of the 2018 Lion Air accident [11] is provided to show the feasibility of the machine-learning proposed approach. The approach proposed allows also to understand how the same type of error is perceived and classified in different sectors.

2 METHODOLOGY

2.1 A Bayesian network to predict human error in complex industries

To build a model of human error in complex industries the Bayesian networks proposed by by some of the authors has been used [6]. Bayesian Networks are a probabilistic tool that can be presented in the form of a directed acyclic graph made of nodes (variables) connected by links. The open source Bayesian Network toolbox [12] implanted in OpenCossan [12]-[14] has been used to analyse and evaluate the developed model. The probability values denoting the degree of dependency within the nodes are stored in a conditional probability table, thus each state of the child is provided given each of the states of the parents. The product of all the conditional and unconditional probabilities specified in the network is governed by the chain rule for Bayesian networks [15].

In [6], to build the structure of the Bayesian network the authors have used the dependency among the variables proposed in [8]. This arrangement of parents and children nodes connected by links allows predictive and diagnostic calculations. Therefore, not only human error probabilities can be predicted but also the factors that contribute to those error can also be further investigated. A simplified version of the originally developed model is shown in Figure 1 where the nodes are related to the CREAM taxonomy (Cognitive reliability and error analysis method) for human errors and organisational, technological and individual factors [16]. The CREAM's features adopted in the model are shown in Table 1. The probability values for each node are based on MATA-D (Multiattribute Technological Accidents Dataset), a dataset created by experts (risk analysts) [3], after reading accident reports and classifying them as boolean values (0 for absent, 1 for present) according to the features described in Table 1.

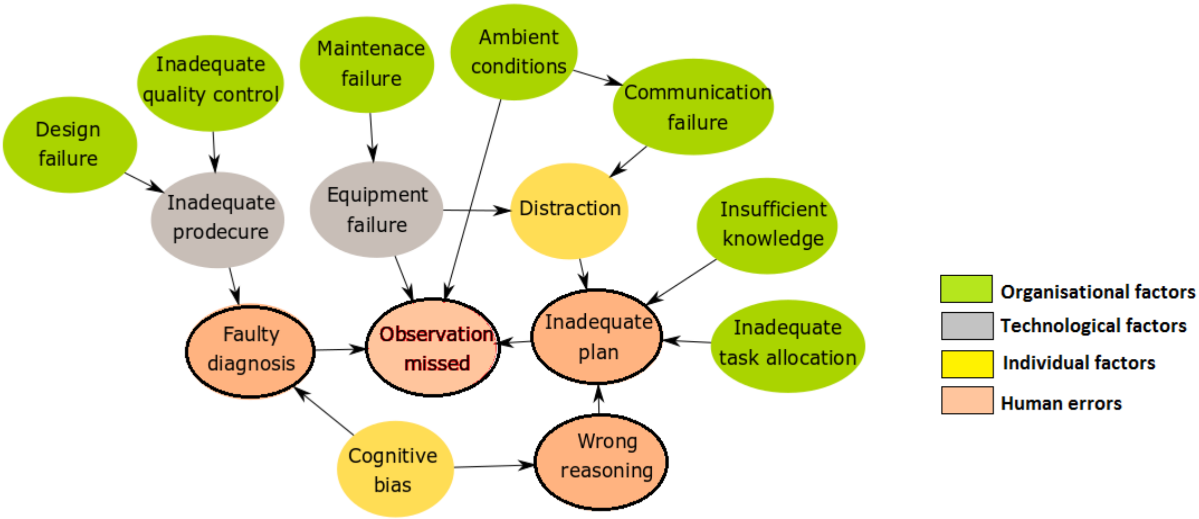


Figure 1 – Simplified representation of the model for Human error derived from [6].

Table 1. CREAM features of human factors adoped in the proposed Human error model.

Organisational Factors	**Technological Factors**	**Individual factors**	**Human Errors**
Communication failure	Equipment failure	**Permanent related**	Cognitive Errors
Missing information	Software fault	Functional impairment	Observation missed
Maintenance failure	Inadequate procedure	Cognitive style	False Observation
Inadequate quality control	Access limitations	Cognitive bias	Wrong Identification
Management problem	Ambiguous information	Temporary	Faulty diagnosis
Design failure	Incomplete information	**Temporary related**	Wrong reasoning
Inadequate task allocation	Access problems	Memory failure	Decision error
Social pressure	Mislabelling	Fear	Delayed interpretation
Insufficient skills		Distraction	Incorrect prediction
Insufficient knowledge		Fatigue	Inadequate plan
Adverse ambient conditions		Performance Variability	Priority error
Excessive demand		Inattention	Execution Errors
Inadequate work place layout		Physiological stress	Wrong time
Irregular working hours		Psychological stress	Wrong type
			Wrong Object
			Wrong place

2.2 Updating Bayesian network probabilities via a machine-learning approach

In an ideal situation, the Bayesian network is updated every time a new accident report is released. Hence, human error probabilities are updated with changes in organizational and technological factors. However, each report has two hundred pages on average, and the reading and classification into a taxonomy is a time-consuming task. Therefore, it was idealised that a trained machine could help on this cumbersome task. A semi-supervised training algorithm could automatically classify the report supporting the analysis of an analyst.

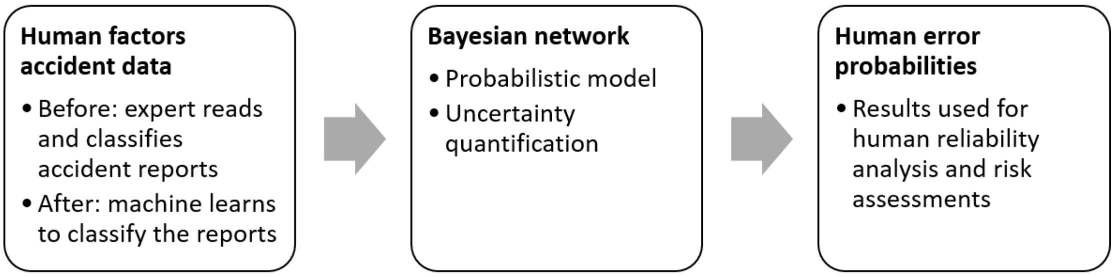

Human factors accident data
• Before: expert reads and classifies accident reports
• After: machine learns to classify the reports
Bayesian network
• Probabilistic model
• Uncertainty quantification
Human error probabilities
• Results used for human reliability analysis and risk assessments

Figure 2 - Conceptual approach to update a human error probability model.

A semi-supervised approach is proposed to analyse accident report and update human error probabilities in the proposed model. The concept of the machine-learning approach is summarised in Figure 2. For this study, the Matlab text analytics toolbox based on the bag of words model [17] is used for extracting text strings from PDF files and preparing the data for the machine-learning algorithm. The Matlab statistics and the machine-learning toolbox is used for transforming text inputs into binary classification adopting Support Vector Machine [18]. A brief background on the models for text selection and machine-learning is here provided.

A bag-of-words model is a way of extracting features from the text, representing it by the vocabulary of known words and a measure of their occurrence. However, it does not provide any information about the order or structure of words – the reason that it is called a "bag" of words. To apply it to a collection of documents, first the data is collected from the text files. Then a vocabulary is prepared by making a list of all the words in the text. To improve the results and save computational time and memory the model ignores case, punctuation, and other frequent words that do not contain relevant information, such as stop words (e.g. 'a', 'the', 'of'). To score the words in each document the presence of known words is marked as a boolean value (0 for absent, 1 for present). Thus, using the list of words previously prepared, the new document is analysed and converted into a binary vector. To extract the features from the documents, the ordering of the words is discarded [17].

The Support Vector Machine is the machine-learning algorithm used, popular due to the little need for adjustments. In the simplest case – when the data has exactly two classes – the Support Vector Machine classifies data by finding the "maximum-margin hyperplane" hyperplane that separates the data points of one class (type 1, represented in the Figure 3 as +) from those of the other class (type -1, represented on Figure 3 as -). Any hyperplane can be written as the set of points **x** satisfying:

$$\mathbf{w}\cdot\mathbf{x} - \mathrm{b} = 0 \qquad \text{(Equation 1)}$$

where **w** is the normal vector to the hyperplane. The parameter b/||**w**|| determines the offset of the hyperplane from the origin along the normal vector **w** as shown in Figure 3. The hyperplanes that defines the classes are can be described by the following equations:

$$\mathbf{w} \cdot \mathbf{x} - b = 1 \qquad \text{(Equation 2)}$$
$$\mathbf{w} \cdot \mathbf{x} - b = -1 \qquad \text{(Equation 3)}$$

The support vectors are the data points that are closest to the separating hyperplane; these points are on the boundary of the slab. As it is a supervised learning model the Support Vector Machine has to be trained first and then cross-validate the classifier. After that, the trained machine can be used to predict or classify new data. In addition, to obtain satisfactory predictive accuracy, various Support Vector Machine kernel functions can be used.

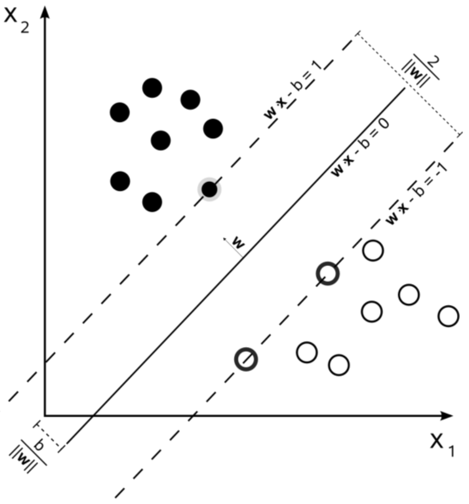

X_2
X_1
$\mathbf{w} \cdot \mathbf{x} - b = 1$
$\mathbf{w} \cdot \mathbf{x} - b = 0$
$\mathbf{w} \cdot \mathbf{x} - b = -1$
$\frac{2}{\|\mathbf{w}\|}$
$\frac{b}{\|\mathbf{w}\|}$
$\mathbf{w}$

Figure 3 – Conceptual illustration of the support vector machine [19]

2.3 Machine-learning tool overview

The proposed machine-learning approach is trained using accident reports classified beforehand by experts according to CREAM taxonomy with the aim of being able to predict automatically the features of similar accident reports. A simplified workflow of the proposed approach is shown in Figure 4.

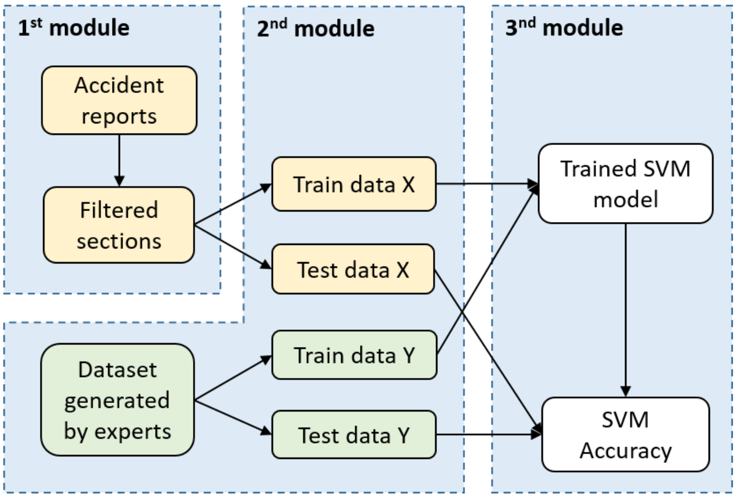


Figure 4 - Simplified workflow of the proposed approach

In the first module of the tool, accident investigation reports are used as inputs. In this study, the documents were in PDF (portable document format). It is important to use an Optical Character Recognition (OCR) software on accident reports that were shared as image files, in order to convert them in text files. After this pre-treatment, the accident reports are scanned and relevant sections are identified for the sake of efficiency. In the current implantation, the semi-supervised approach selects the *recommendation*, *lessons learned*, and *advice* sections of the incident reports. Using a scoring system, the most likely starts and ends of the target sections are identified, and the sections' texts are sent to later code.

The second module of the tool aggregates the aforementioned section texts with the MATA-D dataset information needed to begin machine learning. This dataset contains human factors features to be used as desired outputs for machine-learning. Using another scoring system, the tool takes each accident report's file name and finds the most likely corresponding entry in the MATA-D dataset (as each report listed in the dataset has a correspondent PDF file). This gives the machine-learning component the desired output (i.e. the correct categories) for each incident report. This module's output is a combination of selected section texts and the known human factors features of them.

In the third and last module, the machine-learning model based on support vector machine is trained and tested using the data input from the previous two modules. The section texts are converted into BagOfWords objects as X. The features extracted from the MATA-D data serves as the Y. The module partitions the X and Y data into a training set (90% of total) and a testing set (10% of total). For each CREAM feature, an SVM model is trained using training X and Y sets, then it is tested using the testing X and Y sets. At the same time, run information is recorded and overall accuracy of all test sets in all categories is calculated.

2.4 Accuracy of the machine-learning model created

Each accident report was treated as a document, and the set of accident reports of one specific investigation body was treated as a corpus of documents, The current collection of reports comes from different organizations with considerably different formats and vocabularies. The formats range from a few concise pages in Chemical Safety Board reports to a 200-page letter to the US president on the BP oil spill. In this paper, two corpora used were: the US National Transportation Safety Board (NTSB), that investigates aviation accidents, and the U.S. Chemical Safety and Hazard Investigation Board (CSB), that investigates industrial chemical accidents.

If the machine-learning model is only trained with the NTSB reports, the overall accuracy of the test sets is approximately 85%. If the model is trained with US Chemical Safety and Hazard Investigation Board reports, the accuracy is approximately 91%. This is possibly due to the different number of training data for both corpus; the MATA-D dataset had classified 39 CSB reports and 13 NTSB reports, among a total of 238 accident reports from different industry sectors. However, 85% is considered an equally good result for the classification of narrative reports into a taxonomy, especially if considered that the inter-rater reliabilities within experts are considered acceptable if the label accuracy is above 70% [9].

3 CASE-STUDY – 2018 ACCIDENT WITH BOEING 737 MAX 8 AIRCRAFT

On 2018, an accident with a Lion Air plane has lead to 188 fatalities (two pilots, five flight attendants and 181 passengers) [11]. Five months later, in 2019, an Ethiopian Airlines plane has crashed minutes after take-off, killing all 157 people on board [20][20]. The fact that both planes were the same model, a Boeing 737-8 MAX, concerned civil society and safety regulators about the possible common flaws on all planes within the same model, resulting in all 387 Boeing 737 Max 8 planes grounded globally [21]. This illustrates the importance of learning from accidents before making informed decisions.

For this research the preliminary accident report of the Lion Air Aircraft flight [11] has been tested with the newly developed and trained machine-learning tool, after training the machine with two different training sets: a set with only aviation accident investigation reports and a set with only with chemical industries accident investigation reports. All of the documents were previously classified by an expert within the CREAM human factors taxonomy as shown in Table 1.

3.1 Major findings

The results obtained by the machine after being trained by NTSB (aviation) and CSB (chemical) accident reports can be compared in Table 2 for human errors and Table 3 for the factors that might trigger them. The performance of the machine-learning based tool depends on the quality of the training data. For instance, if the machine learning tool is trained only with aviation reports, it classify four types of human errors against only one if it is trained by chemical accident reports. Also, among the three types of factors (organizational, technological and individual) that may trigger human errors, the machine results after being trained by aviation reports focus much more on individual factors than when trained with chemical reports – giving more weight on the human responsibility upon the system, than the system upon the human. This trend can be better identified in Figure 5, after joining the 53 features into their four highest levels. There are some possible reasons why the training provided by the chemical accident reports are more emphatic on organisational factors, but one is important to discuss: the results on the preliminary report of Lion Airlines flight accident might be describing more about the training corpus them about the actual report. This means that chemical industries might have much more organisational factors initiating accident events than in aviation. This is certainly true in the case of the ‘maintenance failure’ factor. It is possible that a maintenance error initiates an event on a flight, but in chemical industries the probability is much higher as maintenance tasks can be executed while the system operates. Accordingly, it is understandable why human errors and individual factors are much more explored in aviation accident reports than in chemical plants. In aviation, the investigation is focused in the cockpit, on the crew performance. On the other hand, it is not clear to which extent investigators are not digging more to the organisational and technological factors that are triggering the human errors of the crews.

Table 2 – Human errors identified

			Trained with aviation reports (NTSB)	**Trained with chemical accident reports (CSB)**
Human errors	**Execution**	**'Wrong Time'**	**Yes**	**Yes**
		'Wrong Type'	0	0
		'Wrong Object'	0	0
		'Wrong Place'	**Yes**	0
	Observation	'Observation Missed'	0	0
		'False Observation'	0	0
		'Wrong Identification'	**Yes**	0
	Interpretation	'Faulty diagnosis'	**Yes**	0
		'Wrong reasoning'	0	0
		'Decision error'	0	0
		'Delayed interpretation'	0	0
		'Incorrect prediction'	0	0
	Planning	'Inadequate plan'	0	0
		'Priority error'	0	0

Table 3 – Organizational, Technological and Individual factors that may trigger human errors

			Trained with aviation reports	**Trained with chemical accident reports**
Organisational factors	**Communication**	'Communication failure'	**Yes**	0
		'Missing information'	0	**Yes**
	Organisation	'Maintenance failure'	0	**Yes**
		'Inadequate quality control'	**Yes**	**Yes**
		'Management problem'	0	**Yes**
		'Design failure'	0	**Yes**
		'Inadequate task allocation'	0	**Yes**
		'Social pressure'	0	0
	Training	'Insufficient skills'	0	**Yes**
		'Insufficient knowledge'	0	0
	Ambient Conditions	'Temperature'	0	0
		'Sound'	0	0
		'Humidity'	0	0
		'Illumination'	0	0
		'Other'	0	0
		'Adverse ambient conditions'	0	0
	Working Conditions	'Excessive demand'	0	0
		'Inadequate work place layout'	0	0
		'Inadequate team support'	0	0
		'Irregular working hours'	0	0
Technological factors	**Equipment**	'Equipment failure'	0	**Yes**
		'Software fault'	0	0
	Procedures	'Inadequate procedure'	**Yes**	0
	Temporary Interface	'Access limitations'	0	0
		'Ambiguous information'	0	0
		'Incomplete information'	0	**Yes**
	Permanent Interface	'Access problems'	0	0
		'Mislabelling'	0	0
Individual Factors	**Temporary Person Related Factors**	'Memory failure'	0	0
		'Fear'	0	0
		'Distraction'	**Yes**	**Yes**
		'Fatigue'	**Yes**	0
		'Performance Variability'	**Yes**	0
		'Inattention'	**Yes**	0
		'Physiological stress'	0	0
		'Psychological stress'	0	0
	Permanent Person Related Factors	'Functional impairment'	0	0
		'Cognitive style'	0	0
		'Cognitive bias'	**Yes**	0

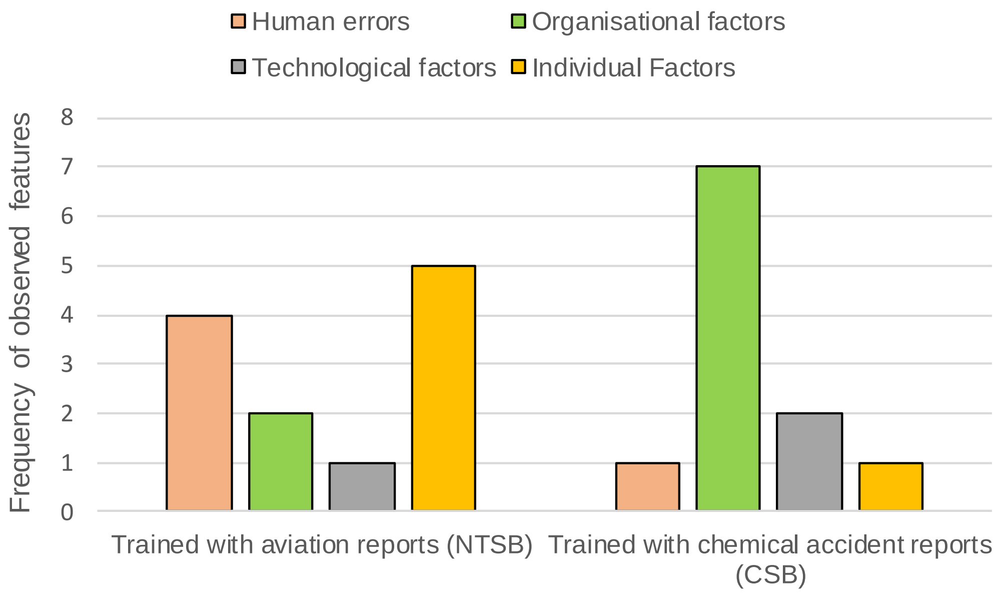


Figure 5 - Observed features after the machine is trained in different types of report

Another important aspect to be considered is the comparison of the present results to what is being communicated by the media and specialists in the area. Although recent news from media (e.g. [21]) and specialists opinion (e.g. [22]) accounts for the possible inadequacy of the software installed on the plane, the developed machine-learning tool has not identified the feature 'software fault' of the taxonomy. Some possible reasons for the lack of 'software fault' identification by the machine are described below:

- The document tested is a preliminary investigation report for the accident occurred to Lion Airlines (on 28 October 2018). There are few mentions to 'AoA' sensor but they do not state a definite problem with it, as illustrated by this sentence extracted from the report: "*The investigation will perform several tests including the test of the 'AoA' sensor and the aircraft simulator exercises in the Boeing engineering simulator. The investigation has received the QAR data for flight for analysis*". Thus, as the data was not yet evaluated and a final report was not issued, the software problem pointed by the media is not official – and it was possible to be perceived as one of the causes only after the accident occurred to Ethiopian Airlines (on 10 March 2019).
- The points stated about the sensors, on this preliminary report, use lots of acronyms or field-specific words possibly not yet related in other accident investigation reports. This could be tackled by training the machine also in the acronyms and specific words for each field, such as those reported in the Aviation Safety Reporting System (ASRS). This might also be improved by using a machine-learning strategy that accounting for the order of the words.
- The model and machine-learning tool developed has not yet achieved 100% accuracy. If it was, maybe could detect the software problem on the preliminary report. It is currently achieving 85% accuracy (if the machine is trained with aviation NTSB reports) and 91% accuracy (if trained with Chemical CSB reports).

4 CONCLUSIONS

This study shows the feasibility of implementing a machine-learning tool to update the Bayesian network probabilities by scanning new reports without the necessity of the time consuming and expensive approach required by the traditional task. The proposed approach is based on text-recognition and text-classification, combined with support vector machine for classifying text according to predefined taxonomy to create a “virtual risk expert”. This allows a real-time update of the model parameter available and it can be of fundamental importance to identify main causes of patterns across accidents.

The case study about the Boeing 737 MAX-8 plane accident has been presented showing that new evidence can be included in the Bayesian network proposed and new human error probabilities can be generated. The results of the analysis show that human factors are revealed when the model is trained using data from the chemical industry and not only from aviation, indicating the importance of cross-discipline knowledge transfer.

ACKNOWLEDGEMENTS

We would like to acknowledge the efforts of Jack Tully and Hao Xu, final year undergraduate students of the University of Liverpool that have contributed to improve and test the code. This research has been supported by ANP, the Brazilian Oil & Gas Regulator, and by EPSRC under the grant EP/R020558/1.

REFERENCES

[1] Center for Chemical Process Safety (CCPS), 2010. *Guidelines for Risk Based Process Safety*. John Wiley & Sons.

[2] Drupsteen, L., Groeneweg, J. and Zwetsloot, G.I., 2013. Critical steps in learning from incidents: using learning potential in the process from reporting an incident to accident prevention. *International journal of occupational safety and ergonomics*, *19*(1), pp.63-77.

[3] Moura, R.; Beer, M.; Patelli, E.; Lewis, J. and Knoll, F., 2017. Learning from accidents: interactions between human factors, technology and organisations as a central element to validate risk studies. *Safety Science*, *99*, pp.196-214. DOI: 10.1016/j.ssci.2017.05.001

[4] Moura, R.; Beer, M.; Patelli, E. &amp; Lewis, J. 2017 Learning from major accidents: graphical representation and analysis of multi-attribute events to enhance risk communication. *Safety Science*, 99, 58-70 DOI: 10.1016/j.ssci.2017.03.005

[5] Zio, E., 2018. The future of risk assessment. *Reliability Engineering & System Safety*, *177*, pp.176-190.

[6] Morais, C., Moura, R., Beer, M., Patelli, E., 2019 (in press) Analysis and estimation of human errors from major accident investigation reports. ASCE-ASME *Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*.

[7] Estrada-Lugo H.D., Tolo S., de Angelis M. and Patelli, E. 2019 (in press) An inference method for bayesian networks with probability intervals ASCE-ASME *Journal of Risk and Uncertainty in Engineering Systems Part B: Mechanical Engineering*

[8] Moura, R., Beer, M., Patelli, E., Lewis, J. and Knoll, F., 2016. Learning from major accidents to improve system design. *Safety science*, 84, pp.37-45. DOI: 10.1016/j.ssci.2015.11.022

[9] Robinson, S., 2018. Multi-label classification of contributing causal factors in self-reported safety narratives. Safety, 4(3), p.30.

[10] Van Gulijk, Coen; HUGHES, Peter; FIGUERES-ESTEBAN, Miguel. The potential of ontology for safety and risk analysis. In: Proceedings of ESREL 2016. CRC Press, 2016.

[11] Transportasi, K.N.K., 2018. Preliminary KNKT.18.10.35.04 Aircraft accident investigation report, PT. Lion Mentari Airlines, Boeing 737-8 (MAX). Ministry of Transportation, Indonesia, Report.

[12] Tolo, S., Patelli, E. and Beer, M. 2018, An open toolbox for the reduction, inference computation and sensitivity analysis of Credal Networks. *Advances in Engineering Software*, 115, 126-148. DOI: 10.1016/j.advengsoft.2017.09.003

[13] Patelli, E., 2016. COSSAN: a multidisciplinary software suite for uncertainty quantification and risk management. In *Handbook of uncertainty quantification*, pp.1-69. DOI: 10.1007/978-3-319-11259-6_59-1

[14] Patelli, E.; George-Williams, H.; Sadeghi, J.; Rocchetta, R.; Broggi, M. and de Angelis, M. 2018, OpenCossan 2.0: an efficient computational toolbox for risk, reliability and resilience analysis Proceedings of the joint ICVRAM ISUMA UNCERTAINTIES conference, http://icvramisuma2018.org/cd/web/PDF/ICVRAMISUMA2018-0022.PDF

[15] Tolo, S. and Patelli, 2015, A computational tool for Bayesian network enanched with reliability methods, Proceedings of the 1st ECCOMAS Thematic Conference on Uncertainty Quantification in Computational Sciences and Engineering Crete Island, Greece, 25-27 May 2015, *Eccomas Proceedia* ID: 4316, 908-923

[16] Hollnagel, E., 1998. *Cognitive reliability and error analysis method (CREAM)*. Elsevier.

[17] Brownlee, Jason Brownlee. "A Gentle Introduction to the Bag-of-Words Model." Machine Learning Mastery, https://machinelearningmastery.com/gentle-introduction-bag-words-model/, website visited on 1st May 2019

[18] Malab documentation. Support Vector Machines for Binary Classification. https://www.mathworks.com/help/stats/support-vector-machines-for-binary-classification.html. Website visited on 1st May 2019

[19] Wikipedia https://commons.wikimedia.org/wiki/File:Svm_max_sep_hyperplane_with_ margin.png.

[20] BBC News. https://www.bbc.co.uk/news/business-47523468. Website visited on 3rd April 2019

[21] BBC News. https://www.bbc.co.uk/news/business-48038026?intlink_from_url= https://www.bbc.co.uk/news/topics/c2g0x3qg9q1t/boeing-737-max-8&link_location=live-reporting-story. Website visited on 25th April 2019.

[22] Gregory Travis Specialist opinion on Boeing 737 MAX-8 failures. IEEE Spectrum. 19 April 2019. Available at: https://spectrum.ieee.org/aerospace/aviation/how-the-boeing-737-max-disaster-looks-to-a-software-developer .

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

RECOMMENDER TECHNIQUES FOR SOFTWARE WITH RESULT VERIFICATION

Ekaterina Auer[1] **and Wolfram Luther**[2]

[1]University of Applied Sciences Wismar,
D-23966 Wismar, Germany
e-mail: ekaterina.auer@hs-wismar.de

[2] University of Duisburg-Essen
D-47048 Duisburg, Germany
e-mail: luther@inf.uni-due.de

Keywords: software, scientific recommender systems, initial value problem, ordinary differential equations.

Abstract. *Methods with result verification such as interval analysis or affine arithmetic have been used successfully at least since the 1970s not only for dealing with the automated proofs that simulation results obtained using computers are correct, but also for taking into account the influence of bounded uncertainty in the input on the outcome of a simulation. There are many packages developed for providing basic arithmetic computations on different platforms, for example, filib++ in C++*[1]*, PyInterval in Python*[2]*, or such exotic implementations as JuliaIntervals in Julia programming language*[3]*, and this is only a small choice of tools for interval arithmetics. Moreover, there are packages for higher level algorithms such as solving initial value problems for ordinary differential equations (e.g., VNODE-LP), global optimization or linear/nonlinear systems of equations (e.g., inside C-XSC Toolbox). However, despite this abundance of software solutions, set-based methods with result verification are rarely used by an ordinary engineer for dealing with bounded uncertainty. In our opinion, one reason for this unpopularity is that engineers do not have time to compare the existing tools and choose the package that is most suitable for their task. To address this problem, we suggest using automatic recommendations.*

In this paper, we focus on software for solving initial value problems since it is important in many application areas such as biomechanics or automatic control. We show how modern concepts from the area of recommender systems can be employed to obtain an automatic suggestion about what tool to use for a given application and what prerequisites are necessary to be able to do so. We discuss in general what kind of data, metadata, quality criteria, metrics, and visualizations are required to be able to compare and recommend software with result verification. Finally, we present algorithms for recommendation and illustrate their functionality.

[1] `www2.math.uni-wuppertal.de/~xsc/software/filib.html`

[2] `pypi.org/project/pyinterval`

[3] `github.com/JuliaIntervals`

1 INTRODUCTION

Digital assistance becomes more and more ubiquitous in everyday private and working life. Therefore, topics with applications to ambient intelligence and smart environments are of particular interest in the field of information science. Research trends focus on cross-cutting issues such as standardization, verification and validation assessment, or development of formal testing and quality criteria concerning, for example, reliability, performance and user satisfaction, which makes devising versatile metrics and agreeing on unified vocabularies across disciplines especially important. As demonstrated in [8], new software engineering priorities are in such technologies as search engines, recommender systems, and general data mining techniques.

The goal of recommender systems (RS) [2] is to aid consumers while they select from a variety of products, mostly with the aim to increase product sales (e.g., for online shops) or otherwise promote a given business (e.g., for social networks). Depending on the intended application, a number of other – operational or technical – goals can be set concerning, for example, relevance, novelty or diversity of recommendations. For generating suggestions, it is assumed that meaningful rules can be discerned about the way the consumers (users) select the products (items) or, vice versa, about the products most suitable for (a group of) customers. To achieve the mentioned goals, RS might take into account such characteristics as the intended product use, customer behavior, or product ratings and try to compile a ranked list from the multitude of offers (or predict a user's rating for a given item) according to criteria specified beforehand. Beyond the classical application in the area of e-commerce, RS are also in high demand for digital products, for media (including such different aspects as movies, news, or scientific publications), for healthcare, learning, or artificial-intelligence-based services. This new perspective requires novel, formally described standards, quality criteria and metrics [66] to determine the ranking in the sense of a multi-criteria optimization supported, if possible, by test environments (e.g., for software tools) and evaluation guidelines for recommendations.

Whereas evaluation of recommendations is a fairly well-known topic (cf. [2], Chapter 7), the related notions of testing and test environments might require a further explanation. Their obvious goal is direct certification of a RS, as, for example, in case of the data set and precise test descriptions provided in the context of the Netflix prize [4]. On the other hand, their development might be motivated by human factors: While some consumer groups might make decisions taking into account just the quality criteria, fulfillment of requirements, known benchmarks results, or recommendations from peers, other consumer groups, in particular, professionals, might want to parameterize and try out different digital products in a common test environment themselves to find the right product or the right solution in a given situation and for a specified task. A new research direction is therefore scientific RS (SRS) for digital products supported by test environments.

In our view, a scientific RS should possess the following characteristics:

1. It is a recommender for scientists providing a ranked list of items (e.g., software tools) suitable for the user's task.
2. It is usually knowledge based, context aware and multi-criteria; its item ratings evolve with time (e.g., requirements for software quality, its usability and the degree of interaction might evolve with time).
3. It provides recommendations which are relevant for a given user with respect to predefined (or inferred) criteria.

[4] `www.netflixprize.com`

4. It is flexible in the sense that (groups of) items or users with new attributes can be added easily to the set-up.
5. It is reliable, for example, is based on formally described criteria, uses reliable algorithms, provides explanations of recommendations for credibility and traceability of decisions, and takes care of good provenance of data (e.g., through common test conditions). Trustworthiness is supported through recommendation rating by users.
6. It can be supported by a test environment, which can take care of the cold-start problem and active learning.

By contrast, such RS subgoals as novelty, surprise, or diversity (cf. [2], page 3) need not be considered. Although context-aware, SRS does not necessarily need to take into account such domain-specific challenges as location and the social component. Social media for scientists such as `www.academia.edu` or `www.researchgate.net` belong to the general class of social RS such as `www.facebook.com` and might contain subcomponents which can be classified as SRS (e.g., recommending project partners).

In this paper, we apply the general principles and techniques of (scientific) recommendation in a specific context of verified software. After giving a brief overview of the state of the art in SRS along with suggestions of how it can be employed for our purposes (Section 2), we illustrate the concepts summarily in Section 2.2 using the application area of visual analytics, where automatic choosing of the appropriate visualization technique is a common software feature (often without the name of recommendation). Our focus is on recommending techniques for verified solution of initial value problems (IVPs) for ordinary differential equations (ODEs), an area where we would like to offer users similar options. In Section 3 we point out what is necessary to recommend a scientific tool in general. In Section 4, we describe first the available verified IVP solvers along with the necessary conditions and rules to be able to give a recommendation. Moreover, we provide and illustrate a specific algorithm using a small data set. Finally, we point out possible improvements and further research directions. Conclusions are in the last section.

2 SCIENTIFIC RECOMMENDER SYSTEMS

The purpose of scientific RS is to recommend an item (a ranked list of k items) to a scientist in a broad sense (e.g., engineer, researcher, teacher, student). This item can be a digital product, for example, a software tool or a technique, from various application areas such as visualization or education.

One classical formulation of a recommender problem is to determine a list of recommendations (for items) based on preferences and needs of a user (group), which we see as the most relevant formulation for a scientific recommender. Three general RS techniques are collaborative filtering (using information about user-item interaction), content-based methods (using information about attributes of users and items), and knowledge-based methods (using explicit information about user requirements). SRS recommend items for scientific tasks that fulfill predefined requirements and constraints and make use of all of the mentioned general techniques where necessary. To establish a list of recommendations, SRS often rely on object or case-based approaches accompanied by filtering and learning algorithms, similarity measures to compare items (cases), quality criteria and metrics to select and rate suggestions, and intelligent algorithms to match users' quality criteria with product properties/descriptions or to find people with similar interests/profiles and expectations for the product.

In this section, we identify the most common SRS topics first. After that we describe the subtopic of visual analytics in more detail. Note that our literature review is not supposed to be complete and has a narrow focus. For general overviews of RS and their research topics, see, for example, [2, 32], and for SRS [22, 25, 53, 58, 59, 49], which are mainly about RS in education and paper recommending. In [50, 14, 26], the state of the art is described for aspect-based RS which make personalized recommendations taking into account the users' opinions about aspects of the rated items extracted from their reviews.

2.1 Topics for scientific recommendation

Below, we identify research topics in scientific recommendation (in capital letters) and exemplify them by (a) relevant publication(s). The considered SRS mainly use a case database with benchmarks for problems and their solutions; retrieve similar tasks; reuse, adapt and revise solutions and retain new cases and new user preferences. That is, they can be classified as mostly knowledge-based and employ appropriate quality criteria and metrics. Case-based and feature-based SRS continue to evolve by including new user groups, their tasks and environments as shown in [39].

SRS-SE recommend relevant activities for software engineering (SE) tasks and support developers during programming of software components by providing "information items estimated to be valuable for a software engineering task in a given context" [56]. Specifically, SRS-RE offer help in the area of requirements engineering (RE) considered to be "one of the most critical places in software development" [22] by employing the whole range of recommender techniques from collaborative filtering to social media related algorithms.

SRS-P make a choice of relevant papers (P) from a specified scientific field. Scientific paper recommender systems are extensively described in [59], supplemented by several new metrics and a comparative/contrasting definition of various recommendation tasks. An important subtask is to extract semantic relations between keywords from scientific articles in order to support users in the process of browsing and searching for content in a meaningful way [38]. SRS-E help students and teachers to make choices (e.g., of suitable courses) in the educational (E) context [23]. Further tasks for SRS-E are given in [4], where the authors describe an RS that can be applied for finding experts in academia, for example, supervisors for students' qualifications or research, reviewers for conferences, journal or project submissions, or partners for R&D proposals. SRS-STI, that is, RS for scientific and technical information (STI), are addressed in [48]. Here, a more general point of view is adopted by combining the angles of SRS-P (e.g., scientific libraries), SRS-E (e.g., e-learning) and others. The privacy issues and the cold start problem are addressed and several algorithms for the generation of behavior-based recommendations are explored there.

The next topic is the one of the most relevant w.r.t. the goal of this paper. SRS-PSE provides recommendation for problem-solving environments (PSE) [28, 27, 67]. In [67], the project CB-Matrix is described – an early development in the area of devising "intelligent recommender components" to assist scientists in choosing and applying scientific tools. The developers of the project PYTHIA [28, 27] start out by recommending software/methods for partial differential equations and then extend their methodology to enable users to prototype their own recommenders on the basis of their own databases and specifications for interaction with underlying execution environments. The resulting customizable web-based platform MyPYTHIA does not seem to be freely accessible online anymore[5]. MyPYTHIA leaves the problem of a

[5] The service swMATH `http://swmath.org`, which is a SRS-P itself and provides information on mathematical software based on the analysis of publications [9], only supplies links to papers for the keyword

common test environment out of consideration [27]. SRS-R (SRS for reliable (R) or verified software/hardware) and SRS-VA (SRS for visual analytics, VA), which are in the focus of this paper, can be considered as subtopics within this general setting. Even if it is not explicitly termed as a RS in such publications as [7], the mechanism behind choosing visualization techniques based on optimization of a metric w.r.t. quality criteria can be seen as such. More information about this topic is in Section 2.2. SRS-R deal with reliable hardware and software components, use reliable algorithms and include evaluation strategies for the system outcome, even if ground truth to assess accuracy is missing. This topic is described in detail in Section 4. Finally, SRS-AS, RS for assistive software (AS) [24], enable existing interoperability architectures to automatically select the most suitable assistive software for a given interaction with a specific electronic target device taking into account the user's benefit and disabilities.

Knowledge-based RS often employ ontologies (constructed beforehand in a 'intelligence' step, e.g., from user reviews) for generating recommendations. Ontologies provide a structured framework for modeling concepts and relationships between scientific domains of expertise. They are a prerequisite for development of domain knowledge metadata bases for modeling, communicating and sharing knowledge among people (or problem-solving applications). A lot of work has been done in this field, also from the angle of artificial intelligence. For example, PROTEGE-II [62] is an implementation of a methodology for building knowledge-based and domain specific knowledge acquisition systems. The tool provides protocol-based decision support in a specific medical domain. Another tool, CEDAR OnDemand [11] allows users to enter ontology-based metadata conveniently through existing web forms from their own repositories. The web page contents is analyzed to identify the text input fields and associate them with automatically recommended ontologies. Finally, there are modeling languages based on ontologies. For example, the publication [17] shows how to employ the unified problem-solving method development language (UPML) as a comprehensive framework for modeling libraries of methods. UPML provides a hierarchy of concepts to specify knowledge components. In particular, the description of a method includes a competence (defined by a set of input and output role descriptions as well as preconditions and postconditions, e.g., formulas for inputs and outputs), a separate method ontology (definitions of the concepts and relationships of a method) and its associated operational description.

2.2 Recommending a visualization

A publication on scientific recommenders would not be complete without mentioning the field of visualization, which is a very extensive topic. A lot of work has been done on choosing the right visualization for the problem at hand, often without explicitly calling it a recommendation. In this section, we mention the most important publications in this area from our point of view and describe an application of such techniques in the field of steel production.

2.2.1 Short overview of recommender tools in visualization

The general goal of SRS-VA is to automatically suggest a visualization providing insight about the data under consideration, ideally taking into account their characteristics and domain as well as individual user preferences. Accordingly, approaches to visualization recommendation can be classified loosely into four categories [34]: RS based on data characteristics, RS (additionally) using representational goals, RS employing domain knowledge to improve recommendations, and RS relying on explicit interaction with users to infer their preferences. The first group can be considered as the most widely spread one since it was explored long before the term RS had been applied to VA. The others appeared as a result of cross-cutting research in such areas as RS, data science, information visualization, and artificial intelligence. The boundaries between groups are not sharp so that there are methods using (parts of) techniques from the other groups. Below, we exemplify the concepts with appropriate RS references. For more information, see [34].

In the first group, the authors of [40, 60] suggest encoding ordered sets of user-specified data and metadata descriptors by visual variables (e.g., size, texture, color, shape). They develop a compositional algebra to enumerate the space of encodings and apply a set of visual integrity criteria to prune and rank the set of visualizations. This approach resulted in algebraic specification language VizQL with the help of which both the structure of a view and the complying queries can be specified and used to fill the structure with data. Moreover, the module Show Me [40] introduces a set of heuristics to extend automatic presentation to the generation of tables of views (small multiple displays) and recommend chart types. This research is implemented in a commercial tool Tableau[6]. An example of a free tool from the same class is the web application Voyager [68, 69]. The Voyager approach uses statistical and perceptual measures for finding out interesting relationships between data and transformations and allows for automatic generation and interactive steering of views as well as refinement of multiple recommendations.

One of the research goals in the second class of SRS-VA is automating generation of user tasks from natural language descriptions instead of creating them manually [34]. In the latter case, there is a connection with formal modeling methods for user interfaces/interaction. In the former case, advanced linguistic techniques are necessary. An example here is the tool Improvise[7]. The tool SemViz[8] [46] belongs to the third group and uses knowledge ontologies from the semantic web for adaptive semantics visualization. Similarly, a knowledge base of various ontologies is used in [65] to recommend visualizations. Here, the whole range of techniques from the previous classes is employed: Although rule-based and functional requirements govern discovering and ranking of potential mappings, such factors as device characteristics, data properties and descriptions of tasks influence the pre-selection and the final ranking. Finally, tools like VizRec [45] or VizDeck [35] belong to the last class and employ information about perceptual guidelines and explicit feedback about user preferences.

The publication [7] gives a comprehensive overview of metrics to compute the quality of a visualization which have been introduced and discussed for different information visualization techniques in recent years. The *quality-metric-driven automation* layer added to the visual analytics pipeline can serve directly as the basis for making data characteristic oriented recommendations, which is (implicitly) suggested to be done by multi-criteria optimization. In particular, the authors cover node-link diagrams and matrix representations for relational data; parallel coordinates and pixel-based techniques for multi-dimensional data; scatter plots and scatter matrices for high-dimensional data; TreeMaps for hierarchical data; radial visualizations when focusing on one dimension (e.g., a person); glyphs, line and bar charts for uncertainty visualization; and, finally, typographic visualizations and tag clouds for visual representation of text data. Additionally, geo-spatial data visualizations are examined separately as a case of special purpose visualization. As the authors explain, the selection focuses on fields in which quality criteria and quality metrics along with their underlying concepts, tasks and evaluation efforts are (semi-)formally described and can be examined analytically. Moreover, they present a high-level overview of visual exploration goals supported by the majority of metrics, for ex-

[6] www.tableau.com

[7] www.cs.ou.edu/~weaver/improvise

[8] knoesis.org/semviz

ample, clutter reduction filtering out noisy views, identifying data groups and partition clusters, establishing relations between dimensions, filtering out outliers, or preserving original data properties in the mapping process while reducing the number of dimensions. However, no user studies are conducted that compare the different metrics for different tasks and different data characteristics from a human-centered point of view.

2.2.2 Challenges in visualization recommender research

The directions of research in SRS-VA are aimed towards stronger involvement of human factors (e.g., higher interactivity) and domain specifics in generation of recommendations which might necessitate higher use of formal languages, standards or ontologies (e.g., for encoding tool and task categories). Filters concerning user experience and further (better) quality criteria and metrics to rank recommendations remain topics of interest [57]. A further challenge is that there exist many tools and methodologies for visualization, which requires possibly expensive filtering, which in turn influences the efficiency. Besides, finding competent users is necessary for selecting the right quality parameters out of a large number and for specifying optimization goals. Finally, although extensive research in this direction exists [7], it is a challenge to devise computable quality measures for optimization. For that, representative situations and datasets, users, tasks, and quality criteria are necessary. Quality measures could be derived from evaluation studies concerning task categories, user experience and interaction styles; concerning visualization tools (with the focus on performance, accuracy, usability, result presentation readability, integrity); and concerning data and metadata quality.

2.2.3 An application to steel production

In the previous subsections, we described recommendation methods relying heavily on formalizing different concepts in visualization. It is also possible to approach the task empirically, which overlaps somewhat with the class of RS based on explicit user interaction. For example, users are urged to explore a small number of parameter variants using large singles and small multiples as alternative views in [21], allowing for efficient data analysis.

A similar approach is adopted by the inclusion processing framework viewer IPFViewer 2.0, developed under guidance of the second author and employed in the area of steel production for analyzing (big) data collected about non-metallic inclusions and other defects in steel samples [61]. Extensive interviews were conducted with experts after the initial version 1.0 had appeared, during which alternative visualization concepts (i.e., various forms of multiple views) had been shown to users. IPFViewer was adapted to the outcome of the survey in its version 2.0, which in turn was evaluated again by the same experts. That is, a number of visualization recommendations (for highly specialized experts) were generated and evaluated comparatively through user feedback w.r.t. their suitability.

IPFViewer 2.0 takes into account process parameters such as intentional settings or measurements taken during monitoring of various steel grades and their metadata, defect parameters, descriptors and volume data for each defect, isoperimetric shape factors (e.g., volume or surface area), sample parameters (e.g., milling machine slices of the steel surface), and statistical descriptors of the defects (e.g., the sample cleanliness). It performs 3D reconstruction of cracks, non-metallic inclusions or pores. The tool can analyze the ensemble data set in various ways, for example, detect outliers to identify samples that differ from the others by position, size, type and number. To rate steel quality, it carries out trend analysis to study the influence of different process parameters on the steel samples and variance analysis to examine natural fluctuations within the samples and desired variations that result from process parameters. When required, IPFViewer relies on incremental, approximate analysis techniques to ensure the responsiveness of the application while sufficient precision is guaranteed for queries with fast response times.

The steel production facility workers are now able to quickly and interactively analyze data with millions of data rows. The resulting data tree is visualized as a huge grid in a scrollable area. Each grid cell incorporates a multiple view system with such standard visualization techniques as scatter plots, bar charts and trend graphs. Steel experts examine the histogram about defect diameter and the largest found defects to evaluate a sample quickly without having to analyze each defect manually. They can also scroll through all the samples and compare them, create and save various layouts that visualize different aspects of the data in order to confirm or refute hypotheses.

3 WHAT IS NECESSARY TO RECOMMEND A SCIENTIFIC TOOL?

In Figure 1, three general steps most SRS have to undergo to generate a recommendation are shown. The purpose of the first one is to extract information which describes a given user's request for a recommendation. This retrieval can be automatic (e.g., identifying keywords from texts), interaction-based (e.g., asking users to enter keywords) or manual (e.g., rigidly fixing the keywords). Here, the base data/metadata set is produced. In the next step, new information is generated, for example, taking into account similarity measures or based on ontologies, possibly including machine learning algorithms. This produces candidates for recommendation. Finally, the candidates are ranked according to predefined criteria or metrics and the resulting list of recommendations is conveyed back to the users. An important additional step is evaluation of the produced recommendations. For example, if there is a common testing environment for the items of interest, the recommendation can be validated additionally and the feedback about these validation results reused at the intelligence step. There are also other possibilities for evaluation such as user studies, see [2], Chapter 7.

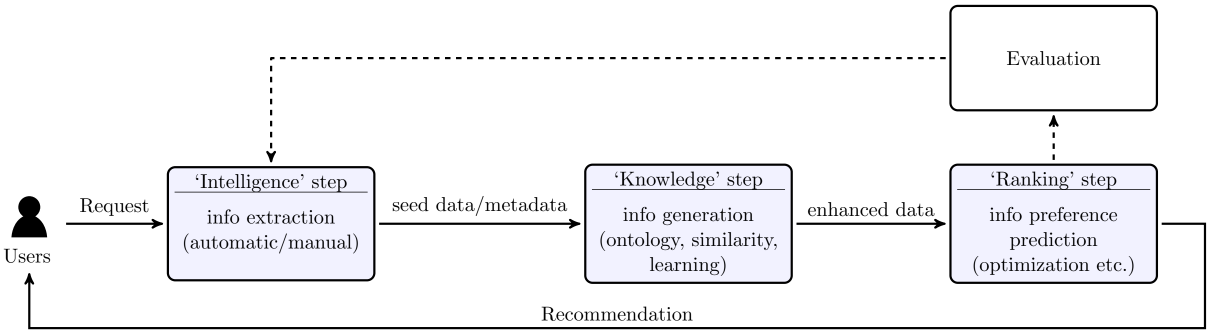


Figure 1: General stages in RS. Steps strictly belonging to a RS are shown in blue.

To illustrate these steps and to see what is necessary to implement a scientific recommender, let us consider a relatively simple example from [67]. The task of the RS from [67] is to recommend data structures (e.g., block matrices) for solving large sparse linear systems based on previously solved use cases. At the first step, features of matrices are determined (e.g., number of non-zeros, degree of bandedness) and a database of past information on pairs "matrix"-"data structure" is created. Possibly, the data have to be normalized beforehand. At the next step, a predefined similarity metric is used to be able to determine matrices similar to a given new one. To improve similarity detection, tests based on genetic algorithms can be carried out here to automatically determine a 'good' set of feature weights. At the third step, the most suitable data structures for these similar matrices are determined and ranked for the recommendation according to a 'measure of goodness' (i.e., performance in flops). The recommendations are evaluated in cross-validation tests. As shown in [27], such principles can be generalized and used to generate recommenders themselves.

Since the goal of a SRS is to provide a ranked list of tools best suited for a given user task as explained earlier, we describe a possible approach to choosing the most suitable item (or a list of them) and relate it to the scientific tool context. An interesting ranking method based on keywords is introduced in [51, 52]. A more sophisticated ranking algorithm is described in [70]. For textbook approaches in network context, see [2], Chapter 10.

As already explained in relation to VA, *multi-objective optimization* can play an important role as an RS technique and be seen independently of visualization context. A quality function q and the associated algorithm A are defined in dependence on the problem-solving tool v, several descriptors (e.g., those pertaining to data, users, and tasks) and some side conditions. As proposed in [7], the algorithm A has to solve a multi-objective optimization problem in order to find a problem-solving instance v (or a ranked list L of such tools v) to maximize (minimize) q. The choice of parameters to optimize and their ranges depends on the task definition, requirements, equality and inequality constraints, valid standards or measurements/experiments for validation. This choice can be made by trial and error, searching or filtering. The function q is characterized by quality criteria and quality metrics defined in the context of the user group and its profile, the task and its model, the data, metadata and data types mapped to the tool v belonging to a predefined set of computer-based problem solvers, the hardware and its interfaces. The quality criteria encompass performance of the task completion including effectiveness and efficiency, reliability criteria for the input data that need to be mapped by v to the outcome space accurately and efficiently. If the optimization problem can be solved automatically, then the problem solving tool v or tool selection and its/their quality metric parameters fulfill the requirements and side conditions and can be recommended to the user (group) for drawing conclusions and making decisions. Typically, requirements concern the solution or its enclosure under uncertainty in parameters. The success of this approach depends on whether an effective and efficient implementation A of the target function q and its computation can be provided.

In order to produce L, a strategy similar to [59] can be employed. There, a similarity-based diversity metric m_{div} is considered for a set P of scientific papers p_i as a normalized sum

$$1-m_{\text{div}} := c^{-1}\cdot\sum_{i\neq j} m_{\text{sim}}(p_i,p_j) \in [0,1], \text{ where } c = (|P|(|P|-1))\cdot\max_{i\neq j}\ m_{\text{sim}}(p_i,p_j)\ , \quad (1)$$

which can be adapted to scientific tools. That is, such tools can be seen as similar if their values for a given quality measure differ only slightly on a set B of benchmark problems. Further, the term 'coverage' is introduced in [59] as "the extent to which all important aspects and subtopics of a scientific field are covered by a set of papers". It is an open research topic to develop a method to similarly cover a given problem space with a small number of scientific tools that solve the benchmark problems w.r.t. given requirements, constraints and quality criteria, for example, performance, accuracy, and usability. That means describing the problem space as a multidimensional space T depending on problem descriptors and compiling a small number of tools in a list L that clusters the space T. This can be done by using an average similarity distance

$$Av(d_{\text{sim}}(L)) := \frac{\sum_{i \neq j} d_{\text{sim}}(v_i, v_j)}{|L|(|L| - 1)} \tag{2}$$

for a set $L \subseteq T$ of scientific tools v_i and the similarity distance d_{sim} by constructing a sequence $Av(d_{\text{sim}}(L_i))$ with further tools $v_{i+1} \in T$ as far as possible from the already taken item set $\{v_1, v_2, \ldots, v_i\} \subseteq T$. For this, we arrange the distances according to their value in descending order and start with two most distant items v_1, v_2. The process can be terminated if the last distance $d(L_i, v_{i+1})$ falls below a certain limit c. Then for each $v \in T$ there is at least one item $v_i \in L$ with $d_{\text{sim}}(v_i, v) = c$.
To summarize, what is needed for SRS is the following.

Database User features, item features, user-item information
Information generation strategies Classification, ontology
Metrics Means for establishing similarity and goodness (quality criteria, weights to reflect situational context)
Ranking algorithms based on metrics such as the similarity-based diversity metric (1) or the the cosine metric (6)
Common test environment (optional): Database generation, evaluation.
Generalization (optional): Means to decouple a recommender from the actual feature vector or metric/criteria instantiation

4 A SCIENTIFIC RECOMMENDER FOR IVP SOLVERS

In this section, we describe an algorithm to recommend verified initial value problem solvers (IVPS) for ODEs and its possible generalizations. Verified methods [44] are constructed in such a way as to provide a mathematical guarantee that a solution obtained on a computer is correct. IVPS generate numerical sets that are mathematically proved to contain exact solutions. They are useful in different contexts, for example, for computer-assisted proofs [63, 64] or for propagating bounded uncertainty through systems [54]. There are many free libraries implementing verified IVPS techniques, which we describe in some more detail in Subsection 4.1. However, an average engineer is disinclined to use them, main reason being the difficulty to choose the right method for a given problem without having the full knowledge about the subject. Some verified methods might be too simple to be used for an advanced application leading to very conservative or pessimistic results; other methods might be too prohibitive computationally. This led us to the idea of implementing a common web-based environment for testing such verified IVPS, which can serve as a basis for a recommender [6].

As far as we know, there are no comparable recommenders for non-verified, normal floating-point arithmetic based IVPS. Such recommendation portals as MyPYTHIA [27] could probably have been used for that purpose but does not seem to be online anymore. Note that the MyPYTHIA application to partial differential equations described in [27] could serve as a good basis for the appropriate IVPS recommender. There are several web platforms for gathering benchmarks, testing and comparing traditional non-verified IVPS, most notably TEST SET at `pitagora.dm.uniba.it/~testset/` building on research from DETEST [29] and similar. Besides, the service swMATH [16] semi-automatically manages the existing Web information about mathematical software. However, verified IVPS have to be compared based on different criteria: for example, they always produce “correct” solutions, that is, the reliability of the results does not need to be tested. After describing available software for verified solution of IVPs and our testing environment VERICOMP, we analyze existing literature concerned with testing and comparing them to identify possible quality criteria and problem classification in Subsection 4.2. The rest of this section is devoted to our recommender algorithm (including an illustration) and possible improvements.

4.1 Verified IVP solvers and VERICOMP

A number of most widely known IVPS are summarized in Table 1 along with some of the newer tools. Some of these IVPS are also suitable for computing solutions for hybrid system dynamics (e.g., Flow*), algebraic-differential equations (CORA) or Poincaré maps (Isabelle), some of the tools additionally provide non-verified solutions (e.g., CAPD) or the use of multiple-precision arithmetic (e.g., kv).

It can be seen from the table that the IVPS are based on very different algorithms (Column 4) with different data structures implemented in different programming languages (Column 2) using different verification concepts (Column 3). A further point is that their performance often depends on the right choice of their settings, which should be preferably tuned to the given problem by their respective developers (Column 5). For the sake of presentation clarity, we only show parameters which are important in our opinion. The time span for the simulation, that is, the initial and the final integration time, is also an important setting and can be specified within all IVPS. Although most of the solvers use only result verification, a lot of effort has been devoted to formal verification of solvers' codes [31, 41] recently. VNODE is a tool relying on the concept of literate programming [37, 47] for code verification. Literate programming allows a human expert to assess in a comfortable way if a code is correct. The list of solvers is not complete, for more software consult, for example, `cps-vo.org/group/ARCH/ToolPresentations`.

A forum for comparing software for verification of continuous and hybrid dynamical systems is offered by the workshop ARCH [1] and its friendly competition[9]. One of the aims is to establish a curated set of benchmarks submitted by academia and industry. This extensive information service gathers and makes accessible benchmark problems, tool presentations, and experience reports in form of papers. However, the approach has shortcomings. The workflow of the competition is to join a group first, then determine the set of problems from the ARCH pdf repository, perform the tests, and, finally, prepare a report. This workflow is not automatized, the responsibility for the correct implementation and testing with the benchmarks for given tools lies with the user/developer, there is no common testing environment, and the results from different reports are not immediately reusable since stored in papers and not in any kind of a common database precluding an automatized recommendation. The long-term goal of our web-based platform VERICOMP [6] is to provide such a common, automatized, recommender-enhanced comparison environment.

VERICOMP is a service for actually comparing verified initial value problem solvers for systems of ordinary differential equations using common comparison conditions. One possibility to employ it is for developers of new IVPS in order to relate their tool to the state of the art. Another possibility is to employ it to decide what solver is the best for a given problem. For users be able to do so at a glance, VERICOMP uses work-precision diagrams, solution plots, and text tables. Here, developing further VA strategies for representing these heterogeneous, large data is our future work. The gathered information is stored in a MySQL database.

The tests with verified IVPS might take considerable time. Therefore, a recommendation should be provided based on similar problems from the database. The RS results can

[9] `cps-vo.org/group/ARCH/FriendlyCompetition`

be additionally validated by actually performing the available tests. Three solvers Vnode-LP, ValEncIA-IVP, and RiOT with various parameters were provided for testing in the old version of VERICOMP under `vericomp.inf.uni-due.de`, the service of which is unfortunately discontinued. VERICOMP 2.0[10] taking into account generalizing features from Section 4.4 is under construction. At the moment, only the feature of adding IVPs to the database and browsing them is accessible. We work on implementing the functionalities discussed in Sections 4.2,4.3.

It is our long-term goal to provide a common environment for testing all the verified solvers mentioned above, which means that a semi-automatic procedure for adding a new solver to VERICOMP is needed. However, this is extremely difficult due to differences in interfaces, concepts, programming languages and platforms. Therefore, we concentrate on the intermediate goal of providing a database which needs to be filled by IVPS' developers themselves. This leads to the lesser challenge of providing the set of problems in the form well suited for running tests with them with different software, which we work on at the moment. This requires a survey on opinions of expert users.

4.2 Existing comparisons and quality

Emerging and old verified software is permanently being tested and compared with some standard benchmarks. For example, VERICOMP's problem database was used as a benchmark for new IVPS in [18, 19]. More tests are described and made available through papers at ARCH. However, every paper presenting a new solver (e.g., [31], [41]) features tests and comparisons according to some criteria relevant for the authors and using benchmarks the authors consider appropriate. To be able to have an overview over the whole range of available possibilities, it is necessary to standardize the comparison criteria and the tests as well as devise various benchmark problem sets. In this subsection, we analyze the publications [12, 19, 20, 31, 33, 18] from Table 1 with the goal to establish a common set of quality criteria and testing aims. Besides, we name the problems most commonly used as benchmarks.

The *quality criteria* can concern performance, accuracy, efficiency, or usability. Usability is often of a minor significance in IVPS tests, which is not entirely justified since such factors as the ease of interfacing an IVPS with a given application might play a crucial role in practice. In verification context, accuracy means the degree of pessimism in the resulting enclosure (e.g., overestimation). Overestimation is not always easy to characterize: the width of the resulting enclosure is only a good indicator for that if all problem parameters are crisp and do not contain any uncertainty [6]. More research on overestimation characterization for tests is necessary. Statistics on the following quality criteria were actively gathered by the old version of VERICOMP:

C4 wall clock time (t_{c}) at a predefined integration time t_{out} (performance),
C5 user CPU time t_{us} w.r.t. overestimation e_{us} at t_{out} (efficiency), and
C6 time to break-down (t_{bd}, accuracy), possibly bounded from above by a certain limit $t_{\max}$.

Here, e_{us} is mainly (but not always) assumed to be characterized by the resulting enclosure width [5]. These criteria allowed us to produce quite accurate recommendations using the algorithm from Section 4.3, see [5]. Besides, the following further criteria can make sense [30]:

C1 Number of arithmetic operations at a time step

[10] `vericomp.fiw.hs-wismar.de`

C2 Number of function/ Jacobian, etc./ inverse matrix evaluations
C3 Overhead (the overall user CPU time minus the user CPU time for function evaluations [30])
C7 Total number of steps and the number of accepted steps

Besides C4 and C6, the most widely used quality characteristics concerning performance, accuracy, and efficiency, resp., in publications from Table 1 are

C8, C9 User CPU time and the width of the enclosure at t_{out}
C10 CPU time to achieve a certain prescribed enclosure width over the time span $[0, t_{\text{out}}]$

As the eleventh criterion **C11**, usability based on empirical (online) studies should be introduced. The *testing aims* in the considered papers are to characterize a given solver w.r.t. the state of the art. However, users also like to find a good IVPS for their given application (cf. experience reports at ARCH). This goodness can be characterized by various scenarios, for example, an offline simulation with very high accuracy (e.g., for particle colliders) or fast online verified simulation over short time spans (e.g., for control).

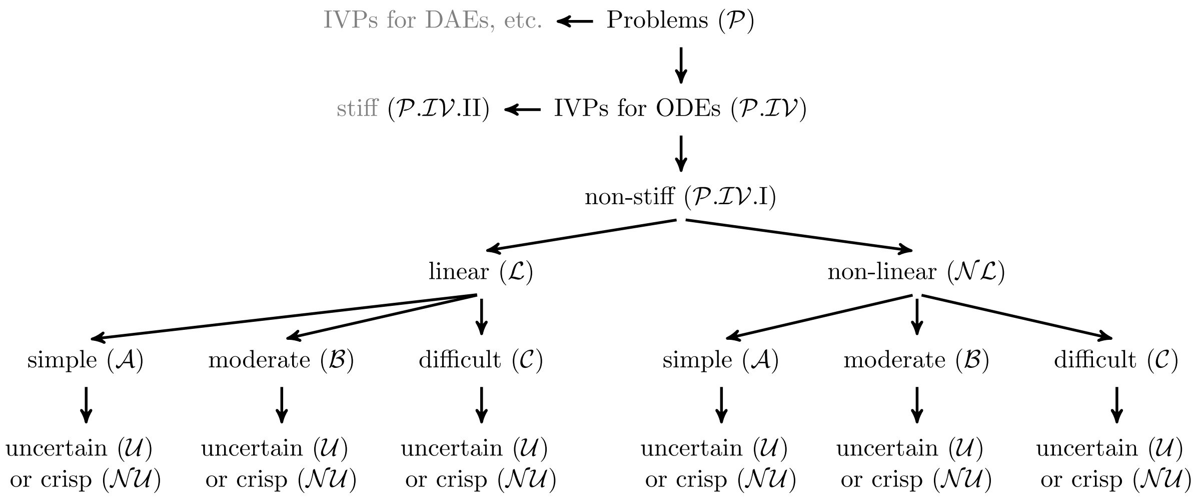

IVPs for DAEs, etc.
Problems ($\mathcal{P}$)
stiff ($\mathcal{P}.\mathcal{IV}$.II)
IVPs for ODEs ($\mathcal{P}.\mathcal{IV}$)
non-stiff ($\mathcal{P}.\mathcal{IV}$.I)
linear ($\mathcal{L}$)
non-linear ($\mathcal{NL}$)
simple ($\mathcal{A}$)
moderate ($\mathcal{B}$)
difficult ($\mathcal{C}$)
simple ($\mathcal{A}$)
moderate ($\mathcal{B}$)
difficult ($\mathcal{C}$)
uncertain ($\mathcal{U}$) or crisp ($\mathcal{NU}$)
uncertain ($\mathcal{U}$) or crisp ($\mathcal{NU}$)
uncertain ($\mathcal{U}$) or crisp ($\mathcal{NU}$)
uncertain ($\mathcal{U}$) or crisp ($\mathcal{NU}$)
uncertain ($\mathcal{U}$) or crisp ($\mathcal{NU}$)
uncertain ($\mathcal{U}$) or crisp ($\mathcal{NU}$)

Figure 2: Classification of benchmark problems for verified IVPS [6].

The most common *benchmarks* used for solver characterization in the considered papers are the Lorenz system, the Rossler system, (Lotka-)Volterra equations, the oil reservoir problem, the harmonic oscillator as well as problems from the DETEST benchmarks. Several papers use old VERICOMP version benchmarks, one difficulty being that the problem IDs changed in the new version. Besides, the benchmark set consisting of over 73 problems needs structuring. For example, automatically extractable, distinct problem benchmark sets for asserting common ground (e.g., the five problems mentioned above), for advanced testing, or for practice-oriented testing can be devised. For more details about the form of the mentioned test problems, see VERICOMP under `vericomp.fiw.hs-wismar.de`. For the benchmarks, we suggest using the classification in Figure 2. The main classes of linear and non-linear problems have three subclasses: simple (possibly with exact expressions for solutions), moderate (w.r.t. their dimension, order, etc.) and complex or difficult problems. In each of these subclasses, we differentiate between problems with uncertain and crisp parameters. The use of this classification was justified in [5]. Further problem classes such as ODEs with delays or non-smooth right sides as well as hybrid systems or systems of differential-algebraic equations can be incorporated into it.

However, the classification needs to remain flexible to be able to reflect factors which interest users at a given moment, for example, chaoticity or cooperativity.

The considered papers mostly use tables and trajectory plots to present comparison results. Only [18] uses other visual aids (spider diagrams) for that purpose. Occasionally, just one set of IVPS parameters (cf. Table 1, Column 5), which additionally varies from example to example sometimes, is employed and the strategy behind the choice of settings is not always clear. A comparison using these settings consistently would be more interesting from the point of view of finding out and recommending such settings automatically.

4.3 A recommender for IVPS for ODEs in VERICOMP

We can think of the following tasks a recommender for IVPS might be required to solve:

1. Recommend a solver (a ranked list of solvers) for a new user problem under consideration of their specific tasks (online/offline simulation, etc.)
2. Find a coverage set L for a problem set B

In this subsection, we describe a formal basis for such a IVPS recommender. We plan to make it accessible in the near future.

Following [36], we define a SRS for IVPS as a 6-tuple $< U, T, L, K, P, S >$, where U represents the user, T is the entity set (of items) , $L \subseteq T$ is the set of recommended items, $K = K(P, T, S)$ is the context, P stands for the user profile and S for the situation. To produce a recommendation, we maximize a certain utility function χ depending on the user, the context, and the set of recommended items.

To solve the recommendation tasks mentioned above, we identify U with the problem a user wants to solve. Therefore, P coincides with the problem characteristics defined by the classification in Figure 2. T contains solvers characterized by their specific settings and S is described by the type of application the users have in mind for their problems (e.g., online/offline simulation). Note that the context K is independent of T, because the number of solvers does not change during a session. The utility function can be a weighted sum of normalized values for each criterion **C1**,. . .,**C11**:

$$\chi(v,u) = \sum_{i=1}^{m} \omega_i n(C_i(v,u)), \quad v \in T, \quad u \in U, \quad \sum_{i=1}^{m} \omega_i = 1, \quad m = 11, \tag{3}$$

where ω_i are the weights, $n(\cdot)$ a normalizing function, and $C_i(v,u)$ a function returning the value of the criterion i for solver v and problem u, for example, as shown in Eqs. (5). Note that we assume that v is not merely one of the solvers, but rather a solver with certain settings (e.g., ValEncIA with the stepsizes of 0.025, 0.0025 is represented by two separate items in T). To produce a recommendation, we use the multiattribute utility collaborative filtering with the given criteria and weighting according to the situation $s \in S$ [43]. The first step in the process of filtering is to establish similarity to a (group of) problem(s) u from the database U with the help of a measure $\mu(u)$. In our case, we can define a simple $\mu(u) := \mu(l(u), c(u), f(u))$ as depending on the linearity $l : U \mapsto \{\mathcal{L}, \mathcal{NL}\}$, complexity $c : U \mapsto \{\mathcal{A}, \mathcal{B}, \mathcal{C}\}$, and the presence of uncertainty $f : U \mapsto \{\mathcal{U}, \mathcal{NU}\}$. It returns all problems from the class uniquely defined by the values $l(u)$, $c(u)$, $f(u)$:

$$\begin{aligned} \mu \quad &: \quad U \mapsto 2^U \\ \mu(u) \quad &:= \quad \{\tilde{u} \in U | l(\tilde{u}) = l(u), c(\tilde{u}) = c(u), f(\tilde{u}) = f(u)\} \ . \end{aligned}$$

The set $\mu(u)$ constitutes the user profile P. The next step is weighting: according to the situation s, a vector with weights $(\omega_1, \ldots, \omega_m)$ is determined by the function ν as

$$\begin{aligned} \nu \quad &: \quad S \mapsto [0,1]^m \\ \nu(s) \quad &:= \quad (\omega_1, \ldots, \omega_m) \ . \end{aligned}$$

The third step is finding the appropriate neighborhood for the problem u. In our case, this neighborhood coincides with P. In the final step of the recommending process, we rate the available solvers $v \in T$ with the help of the normalizing function

$$n : \mathbf{R}^+ \to [0,1], \quad n_{k_1,k_2}(x) = \frac{1}{1 + e^{1-(x-k_1)/k_2}}, \quad k_1, k_2 \geq 0 \ , \tag{4}$$

with k_1 and k_2 being real heuristic parameters depending on the data, and the function χ from Eq. (3). We use $k_1 = 40$ and $k_2 = 10$ in the following.

To see how the recommendation work, consider the following example. Let an engineer be interested in simulating a non-linear, simple problem with uncertainty in a verified way. Suppose the similarity measure μ returns a set consisting of two problems u_1 ($\dot{x} = -\frac{1}{2}x^3$, $x(0) = [0.5, 1.5]$) and u_2 ($\dot{x}_1 = 1$, $\dot{x}_2 = x_2 \cos(x_0)$, $x_1(0) = 0$, $x_2 = [0.9, 1.25]$) belonging to the class $\mathcal{P}.\mathcal{IV}.\mathcal{I}.\mathcal{NL}.\mathcal{A}.\mathcal{U}$. Suppose further that the data concerning the quality criteria **C4**, **C5**, and **C6** shown in Table 2 for three solvers RiOT, Valencia, and VNODE with three different settings are recorded for these problems in the database. Finally, suppose that the engineer's goal is to simulate the problem online over short time intervals, which defines the situation s. If we restrict ourselves to the three criteria for which the data are recorded, the wall clock time (**C4**) and the relation of user CPU time to overestimation (**C5**) are equally important, whereas the time to break-down and the width of the enclosure there (**C6**) do not play much of a role. Accordingly, the assigned weights for $i = 4, 5, 6$ are 0.4, 0.4, 0.2, resp., and zero otherwise. The neighborhood is the set $\{u_1, u_2\}$.

Now we are ready to rate the solvers. A pre-normalization requirement is that bigger criterion values should correspond to better performance. Therefore, they are computed as follows for a solver v and a problem u:

$$\begin{aligned} C_4(v,u) &= 1/t_{\text{c}} \\ C_5(v,u) &= 1/(e_{\text{us}} \cdot t_{\text{us}}), \\ C_6(v,u) &= t_{\text{bd}}/e_{\text{bd}} \ . \end{aligned} \tag{5}$$

The ratings obtained using these definitions in formula (3) are shown in the last column of Table 3 (rounded). A higher rating ($0 \leq \chi(v,u) \leq 1$) indicates better performance. Note that the problem was not actually simulated to make the recommendation. From Table 3, it is clear that the criterion values are always the highest for VNODE with the 15th order of Taylor expansion in the situation s, a recommendation in good accordance with the data in Table 2. The averaging order plays a role: The criterion values can be computed as given in Eqs. (5) for each problem and averaged. This value can be then used in the formula for the rating χ. Alternatively, ratings $\chi(v_i, u_i)$ can be computed for each benchmark u_i and averaged afterwards, which usually provides a better separation between the ratings (therefore, our formula for χ is given such that this dependence is recorded directly). The main difficulty is to find a good normalizing function for the broad range of criterion values (5). Users can validate the recommendation by running the standard test on the problem u. Note that recommendations depend greatly on the information in the data base. Besides, they are produced for a *problem class* and not for a *particular problem*, making a flexible classification procedure a must. Our further work includes improving this recommender principle through the use of better metrics (e.g., (1),(6)), averaging (e.g., (2)) and normalizations.

4.4 Improvements

Recommendations depend significantly on problem and solver features. The problem features in VERICOMP are the right side of the IVP, initial conditions, the integration time interval, parameters, the exact solution (if available), the assignment to a problem class and a textual description concerning, for example, the origin of the problem. It is obvious from Section 4.3 that the set of these features and the classification must be as flexible as possible. A topic for our future work is to investigate if a finer theoretical classification (or set of features) similar to that for partial-differential equations from [27] is also useful (and not too inefficient) in our context. In [27], not as much attention is paid to solver-oriented features as to problem-centric ones. However, we also aim to recommend specific solver settings, which makes a careful study of IVPS characteristics necessary. At the moment, solvers are characterized mainly by their names, parameters (cf. Column 5 from Table 1) with their default values, and a textual description of their methods.

Our long-term (and somewhat ideal) goal is to provide a common environment for testing and recommending IVPS under the same conditions. This goal necessitates development of a semi-automatic procedure for adding a new solver to the solver database, which is a complex task for verified solvers since they lack common interfaces and are very different in their underlying concepts. A more manageable task would be to provide (template of) a solver database to be filled by an IVPS developer, which would leave common test conditions out the consideration. A useful feature in this case would be to convert the available benchmark sets (common ground, application or practice-oriented, see Section 4.2) into the syntax supported by a given solver or, at least, easy to use with it. Similar converters exist, for example, for hybrid systems (cf. HyST[11]). However, they seem to develop a new converter for each new solver. A more interesting approach would be to automatize this process along with the database generation itself, for example, through the use of XML specifications.

If we have a large (and extendable) set of solvers and want to adapt the problem classification in an easy way if necessary, we can consider descriptions in form of keywords. An approach retrieving information from relational and unstructured data might be useful as a filtering and structuring pre-step for the SRS routines described in Section 4.3. Following [51, 52], we assume that each request r for a solver v is defined by a keyword vector $r = (r_1, \dots r_J)^{\mathrm{T}}$ and a corresponding numeric vector $u = (u_1, \dots, u_J)^{\mathrm{T}}$, where $u_j = 1$ for each j-th keyword constituting the request r, $j = 1 \dots J$. Each tool v has a profile represented by a feature vector c_r built from keywords s_i and their weights ω_i^k, $i = 1, \dots, I$, $k = 1, \dots, K$. The weights influence the rating of the tool v w.r.t. each descriptor s_i and incorporate various factors (benchmarks, actuality, fitting to the problem dimensions). For example, they can reflect the degree of belief in s_i according to its provenance. The weights are normalized and aggregated over $k = 1 \dots K$ to an average weight $\omega_i(v)$ only if s_i is equal to a keyword from the request, $s_i = r_j$, resulting in J such weights. Those are used in the cosine similarity measure

[11] `www.verivital.com/hyst/`

$$d_v := \frac{\sum_{j=1}^{J} u_j \cdot \omega_j}{\sqrt{\sum_{j=1}^{J} \omega_j^2} \cdot \sqrt{\sum_{j=1}^{J} u_j^2}}, \tag{6}$$

that describes the similarity between the request vector r and the feature vector c_r characterizing the qualification of the tool v to solve the task. The final ranking is a descending list sorted according to d_v.

Consider a small illustration for this algorithm in our context. Suppose we are interested in the subset of seven items (solvers with their settings) from Table 2[12]. Assume further that we want to find a good solver for linear IVPs applicable in real time. That is, r consists of two keywords r_1 ="linear", r_2 ="online" and $u = (1\ 1)^{\mathrm{T}}$. Let the database on the seven items contain the information shown in Table 4. Here, the weights $\omega_i^1 \in [0, 1]$ reflect how relevant or good the solver is w.r.t. the meaning of a given keyword, whereas the weights $\omega_i^2 \in [0, 1]$ show the degree of confidence in this assessment according to its provenance. For example, ω_i^2 can be set to one if tests were performed using a common environment (e.g., VERICOMP); to some other number less than one if the assessment is supplied by the developer of the tool; and still a smaller number if it is based on information extracted (automatically) from publications or similar. Accordingly, the weights ω_i^2 for RiOT with the order 11 are less than one since the tests for this setting were not carried out in VERICOMP. Suppose there is no information about how well Valencia 0.025 performs with linear systems. The last column of Table 4 shows the ratings obtained using formula (6). For example, the average weight for the keyword "linear" for RiOT 5 is $\frac{0.6+1}{2} = 0.8$ (we can omit normalization here), the weight vector consists of two components $(0.8\ 0.6)^{\mathrm{T}}$ and $d_{\text{RiOT}\,5} = \dfrac{1.4}{\sqrt{0.8^2 + 0.6^2} \cdot \sqrt{2}} = 0.9899$ (truncated). Since a keyword is missing from the description of Valencia 0.025, no rating is generated. According to Table 4, the suitable solvers are Valencia with the stepsize 0.025 and VNODE with orders of Taylor expansion 15 and 20. This smaller set of solvers can be chosen for actual test runs, from the results of which recommender algorithms both from this section and Section 4.3 can profit. Better results can be achieved with more sophisticated normalizing and weighting strategies. Keywords and actual values of weights for them can be assigned on the basis of the previously computed quality criteria or such linguistic descriptions as 'good' or 'bad' extracted from papers. The results can be fed back to run tests and update the database.

5 CONCLUSIONS

In this paper, we presented the state of the art in the field of scientific recommender systems with a focus on visual analytics and problem-solving environments. We identified concepts, components and approaches necessary to recommend a scientific tool. Finally, we described in detail a possibility to recommend a solver for initial value problems depending on a user's problem, partially supported by a testing environment VERICOMP accessible online. In particular, we discussed various quality criteria, metrics, problem classifications and problem/solver features based on an extensive literature study. The recommender algorithm was illustrated using a small example.

Our future work concerns implementing the discussed options in the common test environment VERICOMP. The possibility of manual filling of VERICOMP's database with simulation results for a new solver by a registered user is under construction in its new version. Moreover, we plan to make the recommender available in the near future. Our middle-term goals

[12] Obviously, this pre-step makes more sense if there are many more solvers in the database

are to make the database construction more flexible by allowing it to be generated from XML-like descriptions and to provide templatized converters of benchmark sets to respective solvers' syntaxes.

REFERENCES

[1] *ARCH17. 4th International Workshop on Applied Verification of Continuous and Hybrid Systems*, 2017.

[2] Ch. C. Aggarwal. *Recommender Systems: The Textbook.* Springer Publishing Company, 1st edition, 2016.

[3] M. Althoff. An introduction to CORA 2015. In *Proc. of the Workshop on Applied Verification for Continuous and Hybrid Systems*, 2015.

[4] M. Angelova, V. Devagiri, V. Boeva, P. Linde, and N. Lavesson. An expertise recommender system based on data from an institutional repository (DiVA). In *Proc. ELPUB*, Toronto, Canada, 2018.

[5] E. Auer. *Result Verification and Uncertainty Management in Engineering Applications.* Verlag Dr. Hut, 2014. Habilitation Monograph.

[6] E. Auer and A. Rauh. VERICOMP: a system to compare and assess verified IVP solvers. *Computing*, 94(2):163–172, Mar 2012.

[7] M. Behrisch, M. Blumenschein, N. W. Kim, L. Shao, M. El-Assady, J. Fuchs, D. Seebacher, A. Diehl, U. Brandes, H. Pfister, T. Schreck, D. Weiskopf, and D. A. Keim. Quality metrics for information visualization. *Computer Graphics Forum*, 37(3):625–662, 2018.

[8] B. W. Boehm. Some future software engineering opportunities and challenges. In *The Future of Software Engineering*, pages 1–32. Springer, Berlin, Heidelberg, 2011.

[9] S. Bönisch, M. Brickenstein, H. Chrapary, G.-M. Greuel, and W. Sperber. swMATH - A new information service for mathematical software. In *Intelligent Computer Mathematics - MKM, Calculemus, DML, and Systems and Projects 2013, Part of CICM 2013, Bath, UK, July 8-12, 2013. Proceedings*, pages 369–373, 2013.

[10] O. Bouissou and M. Martel. A Runge-Kutta method for computing guaranteed solutions of ODEs. In *12th GAMM - IMACS International Symposium on Scientific Computing, Computer Arithmetic, and Validated Numerics, SCAN'06, Duisburg, Germany*, 2006.

[11] S. A. C. Bukhari, M. Martínez Romero, M. J. O'Connor, A. L. Egyedi, D. Willrett, J. Graybeal, M. A. Musen, K.-H. Cheung, and S. H. Kleinstein. CEDAR OnDemand: A browser extension to generate ontology-based scientific metadata. *BMC Bioinformatics*, 19(1):268:1–268:6, 2018.

[12] F. Bünger. Shrink wrapping for Taylor models revisited. *Numerical Algorithms*, 78(4):1001–1017, 2018.

[13] M. Capinski, J. Cyranka, Z. Galias, T. Kapela, M. Mrozek, and P. Zgliczynski. Computer assisted proofs in dynamics group. `capd.ii.uj.edu.pl`.

[14] L. Chen, G. Chen, and F. Wang. Recommender systems based on user reviews: The state of the art. *User Modeling and User-Adapted Interaction*, 25(2):99–154, June 2015.

[15] Xin Chen, Erika Ábrahám, and Sriram Sankaranarayanan. Flow*: An analyzer for non-linear hybrid systems. In Natasha Sharygina and Helmut Veith, editors, *Computer Aided Verification*, pages 258–263, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[16] H. Chrapary and W. Dalitz. Software products, software versions, archiving of software, and swMATH. In *Mathematical Software - ICMS 2018 - 6th International Conference, South Bend, IN, USA, July 24-27, 2018, Proceedings*, pages 123–127, 2018.

[17] M. Crubézy and M. A. Musen. Ontologies in support of problem solving. In Steffen Staab and Rudi Studer, editors, *Handbook on Ontologies*, pages 321–341. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[18] J. A. dit Sandretto and A. Chapoutot. Validated explicit and implicit Runge-Kutta methods. *Reliable Computing electronic edition*, 22, July 2016.

[19] T. Dzetkulič. Rigorous integration of non-linear ordinary differential equations in Chebyshev basis. *Numerical Algorithms*, 69(1):183–205, May 2015.

[20] I. Eble. *Über Taylor-Modelle*. PhD thesis, Universität Karlsruhe, 2007.

[21] St. van den Elzen and J. J. van Wijk. Small multiples, large singles: A new approach for visual data exploration. *Computer Graphics Forum*, 2013.

[22] A. Felfernig, G. Ninaus, H. Grabner, F. Reinfrank, L. Weninger, D. Pagano, and W. Maalej. An overview of recommender systems in requirements engineering. In *Managing Requirements Knowledge.*, pages 315–332. Springer, Berlin, Heidelberg, 2013.

[23] A. Geyer-Schulz, M.l Hahsler, and M. Jahn. Educational and scientific recommender systems: Designing the information channels of the virtual university. *International Journal of Engineering Education*, pages 153–163, 2001.

[24] E. Gómez-Martínez, M. Linaje, F. Sánchez-Figueroa, A. Iglesias-Pérez, J. C. Preciado, R. González-Cabero, and J. Merseguer. A semantic approach for designing assistive software recommender systems. *Journal of Systems and Software*, 104(C):166–178, June 2015.

[25] A. I. Guseva, V. S. Kireev, P. V. Bochkarev, I. A. Kuznetsov, and S. A. Philippov. Scientific and educational recommender systems. *AIP Conference Proceedings*, 1797(1):020002, 2017.

[26] M. Hernández-Rubio, I. Cantador, and A. Bellogín. A comparative analysis of recommender systems based on item aspect opinions extracted from user reviews. *User Modeling and User-Adapted Interaction*, Nov 2018.

[27] E. N. Houstis, A. C. Catlin, N. Dhanjani, J. R. Rice, N. Ramakrishnan, and V. S. Verykios. MyPYTHIA: A recommendation portal for scientific software and services. *Concurrency and Computation: Practice and Experience*, 14(13-15):1481–1505, 2002.

[28] E. N. Houstis, A. C. Catlin, J. R. Rice, V. S. Verykios, N. Ramakrishnan, and C. E. Houstis. PYTHIA-II: A knowledge/database system for managing performance data and recommending scientific software. *ACM Trans. Math. Softw.*, 26(2):227–253, June 2000.

[29] T. E. Hull, W. H. Enright, B. M. Fellen, and A. E. Sedgwick. Comparing Numerical Methods for Ordinary Differential Equations. *SIAM Journal on Numerical Analysis*, 9(4):603–637, 1972.

[30] T. E. Hull, W. H. Enright, B. M. Fellen, and A. E. Sedgwick. Comparing numerical methods for ordinary differential equations. *SIAM Journal on Numerical Analysis*, 9(4):603–637, 1972.

[31] F. Immler. *A Verified ODE Solver and Smale's 14th Problem*. PhD thesis, TU Müchen, 2018.

[32] D. Jannach, M. Zanker, M. Ge, and M. Gröning. Recommender systems in computer science and information systems – a landscape of research. In Christian Huemer and Pasquale Lops, editors, *E-Commerce and Web Technologies*, pages 76–87, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[33] M. Kashiwagi. Verified numerical computation and `kv` library. `http://verifiedby.me/kv/index-e.html`.

[34] P. Kaur and M. Owonibi. A review on visualization recommendation strategies. In *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017)*, pages 266–273, 2017.

[35] A. Key, B. Howe, D. Perry, and C. Aragon. VizDeck: Self-organizing dashboards for visual analytics. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 681–684, New York, NY, USA, 2012. ACM.

[36] A. Klahold. *Empfehlungssysteme: Grundlagen, Konzepte und Lösungen*. Vieweg Teubner, 2009. In German.

[37] D. E. Knuth. Literate Programming. *The Computer Jour.*, 27(2):97–111, 1984.

[38] B. Latard, J. Weber, G. Forestier, and M. Hassenforder. Towards a semantic search engine for scientific articles. In *Research and Advanced Technology for Digital Libraries - 21st International Conference on Theory and Practice of Digital Libraries, TPDL 2017, Thessaloniki, Greece, September 18-21, 2017, Proceedings*, pages 608–611, 2017.

[39] F. Lorenzi and F. Ricci. Case-based recommender systems: A unifying view. In Bamshad Mobasher and Sarabjot Singh Anand, editors, *Intelligent Techniques for Web Personalization*, pages 89–113. Springer Berlin Heidelberg, 2005.

[40] J. Mackinlay, P. Hanrahan, and Ch. Stolte. Show Me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1137–1144, November 2007.

[41] A. Mahboubi, G. Melquiond, and Th. Sibut-Pinote. Formally verified approximations of definite integrals. *J. Autom. Reasoning*, 62(2):281–300, 2019.

[42] K. Makino and M. Berz. Suppression of the wrapping effect by Taylor model-based validated integrators. MSUHEP 40910, Department of Physics, Michigan State University, East Lansing, MI 48824, 2004.

[43] N. Manouselis and C. Costopoulou. *Personalization Techniques and Recommender Systems*, chapter Experimental Analysis of Multiattribute Utility Collaborative Filtering on a Syntetic Data Set, pages 111–133. World Scientific Publishing Company, 2008.

[44] R. E. Moore, R. B. Kearfott, and M. J. Cloud. *Introduction to Interval Analysis*. Society for Industrial and Applied Mathematics, Philadelphia, 2009.

[45] B. Mutlu, E. Veas, and Ch. Trattner. VizRec: Recommending personalized visualizations. *ACM Trans. Interact. Intell. Syst.*, 6(4):31:1–31:39, November 2016.

[46] K. Nazemi, R. Retz, J. Bernard, J. Kohlhammer, and D. W. Fellner. Adaptive semantic visualization for bibliographic entries. In *Advances in Visual Computing - 9th International Symposium, ISVC 2013, Rethymnon, Crete, Greece, July 29-31, 2013. Proceedings, Part II*, pages 13–24, 2013.

[47] N.S. Nedialkov. *Implementing a Rigorous ODE Solver Through Literate Programming*, volume 3 of *Mathematical Engineering*, pages 3–19. Springer, Heidelberg, 2011.

[48] A. W. Neumann. *Recommender Systems for Information Providers – Designing Customer Centric Paths to Information*. Springer, 2009.

[49] Ch. Obeid, I. Lahoud, H. El Khoury, and P.-A. Champin. Ontology-based recommender system in higher education. In *Companion Proceedings of the The Web Conference 2018*, WWW '18, pages 1031–1034, Republic and Canton of Geneva, Switzerland, 2018. International World Wide Web Conferences Steering Committee.

[50] D. H. Park, I. Y. Choi, H. K. Kim, and J. K. Kim. A review and classification of recommender systems research. In *International Conference on Social Science and Humanity IPEDR vol. 5*, pages 290–294, Singapore, 2011. IACSIT Press.

[51] J. Protasiewicz. Support system for selection of reviewers. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, pages 3062–3065, San Diego, CA, USA, 2014.

[52] J. Protasiewicz, W. Pedrycz, M. Kozlowski, S. Dadas, T. Stanislawek, A. Kopacz, and M. Galźźewska. A recommender system of reviewers and experts in reviewing problems. *Know.-Based Syst.*, 106(C):164–178, August 2016.

[53] A. S. Raamkumar, S. Foo, and N. Pang. Can I have more of these please? assisting researchers in finding similar research papers from a seed basket of papers. *The Electronic Library*, 36(3):568–587, 2018.

[54] A. Rauh and E. Auer. *Modeling, Design, and Simulation of Systems with Uncertainties*, volume 3 of *Mathematical Engineering*. Springer, Heidelberg, 2011.

[55] A. Rauh, E. Auer, J. Minisini, and E. P. Hofer. Extensions of VALENCIA-IVP for Reduction of Overestimation, for Simulation of Differential Algebraic Systems, and for Dynamical Optimization. In *Proceedings in Applied Mathematics and Mechanics*, pages 1023001–1023002, 2007.

[56] M. P. Robillard, W. Maalej, R. J. Walker, and Th. Zimmermann. *Recommendation Systems in Software Engineering*. Springer Publishing Company, 2014.

[57] J. Scholtz, C. Plaisant, M. A. Whiting, and G. G. Grinstein. Evaluation of visual analytics environments: The road to the visual analytics science and technology challenge evaluation methodology. *Information Visualization*, 13(4):326–335, 2014.

[58] S. Siebert, S. Dinesh, and S. Feyer. Extending a research-paper recommendation system with bibliometric measures. In *Proceedings of the Fifth Workshop on Bibliometric-enhanced Information Retrieval (BIR) co-located with the 39th European Conference on Information Retrieval (ECIR 2017), Aberdeen, UK, April 9th, 2017.*, pages 112–121, 2017.

[59] L. Steinert. *Beyond Similarity and Accuracy - A New Take on Automating Scientific Paper Recommendations*. PhD thesis, University of Duisburg-Essen, Germany, 2017.

[60] Ch. Stolte, D. Tang, and P. Hanrahan. Polaris: A system for query, analysis, and visualization of multidimensional databases. *Commun. ACM*, 51(11):75–84, November 2008.

[61] M. Thurau, Chr. Buck, and W. Luther. IPFViewer: Incremental, approximate analysis of steel samples. In *Proceedings of SIGRAD 2014, Visual Computing, June 12-13, 2014, Göteborg, Sweden*, pages 1–8, 2014.

[62] S. W. Tu, H. Eriksson, J. H. Gennari, Y. Shahar, and M. A. Musen. Ontology-based configuration of problem-solving methods and generation of knowledge-acquisition tools: application of PROTEGE-II to protocol-based decision support. *Artificial Intelligence in Medicine*, 7(3):257–289, 1995.

[63] W. Tucker. The Lorenz attractor exists. *C. R. Acad. Sci. Paris Sér. I Math.*, 328(12):1197–1202, 1999.

[64] W. Tucker. A rigorous ODE solver and Smale's 14th problem. *Found. Comput. Math.*, 2(1):53–117, 2002.

[65] M. Voigt, M. Franke, and K. Meissner. Using expert and empirical knowledge for context-aware recommendation of visualization components. *IARIA Int. Jour. on Advances in Life Sciences*, 5(1):27–41, 2013.

[66] B. Weyers, E. Auer, and Luther W. The role of verification and validation techniques within visual analytics. *JUCS: Special Issues on Collaborative Technologies and Data Science in Smart City Applications*, 2019. submitted.

[67] D. C. Wilson, D. B. Leake, and R. Bramley. Case-based recommender components for scientific problem-solving environment. In *Proceedings of the Sixteenth IMACS World Congress*, 2000.

[68] K. Wongsuphasawat, D. Moritz, A. Anand, J. D. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Trans. Vis. Comput. Graph.*, 22(1):649–658, 2016.

[69] K. Wongsuphasawat, Z. Qu, D. Moritz, R. Chang, F. Ouk, A. Anand, J. D. Mackinlay, B. Howe, and J. Heer. Voyager 2: Augmenting visual analysis with partial view specifications. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver, CO, USA, May 06-11, 2017.*, pages 2648–2659, 2017.

[70] S. Zhao, D. Zhang, Z. Duan, J. Chen, Y.-P. Zhang, and J. Tang. A novel classification method for paper-reviewer recommendation. *Scientometrics*, 115(3):1293–1313, June 2018.

IVPS	Lang.	VC	Main method	Main parameters
Valencia [55]	C++	RV	Picard iteration, exponential extension	stepsize
VNODE [47]	C++	RV LP	Taylor series (TS), Hermite-Obreschkoff method (HO)	method/ order/ tolerances/ min.stepsize/ stepsize control
CAPD [13]	C++	RV	TS, explicit-implicit HO	method/order/tolerances
COSI-VI [42]	FORTRAN	RV	Taylor models	TM order/ stepsize/ tolerances/ preconditioning/ shrink wrapping
RiOT [20]	C++	RV	Taylor models (TM)	TM order/ bounding method/ stepsize control/ tolerances/ sparsity
verifyode [12]	INTLAB	RV	Taylor models	TM order/ bounder/stepsize control/ tolerances/shrink wrap./ sparsity
Flow* [15]	C++	RV	Taylor models	TM order/stepsize control/ tolerances/no.steps with symbolic remainders/ remainder estimation bound
kv [33]	C++	RV	Power series and affine arithmetics	method/ma precision
[19]	C++	RV	TM, Chebyshev function enclosures	method/ order/ sparsity/ tolerances
DYNIbex [18]	C++	RV	affine arithmetic, Runge-Kutta	RK variant/ order/ stepsize/ tolerances
GRK [10]	OCaml	RV	Runge-Kutta, multiprecision arithmetics	stepsize control/ tolerances
CORA [3]	MATLAB	RV	Reachability analysis with zonotopes/ polytopes	(polynomial) zonotope order/ tolerances
Isabelle [31]	PolyML	FV RV	Affine arithmetic/ zonotopes, Runge-Kutta	max. zonotope order/ stepsize control/ tolerances/ ma precision
[41]	Coq	FV RV	antiderivatives of rigorous polynomial approximations, adaptive domain splitting	ma precision/ target error bound

Table 1: Verified IVPS. ‘VC’ means verification concept, with possibilities ‘RV’ (result verification), ‘LP’ (literal programming), ‘FV’ (formal verification). The abbreviation ‘ma’ means ‘machine arithmetic’, usually floating point.

Solver	u_1					u_2				
	t_{c}	t_{us}	e_{us}	t_{bd}	e_{bd}	t_{c}	t_{u}	e_{u}	t_{bd}	e_{bd}
RiOT 5	3.270s	3.197s	0.448	10	0.130	3.597s	3.466s	0.811	10	0.20
RiOT 10	13.030s	12.763s	0.443	10	0.057	0.860s	0.842s	0.811	10	0.20
RiOT 15	40.883s	40.607s	0.443	10	0.055	0.918s	0.886s	0.811	10	0.20
V 0.025	0.045s	0.042s	2.987	1.300	5.85	0.260s	0.257s	0.850	10	309.55
V 0.0025	0.287s	0.282s	2.905	1.17	3.69	1.528s	1.521s	0.815	10	249.32
V 0.00025	2.794s	2.780s	2.897	1.19	3.77	1m30.844s	1m30.726s	0.812	10	243.87
VNODE 15	0.014s	0.009s	0.887	6.36	151.77	0.047s	0.041s	0.811	10	0.203
VNODE 20	0.014s	0.007s	0.987	3.81	218.18	0.047s	0.042s	0.811	10	0.203
VNODE 25	0.015s	0.009s	1.138	2.59	270.42	0.046s	0.039s	0.811	10	0.203

Table 2: Test run data on the problems u_1, u_2 from the database of the old version of VERICOMP. V stands for Valencia. **C4** is defined by t_{c}, **C5** by t_{us} and e_{us}, and **C6** by t_{bd} and e_{bd}. Besides, $t_{\text{out}} = 1\text{s}$ and $t_{\max} = 10\text{s}$.

v_i	$C_4(v_i, u_i)$ (u_1/u_2)	$C_5(v_i, u_i)$ (u_1/u_2)	$C_6(v_i, u_i)$ (u_1/u_2)	$\frac{\chi(v_i,u_1)+\chi(v_i,u_2)}{2}$
RiOT 5	0.007/0.007	0.007/0.007	0.936/0.500	0.149
RiOT 10	0.007/0.008	0.007/0.008	0.999/0.500	0.155
RiOT 15	0.007/0.007	0.007/0.008	0.999/0.500	0.155
V 0.025	0.058/0.009	0.014/0.010	0.007/0.007	0.020
V 0.0025	0.009/0.007	0.008/0.007	0.007/0.007	0.007
V 0.00025	0.007/0.007	0.007/0.007	0.007/0.007	0.006
VNODE 15	0.89/0.05	0.99/0.11	0.01/0.48	0.462
VNODE 20	0.89/0.05	0.99/0.11	0.01/0.48	0.461
VNODE 25	0.84/0.05	0.99/0.13	0.01/0.48	0.453

Table 3: Recommendation process with three criteria and nine solvers, based on the problems u_1 and u_2, using the normalizing function $n_{40,10}$ and rating averaging.

Solver	Keyword	ω_1^1	ω_1^2	Keyword	ω_2^1	ω_2^2	Keyword	ω_3^1	ω_3^2	Rating
RiOT 5	linear	0.6	1	nonlinear	0.8	1	online	0.2	1	98.99
RiOT 10	linear	0.5	1	nonlinear	0.9	1	online	0.1	1	98.83
RiOT 11	linear	0.5	0.6	nonlinear	0.95	0.6	online	0.1	0.6	97.61
V 0.025				nonlinear	0.2	1	online	0.6	1	–
V 0.0025	linear	0.6	1	nonlinear	0.2	1	online	0.56	1	99.99
VNODE 15	linear	0.9	1	nonlinear	0.7	1	online	0.7	1	99.84
VNODE 20	linear	0.95	1	nonlinear	0.8	1	online	0.6	1	99.51

Table 4: Solver descriptions in terms of feature vectors c_r. Ratings are multiplied by the factor 100 for better presentation.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

UNCERTAINTY ANALYSIS OF A CAR CRASH SCENARIO USING A POSSIBILISTIC MULTI-FIDELITY SCHEME

Markus Mäck and Michael Hanss

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
e-mail: {markus.maeck,michael.hanss}@itm.uni-stuttgart.de

Keywords: Possibility Theory, Multi-Fidelity Modeling, Fuzzy Arithmetic, Crash Simulation

Abstract. *This contribution deals with an efficient model-based uncertainty-propagation scheme in possibility theory using models of different fidelity. The aim is to enable a possibilistic description of polymorphic uncertainty and its propagation through large-scale, complex and computationally expensive real-world applications. The possibilistic uncertainty description not only provides bounds for the response statistics of the model, but can also model and process incomplete knowledge and ignorance. For the proposed multi-fidelity scheme, the functional dependency between the costly, but accurate high-fidelity model and the much cheaper, but less accurate low-fidelity model are exploited in such a way that only a few expensive high-fidelity evaluations are needed to correct the potentially poor low-fidelity approximation. Finally, the approach is applied to an automotive car crash scenario, highlighting its potentials regarding uncertainty quantification in real-world applications.*

1 Introduction

The numerical quantification and propagation of uncertainty described through possibility theory [1] entails tremendous computational costs if the model to be investigated is of high complexity and large scale. Typically, in mechanical and civil engineering a model is described by partial differential equations (PDEs) and needs to be evaluated several times if uncertainty in the model parameters is to be considered additionally. This repeated evaluation of the deterministic model renders uncertainty analysis nearly infeasible in the case of expensive, large-scale applications unless appropriate steps are taken to reduce the computation time.

In the context of uncertainty quantification using probability theory, multi-fidelity schemes have emerged over the past decades [2], yielding impressive results [3]. Only recently, as a possibilistic counterpart, a novel strategy for model-based propagation of possibilistic uncertainty has been introduced, using models of different fidelity [4]. In this context, the low-fidelity models can be derived directly from the high-fidelity model by, for example, applying simplification of the geometry or idealization of the physical properties. The possibilistically described uncertain quantities of interest of the low-fidelity model, which might be rather poor estimations, are then corrected by exploiting the dependency between the high- and low-fidelity model in a possibilistic way. This results in a highly flexible and efficient strategy for propagating possibilistic uncertainty through large-scale systems, as illustrated by the automotive crash example presented in the sequel.

2 Possibilistic Uncertainty Description

A possibility measure $\Pi : 2^{\Omega} \to [0, 1]$ is a mapping from the universe of discourse onto the unit interval, which can be expressed by a set of nested confidence intervals and can be interpreted as an upper probability bound [5]. Thus, possibility theory can account for probability distributions with ill-known properties in the context of imprecise probability. A possibility measure fulfills $\Pi(\emptyset) = 0$, $\Pi(\Omega) = 1$ and $\Pi(A \cup B) = \max(\Pi(A), \Pi(B))$ in an axiomatic way. Moreover, a possibility distribution Π_{ξ}, associated with an uncertain variable (or vector) $\xi : \Omega \to \mathbb{R}$ is a set function whose range is in $[0, 1]$ and can be expressed by its possibility density function π_{ξ} via

$$\Pi_{\xi}(U) = \Pi(\{\omega : \omega \in \Omega, \xi(\omega) \in U\}) = \sup_{x \in U} \pi_{\xi}(x) \quad \forall U \subseteq \mathbb{R}, \tag{1}$$

reflecting the possibility of ξ taking a value in U. In contrast to probability theory and as a result of the maxitivity property of the possibility measure, a second measure, the necessity N, is introduced in order to fully characterize the uncertainty of $\xi \in U$. It can be derived from the possibility measure via $\mathrm{N}_{\xi}(U) = 1 - \Pi_{\xi}(\Omega \backslash U)$.

3 Efficient Numerical Propagation of Possibilistic Uncertainty

The numerical propagation of possibilistic uncertainty is realized by the extension principle introduced by Zadeh [6]. Let ξ be an uncertain input variable with its corresponding possibility density function π_{ξ}. The possibility density function of the uncertain output variable ζ, obtained by propagating ξ through a model $h : \mathcal{X} \subseteq \mathbb{R}^n \to \mathcal{Z} \subseteq \mathbb{R}$, which maps the input $x \in \mathcal{X}$ onto an output $z \in \mathcal{Z}$, is given by

$$\pi_{\zeta}(z) = \sup \pi_{\zeta}(h^{-1}(\{z\})) = \sup_{x \in h^{-1}(\{z\})} \pi_{\xi}(x) \quad \forall z \in \mathcal{Z} \tag{2}$$

under the condition that the supremum of the empty set is zero. The model h is called *high-fidelity* model if it provides the output $z_{\text{hi}} \in \mathcal{Z}$ with the necessary accuracy for the task at hand. The optimization problem involved in Eq. 2 requires a repeated evaluation of the high-fidelity model which results in tremendous computational effort if it is expensive to evaluate.

3.1 Surrogate Modeling

In possibilistic uncertainty quantification, the expensive high-fidelity model is usually replaced by a surrogate model which, for example, can be constructed by an interpolation or regression scheme yielding $h(x) \approx \sum_i \nu_i \phi_i(x) =: s(x)$, with ϕ_i denoting the basis functions and ν_i the respective coefficients of the chosen scheme. The surrogate model does not have any physical meaning since it merely reflects the input/output behavior of a high-fidelity black-box model. By the use of the surrogate $s(x)$, the possibility density function of the uncertain quantity of interest ζ can be approximated as

$$\pi_\zeta(z_{\text{hi}}) \approx \sup_{x \in s^{-1}(z_{\text{hi}})} \pi_\xi(x) \quad \forall z_{\text{hi}} \in \mathcal{Z}. \tag{3}$$

If a surrogate model is available, the actual uncertainty analysis is rather inexpensive because it is only based on interpolation. However, the construction of an appropriate surrogate, i.e. the computation of ν_i, can still be computationally costly, especially if the underlying high-fidelity model is non-linear, discontinuous, or exhibits a strongly oscillating behavior.

3.2 Multi-Fidelity Modeling

For the possibilistic multi-fidelity approach, *low-fidelity* versions of the high-fidelity model, which are significantly cheaper to evaluate but also yield less accurate results, can be achieved according to the strategies presented in [7]. In the following, the low-fidelity model is defined by the mapping $g : \mathcal{X} \subseteq \mathbb{R}^n \to \mathcal{Z} \subseteq \mathbb{R}$, which maps an input $x \in \mathcal{X}$ onto an output $z_{\text{lo}} \in \mathcal{Z}$, and its possibility density function reads

$$\pi_\eta(z_{\text{lo}}) = \sup_{x \in g^{-1}(\{z_{\text{lo}}\})} \pi_\xi(x) \quad \forall z_{\text{lo}} \in \mathcal{Z}, \tag{4}$$

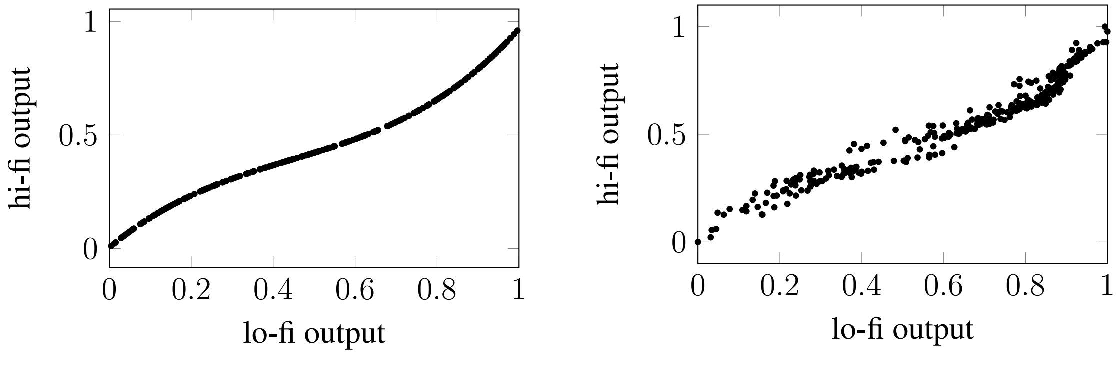
with η as the corresponding possibilistic low-fidelity output variable. If there exists a strong functional dependency between high- and low-fidelity model, as exemplified in Figure 1a, then the high-fidelity output can be written in the form of

$$z_{\text{hi}} = f(z_{\text{lo}}) = f(g(x)), \tag{5}$$

where the function f captures the potentially unknown functional dependency between the high- and the low-fidelity models. Inserting Eq. (5) in Eq. (2) yields

$$\pi_\zeta(z_{\text{hi}}) = \sup_{x \in h^{-1}(z_{\text{hi}})} \pi_\xi(x) = \sup_{x \in g^{-1}(f^{-1}(z_{\text{hi}}))} \pi_\xi(x), \tag{6}$$

which means that for a given f the entire uncertainty analysis can be carried out entirely on the low-fidelity model. The functional dependency can be identified by applying regression or interpolation schemes that use only few high-fidelity model evaluations, resulting in $f(z_{\text{lo}}) \approx \mathcal{I}(f(z_{\text{lo},i}))$. In practice, however, it is difficult to derive low-fidelity models which exhibit a strong functional dependency with the associated high-fidelity model.



(a) Strong functional dependency

(b) Weak functional dependency

Figure 1: Different degrees of dependency between high- and low-fidelity model outputs.

In general, the pointwise evaluation of the two models using the same values of the input parameters x_i describes the dependency as $z_{\text{hi},i} = f(z_{\text{lo},i}) + \delta_i$ where δ_i denotes some non-random perturbation, as exemplarily shown in Figure 1b. Consequently, the dependency can be captured by the conditional possibility density function $\pi_{\zeta\,|\,\eta}$. According to [8], the marginal density function of the high-fidelity solution π_ζ can then be computed by the conditional density function and the marginal density function of the low-fidelity model π_η via

$$\pi_\zeta(z_{\text{hi}}) = \max_{z_{\text{lo}}} \left(\min\left(\pi_{\zeta\,|\,\eta}(z_{\text{hi}}\,|\,z_{\text{lo}}), \pi_\eta(z_{\text{lo}})\right)\right). \tag{7}$$

As in the strongly dependent case, the conditional density function, which represents the functional dependency in a possibilistic way, is to be learned using a number of high- and low-fidelity model evaluations. For example, the linear combination of Gaussian basis functions

$$f(x, z_{\text{lo}}) \approx \sum_{j=1}^{m} a_j(x)\mathrm{e}^{-b_j(x)(c_j(x)-z_{\text{lo}})^2} \tag{8}$$

can be used as a non-parametric approach, where the coefficients $\{a_j, b_j, c_j\}$ are additionally considered as possibilistic variables and need to be determined. Consequently, the conditional possibility distribution can be estimated by

$$\pi_{\zeta\,|\,\eta}(z_{\text{hi}}\,|\,z_{\text{lo}}) = \sup_{x:z_{\text{hi}}=f(x,z_{\text{lo}})} \pi_\xi(x) \tag{9}$$

using the approximation in Eq. (8). The coefficients of the basis function can be obtained by solving an optimization problem to determine the most specific solution of the conditional density function which, at the same time, coincides with the constraints derived in [4]. For this purpose, the density functions of the uncertain variables can be parameterized using a number of independent shape parameters (three in case of a triangular distribution) as design parameters for the optimization.

Equation (6) is perfectly consistent with the formulation in Eq. (7). Hence, the description of the functional dependency in a possibilistic manner is a generalization that comprises the deterministic formulation as a special case, as shown in the following. Let ξ be an uncertain input variable, and η and ζ be the uncertain output variables of the low- and high-fidelity model, respectively. Furthermore, let there exist a strong functional dependency between the high- and low-fidelity output quantities, i.e. $z_{\text{hi}} = f(x, z_{\text{lo}}) = f(z_{\text{lo}})$. Then, it holds

$$\begin{aligned}
\pi_\zeta(z_{\text{hi}}) &= \max_{z_{\text{lo}}} \left(\min\left(\pi_{\zeta\,|\,\eta}(z_{\text{hi}}\,|\,z_{\text{lo}}), \pi_\eta(z_{\text{lo}})\right)\right) \\
&= \max_{z_{\text{lo}}} \left(\min\left(\sup_{x:z_{\text{hi}}=f(x,z_{\text{lo}})} \pi_\xi(x), \sup_{x\in g^{-1}(z_{\text{lo}})} \pi_\xi(x)\right)\right) \\
&= \max_{z_{\text{lo}}} \left(\min\left(\sup_{x:z_{\text{hi}}=f(z_{\text{lo}})} \pi_\xi(x), \sup_{x\in g^{-1}(z_{\text{lo}})} \pi_\xi(x)\right)\right) \\
&= \max_{z_{\text{lo}}} \begin{cases} \sup\limits_{x\in g^{-1}(z_{\text{lo}})} \pi_\xi(x) & \forall z_{\text{lo}} \in f^{-1}(z_{\text{hi}}), \\ 0 & \text{else.} \end{cases} \\
&= \sup_{x\in g^{-1}(f^{-1}(z_{\text{hi}}))} \pi_\xi(x).
\end{aligned} \tag{10}$$

In summary, rather than approximating the deterministic input-output behavior $h(x) \approx \mathcal{F}(h(x_i))$, which then can be used within the extension principle, the multi-fidelity scheme directly approximates the possibilistic quantities of the model response $\pi_\zeta(z_{\text{hi}}) \approx \mathcal{F}(\pi_\eta(z_{\text{lo},i}))$ depending on the low-fidelity solution. However, this approach fails if the corresponding high- and low-fidelity models are functionally independent.

4 Applications

The following chapter illustrates the proposed multi-fidelity approach by two applications. While the first one deals with an academic example and assumes a strong functional dependency between the high- and the low-fidelity model, the second one deals with a real-world example, namely the possibilistic investigation of an automotive crash scenario, and illustrates the application of the multi-fidelity approach in the case of a weak functional dependency.

4.1 Example 1: Academic Example (Strong Dependency)

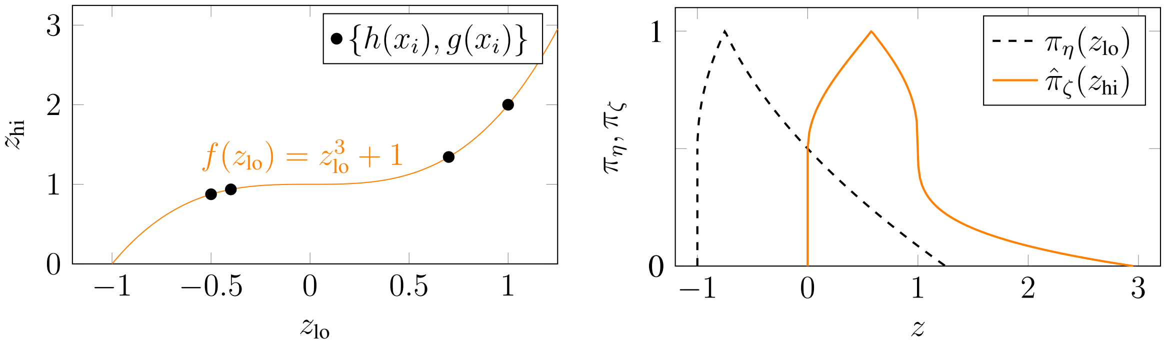
Let g be a mapping $g : \mathbb{R} \to \mathbb{R}$ which describes an arbitrary low-fidelity model whose output is given by $z_{\text{lo}} = g(x) = (x - \frac{1}{2})^2 - 1$, and let $h : \mathbb{R} \to \mathbb{R}$ be an unknown and expensive to evaluate high-fidelity model. The possibility density function of some uncertain input variable ξ shall be given by

$$\pi_\xi(x) = \begin{cases} 1 - |x| & \forall x \in [-1, 1], \\ 0 & \text{else.} \end{cases} \tag{11}$$

Consequently, the possibility density function of the uncertain output variable η associated with the low-fidelity model can be computed by

$$\pi_\eta(z_{\text{lo}}) = \sup_{x\in g^{-1}(z_{\text{lo}})} \pi(x) = \begin{cases} 1 - |\frac{1}{2} - \sqrt{z_{\text{lo}} + 1}| & \forall z_{\text{lo}} \in [-1, \frac{5}{4}], \\ 0 & \text{else} \end{cases} \tag{12}$$

when applying the extension principle in Eq. (2), see Figure 2b. The low-fidelity solution is considered as a poor approximation of the output quantity of the more sophisticated, but



(a) Functional dependency

(b) Low- and high-fidelity solution

Figure 2: Functional dependency of the high- and low-fidelity model and respective output possibility density functions.

expensive high-fidelity model h. In the following, a strong cubic dependency between the two models shall be assumed. Evaluating the low-fidelity as well as the high-fidelity model using the same (arbitrary) input parameters x_i, $i = 1, \dots, 4$, enables the fitting of a cubic polynomial resulting in $z_{\text{hi}} = f(z_{\text{lo}}) = z_{\text{lo}}^3 + 1$, see Figure 2a. The estimated solution for the uncertain variable ζ associated with the high-fidelity model can then obtained using the multi-fidelity approach in Eq. (6) as

$$\hat{\pi}_\zeta(z_{\text{hi}}) = \sup_{x \in g^{-1}(f^{-1}(z_{\text{hi}}))} \pi_\xi(x) = \begin{cases} 1 - \left|\frac{1}{2} - \sqrt{(z_{\text{hi}} - 1)^{\frac{1}{3}} + 1}\right| & \forall z_{\text{hi}} \in [0, \frac{189}{64}], \\ 0 & \text{else} \end{cases} \tag{13}$$

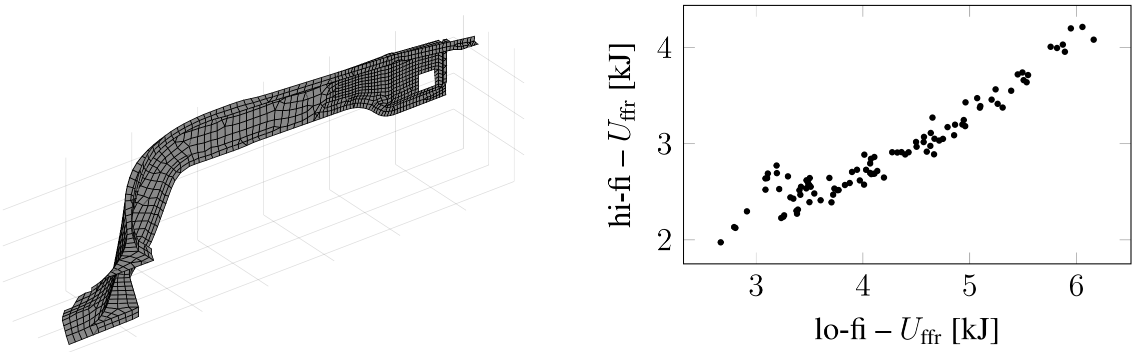
which is shown in Figure 2b and which is in this case identical to the high-fidelity solution π_ζ.

4.2 Example 2: Automotive Crash Example (Weak Dependency)

In the following, an automotive, non-overlapping, frontal crash scenario is investigated in order to emphasize the potential of the presented possibilistic multi-fidelity approach. The crashworthiness of vehicles, i.e. their passive safety, is the deciding factor in protecting the driver and the passengers during and after the frontal impact. Accordingly, the passive safety systems have to be designed for the different types of collision that can occur. In case of a frontal crash, the crumple-zone structure of the car plays a vital role in passenger safety. Thanks to its plastic deformation, the kinetic energy of the car can be absorbed and the acceleration acting on the passengers can be reduced. This lifesaving function requires the design of the crumple zone to be accomplished with reasonable care. Against this background, several car safety programs have been established all around the world over the past decades, defining the current safety standards for automotive crash scenarios. The countries of the European Union established the Euro New Car Assessment Program (Euro NCAP), which defines a set of standardized car crash scenarios to be passed by new cars. The Euro NCAP, however, only defines a small number of scenarios which do not coincide with the numerous imaginable scenarios in reality.

In the following, the energy-absorbing capabilities of the crumple-zone structure are numerically investigated with respect to only partial knowledge about potential passengers, loading and initial velocity. For this reason, the mass of the co-driver, the mass of loading in the trunk as well as the velocity are modeled in terms of uncertain variables with a triangular shape. The

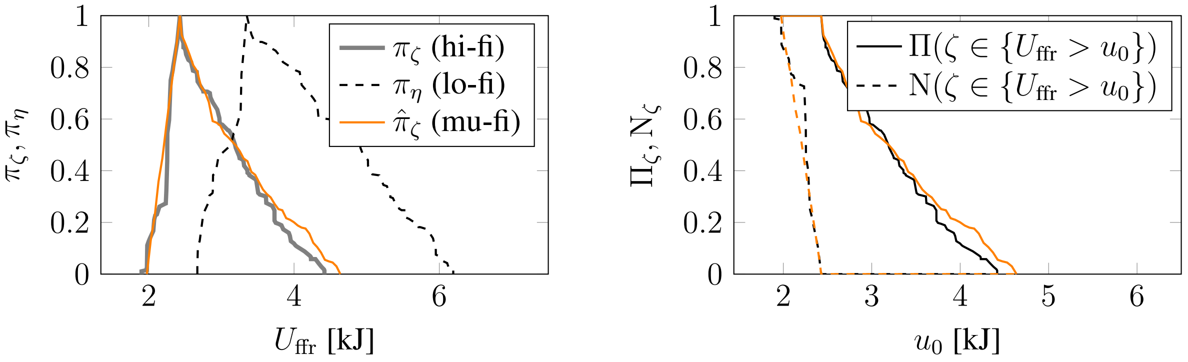
masses of the co-driver and the loading together exhibit a worst-case interval of [0, 300] kg as well as a nominal value of 100 kg. The velocity varies within [30, 50] km/h with a nominal value of 36 km/h. The finite element model used for the investigation has been developed by The National Crash Analysis Center (NCAC) of The George Washington University under a contract with the FHWA and NHTSA of the US DOT. However, it has been completely remeshed and additionally modified in order to increase its numerical robustness and to achieve better results. The finite element simulation was executed in parallel on several local computing platforms using the commercial finite element code LS-DYNA. The model, which in the following will be referred as the high-fidelity model, consists of 47,232 deformable elements, and a single deterministic model evaluation on a computing platform with Intel Xeon CPU E5-2667 processors takes around 9,141 s. A low-fidelity version of this model is derived by, first, using extensive mass-scaling to increase the critical time step for a stable simulation and, second, by converting deformable parts at the backside of the car into rigid bodies, reducing the number of deformable elements to 27,287. Consequently, the solution time of a single deterministic model evaluation of the low-fidelity model decreases to 1,514 s, which is about six times cheaper than the respective high-fidelity evaluation. Note that the crumple-zone structure remains unchanged.
As the possibilistic quantity of interest, the maximum energy absorption U_{ffr} of one of the front frame rails is chosen, which is an essential part of the crumple-zone structure and shown in Figure 3a. The possibilistic density functions of the absorbed energy of the low-fidelity model, i.e. π_η, and of the high-fidelity model, i.e. π_ζ, which acts as the reference solution, are both calculated using a sampling approach according to [4] with 5,000 model evaluations each. The functional dependency between the low- and the high-fidelity model, identified on the basis of 100 model evaluations, is shown in Figure 3b. There exists only a weak functional dependency between the two models; thus, the approach in Eq. (7) is used, in which the conditional density function $\pi_{\zeta\,|\,\eta}$ needs to be estimated using Eq. (8). The obtained solutions using the low-fidelity model, the high-fidelity model, as well as the multi-fidelity approach are summarized in Figure 4a. Apparently, the low-fidelity solution is only a poor approximation and in the deterministic case only of limited use. However, by applying the presented multi-fidelity scheme and exploiting the dependencies between the low- and high-fidelity models, the obtained solution is in excellent agreement with the high-fidelity solution, using overall only few high-fidelity evaluations.



(a) Front frame rail (ffr) of the car model

(b) Functional dependency of the low- and high-fidelity solution

Figure 3: Investigated part of the car and functional dependency between the high- and the low-fidelity model.



(a) Possibility density functions of the high-fidelity (hi-fi), low-fidelity (lo-fi) and multi-fidelity (mu-fi) solution

(b) Possibility and necessity of the multi-fidelity solution (- - -/——) the high-fidelity solution (- - -/——)

Figure 4: Results of the proposed multi-fidelity approach for the automotive crash example.

Often, one is interested in whether an uncertain output surpasses or falls below a specific value. Especially in reliability analysis, this is a frequently asked question. In this example, like in the context of robust design, one is interested in the possibility and the necessity of the absorbed energy U_{ffr} surpassing a given threshold u_0 which is, for example, specified by the development engineer. The corresponding possibility distribution can be directly derived from Eq. (1) for a given possibility density function and is shown for the high-fidelity and multi-fidelity solution in Figure 4b. According to [9], the possibility/necessity pair defines a family of probabilities and can be interpreted as lower and upper bounds of the probability of the event.

5 Conclusion and Outlook

In this work, an efficient possibilistic multi-fidelity framework is presented. It enables the propagation of possibilistically modeled uncertainty through large-scale models. For the presented approach, several a-priori assumptions have been made, representing starting points for further investigations. First, the number of points selected to establish the functional dependency was chosen somewhat arbitrarily. It would be useful to determine the most appropriate amount of high-fidelity evaluations required for this task because it strongly affects the computation time needed for the multi-fidelity approach. Second, the approximation of the conditional density function is carried out using only Gaussian basis functions. Potentially there exist more appropriate strategies for this estimation. Moreover, for the low-fidelity version, an even more simplified model can be used. In the authors' opinion, low-fidelity models which are up to 100 times cheaper than the high-fidelity models should be achievable while still being able to exhibit a sufficient functional dependency for successfully applying the multi-fidelity approach.

6 Acknowledgment

The authors gratefully acknowledge the support of this research by the German Research Foundation DFG in the framework of the priority programme 1886 *Polymorphic Uncertainty Modelling for the Numerical Design of Structures*. Additionally, the authors would like to thank Dipl.-Math. Dennis Grunert for providing the scripts for the inclusion of rigid bodies in LS-DYNA.

REFERENCES

[1] Didier Dubois and Henri Prade. *Possibility Theory – An Approach to Computerized Processing of Uncertainty*. Plenum Press, 1988.

[2] Phaedon-Stelios Koutsourelakis. Accurate uncertainty quantification using inaccurate computational models. *SIAM Journal on Scientific Computing*, 31(5):3274–3300, 2009.

[3] Jonas Biehler, Michael W. Gee, and Wolfgang A. Wall. Towards efficient uncertainty quantification in complex and large-scale biomechanical problems based on a bayesian multi-fidelity scheme. *Biomechanics and modeling in mechanobiology*, 14(3):489–513, 2015.

[4] Markus Mäck and Michael Hanss. A multi-fidelity approach for possibilistic uncertainty analysis. In *Proc. of the 7th International Conference on Uncertainties in Structural Dynamics – USD2018*. KU Leuven, Belgium, 2018.

[5] Didier Dubois and Henri Prade. When upper probabilities are possibility measures. *Fuzzy Sets and Systems*, 49(1):65–74, 1992.

[6] Lotfi A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3):199–249, 1975.

[7] Jonas Biehler, Markus Mäck, Jonas Nitzler, Michael Hanss, Phaedon-Stelios Koutsourelakis, and Wolfgang A. Wall. Multi-fidelity approaches for uncertainty quantification. *Surveys for Applied Mathematics and Mechanics (GAMM-Mitteilungen)*, 2019. (accepted).

[8] Ellen Hisdal. Conditional possibilities independence and noninteraction. *Fuzzy Sets and Systems*, 1(4):283–297, 1978.

[9] Didier Dubois. Possibility theory and statistical reasoning. *Computational Statistics & Data Analysis*, 51(1):47–69, 2006.

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

STOCHASTIC NON-PARAMETRIC IDENTIFICATION IN COMPOSITE STRUCTURES USING EXPERIMENTAL MODAL DATA

S. Chandra, K. Sepahvand, C.A. Geweth, F. Saati, and S. Marburg

Chair of Vibroacoustics of Vehicles and Machines,
Department of Mechanical Engineering, Technical University of Munich,
85748 Garching b. Munich, Germany.
e-mail: sourav.chandra@tum.de.

Keywords: Stochastic inverse identification, Elastic parameter, Generalized Polynomial Chaos, statistical moment, Composite Structure.

Abstract. *In the stochastic structural analysis of composite structure, the probabilistic knowledge of the uncertain parameters are essential. Variability of the manufacturing process of the composite structure introduces the uncertainty to the elastic parameters. It is easier to identify the uncertainty of the material parameters using stochastic inverse process. An efficient stochastic inverse identification of the elastic parameters of laminated composite plate using generalized Polynomial Chaos (gPC) theory is presented in this paper. A data set of measured eigen frequencies and mass density are used for stochastic inversion processes. Stochastic identification of the elastic parameters of composite plate transforms into estimation of deterministic coefficients of gPC expansion for the elastic parameters. A robust optimization technique by minimization of the quadratic difference between statistical moments is used to estimate the deterministic coefficients of the gPC expansion. These coefficients can effectively construct the distributions of the uncertain elastic parameters. Evaluation of the deterministic coefficients by higher order statistical moments minimization, can efficiently simulate the randomness of the experimental eigen frequencies.*

1 INTRODUCTION

Knowledge of material parameters of a structural system are essential before inserting into the forward model to asses the structural responses of the dynamic system. The values of the material parameters of composite structure in terms of elastic moduli, shear moduli, Poisson's ratio and mass density are often described by the manufacturer. Prior to inserting these material parameters into realistic forward simulation, effect of uncertainty should be accounted. A real dynamic system consists of two fundamental uncertainties such as parameter uncertainty and model uncertainty. Parameter uncertainty for the composite structure propagates to the system due to inherent randomness of the elastic moduli, Poisson's ratio and mass density. The random fiber orientations, variation of the thickness and variation of the fabrication procedure introduce the parameter uncertainty to the dynamic system of the composite structure. Whereas, boundary conditions in the mathematical model, using homogenized theory, to evaluate the effective elastic moduli of the composite material are responsible for modeling uncertainty of the system. Direct measurement of the uncertainty of the elastic parameters are not able to represents these variabilities efficiently. The elastic parameters can be evaluated by inverse identification based on the concept of error minimization of the experimental responses and simulated responses. The deterministic inverse problems [1, 2, 3] are involved in identifying the elastic parameters of the composite plate from a single experimental modal data, based on various optimization algorithms. Single measurement is not sufficient to capture the uncertainty and variability associated with the parameters and the model. The reliable prediction of overall dynamic behavior of the composite structure is possible by incorporating the uncertainty of the material parameters within the finite element (FE) framework.

Uncertainty of elastic parameters can be evaluated by establishing stochastic relation between uncertain elastic parameters and set of measured responses. A well suited FE based stochastic inverse method is employed to identify the variability of the elastic parameters of the composite structure. Rikards et al. [4] presented various methods to identify the elastic properties of laminated material using experimental data. Lauwagie [5] discussed various optimization techniques adopted for material properties identification of laminated composite materials in inverse process. Stochastic inverse technique involved to identify the probabilistic parameters such as mean and variance, of the elastic constants of laminated composite structure based on probabilistic representation of modal responses. Bayesian inverse updating method offers a wide range of flexibility to multi-parameter model identification from a sufficient number of experimental data sets. Basically, posterior distribution of the material parameters are inferred from assumed prior by evaluating the likelihood of the elastic parameters. In recent years, several applications [6, 7, 8] of Bayesian inference technique in inverse problem have appeared. The evaluation of the integral is the most challenging part in multi-parameter Bayesian inverse inference. However, Markove Chain Monte Carlo (MCMC) became an efficient alternative to determine the posterior density without evaluating the integral. Various sampling based approaches such as, Metropolis-Hasting (M-H) algorithm and Gibbs sampler [9, 10] have developed for the improvement of MCMC algorithm. However, sampling based inverse identification often suffer due to computational efficiency. Nagel [11] discussed Bayesian inverse problems with a direction to overcome the limitations of sampling based technique for determining the posterior probability density functions of the system parameters. In recent years, spectral stochastic formulation has been proposed in combination with the Bayesian inference [12]. Rosić et al. [13] proposed a linear Bayesian estimation of the unknown parameters in combination with the Karhunen-Loève and Polynomial Chaos expansions without using any sampling technique such as, MCMC. This method can effectively update non-Gaussian uncertainties. The introduction of Galerkin projection technique using generalized Polynomial Chaos (gPC) theory [14, 15] transfer the inverse problem as a deterministic one which involves to identify the unknown gPC coefficients instead of probabilistic parameter of the quantity. Sepahvand and Marburg [16, 17] have efficiently estimated the elastic parameters of the orthotropic material via stochastic inverse method using non-sampling based gPC expansion technique. Non-Gaussian experimental modal data are used to identify elastic parameters. Literature review described the application of the stochastic non-parametric identification for the laminated composite structure and efficiency of the inverse algorithm to predict the wide range of uncertainty in the case of laminated composite structure.

This paper aims to identify the uncertainty of the elastic moduli and shear modulus for laminated composite plate using experimental modal frequencies. The method involves to evaluate the deterministic coefficients of the uncertain elastic parameters through minimization of statistical moments, calculated from measured eigen frequencies and corresponding statistical moments derived from simulated eigen frequencies, using gPC expansion method. Moreover, present paper also reported the efficiency of parameter identification by implementation of higher order statistical moments minimization technique. The in-situ randomness of the mass density of the composite material is determined and is considered as an input to the stochastic inverse model.

2 FE MODEL OF LAMINATED COMPOSITE PLATE

In the present formulation of the forward model of laminated composite plate, the classical thin plate theory [19] is assumed . The assumption neglects the effect of transverse shear deformation. The relation between stress σ' and strain ε' for orthotropic layer with reference to the principal material axes (1, 2, 3) are presented by the generalized Hooke's law as

$$\sigma' = \mathbf{C}\varepsilon'. \tag{1}$$

Here, $\mathbf{C}$ is the stress-strain relationship matrix along the principal material axes of the k^{th} lamina. The elements of the $\mathbf{C}$ matrix for the k^{th} layer is expressed as

$$\begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{13} \end{bmatrix}_k = \begin{bmatrix} C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{22} & 0 & 0 & 0 \\ 0 & 0 & C_{33} & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & C_{55} \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_{12} \\ \varepsilon_{23} \\ \varepsilon_{13} \end{bmatrix}_k , \tag{2}$$

where $C_{11} = E_{11}/(1-\nu_{12}\nu_{21})$, $C_{12} = \nu_{21}E_{11}/(1-\nu_{12}\nu_{21})$, $C_{22} = E_{22}/(1-\nu_{12}\nu_{21})$, $C_{33} = G_{12}$, $C_{44} = G_{23}$ and $C_{55} = G_{13}$. Here, E_{ii}, G_{ij} and ν_{ij} are the set of elastic constants such as Young's moduli, shear moduli and Poisson's ratio of the laminated composite plate, respectively. The stress σ and strain ε relationship is redefined with reference to the laminate axes (x, y, z) of the composite plate as

$$\sigma = \mathbf{Q}\varepsilon, \tag{3}$$

in which

$$\mathbf{Q} = \mathcal{T}^{-1}\mathbf{C}\mathcal{T}, \tag{4}$$

and $\mathcal{T}$ is transformation matrix [20] to relate the principal material axes and laminate axes. The element stiffness matrix of the laminated composite plate takes the form

$$K_e = \int_{A_e} B^T D B \mathrm{d}A_e, \tag{5}$$

in which B is the strain-displacement matrix written as

$$\bar{\varepsilon} = B\delta. \tag{6}$$

Here, $\bar{\varepsilon}$ is the strain and curvature vector and δ is the nodal displacement vector of the composite plate. The mid-plane stress resultant $\bar{\sigma}$ and strain $\bar{\varepsilon}$ of the laminate are related by stiffness matrix D [19] as

$$\bar{\sigma} = D\bar{\varepsilon}, \tag{7}$$

where

$$D = \begin{bmatrix} A_m & A_c & 0 \\ A_c & A_b & 0 \\ 0 & 0 & A_s \end{bmatrix}. \tag{8}$$

In the above matrix, sub-components A_m, A_c, A_b and A_s represent membrane stiffness, membrane-bending coupling stiffness, bending stiffness and shear stiffness, respectively. Here,

$$A_i = \sum_{j=1}^{l} \int_{z_{k-1}}^{z_k} (Q_i)^k (1, z, z^2) dz, \quad i = m, c, b \tag{9}$$

$$A_i = \sum_{j=1}^{l} \int_{z_{k-1}}^{z_k} \kappa (Q_i)^k dz, \quad i = s, \kappa = 5/6 \tag{10}$$

where κ is shear correction factor [21] and l is numbers of orthotropic layers in composite plate. The elemental mass matrix is written as

$$M_e = \int_{A_e} N^T \rho N \mathrm{d}A_e, \tag{11}$$

where N is interpolation matrix and ρ is the inertia matrix. The global stiffness matrix K and the global mass matrix M are developed after assembling the elemental stiffness and mass matrices, K_e and M_e, respectively. Therefore, undamped modal analysis involves the solution of

$$[\lambda_i^2 M + K]\phi_i = 0, \quad i = 1, 2, ..., n \tag{12}$$

to extracts the modal frequency λ_i and mode shape ϕ_i of the laminated composite plate with n numbers of degrees of freedom (DOF) in FE model. Generalized forward model of the composite plate can be defined as

$$d = G(m). \tag{13}$$

Herein, m denotes vector of elastic parameters of the model and d is set of simulated data for ideal case. The forward model operator G predicts the model output data set d in terms of eigen frequencies as a function of model parameters m. In the present paper, model parameters are E_{ii} and G_{ij} and the forward model yield the data output in the form of modal frequency λ_i.

3 POLYNOMIAL CHAOS EXPANSION

Assume a probability space (Ω, $\mathcal{U}$, $\mathcal{P}$) in which Ω is the sample space, $\mathcal{U}$ is the σ-algebra over Ω, and $\mathcal{P}$ is the probability measure on the sample space $\mathcal{U}$. Consider a random parameter $\mathcal{X}(\omega)$ with random outcome $\omega \in \Omega$. Such random parameter can effectively be presented by gPC expansion method by projecting it onto a stochastic space spanned by a random orthogonal polynomials. The random parameter $\mathcal{X}$:$\Omega \rightarrow \mathbb{R}$ with finite variance possess the following representation in compact form [15]

$$\mathcal{X}(\omega) = \sum_{i=0}^{P} a_i \Psi_i(\xi), \tag{14}$$

where a_i are unknown deterministic coefficients and $\Psi_i(\xi)$ are the multivariate orthogonal basis functions given by the product of the corresponding univariate polynomial form. Total number of terms in the n dimensional p^{th} order truncated gPC expansion is ($P + 1$), where

$$P + 1 = \frac{(n+p)!}{n!p!}. \tag{15}$$

One-dimensional orthogonal polynomial can be represented by standard normal random vector $\xi = \{\xi_i\}, i = 1, 2, ...N$ in a particular sample space such, that $\xi_i \in \Omega_i$. The orthogonal relationship of the multidimensional polynomial function $\Psi = \{\Psi_i(\xi)\}$ is written as

$$\mathbb{E}[\Psi_i, \Psi_j] = \mathbb{E}[\Psi_i^2]\delta_{ij} = p_i^2\delta_{ij}, \qquad i, j = 0, 1, 2, ...N, \tag{16}$$

in which δ_{ij} represents Kronecker delta and p_i^2 is the norm of the polynomial. Due to the orthogonal properties of the gPC expansion, the unknown coefficients a_i can be calculated by projecting onto the orthogonal set of polynomial chaos, such that

$$a_i = \frac{\langle \mathcal{X}(\omega)\Psi_i \rangle}{\langle \Psi_i^2 \rangle}, \tag{17}$$

where $\langle \Psi_i^2 \rangle$ denote the inner products in the Hilbert space in the L^2 norm. This orthogonal projection minimizes the error on the space spanned by $\{\Psi\}_{k=0}^{P}$ and evaluate the deterministic coefficient a_i. Once, a_i is known, statistical properties of the uncertain parameters can be evaluated. For instance, the expected value $\mu_{\mathcal{X}}$ and variance $\sigma_{\mathcal{X}}^2$ are evaluated as

$$\mu_{\mathcal{X}} = a_0\,, \qquad\qquad \sigma_{\mathcal{X}}^2 = \sum_{i=1}^{P} a_i^2 p_i^2. \tag{18}$$

Identification of the statistical properties of the uncertain parameters require calculation of the gPC coefficients. Therefore, stochastic inverse method is employed to obtain the orthogonal basis function of uncertain parameters via uncertainty propagation of the measured structural responses. The technique of inverse stochastic identification of uncertain parameters from measured modal data is discussed in the next section.

4 STOCHASTIC INVERSE MODEL

For identifying probabilistic properties of the elastic parameters of the dynamic system, statistical informations of the modal responses are known a priori. Assume that, probabilistic measure of the system parameters m are represented by gPC expansion. Therefore, calculation of statistical properties of the elastic parameters m is transferred into evaluation the finite set of unknown gPC coefficients $\mathfrak{m}_i$ using statistical properties of the measured modal data d. An optimization procedure is adopted to evaluate the unknown gPC coefficients $\mathfrak{m}_i$ as a design variable. Experimental modal frequencies are represented in the form of gPC expansion as

$$d(\xi) = \sum_{i=0}^{N} \mathfrak{d}_i \Psi_i(\xi). \tag{19}$$

The deterministic coefficient $\mathfrak{d}_i$ of the gPC expansion is estimated by minimization of statistical moments calculated from measured frequencies and gPC expansion. The stochastic inverse problem can be defined with reference to the Eq. (13) as

$$\sum_{i=0}^{N} \mathfrak{m}_i \Psi_i(\xi) = G^{-1}\Bigg(\sum_{i=0}^{N} \mathfrak{d}_i \Psi_i(\xi)\Bigg). \tag{20}$$

Here, $G^{-1}(\cdot)$ is the inverse structural operator in terms of FE model. Moreover, direct evaluation of the inverse operator is impossible and leads the inverse problem to an optimization problem with $\mathfrak{m}_i$ as a design variable. The optimization function $\mathscr{F}$ is defined as the sum of the quadratic difference between the central moments calculated from the simulated stochastic modal data and measured modal data as

$$\mathscr{F} = \frac{1}{2}\sum_{j=1}^{neig}\Bigg[(\mu_{\mathfrak{D}_j} - \mu_j^{exp})^2 + \sum_{r=1}^{k}\Bigg\{E[\mathfrak{D}_j - \mu_{\mathfrak{D}_j}]^r - \gamma_{j_r}\Bigg\}^2\Bigg]. \tag{21}$$

In this equation, $\mu_{\mathfrak{D}_j}$ is the expected value of the simulated j^{th} modal frequency, $E[\cdot]^r$ is the r^{th} order central moment of the simulated modal frequency and γ_{j_r} is the r^{th} order central moment of measured j^{th} modal frequency. The number of eigen modes is denoted by $neig$.The expected value of the j^{th} modal frequency is described by μ_j^{exp}. The stochastic representation of the simulated j^{th} eigen frequency with reference to the gPC expansion of uncertain parameters and forward structural operator is presented as

$$\mathfrak{D}_j = G\Bigg(\sum_{i=0}^{N} \mathfrak{m}_i \Psi_i(\xi)\Bigg)_j. \tag{22}$$

The Eq. (21) is rewritten with reference to the Eq. (22) as

$$\mathscr{F} = \frac{1}{2}\sum_{j=1}^{neig}\Bigg[(\mu_{\mathfrak{D}_j} - \mu_j^{exp})^2 + \sum_{r=1}^{k}\Bigg\{E\Bigg[G\Bigg(\sum_{i=0}^{N} \mathfrak{m}_i \Psi_i(\xi)\Bigg)_j - \mu_{\mathfrak{D}_j}\Bigg]^r - \gamma_{j_r}\Bigg\}^2\Bigg]. \tag{23}$$

The optimization algorithm determines the best solution for the gPC coefficients $\mathfrak{m}_i$ of the system parameters by the functional minimization of cost function $\mathscr{F}$.

5 NUMERICAL PROCEDURE

The proposed solution procedure for inverse identification of uncertain elastic parameters from experimental modal frequencies involves estimation of deterministic coefficients of gPC expansions for the parameters using a stochastic inverse model. A FE model is developed to evaluate the structural responses of the composite plate and considered as a forward structural operator. The detailed procedure of numerical simulation is summarized herein.

- Measure the eigen frequencies for each sample of composite plate.
- Measure weight of each sample of composite plate and derive mass density of composite material.
- Evaluate the deterministic coefficients for the measured eigen frequencies and mass density based on minimization of the error function between statistical moments of the measured data and the same is derived from the gPC expansion of the quantity.
- Construct the probability distribution functions (PDFs) of the eigen frequencies and mass density based on gPC expansion method and compare with the measured distributions.
- Construct the truncated gPC expansion for the identifiable parameters with the initial approximation of deterministic coefficients.
- Estimate the gPC coefficients of the eigen frequencies employing the stochastic FE forward model by using initial values for the unknown coefficients of the parameters.
- Evaluate the error function between the r^{th} order central moments calculated from gPC expansion coefficients of the eigen frequencies and corresponding central moments calculated from the experimental data.
- A constrained optimization procedure is adapted to update the initially approximated unknown coefficients of the identifiable parameters by minimization of the cost function.
- Construct the PDFs of elastic parameters using the updated coefficients of the gPC expansion.

6 NUMERICAL RESULTS

A set of modal frequencies of 100 numbers, 12 layers glass-fiber epoxy composite plate with identical dimension of $250 \times 125 \times 2$ are measured in free-free boundary condition. Each plate is suspended using two thin elastic wires to approximate the free-free boundary condition. The composite plate is excited by impulse hammer and responses are collected at 35 points of each plate by an accelerometer. A post-processing software is employed to derive the modal responses i.e., eigen frequencies, mode shapes and modal damping ratios. The weight of each plate is measured precisely to evaluate the uncertainty of the mass density. First 4 modes of the eigen frequencies are considered for identification of the elasticity moduli E_{11} and E_{22} and shear modulus G_{12} of the composite plate. To avoid modal coupling and corresponding error only first 4 eigen frequencies are considered for the identification process. The initial 6 rigid modes are neglected in the analysis. The uncertainty of the mass density is also considered in the identification process. The PDFs of the measured eigen frequencies and corresponding stochastic representations are shown in Figure 1. Third order gPC expansion, employing one dimensional Hermite polynomial H_i, is used to construct the experimental eigen frequencies [18] as

$$d(\xi) = \sum_{i=0}^{3} \mathfrak{d}_i H_i(\xi). \tag{24}$$

The gPC coefficients $\mathfrak{d}_j$ for first 4 eigen frequencies are estimated by minimization of the statistical moments derived from experimental data and gPC expansion via an optimization procedure and is presented in Table 1. The reconstructed PDFs as shown in Figure 1 using gPC expansion are fitted well with the experimental distributions. Third order gPC expansion is well suited to represent the nature of variability of the experiential eigen frequencies. Nine set of collocation points (0, ±0.742, ±2.334, ±1.3556, and ±2.875) are selected from the roots of the 4^{th} order and 5^{th} order Hermite polynomials. To check the Gaussian nature, the best fitted normal distribution are plotted against each experimental eigen frequencies. It is observed that first and third experimental eigen frequencies are Gaussian in nature whereas, other two measured eigen frequencies are non-Gaussian. However, 3rd order gPC expansion using Hermite polynomial can efficiently described the non-Gaussian nature of the eigen frequencies. Experimental mass density is represented by 3rd order gPC expansion and shown in Figure 2. The deterministic coefficients representing 3rd order gPC expansion for the mass density are 2.1143, 0.0540, 0.0075, and 0.0023, respectively. The deterministic coefficients for the uncertain elastic parameters

Eigen freq.	$\mathfrak{d}_0$	$\mathfrak{d}_1$	$\mathfrak{d}_2$	$\mathfrak{d}_3$
λ_1	115.489	4.536	0.366	0.011
λ_2	144.805	5.771	0.231	0.4828
λ_3	275.165	8.761	1.083	0.338
λ_4	395.763	14.867	0.465	1.311

Table 1: The gPC coefficients of the first 4 eigen frequencies (Hz) from experimental measurement

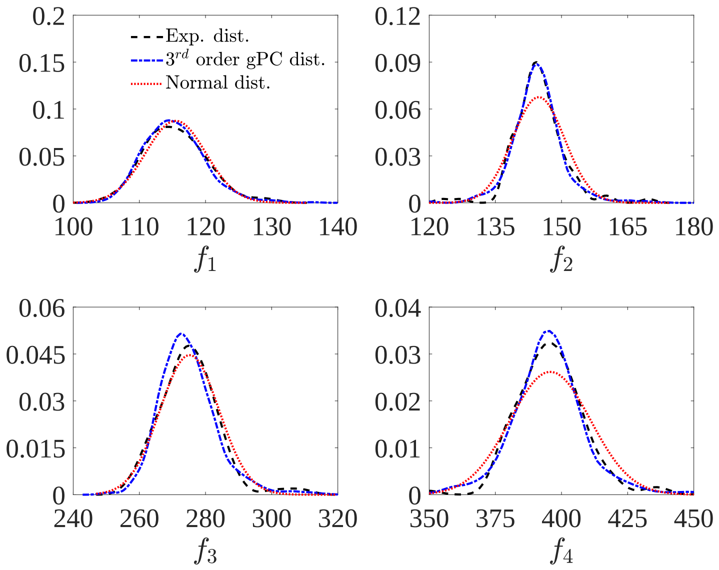


Figure 1: Stochastic representation of the experimental eigen frequencies (Hz) and comparison with the normal distribution

are estimated by employing 3rd order gPC expansion by minimization of the cost function as stated in Section 5. The representation of the identified material parameters $\mathfrak{m}$ in terms of gPC expansion is

$$\mathfrak{m}(\xi) = \sum_{i=0}^{3} \mathfrak{m}_i \varPsi_i(\xi), \qquad \mathfrak{m} = \{E_{11}, E_{12}, G_{12}\}. \tag{25}$$

Two cases of optimization procedure are adopted herein. The cost functions for the two cases are developed to minimize of errors: upto 3rd order central moment and upto 4th order central moment. The identified deterministic coefficients of the elastic moduli E_{11}, E_{22} and shear

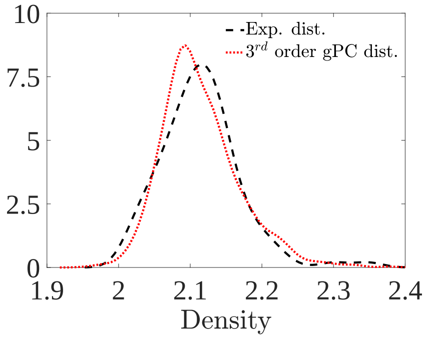


Figure 2: Stochastic representation of the experimental mass density (gm/cm^3)

modulus G_{12} are listed in Table 2. The first coefficient $\mathfrak{m}_0$ of the gPC expansion represents the expected value of the elastic parameters for the epoxy based glass-fiber reinforced composite laminated plate. The standard deviation of the elastic parameters are also calculated in Table 2 referring Eq. (18). The PDFs of the uncertain elastic parameters are constructed using gPC expansion and are presented in Figure 3. To check the accuracy of the gPC constructed PDFs, PDFs of the first 4 eigen frequencies are reconstructed with the application forward stochastic model and are shown in Figure 4. The histograms of the experimental eigen frequencies are depicted in this Figure. The reconstructed PDFs of the eigen frequencies considering identified gPC coefficients can well represent the uncertainty of experimental eigen frequencies. Moreover, gPC coefficients calculated by minimization of error functions derived from 4^{th} order central moments represent better accuracy over the 3^{rd} order moments minimization. The gPC coefficients evaluated from the higher order statistical moment minimization technique can predict the uncertainty of the eigen frequency with reasonably higher accuracy. The non-Gaussian nature of the 2^{nd} and 4^{th} eigen frequencies are well estimated by the identified coefficients using higher order statistical error minimization technique specifically near the tail region.

	Parameters	$\mathfrak{m}_0$	$\mathfrak{m}_1$	$\mathfrak{m}_2$	$\mathfrak{m}_3$	σ
Upto 3^{rd} order central moment minimization	E_{11} (GPa)	69.398	6.514	0.837	0.158	6.632
	E_{22} (GPa)	27.141	3.023	0.381	0.017	3.071
	G_{12} (GPa)	6.117	0.576	0.080	0.022	0.589
Upto 4^{th} order central moment minimization	E_{11} (GPa)	68.714	7.041	0.684	0.898	7.440
	E_{22} (GPa)	27.401	2.681	0.655	0.080	2.843
	G_{12} (GPa)	6.122	0.578	0.078	0.010	0.589

Table 2: The gPC coefficients of the uncertain elastic parameters

7 CONCLUSIONS

The identification of stochastic behavior of the elastic parameters of the laminated composite plate using non-sampling based stochastic inverse process is presented in this paper. Collocation based non-intrusive gPC expansion method is used to identify the randomness of the elastic parameters. The identification of uncertainty of the elastic parameters transforms into the identification of the unknown deterministic coefficients of the elastic moduli and shear modulus for the laminated composite plate. The experimental distribution of the first 4 eigen frequencies and mass density of the composite plate are used as inputs for the stochastic inverse identification algorithm. An optimization technique is adopted to estimate the deterministic coefficients of the uncertain elastic parameters by minimization of the cost function. The cost function is devel-

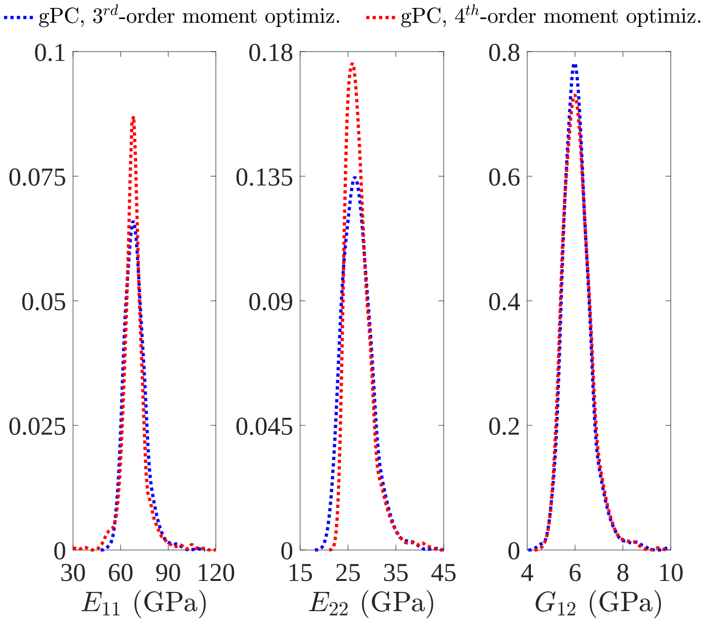


Figure 3: PDF of the identified elastic parameters

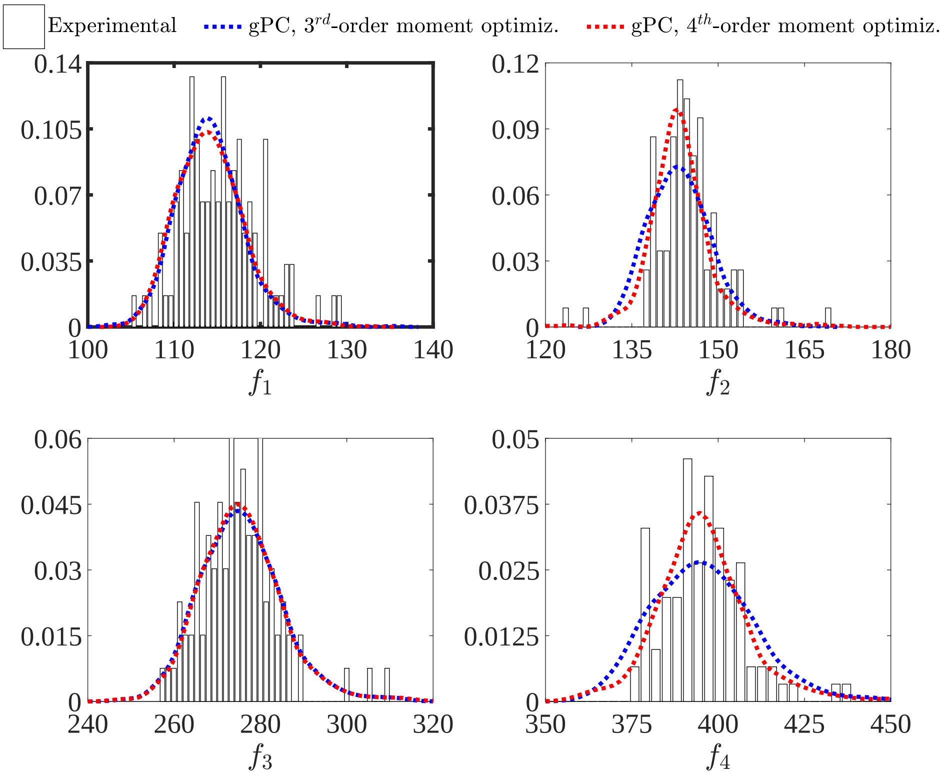


Figure 4: Reconstruct the PDFs of the eigen frequencies (Hz) from the identified gPC expansions of the elastic parameters

oped by summing up the quadratic difference between experimental statistical moments of the eigen frequencies and the simulated statistical moments of the eigen frequencies derived using the gPC expansion method. The use of Hermite polynomial in the gPC expansion method can efficiently inferred the distributions of the elastic parameters from the combination of Gaussian and non-Gaussian experimental eigen frequencies. The accuracy of the inverse identification is increased with the incorporation of the higher order statistical moments in the optimization process. The reconstructed PDFs of the eigen frequencies can efficiently predict the uncertainty of the experimental eigen frequencies, specifically with the application of the higher order statistical moment optimization.

8 ACKNOWLEDGMENT

The first author gratefully acknowledge the financial support extended by the Deutscher Akademischer Austauschdienst (DAAD).

REFERENCES

[1] C.M. Mota Soares, M. Moreira de Freias, A.L. Araújo, P. Pedersen. Identification of material properties of composite plate specimens. *Composite Structures*, **25(1-4)**, 277-285, 1993.

[2] A.K Bledzki, A. Kessler, R. Rikards, A. Chate. Determination of elastic constants of glass/epoxy unidirectional laminates by the vibration testing of plates. *Composites Science and Technology*, **59(13)**, 2015-2024, 1999.

[3] R. Rikards, A. Chate, G. Gailis. Identification of elastic properties of laminates based on experiment design. *International Journal of Solids and Structures*, **38(30-31)**, 5097-5115, 2001.

[4] R. Rikards, A. Chate, W. Steinchen, A. Kessler, A.K. Bledzki. Method for identification of elastic properties of laminates based on experiment design. *Composites: Part B*, **30**, 279-289, 1999.

[5] T. Lauwagie, Vibration-based methods for the identification of the elastic properties of layered materials. Dissertation, Catholic University of Leuven.

[6] M. Tanaka, G.S. Dulikravich. eds. *Inverse Problems in Engineering Mechanics*. Elsevier, 1998.

[7] J. Kaipio, E. Somersalo. *Statistical and Computational Inverse Problems*. Springer, 2005.

[8] R. Aster, B. Borchers, C. Thurber. *Parameter Estimation and Inverse Problems*. Academic Press, 2004.

[9] L. Tierney. Markov Chains for Exploring Posterior Distributions. *The Annals of Statistics*, **22**, 1701-1728, 1994.

[10] W.R. Gilks, S Richardson, D.J, Spiegelhalter eds. *Markov Chain Monte Carlo in Practice* . Springer-science+business media, B.V., 1996.

[11] J.B. Nagel. Bayesian techniques for inverse uncertainty quantification. *PhD Thesis*, University of Bonn, 2017.

[12] R.G. Ghanem, A. Doostan. On the construction and analysis of stochastic models: characterization and propagation of the errors associated with limited data. *Computational Physics*, **217**, 63-81, 2006.

[13] B.V. Rosić, A. Litvinenko, O. Pajonk, H.G. Matthies. Sampling-free linear Bayesian update of polynomial chaos representations. *Computational Physics* **37**, 5761-5787, 2012.

[14] R.G. Ghanem, P.D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Dover Publication, 1991.

[15] K. Sepahvand, S. Marburg, H.-J. Hardtke. Uncertainty quantification in stochastic systems using polynomial chaos expansion. *Applied Mechanics*, **02(02)**, 305353, 2010.

[16] K. Sepahvand, S. Marburg. Identification of composite uncertain material parameters from experimental modal data. *Probabilistic Engineering Mechanics*, 148-153, 2014.

[17] K. Sepahvand and S. Marburg. Non-sampling inverse stochastic numerical-experimental identification of random elastic material parameters in composite plates, *Mechanical System and Signal Processing*, **54-55**, 172-181, 2015.

[18] K. Sepahvand, C.A. Geweth, F. Saati, M. Klaerner, L. Kroll and S. Marburg. Spectral Representation of Uncertainty in Experimental Vibration Modal Data, *Advances in Acoustics and Vibration*, **9695357**, 2018.

[19] J.N. Reddy, Mechanics of Laminated Composite Plates and Shells. *Boca Raton: CRC Press*, 2003.

[20] T. Kant, H. Varaiya, C. P. Arora, Finite element transient analysis of composite and sandwich plates based on a refined theory and implicit time integration schemes, *Composite structures*, **36(3)**, 401-420, 1990.

[21] A.G. Niyogi, M.K. Laha, P.K. Sinha, Finite element vibration analysis of laminated composite folded plate structures, *Shock and Vibration*, **6**, 273-283, 1999.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

GAUSSIAN PROCESSES FOR REGRESSION AND CLASSIFICATION TASKS USING NON-GAUSSIAN LIKELIHOODS

Diego Echeverria-Rios[1], **P. L. Green**[2]

Institute for Risk and Uncertainty, School of Engineering
University of Liverpool, UK
e-mail: sgdechev@liverpool.ac.uk[1], plgreen@liverpool.ac.uk[2]

Keywords: Machine Learning, Gaussian Process, EM algorithm, Non-Gaussian likelihood, Regression, Classification

Abstract. *A solution to non-Gaussian likelihood problems for Gaussian Process (GP) regression is proposed. The present contribution aims to address the scenario where data has been corrupted with noise whose distribution has heavier tails than a Gaussian. A variant of the Expectation-Maximisation (EM) algorithm and a GP are used in a complementary fashion to develop a model that captures the behaviour of such data, by eliminating the effects of 'non-Gaussian components' in GP predictions.*

We model the likelihood function with a mixture of Gaussian distributions which allows us to use a variant of the EM algorithm to classify the observations. The classification outcomes are later used to assemble a sparse dataset which is corrupted only by Gaussian noise. Finally, a GP is trained on the sparse dataset and its predictive distribution is used to simulate the process under study. The behaviour of the proposed model has been evaluated using, synthetic and benchmark datasets, providing comparisons between a standard GP and a GP that assumes an input-dependent noise model (i.e. a Heteroscedastic GP).

1 INTRODUCTION

A Gaussian Process (GP) is a widely used Machine Learning regression technique which aims to mimic and predict the behaviour of a system, quantifying the uncertainties associated with its predictions. Machine Learning regression methods use noisy observations of a process to infer knowledge about a true underlying function. In a standard GP approach, it is assumed that the data has been corrupted with Gaussian noise. However, in real applications, this condition is not always satisfied and data can be corrupted with different sources of noise. Thus, a non-Gaussian distribution may be induced over the set of observations. Under this scenario, a GP is not always able to simulate the actual process, because it is reliant on the assumption of a Gaussian noise model.

Previous works have addressed this problem using, for example, a Mixture of Gaussian Process (MGP), [1, 2], which is a variant of the Mixture of Experts (ME) model [3]. The MGP assumes that each observation has been corrupted independently by Gaussian noise whose variance is constant only across separate regions of input space. Hence, a single GP is assigned to each of these regions and a gating function activates the corresponding GP to interpolate between the given inputs. As part of this solution, some MGP models have adopted the well-known Expectation Maximisation (EM) algorithm to aid classification (E-step) and parameter estimation (M-step) tasks. In [4, 5], accuracy was increased by implementing a so-called 'heuristic parameter estimation' approach in the M-step. Alternatively, Chen et al. [6] derived a precise hard-cut EM algorithm, where sparse datasets were assembled and later used to train a MGP, demonstrating reduced computational cost without compromising the accuracy of the resulting model.

Although these methods estimate the behaviour of a process using data affected by noise whose variance is not constant over the input space, they must implement and train more than one GP, which increases the complexity of the model compared to a standard GP. An example of an approach that, instead, uses a mixture of GPs that globally act over the entire input space is given in Lazaro-Gredilla et al. [7], where a variational approach is used to identify trajectories in multi-object target tracking problems.

The present contribution uses a Mixture of Gaussians to create a noise model with heavier tails than a Gaussian. We apply a variant of the EM algorithm that allows us to learn the noise mixture parameters exclusively from the data, without dividing the input space into separate regions. From the classification task, the labelled observations are used to assemble a sparse dataset that is affected with noise from a single Gaussian distribution. A standard GP is trained with the sparse dataset and its predictive distribution is used to estimate values of the underlying system.

2 A BRIEF DESCRIPTION OF GAUSSIAN PROCESSES

2.1 A Gaussian noise model

In a regression problem, data is arranged as a set of input-output pairs $\{\mathbf{x}_n, y_n\}_{n=1}^{N}$, also known in Machine Learning as the *training dataset*. Each observation, y_n is considered to be a noisy instance of the system under study, $f(\mathbf{x}_n)$, at a given input $\mathbf{x}_n$. With a standard GP approach it is assumed that the noise corrupting each observation is sampled from a Gaussian distribution, such that

$$y_n = f(\mathbf{x}_n) + \epsilon_n \qquad \epsilon_n \sim \mathcal{N}\left(\epsilon_n | 0, \sigma^2\right) \tag{1}$$

Equation (1) induces a normal distribution over the set of observations $\{y_n \in \boldsymbol{y}\,|1 \leq n \leq N\}$, conditional on the system's inputs $\left\{\mathrm{x}_n \in \boldsymbol{X} \,|\, \mathrm{x}_n \in \mathbb{R}^D \text{ and } 1 \leq n \leq N\right\}$, easing the application of a Bayesian approach when a GP prior is defined over $\boldsymbol{f}$, where $f_n \equiv f(\mathrm{x}_n)$. This procedure provides analytic expressions that ease the parameter estimation process and calculations of the GP's mean predictions and predictive uncertainties. For these reasons, the assumed noise model in equation (1) is central to the application of a standard GP. These concepts are detailed and clarified in the following subsection.

2.2 Standard GP models

The set of observations $\boldsymbol{y}$ is a realisation of the stochastic process defined by equation (1). With the true function, f, uncertain, we may choose to define a probability distribution describing what forms f could take. Here, we define a multivariate zero-mean Gaussian prior over $\boldsymbol{f}$:

$$p(\boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{f}\,|\boldsymbol{0}, \boldsymbol{K}\right) \tag{2}$$

where $\boldsymbol{K}$ is a covariance matrix. Our aim is then to use a Bayesian approach to infer a posterior distribution $p(\boldsymbol{f} \mid \boldsymbol{X}, \boldsymbol{y})$, once the training data has been observed [8].

When introducing $\boldsymbol{K}$, we must assure that it is a valid covariance matrix. For this reason each element of the matrix will be described by a positive definite function, also called the *kernel function*, such as

$$k(\mathrm{x}_i, \mathrm{x}_j) = \sigma_f^2 \exp\left\{\frac{(\mathrm{x}_i - \mathrm{x}_j)^2}{2\ell^2}\right\} \tag{3}$$

Equation (3) is the well-known Square Exponential (SE) kernel. It ensures that $\boldsymbol{K}$ is symmetric and positive-semidefinite [9]. Moreover, it assigns correlations that depend on the closeness of the inputs $(\mathrm{x}_i - \mathrm{x}_j)$ (a property that can loosely be called *smoothness*). The kernel in equation (3) is a function of:

σ_f^2 : Vertical Length Scale.
ℓ : Horizontal Length Scale.

To determine a posterior distribution, we first define the probability of witnessing $\boldsymbol{y}$ given $\boldsymbol{f}$:

$$p(\boldsymbol{y} \mid \boldsymbol{f}) = \mathcal{N}\left(\boldsymbol{y} \mid \boldsymbol{f}, \boldsymbol{I}\,\sigma^2\right) \tag{4}$$

where, $\boldsymbol{I}$ is the identity matrix and σ^2 is the variance of the Gaussian noise that corrupted the observations $\boldsymbol{y}$. Marginalising, we can obtain

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y} \mid \boldsymbol{f}) p(\boldsymbol{f}) d\,\boldsymbol{f} \tag{5}$$

From equations (2) and (4), we can write

$$p(\boldsymbol{y}) = \int \mathcal{N}\left(\boldsymbol{y} \mid \boldsymbol{f}, \boldsymbol{I}\,\sigma^2\right) \mathcal{N}\left(\boldsymbol{f}\,|\boldsymbol{0}, \boldsymbol{K}\right) d\,\boldsymbol{f}$$

such that, solving the integral [1], we find that

$$p(\boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{y}\,|0, \boldsymbol{C}\right) \tag{6}$$

where $\boldsymbol{C}$ is,

$$\boldsymbol{C} = \boldsymbol{K} + \boldsymbol{I}\sigma^2 \quad \text{with} \quad C(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j) + \delta_{ij}\sigma^2 \tag{7}$$

As σ performs an analogous role as a kernel parameter, we can define $\boldsymbol{\theta} = \{\sigma_f, \ell, \sigma\}$ as the set of parameters to be estimated. Stated explicitly, the likelihood of $\boldsymbol{\theta}$ is therefore

$$p(\boldsymbol{y}\,|\,\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{y}\,|0, \boldsymbol{C}\right) \tag{8}$$

Equation (8) describes how probable it is to witness $\boldsymbol{y}$, conditional on the kernel parameters, $\boldsymbol{\theta}$, that parameterises $\boldsymbol{C}$. By using a GP, we are defining a relationship between $\boldsymbol{\theta}$ and the underlying function $\boldsymbol{f}$, which is not subject to a specific parametric family. For this reason, a GP is not considered a parametric model. Consequently, even though $\boldsymbol{\theta}$ may be thought of as 'model parameters', they are usually referred to as *hyperparameters* (to specify that the regression method is not restricted to a specific parametric family).

Taking the logarithm of equation (8), the log-likelihood of a standard GP is

$$\ln p(\boldsymbol{y}\,|\,\boldsymbol{\theta}) = -\frac{1}{2}\ln|\,\boldsymbol{C}\,| - \frac{1}{2}\,\boldsymbol{y}^T\,\boldsymbol{C}\,\boldsymbol{y} - \frac{N}{2}\ln(2\pi) \tag{9}$$

The process of finding the parameters, $\boldsymbol{\theta}$, that maximises the log-likelihood function is called Maximum Likelihood Estimation (MLE)[2]. Noting that equation (9) can be evaluated relatively easily, hyperparameter estimates can be realised using a MLE procedure with Gradient Based Methods [10].

2.3 GP predictions

Using the estimated optimum hyperparameters, $\boldsymbol{\theta}$, probabilistic predictions at new inputs can then be generated. For the case where we wish to make a prediction at a single new input $\mathbf{x}_*$, it can be shown that

$$p(y_*|\,\mathbf{x}_*, \boldsymbol{X}, \boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{y}_*\,|\,\boldsymbol{k}_*^T\,\boldsymbol{C}^{-1}\,\boldsymbol{y},\; k_{**} - \boldsymbol{k}_*^T\,\boldsymbol{C}^{-1}\,\boldsymbol{k}_*\right) \tag{10}$$

where, $\boldsymbol{k}_* = [k(\mathbf{x}_*, \mathbf{x}_i), ..., k(\mathbf{x}_*, \mathbf{x}_N)]$ and $k_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$ (for a detailed derivation of the GP predictive density, please refer to [8]).

This concludes the description of a standard GP in which the noise model, equation (1), was key to obtaining the required closed-form equations.

3 A NON-GAUSSIAN NOISE MODEL

To obtain a model that can predict the behaviour of a process when the data has been corrupted with non-Gaussian noise, we propose a noise model that can describe non-Gaussian distributions. To this end, and given that our aim is to preserve the closed-form solutions that are associated with Gaussian distributions, a noise model consisting of a linear superposition of

[1] Working in proportionality and using the completing the square procedure.

[2] The likelihood indicates the probability of witnessing an observation / set of observations as a function of the parameters in the regression model. Choosing parameters that maximise the likelihood function (i.e. 'maximum likelihood') is a standard optimisation criterion for GPs.

Gaussians is considered. By changing the number of Gaussians used in the superposition (or ‘mixutre’), their variance and the location of their means, a wide range of different probability distributions can be represented. Figure 1 shows the superposition of two Gaussians with the same mean $\mu_1 = \mu_2 = 0$, but different standard deviation $\sigma_1 = 0.1, \sigma_2 = 1$. This combination produces a non-Gaussian distribution (with heavy tails), represented by the magenta line.

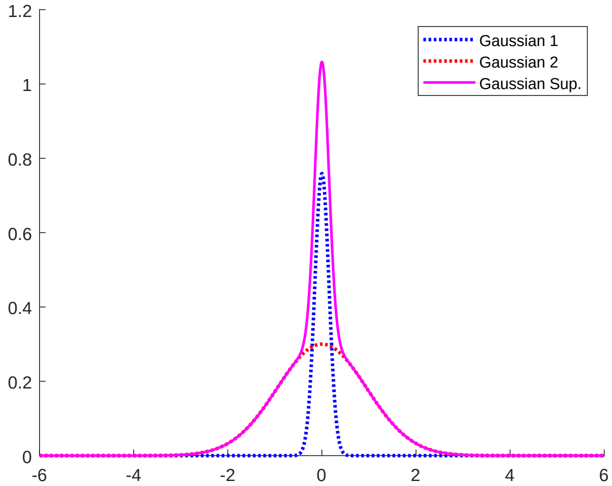


Figure 1: Non-Gaussian PDF formed by the superposition of two normal distributions.

To model a non-Gaussian noise distribution from the mixture of Gaussians, we first assume that the noise corrupting each observation has been generated from one of K Gaussian distributions. Therefore, binary vectors, $\boldsymbol{z}_n \in \mathbb{R}^K, n = 1, ..., N$, where each $\boldsymbol{z}_n$ is defined such that $\{z_{nk} \in \{0, 1\} \,|\, \sum_{k=1}^{K} z_{nk} = 1\}$, are introduced to identify which of the K Gaussian distributions was used to generate the sample of interest. Specifically, $z_{nk} = 1$ indicates that the observation y_n was corrupted by noise drawn from the kth Gaussian.

With this representation, it is now convenient to write the probability that $z_{nk} = 1$ as

$$\Pr(z_{nk} = 1) = \pi_k \qquad \text{satisfying} \quad \sum_{k=1}^{K} \pi_k = 1 \tag{11}$$

where $\pi_1, ..., \pi_K$ are known as the mixture *proportionalities* [11].

Marginalising the joint distribution $\Pr(\boldsymbol{z}_n)p(\epsilon_n|\,\boldsymbol{z}_n)$ over the possible states of $\boldsymbol{z}_n$ we obtain a probability distribution over ϵ_n:

$$p(\epsilon_n) = \sum_{\boldsymbol{z}_n} \Pr(\boldsymbol{z}_n)p(\epsilon_n|\,\boldsymbol{z}_n) \equiv \sum_{k=1}^{K} \Pr(z_{nk} = 1)p(\epsilon_n|z_{nk} = 1) \tag{12}$$

This allows us to write the new noise model as follows:

$$y_n = f(\mathbf{x}_n) + \epsilon_n, \qquad \epsilon_n \sim \sum_{k=1}^{K} \pi_k \, \mathcal{N}\left(\epsilon_n|0, \sigma_k^2\right) \tag{13}$$

Notice from equation (13) that, at each measurement y_n, the random variable ϵ_n is now sampled from a mixture of K Gaussians and thus a non-Gaussian distribution is induced over the full set of observations $\boldsymbol{y}$.

4 A NON-GAUSSIAN LIKELIHOOD

Having defined a noise model in previous section, a corresponding likelihood function is now defined. We start by computing the joint probability that an observation y_n, corrupted by noise drawn from the kth Gaussian distribution, is witnessed. To this end, the product rule can be applied to realise

$$p(y_n, z_{nk} = 1) = \text{Pr}(z_{nk} = 1)p(y_n|z_{nk} = 1) \tag{14}$$

From equations (11) and (13) we can write

$$p(y_n, z_{nk} = 1|f_n) = \pi_k \mathcal{N}\left(y_n|f_n, \sigma_k^2\right) \tag{15}$$

Marginalising over all the possible states of z_{nk} we obtain

$$p(y_n|f_n) = \sum_{k=1}^{K} \pi_k \mathcal{N}\left(y_n|f_n, \sigma_k^2\right) \tag{16}$$

Finally, assuming that the noise corrupting each observation is independent and identically distributed (*iid*), we can write,

$$p(\boldsymbol{y} \mid \boldsymbol{f}) = \prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k \mathcal{N}\left(y_n|f_n, \sigma_k^2\right) \tag{17}$$

To explicitly state the parameters that influence the likelihood shown in equation (17), we first associate $\boldsymbol{f}$ with the parameters in the GP kernel, $\boldsymbol{\theta}$, such that $f \equiv f(\theta)$. We then write

$$p(\boldsymbol{y} \,|\Theta) = \prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k \mathcal{N}\left(y_n|f_n, \sigma_k^2\right) \tag{18}$$

where $\Theta = \{\boldsymbol{\pi}, \boldsymbol{\sigma}, \boldsymbol{\theta}\}$ with , $\boldsymbol{\sigma} = \{\sigma_k | 1 \geq k \geq K\}$ and $\boldsymbol{\pi} = \{\pi_k | 1 \geq k \geq K\}$, grouping the GP kernel parameters and the parameters in the non-Gaussian likelihood together.

As described in the following section a variant of the EM algorithm is used to maximise equation (18) with respect to Θ.

5 THE ERROR-BASED EM ALGORITHM

Of the K Gaussian distributions in equation (13), the standard deviation of the 'narrowest' Gaussian is defined as $\sigma_s = \min\{\sigma_1, ..., \sigma_K\}$. Observations corrupted with noise drawn from $\mathcal{N}(0, \sigma_s^2)$ can be used to form a sparse training dataset $\{\mathbf{x}_s, y_s\}_{s=1}^{S}$. The data $\{\mathbf{x}_s, y_s\}_{s=1}^{S}$ can therefore be used to train a standard GP, where the 'high noise' effects of the non-Gaussian distribution in equation (13) are eliminated from the estimates of $\boldsymbol{\theta}$. Hence, we aim to derive a variant of the EM algorithm that performs the following specific tasks:

- E-step: classify each observation as being corrupted by noise drawn from one of the K Gaussians in equation (13).
- M-step (1): use the E-step outcome and the full dataset to realise MLE estimates of $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$.
- M-step (2): use the E-step outcome to assemble a sparse training dataset, $\{\mathbf{x}_s, y_s\}_{s=1}^{S}$, that when used in the MLE of a GP gives us estimates of the kernel parameters $\boldsymbol{\theta}$.

5.1 The Expectation step

From Bayes Theorem, the conditional distribution $p(z_{ni}|y_n)$ can be written as,

$$\text{Pr}(z_{ni} = 1|y_n) = \frac{\text{Pr}(z_{ni} = 1)p(y_n|z_{ni} = 1)}{p(y_n)} \tag{19}$$

Knowing from equation (11) that $p(z_{ni}) = \pi_i$, we aim to determine an expression for $p(y_n|z_{ni})$. We first note that $p(y_n|z_{ni} = 1) = \mathcal{N}(y_n|f_n, \sigma_i^2)$, where the mean of this distribution is equal to f_n. On the other hand, the conditional probability of witnessing y_n given the Gaussian label $z_{ni} = 1$, is equal to the conditional probability $\mathcal{N}(\epsilon_n|0, \sigma_i^2)$, one can describe the likelihood $p(y_n|z_{ni} = 1)$, in terms of the error, as follows,

$$p(y_n|z_{ni} = 1) = \mathcal{N}\left(\epsilon_n|0, \sigma_i^2\right) \tag{20}$$

Now we can use equation (11) as the prior and equation (20) as the likelihood, to write the posterior distribution (19), as

$$\text{Pr}(z_{ni} = 1|y_n) = \frac{\pi_i \mathcal{N}(\epsilon_n|0, \sigma_i^2)}{p(y_n)} \tag{21}$$

When marginalising equation (20) for all the states of $\boldsymbol{z}_n$, an expression for $p(y_n)$ is obtained, allowing us to rewrite (21) as,

$$\gamma(z_{ni}) \equiv \text{Pr}(z_{ni} = 1|y_n) = \frac{\pi_i \mathcal{N}(\epsilon_n|0, \sigma_i^2)}{\sum_k \pi_k \mathcal{N}(\epsilon_n|0, \sigma_k^2)} \tag{22}$$

Equation (22) is called the *responsibility* [11] By applying equation (22) to each of the N observations, we probabilistically classify each observation according to the Gaussian distribution that generated the corresponding corrupting noise.

From equation (22) we see that, the observation residual ϵ_n is needed and that this cannot be measured without knowing the actual system response, f_n. Consequently, this variant of the EM algorithm requires an initial estimate of the function $\boldsymbol{f}$. As there are no restrictions on how to compute the residual, any regression method can be used to initialise $\boldsymbol{f}$. In this work, we use a standard GP to define, $\boldsymbol{f} \approx \text{GP}(\boldsymbol{X}, \boldsymbol{y})$, which produces the required residual, $\boldsymbol{\epsilon} = \boldsymbol{f} - \boldsymbol{y}$. In fact, as is detailed in the following sections, $\boldsymbol{f}$ is re-estimated recursively as part of the training procedure.

5.2 The Maximisation step (1)

Now that we have realised an estimate for the responsibilities we aim to identify MLE estimates of the hyperparameters $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ by maximising

$$\ln p(\boldsymbol{\epsilon}) = \ln \left[\prod_{n=1}^{N} \sum_{k=1}^{K} \pi_k \, \mathcal{N}\left(\epsilon_n|0, \sigma_k^2\right)\right] \tag{23}$$

where equation (23) follows from equations (18) and equation (20). In the subsequent sections, we describe how, using equation (23), closed-form solutions for the MLE estimates of $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ can be reaslied.

5.2.1 Maximising with respect to the mixture standard deviations

Taking the partial derivatives of equation (23) with respect to $\boldsymbol{\sigma}$ gives

$$\frac{\partial}{\partial \boldsymbol{\sigma}} \ln p(\boldsymbol{\epsilon}) = \sum_{n=1}^{N} \frac{\partial}{\partial \boldsymbol{\sigma}} \ln \left[\sum_{k=1}^{K} \pi_k \, \mathcal{N}\left(\epsilon_n | 0, \sigma_k^2\right) \right] \tag{24}$$

With respect to σ_i (the standard deviation of the ith Gaussian), we therefore have

$$\frac{\partial}{\partial \sigma_i} \ln p(\boldsymbol{\epsilon}) = \sum_{n=1}^{N} \frac{1}{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\epsilon_n | 0, \sigma_k^2\right)} \frac{\partial}{\partial \sigma_i} \sum_{k=1}^{K} \pi_k \, \mathcal{N}\left(\epsilon_n | 0, \sigma_k^2\right) \tag{25}$$

Setting equation (25) equal to zero and recalling the definition of the 'responsibilities' (equation (22)), we can write

$$0 = \sum_{n=1}^{N} \gamma(\mathrm{z}_{ni}) \left[\epsilon_n^2 \sigma_i^{-2} - 1\right] \tag{26}$$

Finally, solving for σ_i gives

$$\sigma_i = \sqrt{\frac{\sum_{n=1}^{N} \gamma(\mathrm{z}_{ni}) \epsilon_n^2}{N_i}} \tag{27}$$

where $N_i = \sum_{n=1}^{N} \gamma(\mathrm{z}_{ni})$.

5.2.2 Maximising with respect to the proportionalities

We now aim to estimate the optimum $\boldsymbol{\pi}$ that maximises equation (23). When finding the MLE of the proportionalities, the constraint $\sum_{k=1}^{K} \pi_k = 1$ has to be considered. The optimisation procedure is therefore achieved using Lagrange multipliers. From the log-likelihood equation (23) the Lagrangian is

$$\mathcal{L}(\boldsymbol{\epsilon}, \lambda) = \ln p(\boldsymbol{\epsilon}) + \lambda \left(\sum_{k=1}^{K} \pi_k - 1 \right) \tag{28}$$

Evaluating the partial derivative, $\frac{\partial \mathcal{L}(\boldsymbol{\epsilon}, \lambda)}{\partial \pi_i}$, and setting the resulting expression equal to zero, we find that

$$0 = \sum_{n=1}^{N} \frac{\pi_i \mathcal{N}\left(\epsilon_\mathrm{n} | 0, \sigma_i^2\right)}{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\epsilon_\mathrm{n} | 0, \sigma_k^2\right)} + \lambda \tag{29}$$

where the responsibility,

$$\gamma(z_{ni}) = \frac{\pi_i \mathcal{N}\left(\epsilon_\mathrm{n} | 0, \sigma_i^2\right)}{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\epsilon_\mathrm{n} | 0, \sigma_k^2\right)}$$

can be identified and substituted into equation (29) to obtain ,

$$0 = \sum_{n=1}^{N} \gamma(z_{ni}) + \lambda \tag{30}$$

Multiplying equation (30) by π_i and rearranging, it can be shown that $\lambda = -N$. Recalling that, $N_i = \sum_{n=1}^{N} \gamma(\mathrm{z}_{ni})$, we then find that

$$\pi_i = \frac{N_i}{N} \tag{31}$$

Notice that, in both equations, (27) and (31), the responsibility term is required, which suggests a recursive solution where the E-step outcome is later used in the M-step. These steps form an optimisation technique that maximises a lower bound (of the log-likelihood function) at each iteration of the EM algorithm. Eventually the method will converge to a local or global solution after a finite number of iterations. The EM algorithm proof of convergence is beyond the scope of this work and for further insight please refer to [11].

5.3 The Maximisation step (2)

Once estimates of the hyperparameters $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ have been realised, the remaining hyperparameters, $\boldsymbol{\theta}$, can then be estimated.

As stated in Section 5, from the K Gaussians in equation (13), the 'narrowest' distribution was defined as having standard deviation $\sigma_s = \min\{\sigma_1, ..., \sigma_K\}$. Then, the observations corrupted with the Gaussian noise $z_{ns} = 1$ are found through the responsibility (22), as follows,

$$\gamma(z_{ns}) = \frac{\pi_s \mathcal{N}\left(\epsilon_{\mathrm{n}} \,|0, \sigma_s^2\right)}{\sum_{k=1}^{K} \pi_k \mathcal{N}\left(\epsilon_{\mathrm{n}} \,|0, \sigma_k^2\right)} \tag{32}$$

Hence, the new training data can be ensemble by the threshold function.

$$t(\gamma(z_{ns})) = \begin{cases} (\mathbf{x}_s, y_s) = (\mathbf{x}_n, y_n) & \text{if } \gamma(z_{ns}) > \frac{1}{K} \\ \text{No action} & \text{otherwise} \end{cases} \tag{33}$$

Therefore, using the sparse dataset $\{\mathbf{x}_s, y_s\}_{s=1}^{S}$ in the standard GP log-likelihood (9), we find that,

$$\ln p(\boldsymbol{y}_{(S)} \,|\, \boldsymbol{\theta}_s) = -\frac{1}{2} \ln |\, \boldsymbol{C}_{(S)} \,| - \frac{1}{2} \boldsymbol{y}_{(S)}^T \, \boldsymbol{C}_{(S)} \, \boldsymbol{y}_{(S)} - \frac{N}{2} \ln(2\pi) \tag{34}$$

where $\boldsymbol{y}_{(S)}$ is the vector of sparse observations and $\boldsymbol{C}_{(S)}$ is the covariance matrix formed by applying our GP kernel to the set of sparse inputs (i.e. $\boldsymbol{X}_{(S)}$).

We can now apply a MLE to equation (34), where the high noise effects within the full set of observations are eliminated from the estimated hyperparameters $\boldsymbol{\theta}_s$. We define this procedure as the M-step(2).

5.4 Making predictions

The EM algorithm variant described in the previous section, not only provides estimates for the hyperparameters $\boldsymbol{\Theta}$, but in addition, it classifies the observations during the E-step. Once training is complete, the estimated kernel parameters $\boldsymbol{\theta}_s$, can be used in the predictive distribution of a standard GP to make new predictions.

From equation (10), it is straightforward to show that

$$p(y_*|\, \mathbf{x}_*, \{\mathbf{x}_s, y_s\}_{s=1}^{S}) = \mathcal{N}\left(\boldsymbol{y}_* \,|\, \boldsymbol{k}_*^T \boldsymbol{C}_{(S)}^{-1} \boldsymbol{y}_{(S)},\; k_{**} - \boldsymbol{k}_*^T \boldsymbol{C}_{(S)}^{-1} \boldsymbol{k}_*\right) \tag{35}$$

where, $\boldsymbol{k}_* = [k(\mathbf{x}_*, \mathbf{x}_s), ..., k(\mathbf{x}_*, \mathbf{x}_S)]$ and $k_{**} = k(\mathbf{x}_*, \mathbf{x}_*)$.

A set of new predictions can be calculated using the predictive mean of equation (35), given by

$$\boldsymbol{\mu}_* = \boldsymbol{K}_*^T \boldsymbol{C}_{(S)}^{-1} \boldsymbol{y}_{(S)} \tag{36}$$

Consequently, the absence of non-Gaussian noise components in the training dataset $\{\mathbf{x}_s, y_s\}_{s=1}^{S}$ allows us to use the predictive mean of a standard GP to make new predictions. Recalling fro the E-step that an estimate of $\epsilon_1, ..., \epsilon_N$ is required to start a new iteration of the training algorithm, we can use equation (36) to estimate $f(\mathbf{x})$ at the N required positions. In other words, once a new estimated of $\boldsymbol{f}$ has been realised, a new estimate of the residuals is readily obtained. With this, $\boldsymbol{\epsilon}$ can then be used in a new iteration of the EM procedure.

This concludes the modelling section of the proposed method.

6 EXPERIMENTS

In this section, the model performance is accessed using two separate cases.

Case 1: a synthetic dataset was generated to assess the suitability of the approach when the noise model truly is represented as a mixture of Gaussians. Knowing the function from which the observations are generated, we investigated the model's predictive performance using the Mean Squared Error (MSE). We compared its performance with a standard GP and a Heteroscedastic GP (HGP - a GP which allows the variance of the noise model to be input-dependent) [3]. The synthetic data allowed us to evaluate the proposed algorithm's ability to perform classification, as the latent variable, z_{nk} from each observation were known. Finally, the classification performance was determined simply by counting the number of labels that had been correctly identified.

Case 2: we test the models behaviour when using a training dataset whose observations have been corrupted with noise whose variance is input-dependent. This second case is interesting as it highlights a scenario where the proposed mixture of Gaussians noise model is erroneous and, in fact, a HGP should be more suitable.

6.1 Case 1

In this experiment 100 realisations of the function, $f(x) = \frac{1}{2}x\sin(x)$, were corrupted following the noise model described in equation (13). To study the model performance when using data corrupted by non-Gaussian noise whose distribution has heavy tails, $K = 2$ sources of noise are used. The Gaussian mixture parameters were set as follows:

Proportionalities:	$\pi_1 = 0.7, \pi_2 = 0.3$
Standard Deviations:	$\sigma_1 = 10, \sigma_2 = 90$

The observations obtained from the corrupted sine function were used to train the proposed model, where results of the regression and classification performances are shown in the left and

[3] The Heteroscedastic GP used in this analysis is a MLE variant of the model proposed by Goldberg in [12], as suggested by Kersting [13].

right sides of Figure (2), respectively. The miss-classified observations are shown in red. In this case, 2 out of the 100 observations were incorrectly classified.

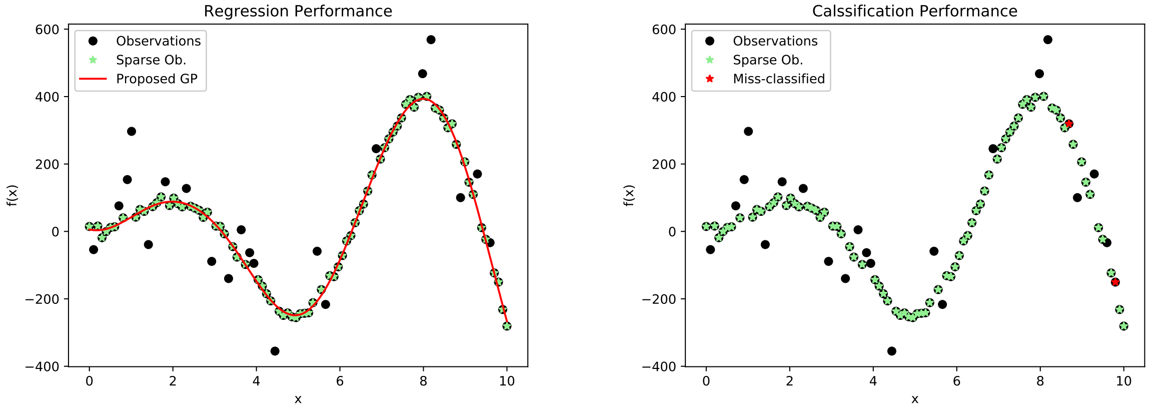


Figure 2: Case 1. The proposed regression model (left) and classification performance (right). In the left-panel the green points correspond to observations corrupted with 'low noise', $\sigma_1 = 10$, and the black points corresponds to observations corrupted with 'high noise', $\sigma_2 = 90$.

To quantify the model's ability to replicate the true sine function, we made 100 predictions at inputs omitted originally from the training data and calculated the MSE with respect to the values of the true underlying function. Comparisons of the MSE between a standard GP and a HGP are shown in Table 1, where the lowest value corresponds to the proposed model. In addition, a visual comparison of the models regression performance is shown in Figure 3. It is clear that the standard GP and the HGP were affected by the high noise observations; deviating their predictive mean from the trajectory of the true underlying function.

Case 1: Mean Square Error	
GP	0.04945
HGP	0.06541
Proposed GP	0.00231

Table 1: MSE of 100 predictions at inputs that were not used as training data.

The inferred noise-model parameters, $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$, are compared in Table 2 with the original noise parameters. It can be seen that the parameter estimates are close to the true values.

Case 1: Noise Parameters				
	Calculated		Original	
Gaussian noise:	1	2	1	2
Proportionalities :	0.7097	0.2902	0.70	0.30
Std. Deviations :	12.340	85.7299	10	90

Table 2: Comparison between the noise parameters that the model calculated and the ones used to generate the non-Gaussian noise.

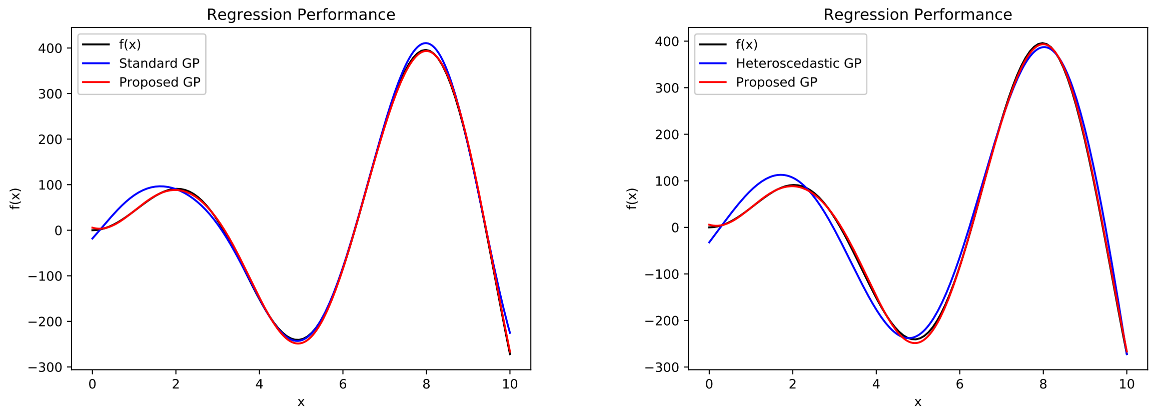


Figure 3: Case 1. Comparisons of the predictions made by the proposed regression model, a standard GP (left) and a heteroscedastic GP (right).

6.2 Case 2

This experiment aimed to asses the regression performance of the proposed method when using data corrupted with noise whose variance is input-dependent. The well-known benchmark data from Silverman et at. [14] was used. This data was collected from research concerning motorcycle helmet efficacy in a crash accident. It consists of 133 observations, corresponding to accelerometer readings that have been taken through a motorcycle crash time line.

It is known from previous works, such as [15], that the motorcycle data from Silverman can be accurately modelled as being heteroscedastic. Furthermore, from [15] and [14] we know that the 'low noise' portion of the data is only found in the first region of the domain (from 0 to 10 ms), as Figure 4 shows. As with case 1, we assume a noise model with $K = 2$ Gaussians. The classification results of the proposed method are shown in Figure 4 (left) and its regression performance is shown in Figure 4 (right). Figure 4 (left) shows how our model classified the data that was corrupted with high-variance noise as outliers and how the remaining sparse observations provided enough information to make reasonable predictions of the system's behaviour. Figure 4 (right) shows that our model follows a similar trajectory as the HGP at values $\leq$ 30ms. After this value the model tries to go through the sparse observations, deviating its trajectory from the HGP. The calculated noise parameters for the motorcycle data experiment are shown in Table 3.

Case 2 illustrates an example where, even though the Mixture of Gaussians noise model is being applied to a scenario where a heteroscedastic noise model is thought to be more appropriate, the proposed method functions relatively well. This is encouraging as it illustrates that, potentially, the proposed method may be relatively robust even when applied to scenarios where the Mixture of Gaussians noise model is, in fact, erroneous. A more-thorough exploration of this property is a topic of future work.

Case 2: Noise Parameters		
Gaussian noise:	1	2
Proportionalities :	0.5590	0.4409
Std. Deviations :	10.406	30.019

Table 3: Calculated noise parameters.

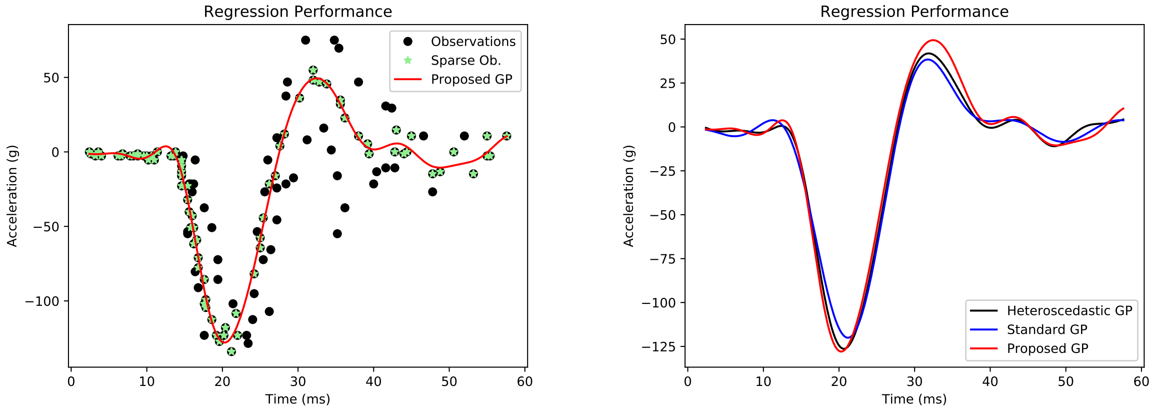


Figure 4: Classification and regression performances of the proposed model.

7 CONCLUSIONS

The current paper explores the situation where Gaussian Process (GP) regression is being performed on a dataset, where observations of the true underlying function have been corrupted with non-Gaussian noise. A method is proposed which, by using a noise model that consists of a mixture of Gaussians, is able to address such a scenario while preserving closed-form expressions for the GP predictions. The approach uses a variant of the Expectation-Maximisation (EM) algorithm whereby, to aid parameter estimation, observations corrupted with high levels of noise are treated as outliers.

Two case studies were investigated. Case 1 concerned data generated from the proposed noise model, while case 2 concerned data where an input-dependent noise model is known to work well. Encouragingly, the model performed well for both bases, indicating that the proposed algorithm may be applicable to a wide-variety of cases where data has been corrupted with non-Gaussian noise.

REFERENCES

[1] V. Tresp, “Mixtures of Gaussian Processes,” in *Advances in Neural Information Processing Systems 13*, 2001.

[2] C. E. Rasmussen and Z. Ghahramani, “Infinite mixtures of Gaussian process experts,” in *Neural Information Processing Systems*, 2002.

[3] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, “Adaptive Mixtures of Local Experts,” *Neural Computation*, 1991.

[4] E. Snelson and Z. Ghahramani, “Sparse Gaussian Processes using Pseudo-inputs,” *Advances in Neural Information Processing Systems 18*, 2006.

[5] C. Stachniss, C. Plagemann, a. Lilienthal, and W. Burgard, “Gas Distribution Modeling using Sparse Gaussian Process Mixture Models,” *Advanced Robotics*, 2008.

[6] Z. Chen, J. Ma, and Y. Zhou, “A precise hard-cut em algorithm for mixtures of Gaussian processes,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8589 LNAI, pp. 68–75, 2014.

[7] M. Lázaro-Gredilla, S. Van Vaerenbergh, and N. D. Lawrence, "Overlapping Mixtures of Gaussian Processes for the data association problem," *Pattern Recognition*, vol. 45, pp. 1386–1395, 2011. [Online]. Available: www.elsevier.com/locate/pr

[8] K. P. Murphy, *Machine Learning: A Probablistic Perspective*. MIT Press, 2012.

[9] C. E. Rasmussen, *Gaussian processes for machine learning.* MIT Press, 2006.

[10] J. S. Arora, "Jan A. Snyman, Practical Mathematical Optimization: An introduction to basic optimization theory and classical and new gradient-based algorithms," *Structural and Multidisciplinary Optimization*, 2006.

[11] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[12] P. W. Goldberg, C. K. I. Williams, and C. M. Bishop, "Regression with input-dependent noise: A Gaussian process treatment," *Group*, vol. 9, no. 7, pp. 1682–1697, 1998. [Online]. Available: http://wrap.warwick.ac.uk/15030/

[13] K. Kersting, C. Plagemann, P. Pfaff, and W. Burgard, "Most Likely Heteroscedastic Gaussian Process Regression," in *International Conference on Machine Learning*, 2007. [Online]. Available: http://gp.kyb.tuebingen.mpg.de/

[14] B. W. Silverman, "Some Aspects of the Spline Smoothing Approach to Non-Parametric Regression Curve," *Source: Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 47, no. 1, pp. 1–52, 1985. [Online]. Available: https://www.jstor.org/stable/pdf/2345542.pdf?refreqid=excelsior{%}3A5436432919e3175899d7c16e85544489

[15] A. D. Saul, J. Hensman, and N. D. Lawrence, "Chained Gaussian Process," in *International Conference on Artificial Intelligence and Statistics (AISTATS)*, vol. 51, 2016.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

ASSESSING MODEL FORM UNCERTAINTY FOR A SUSPENSION STRUT USING GAUSSIAN PROCESSES

Robert Feldmann[1] **and Roland Platz**[2]

[1]System Reliability, Adaptronics and Machine Acoustics SAM, Technische Universität Darmstadt,
Magdalenenstraße 4, D - 64289 Darmstadt, Germany
e-mail: feldmann@sam.tu-darmstadt.de

[2] Fraunhofer Institute for Structural Durability and System Reliability LBF,
Bartningstraße 47, D - 64289 Darmstadt, Germany
e-mail: Platz@lbf.fraunhofer.de

Keywords: Uncertainty Quantification, Structural Dynamics, Model Form Uncertainty.

Abstract. *In this paper, a modular spring-damper system that is integrated into a space truss structure is considered that was developed in the collaborative research centre SFB 805 "Control of Uncertainty in Load-Carrying Structures in Mechanical Engineering" at the Technische Universität Darmstadt. An idealized two degree of freedom (2DOF) model serves as a mathematical model to describe the dynamical system behaviour, yielding a system of two coupled ordinary differential equations (ODE) of second order. Previous own research already addressed the dynamic behaviour of the suspension system as regression curves from experiments for both stiffness and damping behaviour. Combining the regression models with the system equations of a 2DOF model of the modular spring-damper system yielded several model candidates to describe the dynamic behaviour.*

The resulting model form uncertainty is addressed in the framework of a model selection process. The approach employed in this paper uses a simplified form of the Kennedy and O'Hagan framework. Assuming that all models incorporate a model error, measurements of a system can be expressed as a the sum of the simulation model output, a discrepancy function and measurement noise. The discrepancy function gives information about the accuracy of the simulation model. It can therefore be used to compare model candidates and thus assess model form uncertainty. Among the approaches to model the discrepancy function, Gaussian processes (GP) have proofed to be suitable due to their versatility. Hence, for each model candidate, a GP representation for the discrepancy function can be determined based on experimental data. This paper shows the comparison and evaluation of the model candidates' discrepancy functions. Characterization of the underlying GP with regard to its confidence intervals is employed as a measure to select models that represent the dynamic behaviour of the modular spring-damper system most adequately.

1 INTRODUCTION

In engineering science, mathematical models are of utmost importance to predict the dynamic behaviour of structures or to improve structural design. The necessity for an accurate model stems from the need to assess dynamic performance under all environmental and possible operating conditions for stability evaluation, designing robust structures or for appropriate controller design.

During the modelling process, assumptions play a central role. Often, sufficient knowledge about the underlying physics as well as the model parameters and state variables of a system may be missing or the effort to build a more complex model is too high compared to the expected accuracy gain. Therefore, assumptions and simplifications have to be made that determine the form of a model. As a consequence, models may differ in complexity and the underlying physics can be described by the aid of linear, non-linear as well as empiric or axiomatic relations. The evaluation of competing models can be conducted by the quantification of *model form uncertainty* that describes the uncertainty in model selection, when models differ in their functional relationships. It is assumed that all models in the selection set are capable to represent the relevant dynamic behaviour, however, their output differs. Model form uncertainty has to be further differentiated form other types of uncertainty such as numerical uncertainty that arise in simulation models, uncertainty in the parameters of a mathematical model and experimental uncertainty that is due to variability in measurements [1].

Multiple approaches in literature exist to assess model form uncertainty, for example in the framework of verification and validation [2]. In general, verification means inspection of a mathematical model for matching sufficiently with numerical results of a simulation model, while validation addresses the comparison of a mathematical model with measurement gained from the real structure. Validation is typically accompanied by calibration, that is called the process of fitting the mathematical model to the observed data by adjusting its parameters [5]. Recently, model form uncertainty has gained momentum as a research topic and methodological approaches such as using nested sampling [3], a Bayesian inference approach [4] or random matrices [9] have been presented.

This paper builds upon the Kennedy and O'Hagan framework in the context of model form uncertainty quantification and model selection. The concept was published in 2001 [5] and introduced a broadly accepted representation for measurement data of a system

$$y_n = \eta(\boldsymbol{\theta}, \boldsymbol{x}_n) + \delta(\boldsymbol{x}_n) + \varepsilon_n \tag{1}$$

where $y_n \in \mathbb{R}, (n = 1, \dots, N)$ denoted the n-th of a total of N measurements, η is a simulation model output with inputs $\boldsymbol{x}_n \in \mathbb{R}^d$ and calibration parameters $\boldsymbol{\theta}$ like damping coefficients, δ is the discrepancy function and ε_n represents zero-mean normally distributed measurement noise for each measurement n. In contrast to the original paper [5], the models in this paper are assumed to be calibrated, so that $\boldsymbol{\theta}$ can be omitted and the simulation model in (1) simplifies to $\eta(\boldsymbol{x}_n)$. Measurements y_n and respective simulation model evaluations $\eta(\boldsymbol{x}_n)$ for selected input values $\boldsymbol{x}_n$ can now be used to construct a GP to model the sum of the discrepancy function and the measurement error, that are denoted by the difference

$$z_n := \delta(\boldsymbol{x}_n) + \varepsilon_n = y_n - \eta(\boldsymbol{x}_n). \tag{2}$$

A GP is a generalization of the Gaussian probability distribution. Whereas a probability distribution describes random variables that are scalars or vectors, a stochastic process such as a GP governs the properties of functions [8]. A GP constitutes of a so called mean function $\mu(\cdot)$

and a covariance function $c(\cdot,\cdot)$. The mean function is a linear combination of basis functions that are often assumed zero or one and weights, while the covariance function specifies entries of the covariance matrix for the respective input. Both mean function and covariance function have adjustable parameters, called hyperparameters [8]. This notion was adopted due to the non-parametric characteristic of GP[1]. For example in the case of a constant mean function $\mu(\cdot) = \beta$, the constant β is a hyperparameter. For chosen mean and covariance functions, the hyperparameters fully determine a GP. The comparison of confidence intervals of the GP describing the response behaviour of discrepancy functions thus yields a measure to compare competing models when model form uncertainty is present.

This paper is organized as follows: The structural system considered in this paper is presented in section 2 that covers modelling and simulation as well as the model candidates based on regression studies of a previous publication are introduced. Section 3 briefly introduces GP and their specifications and section 4 describes the model selection using confidence intervals of the model discrepancy function.

2 MAFDS AND MATHEMATICAL MODEL

The modular active spring damper system and space truss (German acronym MAFDS) in Fig. 1a is a suspension system that was designed with similar specifications and requirements as an air plane landing gear, although it is not a landing gear substitute. It was developed in the collaborative research centre SFB 805 at the Technische Universität Darmstadt in order to investigate data and model form uncertainty in a load-bearing structural system when predicting the dynamic response. It's main components are an upper truss structure ①, a lower truss structure ② with an elastic foot ⑤, guidance links that enable relative translation of the truss structures in z-direction ③ and a spring-damper component ④ [4]. The upper truss of the MAFDS is fixed on a frame ⑥ that can translate along guidance rails in z-direction. Dynamic drop tests are carried out similar to landing gear testing with a drop height h. Additional weights m_{add} can be added to the frame ⑥.

[1] While in nonparametric models such as GP, the number of parameters grows with the number of training samples, in parametric models such as polynomial models the number of parameters is fixed before training [8].

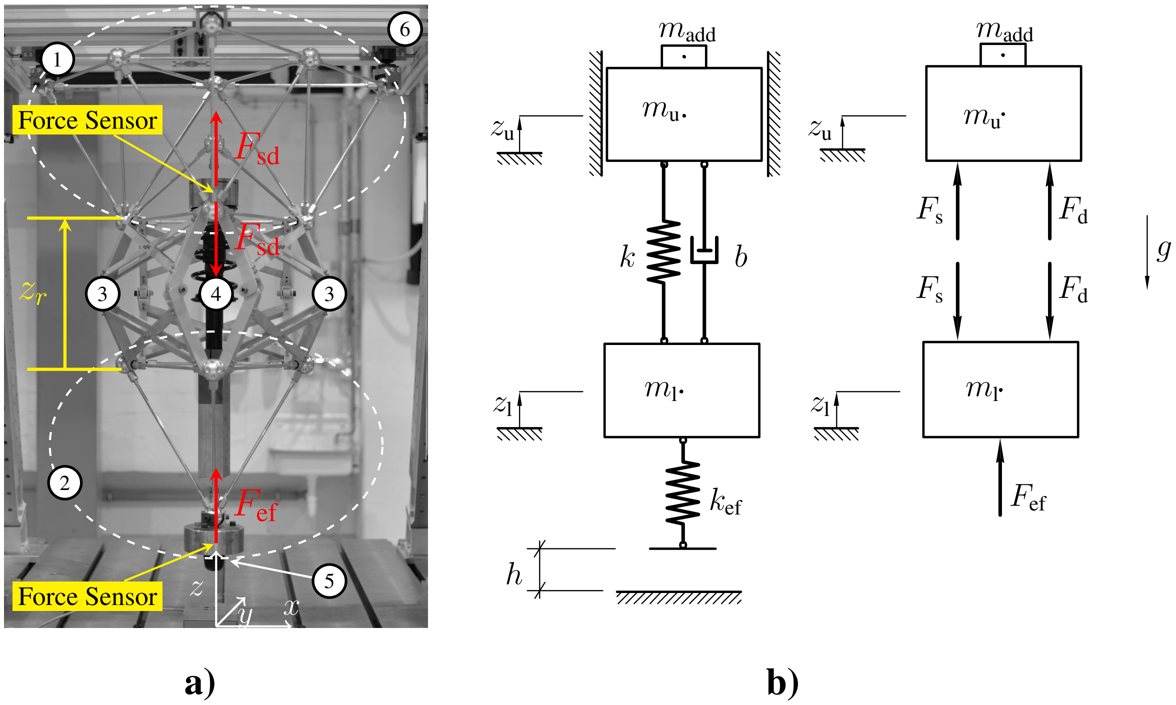


Figure 1: a) The MAFDS with its components and measured forces F_{sd} and F_{ef} of the spring damper and foot as well as the measured relative displacement z_{r}. b) 2DOF model of the MAFDS: k denotes the stiffness and b denotes the damping of the spring-damper-component ④, k_{ef} denotes the stiffness of the elastic foot ⑤ and h and m_{add} identify drop height and added mass, respectively. The spring and damper force are denoted F_{s} and F_{d} respectively. The whole structure is subject to the gravitation g [4].

2.1 Mathematical model

In previous research, the dynamic behaviour of the MAFDS was captured using a 2DOF model, see Fig. 1b. The upper truss, the upper part of the spring-damper component, the frame and the added mass constitute the upper mass m_{u} of the 2DOF model, the lower truss, including the foot and the lower part of the spring damper component are modelled by the lower mass m_{l} in the 2DOF model. The position of both the upper and lower mass is determined by the coordinates z_{u} and z_{l} of the 2DOF model, where $z_{\mathrm{r}} = z_{\mathrm{u}} - z_{\mathrm{l}}$ denotes the relative displacement of upper and lower mass. The system equations are given as:

$$\begin{pmatrix} m_{\mathrm{u}} & 0 \\ 0 & m_{\mathrm{l}} \end{pmatrix} \begin{pmatrix} \ddot{z}_{\mathrm{u}} \\ \ddot{z}_{\mathrm{l}} \end{pmatrix} + \begin{pmatrix} b(\dot{z}_{\mathrm{r}}) & -b(\dot{z}_{\mathrm{r}}) \\ -b(\dot{z}_{\mathrm{r}}) & +b(\dot{z}_{\mathrm{r}}) \end{pmatrix} \begin{pmatrix} \dot{z}_{\mathrm{u}} \\ \dot{z}_{\mathrm{l}} \end{pmatrix} + \begin{pmatrix} k(z_{\mathrm{r}}) & -k(z_{\mathrm{r}}) \\ -k(z_{\mathrm{r}}) & k(z_{\mathrm{r}}) + k_{\mathrm{ef}} \end{pmatrix} \begin{pmatrix} z_{\mathrm{u}} \\ z_{\mathrm{l}} \end{pmatrix} + \begin{pmatrix} (m_{\mathrm{u}} + m_{\mathrm{add}})g \\ m_{\mathrm{l}}\, g \end{pmatrix} = \mathbf{0}. \tag{3}$$

The detailed derivation of the system of coupled ODEs has been omitted here, for further details see [6] and [4]. The model parameters of the system are given in accordance with [4] in Tab. 1.

Table 1: Model parameters

parameter	symbol	value	SI unit
mass of upper structure	m_u	185	kg
mass of lower structure	m_l	41	kg
elastic foot stiffness	k_ef	$22.1 \cdot 10^4$	N/m

2.2 Model candidates

Previous research on the dynamic behaviour of the MAFDS focused on the investigation of the static and dynamic system properties of the spring-damper component [6]. Different regression models were developed for the stiffness function $k(z_\mathrm{r})$ and the damping function $b(\dot{z}_\mathrm{r})$ from experiments to describe the respective forces the spring damper component exercises on the upper and lower mass. For the suspension stiffness regression models (a) and (b), and for the damping function regression models (c) and (d) were defined.

- In regression model (a) the stiffness curve was approximated by piecewise linear polynomials

$$k_\mathrm{a}(z_\mathrm{r}) = \begin{cases} k_{\mathrm{a},1} + k_{\mathrm{a},2} z_\mathrm{r}, & z_\mathrm{r} \leqslant 0.068\,\mathrm{m} \\ k_{\mathrm{a},3} + k_{\mathrm{a},4} z_\mathrm{r}, & z_\mathrm{r} > 0.068\,\mathrm{m}. \end{cases} \tag{4}$$

The regression parameters are given in Tab. 2.

Table 2: Coefficients $k_{\mathrm{a},\cdot}$ for the piecewise first-order polynomial regression model of stiffness

coefficient	value	SI unit
$k_{\mathrm{a},1}$	28	kN/m
$k_{\mathrm{a},2}$	73	kN/m^2
$k_{\mathrm{a},3}$	-1.58	kN/m
$k_{\mathrm{a},4}$	516	kN/m^2

- In regression model (b) a cubic polynomial form was assumed for the stiffness function

$$k_\mathrm{b}(z_\mathrm{r}) = k_{b,1} + k_{b,2} z_\mathrm{r} + k_{b,3} z_\mathrm{r}^2 + k_{b,4} z_\mathrm{r}^3. \tag{5}$$

The regression parameters can be found in Tab. 3.

Table 3: Coefficients $k_{\mathrm{b},\cdot}$ for the cubic polynomial regression model of stiffness

coefficient	value	SI unit
$k_{\mathrm{b},1}$	23	kN/m
$k_{\mathrm{b},2}$	601	kN/m^2
$k_{\mathrm{b},3}$	$-1.49 \cdot 10^4$	kN/m^3
$k_{\mathrm{b},4}$	$1.24 \cdot 10^5$	kN/m^4

- Regression model (c) assumes piecewise power functions

$$b_{\mathrm{c}}(\dot{z}_{\mathrm{r}}) = \begin{cases} b_{\mathrm{c},1}\dot{z}_{\mathrm{r}}, & \dot{z}_{\mathrm{r}} \leqslant 0\,\mathrm{m/s} \\ b_{\mathrm{c},2}\dot{z}_{\mathrm{r}}, & \dot{z}_{\mathrm{r}} > 0\,\mathrm{m/s} \end{cases} \tag{6}$$

for the damping function. The regression coefficients under assumption of the respective regression model for the stiffness function, indicated by upper-case letters $(\cdot)^{\mathrm{a}}$, $(\cdot)^{\mathrm{b}}$, are given in Tab. 4.

Table 4: Coefficients $b^{\mathrm{a}}_{\mathrm{c},\cdot}$, $b^{\mathrm{b}}_{\mathrm{c},\cdot}$ for the piecewise first-order polynomial regression model of damping

coefficient	value	SI unit
$b^{\mathrm{a}}_{\mathrm{c},1}$	24.81	$\mathrm{kNs^2/m}$
$b^{\mathrm{a}}_{\mathrm{c},2}$	1.09	$\mathrm{kNs^2/m}$
$b^{\mathrm{b}}_{\mathrm{c},3}$	4.92	$\mathrm{kNs^2/m}$
$b^{\mathrm{b}}_{\mathrm{c},4}$	1.08	$\mathrm{kNs^2/m}$

- Model (d) describes a cubic polynomial approach for the damping function

$$b_{\mathrm{d}}(z_{\mathrm{r}}) = k_{d,1}\dot{z}_{\mathrm{r}} + b_{d,2}\dot{z}_{\mathrm{r}}^2 + b_{d,3}\dot{z}_{\mathrm{r}}^3. \tag{7}$$

Table 5: Coefficients $b^{\mathrm{a}}_{\mathrm{d},\cdot}$, $b^{\mathrm{b}}_{\mathrm{d},\cdot}$ for the cubic polynomial regression model of damping

coefficient	value	SI unit
$b^{\mathrm{a}}_{\mathrm{d},1}$	3.05	$\mathrm{kNs^2/m^2}$
$b^{\mathrm{a}}_{\mathrm{d},2}$	-5.24	$\mathrm{kNs^2/m^3}$
$b^{\mathrm{a}}_{\mathrm{d},3}$	2.67	$\mathrm{kNs^2/m^4}$
$b^{\mathrm{b}}_{\mathrm{d},1}$	3.16	$\mathrm{kNs^2/m^2}$
$b^{\mathrm{b}}_{\mathrm{d},2}$	-5.81	$\mathrm{kNs^2/m^3}$
$b^{\mathrm{b}}_{\mathrm{d},3}$	3.09	$\mathrm{kNs^2/m^4}$

For the subsequent analysis, the system (3) and the stiffness and damping functions introduced in this section form a set of $P = 4$ simulation model candidates that are given in Tab. 6.

Table 6: Stiffness and damping functions for the four model candidates

model number p	stiffness function	damping function
1	$k_{\mathrm{a}}(z_{\mathrm{r}})$	$b^{\mathrm{a}}_{\mathrm{c}}(\dot{z}_{\mathrm{r}})$
2	$k_{\mathrm{b}}(z_{\mathrm{r}})$	$b^{\mathrm{b}}_{\mathrm{c}}(\dot{z}_{\mathrm{r}})$
3	$k_{\mathrm{a}}(z_{\mathrm{r}})$	$b^{\mathrm{a}}_{\mathrm{d}}(\dot{z}_{\mathrm{r}})$
4	$k_{\mathrm{b}}(z_{\mathrm{r}})$	$b^{\mathrm{b}}_{\mathrm{d}}(\dot{z}_{\mathrm{r}})$

2.3 Initial conditions

For the drop tests, the structure is lifted up by a drop height h. When the system is dropped, it is assumed that the relative displacement z_r approaches zero and the two masses have equal velocities $v_0 = \sqrt{2gh}$ at the moment of impact. A simulation study has shown that this assumption leads only to a negligible error. Therefore, the initial conditions (8) can be adopted. The simulation was carried out in Matlab and a standard ODE solver for non-stiff ODEs (`ode45`) was utilized.

$$z_{\text{u}}(0) = 0 \quad \dot{z}_{\text{u}}(0) = \sqrt{2gh} \tag{8a}$$

$$z_{\text{l}}(0) = 0 \quad \dot{z}_{\text{l}}(0) = \sqrt{2gh} \tag{8b}$$

2.4 System inputs and outputs

Inputs to a simulation model such as η in (1) can be parameters and initial conditions and excitations that are needed to run the simulation model. Calibration parameters $\boldsymbol{\theta}$ can also be varied in the simulation model, but are fixed for the experiment [5]. In this paper, it is assumed that the calibration parameters are known such that the simulation model in (1) simplifies to $\eta(\boldsymbol{x}_n)$. Inputs for the drop tests of the 2DOF Model of the MAFDS are set to be the drop height h and additional weight m_{add} that can be added to the frame ⑥ in Fig. 1a. In analogy of the MAFDS to an aircraft landing gear, the additional weight m_{add} can for example account for additional payload.

$$\boldsymbol{x}_n = (h_n, m_{\text{add},n})^\top \tag{9a}$$

$$\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^\top \tag{9b}$$

are the input vector $\boldsymbol{x}_n \in \mathbb{R}^2$ in (1) and the (40×2) input matrix $\boldsymbol{X}$ for $N = 40$ measurements. For the 2DOF model of the spring damper system, the outputs were chosen to be the maximum relative compression $z_{\text{r,max}}$, the maximum force in the elastic foot $F_{\text{ef,max}}$ and the maximum force on the spring damper system $F_{\text{sd, max}}$, in accordance to [4]. The respective quantities are calculated as

$$z_r = z_{\text{u}} - z_{\text{l}} \tag{10a}$$

$$F_{\text{sd}} = F_{\text{s}} + F_{\text{d}} = k(z_r)z_r + b(\dot{z}_r)\dot{z}_r \tag{10b}$$

$$F_{\text{ef}} = k_{\text{ef}}z_l, \tag{10c}$$

with the spring force F_{s} and the damping force F_{d} depicted in Fig. 1b. The relative displacement z_{r} is set to be zero in a system state where the spring damper component is not deflected. The four model candidates specified in Tab. 6 are simulated for the inputs $\boldsymbol{x}_n$ specified in $\boldsymbol{X}$. The simulation output vectors denote

$$\boldsymbol{h}_{p,z_{\text{r}}} = (\eta_{p,z_{\text{r,max}}}(\boldsymbol{x}_1), \ldots, \eta_{p,z_{\text{r,max}}}(\boldsymbol{x}_N))^\top \tag{11a}$$

$$\boldsymbol{h}_{p,F_{\text{ef}}} = (\eta_{p,F_{\text{ef,max}}}(\boldsymbol{x}_1), \ldots, \eta_{p,F_{\text{ef,max}}}(\boldsymbol{x}_N))^\top \tag{11b}$$

$$\boldsymbol{h}_{p,F_{\text{sd}}} = (\eta_{p,F_{\text{sd,max}}}(\boldsymbol{x}_1), \ldots, \eta_{p,F_{\text{sd,max}}}(\boldsymbol{x}_N))^\top, \tag{11c}$$

where the index $p = 1, \ldots, P$ with $P = 4$ indicates the model number and $\eta_{p,z_{\text{r,max}}}$, $\eta_{p,F_{\text{ef,max}}}$, $\eta_{p,F_{\text{sd,max}}}$ denote the maximum outputs of the respective simulation model. A simulation of model

1 in Tab. 6 for a drop test with inputs $\boldsymbol{x} = (0\,\text{kg}, 0.09\,\text{m})^\top$ as an example yielded maximum output values $\eta_{1,z_\text{r,max}}(\boldsymbol{x})$, $\eta_{1,F_\text{ef,max}}(\boldsymbol{x})$, $\eta_{1,F_\text{sd,max}}(\boldsymbol{x})$ that are depicted in Fig. 2 as horizontal lines.

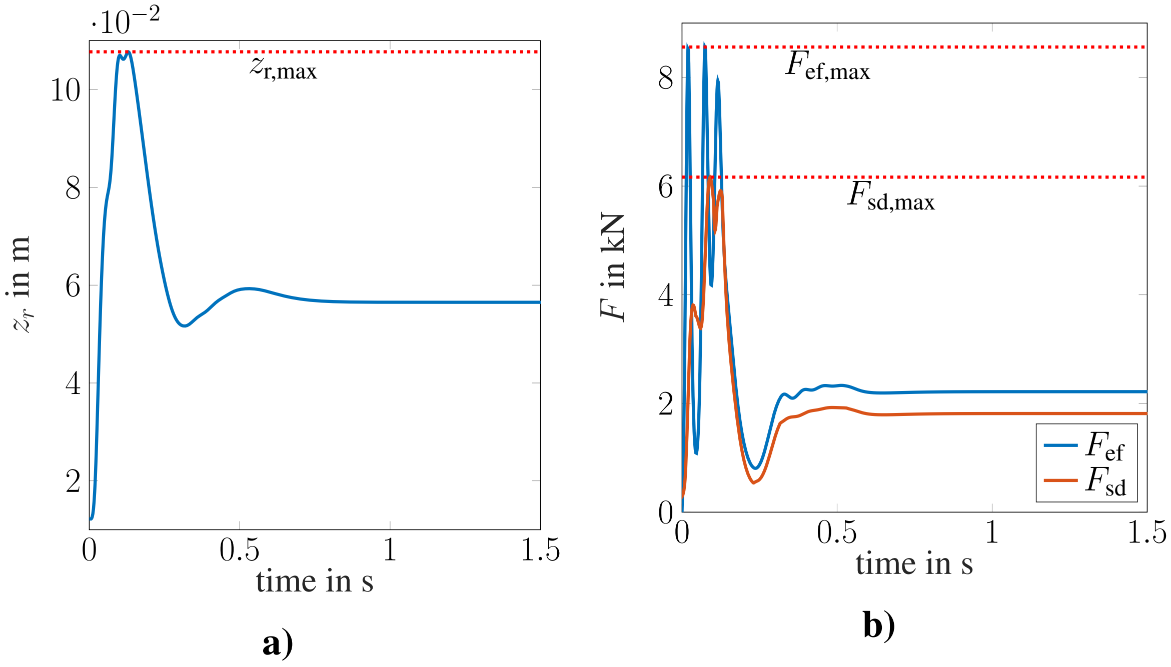


Figure 2: Outputs from a simulated drop test: (a) maximum relative compression $z_\text{r,max}$, (b) maximum force in the elastic foot $F_\text{ef,max}$ and maximum force on the spring damper system $F_\text{sd, max}$.

For the measurement of the relative displacement z_r, displacement sensors were attached between the upper and lower truss. Fig. 1a indicates the locations of the force sensors. The measured output vectors with the quantities in (10) of all N measurements denote

$$\boldsymbol{y}_{z_\text{r}} = (z_\text{r,max,1}, \ldots, z_\text{r,max,N})^\top \tag{12a}$$
$$\boldsymbol{y}_{F_\text{sd}} = (F_\text{sd,max,1}, \ldots, F_\text{sd,max,N})^\top \tag{12b}$$
$$\boldsymbol{y}_{F_\text{ef}} = (F_\text{ef,max,1}, \ldots, F_\text{ef,max,N})^\top. \tag{12c}$$

2.5 Training data for the GP

The input matrix $\boldsymbol{X}$ (9b) contains the input values specified in Tab. 7 for which the measured output vectors from the drop tests $\boldsymbol{y}_{z_\text{r}}$, $\boldsymbol{y}_{F_\text{sd}}$, $\boldsymbol{y}_{F_\text{ef}}$ (12) and the simulation output vectors $\boldsymbol{h}_{p,z_\text{r}}$, $\boldsymbol{h}_{p,F_\text{ef}}$, $\boldsymbol{h}_{p,F_\text{sd}}$ (11) were obtained.

Table 7: Input configurations and number of drop tests

weight in kg	height in m	number of drop tests
0	0.09	6
10	0.07	6
20	0.05	6
40	0.03	6
60	0.05	6
80	0.03	5
100	0.01	5
		$\sum = 40$

The sum of the discrepancy term and the measurement error z (2) for all three outputs and all $P = 4$ model candidates is now represented by the difference vectors

$$\boldsymbol{z}_{p,z_\mathrm{r}} = \boldsymbol{y}_{z_\mathrm{r}} - \boldsymbol{h}_{p,z_\mathrm{r}} = (\delta_{p,z_\mathrm{r}}(\boldsymbol{x}_1) + \varepsilon_{p,z_\mathrm{r},1}, \dots, \delta_{p,z_\mathrm{r}}(\boldsymbol{x}_N) + \varepsilon_{p,z_\mathrm{r},N})^\top \tag{13a}$$

$$\boldsymbol{z}_{p,F_\mathrm{ef}} = \boldsymbol{y}_{F_\mathrm{ef}} - \boldsymbol{h}_{p,F_\mathrm{ef}} = (\delta_{p,F_\mathrm{ef}}(\boldsymbol{x}_1) + \varepsilon_{p,F_\mathrm{ef},1}, \dots, \delta_{p,F_\mathrm{ef}}(\boldsymbol{x}_N) + \varepsilon_{p,F_\mathrm{ef},N})^\top \tag{13b}$$

$$\boldsymbol{z}_{p,F_\mathrm{sd}} = \boldsymbol{y}_{F_\mathrm{sd}} - \boldsymbol{h}_{p,F_\mathrm{sd}} = (\delta_{p,F_\mathrm{sd}}(\boldsymbol{x}_1) + \varepsilon_{p,F_\mathrm{sd},1}, \dots, \delta_{p,F_\mathrm{sd}}(\boldsymbol{x}_N) + \varepsilon_{p,F_\mathrm{sd},N})^\top \tag{13c}$$

where δ_{p,z_r}, δ_{p,F_ef}, δ_{p,F_sd} denote the discrepancy functions of each model p for the respective output and and $\varepsilon_{p,z_\mathrm{r}}$, $\varepsilon_{p,F_\mathrm{ef}}$, $\varepsilon_{p,F_\mathrm{sd}}$ denote measurement noise. The input vector $\boldsymbol{X}$ and each of the twelve difference vectors in (13) constitute the data set to train GPs using the methodology described in the next section. For the sake of simplicity, the vector $\boldsymbol{z}$ refers to any difference vector in the following section.

3 GAUSSIAN PROCESS MODEL

The aim of this section is to give a brief introduction to the modelling of the discrepancy function and measurement noise comprised in the vectors defined in (13) using a GP. A GP is the generalization of the normal distribution in the function space as it describes a distribution over functions.A visual explanation of this can be found in [8]: Thinking of a function as an infinite long vector with each entry representing a function value $f(\boldsymbol{x})$ of an input $\boldsymbol{x}$, a GP describes a multivariate normal distribution over a arbitrary finite number of these vector entries.

Given a set of input values $\boldsymbol{X} = (\boldsymbol{x}_1 \dots \boldsymbol{x}_N)^\top$ as in (9b) and the difference vectors as training outputs $\boldsymbol{z} = (z_1 \dots z_N)$ given in (13) where the $z_n \in \mathbb{R}$ are assumed realizations of a stochastic process f, a GP representation for the data yields the multivariate normal distribution

$$f \sim \mathcal{N}(\mu(\boldsymbol{X}), \boldsymbol{K}) \tag{14}$$

where μ denotes the mean function and $\boldsymbol{K} \in \mathbb{R}^{N \times N}$ denotes the covariance matrix. The covariance matrix is built up element-wise by the covariance function $c\colon \mathbb{R}^2 \times \mathbb{R}^2 \mapsto \mathbb{R}$

$$\boldsymbol{K}(k,l) = c(\boldsymbol{x}_k, \boldsymbol{x}_l), \tag{15}$$

where $\boldsymbol{x}_k, \boldsymbol{x}_l$ with $k, l = 1 \dots N$ denote the input vectors (9a). The mean function $\mu(\boldsymbol{x})$ and the covariance function c chosen in this paper are defined and further elaborated in the following. In a Bayesian framework for regression, the GP given by (14) can be regarded as a *prior* distribution.

3.1 Mean function

The mean function μ models the expectation of the GP. Especially when prior knowledge about the process is available, the mean function can be selected to fit the form of the expected mean. Quite often, the mean function is set to zero. However, since the elements of vectors $\boldsymbol{z}$ in (13) appear to be all negative, it seems reasonable to assume the mean function to be a constant $\beta \in \mathbb{R}$

$$\mu(\boldsymbol{x}) = \beta. \tag{16}$$

3.2 Covariance function

The covariance function c essentially determines the smoothness of the GPs response. There are several possible covariance functions, each of them cater specific requirements for the data to fit. In this paper we will utilize the squared exponential covariance function, that is suitable to describe smooth behaviour of functions. It is defined as

$$c(\boldsymbol{x}_k, \boldsymbol{x}_l) = \sigma_\mathrm{f}^2 \exp\left(-\frac{1}{2}(\boldsymbol{x}_k - \boldsymbol{x}_l)^\top \boldsymbol{M}(\boldsymbol{x}_k - \boldsymbol{x}_l)\right) + \sigma_\mathrm{n}^2 \delta_{kl} \tag{17}$$

where $\boldsymbol{x}_k, \boldsymbol{x}_l$ with $k, l = 1 \dots N$ denote the input vectors (9a) and δ_{kl} denotes the Kronecker delta. The matrix $\boldsymbol{M}$ is set to $\boldsymbol{M} = \boldsymbol{I}\ell^{-2}$ with unity matrix $\boldsymbol{I} \in \mathbb{R}^2 \times \mathbb{R}^2$ and length scale $\ell > 0$ [8]. The signal variance $\sigma_f > 0$ determines how much the function values deviate from the mean value. Larger values for the signal variance lead to larger deviation of the function.

Measurement noise is accounted for by the noise level parameter σ_n in the covariance function (17). It is assumed to be an additive, independent identically distributed Gaussian noise with variance σ_n^2 [8].

3.3 Hyperparameter optimization

In summary, the vector of hyperparameters that governs the behaviour of the GP can be written as $\boldsymbol{\theta}_\text{hyper} = (\beta, \ell, \sigma_\mathrm{f}, \sigma_\mathrm{n})^\top$. In order to represent the data set with input matrix $\boldsymbol{X}$ and training outputs $\boldsymbol{z}$ most adequately with a GP, the optimal set of hyperparameters $\boldsymbol{\theta}_\text{hyper,opt}$ is typically determined by maximizing the log marginal likelihood [8]

$$\boldsymbol{\theta}_\text{hyper,opt} = \underset{\boldsymbol{\theta}_\text{hyper}}{\arg\max} \log(p(\boldsymbol{z}|\boldsymbol{X}, \boldsymbol{\theta}_\text{hyper})). \tag{18}$$

In this paper, a Bayesian optimization scheme is used for the optimization (18). The objective of the Bayesian optimization is to minimize an expensive objective function when stochastic noise in function evaluations is present [11]. The Bayesian approach allows for tracking down potential optima as well as to explore the hyperparameter space in a way that can be customized by an acquisition function [10]. In this paper, optimization of the hyperparameters $\boldsymbol{\theta}_\text{hyper,opt}$ was carried out in Matlab using the function `bayesopt` with the 'Expected Improvement' acquisition function and yielded an optimal set of hyperparameters $\boldsymbol{\theta}_\text{hyper,opt}$.

4 MODEL SELECTION USING CONFIDENCE INTERVALS

This section presents the results of of the GP model's training according to the methodology presented in section 3. For the $P = 4$ models and three outputs in twelve training data sets each with input matrix $\boldsymbol{X}$ and respective training output vectors $\boldsymbol{z}$ given in (9b) and (13) the respective set of optimal hyperparameters $\boldsymbol{\theta}_\text{hyper,opt}$ (18) was obtained. Subsequently, confidence intervals for the respective prior GP are constructed and serve as a measure to assess model form uncertainty in a model selection process.

4.1 Confidence intervals for prior Gaussian process models

As mentioned earlier, a GP is fully determined by its hyperparameters. With the mean and covariance functions (16) and (17) its 95%-confidence interval can be specified with the hyperparameters of the GPs describing the response behaviour of the discrepancy terms δ_{p,z_r}, δ_{p,F_ef}, δ_{p,F_sd} (13). For the twelve sets of optimal hyperparameters $\boldsymbol{\theta}_\mathrm{hyper,opt} = (\beta_\mathrm{opt}, \ell_\mathrm{opt}, \sigma_\mathrm{f,opt}, \sigma_\mathrm{n,opt})^\top$, the lower and upper confidence bounds C_l and C_u respectively are calculated as

$$C_\mathrm{l} = \beta_\mathrm{opt} - 2\sigma_\mathrm{f,opt} \tag{19a}$$

$$C_\mathrm{u} = \beta_\mathrm{opt} + 2\sigma_\mathrm{f,opt}. \tag{19b}$$

Measurement noise $\varepsilon_{p,z_\mathrm{r}}$, $\varepsilon_{p,F_\mathrm{ef}}$, $\varepsilon_{p,F_\mathrm{sd}}$ (13) is neglected in the calculation of the confidence interval for the discrepancy functions. It is assumed that it has been fully captured in the noise level parameter σ_n^2 in the covariance function (17). The values for σ_n are given in Tab. 8.

Table 8: Values of noise level parameter σ_n for the GP models

	noise level parameter σ_n			
output	model 1	model 2	model 3	model 4
$z_\mathrm{r,max}$	0.001m	0.001m	0.001m	0.001m
F_ef	69.6596 N	68.3976 N	61.3605 N	37.8121 N
F_sd	51.8532 N	69.7893 N	4.0560 N	99.9499 N

4.2 Model selection

Fig. 3 shows the confidence intervals according to (19) for the GP prior for all three outputs and all $P = 4$ model candidates. In this paper, the confidence interval shall provide a measure to compare models by the maximum absolute value of the confidence bounds (19) of the GP describing the discrepancy functions for the respective output δ_{p,z_r}, δ_{p,F_ef}, δ_{p,F_sd} (13)

$$C_{\mathrm{max},\delta_{p,z_\mathrm{r}}} = \max(|C_{\mathrm{l},\delta_{p,z_\mathrm{r}}}|, |C_{\mathrm{u},\delta_{p,z_\mathrm{r}}}|) \tag{20a}$$

$$C_{\mathrm{max},\delta_{p,F_\mathrm{ef}}} = \max(|C_{\mathrm{l},\delta_{p,F_\mathrm{ef}}}|, |C_{\mathrm{u},\delta_{p,F_\mathrm{ef,}}}|) \tag{20b}$$

$$C_{\mathrm{max},\delta_{p,F_\mathrm{sd}}} = \max(|C_{\mathrm{l},\delta_{p,F_\mathrm{sd}}}|, |C_{\mathrm{u},\delta_{p,F_\mathrm{sd}}}|). \tag{20c}$$

The lower C_max, the better the model captures the dynamic behaviour of the system with regard to the respective outputs $z_\mathrm{r,max}$, $F_\mathrm{ef,max}$ and $F_\mathrm{sd,max}$. It can therefore be interpreted as a measure of model adequacy and will be used as a metric to compare the $P = 4$ model candidates in the following.

Comparing the confidence intervals in Fig. 3a it can be seen that all models overestimate the maximum relative compression $z_\mathrm{r,max}$ as the confidence bounds for all models are located in the negative domain. Model 2 and model 4 have similar confidence bounds and the values of $C_{\mathrm{max},\delta_{2,z_\mathrm{r}}}$ $C_{\mathrm{max},\delta_{4,z_\mathrm{r}}}$ are highest, while the value of $C_{\mathrm{max},\delta_{3,z_\mathrm{r}}}$ for model 3 is lowest. Therefore, according to the proposed metric, model 3 appears to be the most adequate model with respect to the output $z_{r,max}$.

In Fig. 3b all models overestimate the maximum elastic foot force $F_\mathrm{ef,max}$ and the maximum force in the spring damper component $F_\mathrm{sd,max}$, as again the confidence intervals for the discrepancy functions δ_{p,F_ef}, δ_{p,F_sd} are located in the negative domain. For the outputs $F_\mathrm{ef,max}$ and $F_\mathrm{sd,max}$, model 1 is unequivocally the most adequate, as the values of $C_{\max,\delta_{1,F_{\text{ef}}}}$ and $C_{\max,\delta_{1,F_{\text{sd}}}}$ are clearly lowest. In comparison to model 1, model 3 and especially model 2 and 4 exhibit substantially higher values for $C_{\max,\delta_{3,F_{\text{ef}}}}$, $C_{\max,\delta_{3,F_{\text{sd}}}}$, $C_{\max,\delta_{2,F_{\text{ef}}}}$, $C_{\max,\delta_{2,F_{\text{sd}}}}$, $C_{\max,\delta_{4,F_{\text{ef}}}}$, $C_{\max,\delta_{4,F_{\text{sd}}}}$, respectively, indicating a higher level of model inadequacy.

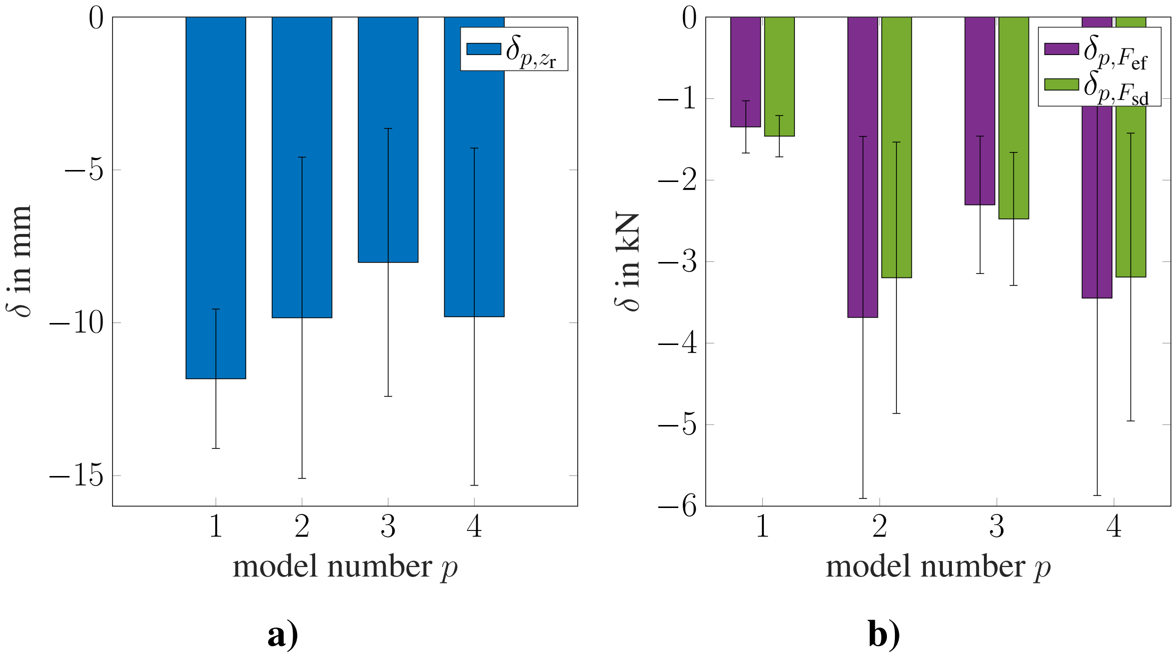


Figure 3: Mean scale β and confidence intervals (19) for the discrepancy function δ for the $P = 4$ model candidates: (a) for model output $z_{\text{r,max}}$, (b) for model outputs F_{ef} and F_{sd}.

Overall, there is no model that consistently ranks first for all regarded outputs after application of the proposed metric. However, model 3 and model 1 showed the highest level of adequacy for the outputs $z_{\text{r,max}}$ in Fig. 3a and $F_{\text{ef,max}}$, $F_{\text{sd,max}}$ in Fig. 3b, respectively. In a model selection process, both models could be pre-selected. In order to advance a model selection it would be advisable to repeat this investigation for supplementary outputs. Also, individual technical requirements should also play a role in the selection process, as this might give more relevance to the results for one or the other output.

The fact, that the discrepancy function exhibits a trend that all models overestimate the dynamic load of the system could be an indication that the initial conditions (8) used to simulate the system are inadequate and do not coincide sufficiently with the initial excitation the system actual experiences during measurements. Friction effects between the frame and the guidance rails (Fig. 1a), that have not been modelled, could lead to a reduction the velocity v_0 at impact, which would shift the simulated outputs $z_{\text{r,max}}$, $F_{\text{ef,max}}$, $F_{\text{sd,max}}$ closer to the observed values and thereby moving the confidence bounds of the discrepancy $\delta_{p,z_{\text{r,max}}}$, $\delta_{p,F_{\text{ef,max}}}$, $\delta_{p,F_{\text{sd,max}}}$ closer to zero. Additional measurements are required to verify this presumption.

5 CONCLUSION

A measure to assess model form uncertainty to assist in the selection of competing models systems is presented. First, GP are fitted to the difference between model outputs and measurements. With the hyperparameters of the GP, confidence intervals for the discrepancy are constructed and compared. Preference is given to the models with the smallest maximum absolute value of the confidence bounds.

ACKNOWLEDGMENTS

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Projektnummer 57157498 – SFB 805.

REFERENCES

[1] Atamturktur, S.; Hemez, F. M.; Laman, J. A. (2012): Uncertainty quantification in model verification and validation as applied to large scale historic masonry monuments. In: *Engineering Structures* 43, S. 221–234. DOI: 10.1016/j.engstruct.2012.05.027.

[2] Roy, Christopher; Oberkampf, William (2010): A Complete Framework for Verification, Validation, and Uncertainty Quantification in Scientific Computing (Invited). In: 48th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition. 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition. Orlando, Florida.

[3] Mthembu, Linda; Marwala, Tshilidzi; Friswell, Michael I.; Adhikari, Sondipon (2011): Model selection in finite element model updating using the Bayesian evidence statistic. In: *Mechanical Systems and Signal Processing* 25 (7), S. 2399–2412. DOI: 10.1016/j.ymssp.2011.04.001.

[4] Mallapur, Shashidhar; Platz, Roland (2019): Uncertainty quantification in the mathematical modelling of a suspension strut using Bayesian inference. In: *Mechanical Systems and Signal Processing* 118, S. 158–170. DOI: 10.1016/j.ymssp.2018.08.046.

[5] Kennedy, Marc C.; O'Hagan, Anthony (2001): Bayesian calibration of computer models. In: *J Royal Statistical Soc B* 63 (3), S. 425–464. DOI: 10.1111/1467-9868.00294.

[6] Locke, Robert; Kupis, Shyla; Gehb, Christopher M.; Platz, Roland; Atamturktur, Sez: Applying Uncertainty Quantification to Structural Systems: Parameter Reduction for Evaluating Model Complexity: *Proceedings of the 37th IMAC, A Conference and Exposition on Structural Dynamics 2019*, Springer International Publishing, 2019. *Accepted and in press.*

[7] Smith, Ralph C. (2014): Uncertainty quantification. Theory, implementation, and applications. *Philadelphia: siam Society for Industrial and Applied Mathematics (Computational science & engineering, 12).*

[8] Rasmussen, Carl Edward; Williams, Christopher K. I. (2008): Gaussian processes for machine learning. 3. print. *Cambridge, Mass.: MIT Press* (Adaptive computation and machine learning).

[9] Soize, C. (2000): A nonparametric model of random uncertainties for reduced matrix models in structural dynamics. In: *Probabilistic Engineering Mechanics* 15 (3), S. 277–294. DOI: 10.1016/S0266-8920(99)00028-4.

[10] Lévesque, Julien-Charles (2018): Bayesian Hyperparameter Optimization: Overfitting, Ensembles and Conditional Spaces. *PhD thesis*. Université Laval, Québec, Canada.

[11] Frazier, Peter I. (2018): A Tutorial on Bayesian Optimization. Online http://arxiv.org/pdf/1807.02811v1.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

INFLUENCE OF DIFFERENT TYPES OF SUPPORT ON EXPERIMENTALLY DETERMINED DAMPING VALUES

Christian A. Geweth[1], Patrick Langer[1], Ferina Saati[1], Kheirollah Sepahvand[1] and Steffen Marburg[1]

[1]Technical University of Munich
Chair of Vibroacoustics of Vehicles and Machines
Boltzmannstraße 15
85748 Garching bei München
e-mail: christian.geweth@tum.de

Keywords: Measurment, Damping, Experimental Modal Analysis, Influence of Support

Abstract. *Identifying the source of any discrepancies between a numerical model and experimental data can be a time consuming and costly undertaking. Individual input parameters of a numerical model can only be determined experimentally. Out of the input parameters usually required for computer-aided structural dynamical models, damping is one of the most challenging ones to obtain. It cannot be measured directly and has to be derived from other measured values in the post-processing. Obtained damping value can be sensitive to the method utilized during the post-processing [1, 2]. Furthermore, boundary conditions which are common in numerical models like ideally free-free or fixed support, can be merely approximated in an experimental setup [3]. As a consequence, the influence of a chosen type of support on the damping of the test specimen cannot be neglected. In this study, the influence of different types of support on the obtained damping values are investigated. For this purpose, the structural dynamical behaviour of several test specimens, each being under several boundary conditions is measured with a laser scanning vibrometer. Each specimen under each boundary condition has been performed out several times in order to identify the reliability and reproducibility of the measurement. From the insights gained in the course of these investigations, suggestions are proposed for the reduction of the measurement effort in damping measurements*

1 INTRODUCTION

The purpose of a numerical model in the field of engineering is to predict the behaviour of a real structure. Hence, the reliability and precision of the model are keys for the usefulness of a numerical model. In order to improve the quality of a numerical model, geometrical and material properties describing the structure in question should have a sufficient precision. Minor changes in the geometry can lead to a noticeable difference in the natural frequencies [4]. Fortunately, several different methods are described in the literature [5, 6] which allow to determine the actual geometry quite reliable. Furthermore, it is relatively easy to determine the mass of a structure for normal problems in engineering with a more than sufficient precision. In case of a homogeneous isotropic material, a realistic density value can be calculated.

In comparison to the accuracy weighing and geometrical measurements, the uncertainty in measuring the Young's modulus is quite large. This even applies to techniques which are known as highly accurate like ultrasonic measurements [7, 8] or inverse modal analysis [9, 10]. The magnitude of uncertainties has been investigated in the recently published literature by Langer et.al [11]. He stated that the margin of error in measuring the Young's modulus of steel specimens is about $\pm 2.3\%$ and the uncertainties in determining the Poisson's ratio is about $\pm 3.2\%$. In comparison, he stated that the averaged measured uncertainty in determining the density is only $\pm 0.171\%$.

All of the above-mentioned margins of errors can be assumed to be relatively small in comparison to uncertainties regarding damping of structures. In the past, a vast amount of research has been published on this topic. Besides the commonly known viscous damping which was introduced by Rayleigh [12], several different approaches on the energy dissipation due to damping emergered in the literature [13, 14, 15, 16, 17, 18]. Experimentally determined damping values can be influenced, due to joint damping [19], by the applied type of support [20]. A common way of supporting a specimen during testing is to approximate free-free boundary conditions by utilising thin elastic strings [3]. The measurements and results in this publication focus on the influence of the mounting position of strings.

This paper is structured as follows: In section 2, the used specimen is introduced as well as the experimental setup and the applied post processing. The results obtained from the measurements are shown in section 3. The final section contains the conclusions drawn from the results and an outlook for future work.

2 MEASUREMENT

The measurements in this study were performed on an aluminium plate. For the purpose of these measurements, the specimen has been suspended at different points to identify any influence of positioning the suspension on the natural frequency as well as on the obtained damping values. Furthermore, the structural response has been recorded at several positions on the specimen. In order to obtain a-priori knowledge about the anticipated mode shapes and natural frequencies of the plate, a FE-Model has been implemented in Abaqus CAE version 6.14.

2.1 Specimen

Several sets of measurements were performed on the above-mentioned plate made out of aluminium alloy AlMg4,5Mn0,7. This plate has a size of $355mm$ x $255mm$ x $13mm$. Around the edge of the plate are a total of 44 holes with an M10 thread. The used alloy was chosen because the material properties are well known and the structural dynamical behaviour of the homogeneous isotropic material is less inflicted with uncertainties than most anisotropic materials. In order to reduce influences of residual stresses, the plate was milled from a solid block of aluminium.

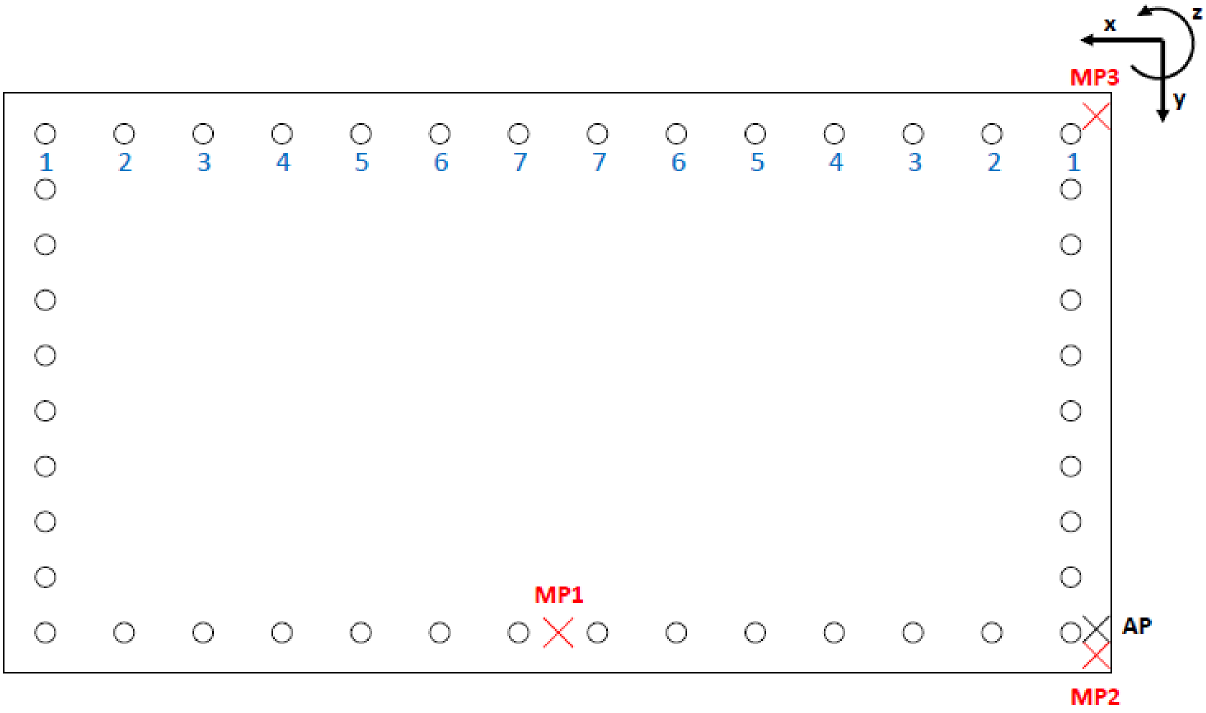


Figure 1: Geometry of the Plate; AP := Point of Excitation; MP := Measurement Point

In figure 1 the schematics of the plate are displayed. The point 'AP' is marking roughly the position of the point of excitation, while the points MP1 to MP3 are marking the positions where the structure's surface velocity has been measured. The blue numbers under the threaded holes are marking different points at which the strings for mounting were attached. Hereinafter, the mounting positions are denoted with the two numbers of the holes, e.g.: '44' means the strings were attached at the two holes marked with a 4.

2.2 Experimental Setup

While choosing an appropriate method of excitation, two key requirements were specified. Firstly, the dynamical behaviour of the specimen should be changed as little as possible by the method of excitation, hence, connecting a shaker to the structure was not an option. Secondly, the excitation should be reproducible and not inflicted by human errors. Since both requirements are fulfilled by an automated impact hammer, the SAM1 produced by NV-Tech-Design was utilised in these measurements [21, 22]. The point of excitation has been placed, as shown in figure 2, in the lower right corner on the back side of the plate. This point has been chosen since the numerical simulations suggested that none of the first ten modes has a nodal line in this point.

Since no additional mass or damping should be added to the specimen, a Laser-Scanning-Vibrometer (Polytec PSV500) was set up to measure the structural response on the impact. Although it is possible to directly transform the measured time data to the frequency domain with the PSV500, this option was consciously disabled.

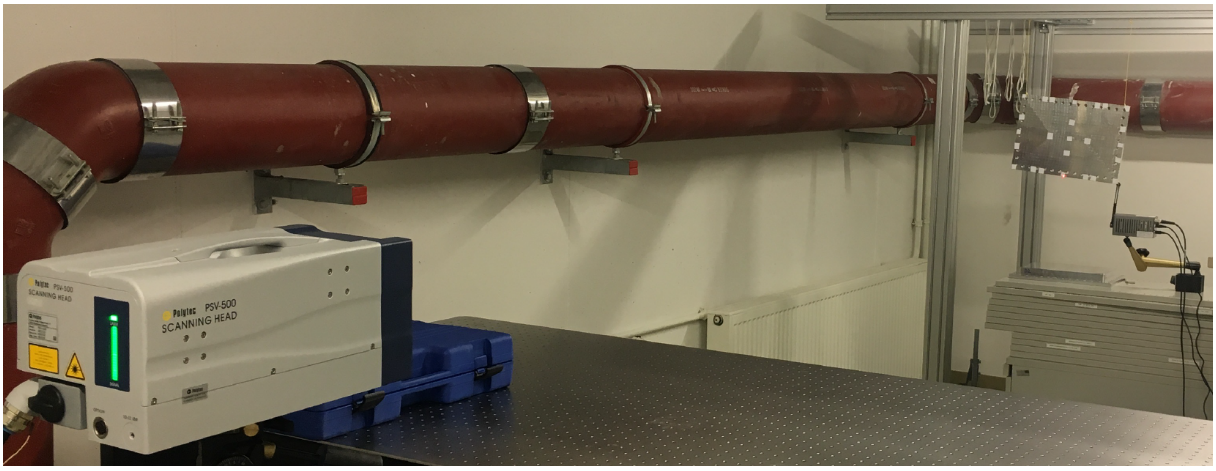


Figure 2: Experimental setup

2.3 Post-Processing

The obtained time data were imported into MATLAB® 2018a. Here, the measured data were checked for possible double-hits or other irregularities in the measurement. Following that, the data were transformed to the frequency domain using the 'fft' function in MATLAB® and exported as *.mat-file. In the second part of the post-processing, these files were imported into ME'SCOPE®. The natural frequencies as well as the damping values were obtained by performing a modal analysis. In order to analyse the influence of the mounting points on the obtained natural frequencies and damping, only the modes which could be clearly identified were further analysed. Hence, in the following chapter, only the modes 2, 3, 6, 7, 8 and 10 are investigated. The damping values and natural frequencies from the three measurement points has been averaged. Unfortunately, the measurement files for the mounting position '33' were corrupted, as a result those are not included in the follow section.

3 RESULTS

The natural frequncies and damping values listed in table 1 are the average values over all measured mounting points for each investigated mode. The natural frequencies of the elastic mode are way higher than the frequencies of the rigid body modes, which could be observed by the naked eye. Furthermore, it is noticeable that the damping of the specimen is quite low.

Mode	Natural frequencies [Hz]	Damping [%]
2	546.2	0.0092
3	1077.5	0.0146
6	1202.8	0.0701
7	1506.4	0.0495
8	2203.0	0.0721
10	2908.4	0.0576

Table 1: Average natural frequency and damping for each investigated mode

The deviation for each mounting position from the average value is calculated for the natural frequencies by equation 1 and for the damping by equation 2. In those two equations, f_{nn} and

ξ_{nn} describe the natural frequency and damping respectively for each mounting point. In figure 3 Δf and $\Delta\xi$ are multiplied by 100 in order to display the deviation in percentage.

$$\Delta f = \frac{f_{nn} - f_{mean}}{f_{mean}} \tag{1}$$

$$\Delta\xi = \frac{\xi_{nn} - \xi_{mean}}{\xi_{mean}} \tag{2}$$

As can been seen in the left graph of figure 3 the natural frequencies for the six investigated modes deviate by less than $\Delta f < \pm 0.05\%$. Even in case of Mode 3, which has the largest differences between the highest obtained natural frequency 1077.9Hz at mounting position '44' and the lowest obtained eigenfrequency 1077.1Hz at mounting position '11' could be considered negligibly small.

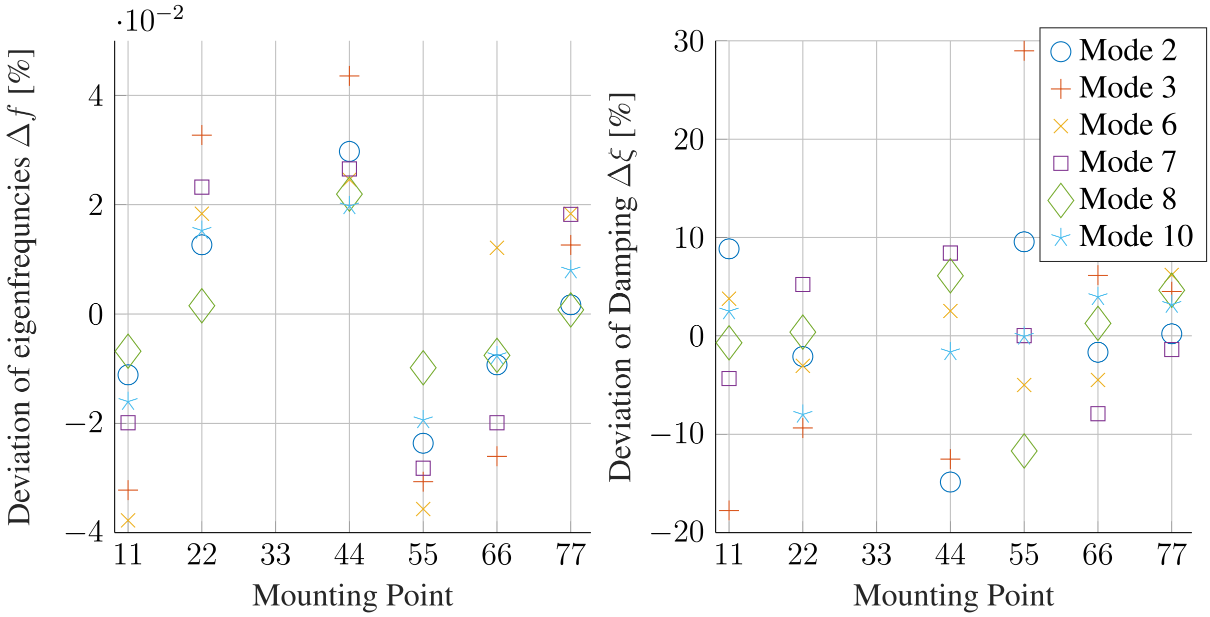


Figure 3: Deviation of eigenfrequencies (left) and damping (right)

The deviation of the obtained damping values, which are displayed in the right graphic of figure 3, variates less than $\Delta\xi < \pm 30\%$. Since the only parameter changed between the measurements were the mounting positions of the strings for the 'free-free' boundary condition, one way of interpreting these results would be that the damping of a structure is highly sensitive to the applied boundary conditions. This would mean, that the necessary effort to experimentally determine realistic damping values would be quite large in comparison to the effort necessary for the natural frequencies. Furthermore, this would mean that especially for light weight structures, it is impossible to identify reliable damping values with sensors, which are mounted directly onto the specimens, i.e. accelerometers. Since aluminium from which the investigated structure has been made from is known to be only lightly damped [23], the previous statements cannot necessarily be transferred on structures with a higher damping.

Another way of interpreting the obtained results is that the observed deviations of the obtained damping values is within the scope of the uncertainties of the applied post-processing. Previous research on simplified models using the same methods as in this paper, showed that the applied post-processing is able to produce reliable results [1]. Nevertheless, the question of the reliability of damping value for each mounting position remains interesting in order to quantify the influence of the applied type of support.

4 CONCLUSIONS and OUTLOOK

In this paper, the influence of slightly different approaches to approximate free-free boundary conditions in experiments on the obtained natural frequencies and damping is investigated. For that purpose, an aluminium plate has been suspended with the same strings at different points on the plate. The obtained results suggest that natural frequencies are hardly influenced by the mounting position of the strings. In contrast, the obtained damping values seemed to be quite sensitive to the mounting position.

Future research focus on the reproducibility and variance of measured damping values for a single mounting position. As a result, it could be possible to quantify the influence of the chosen position for mounting a specimen during experiments.

REFERENCES

[1] Geweth, C.A.; Moscoso Cires, R.; Martínez García, A.; Jagodzinski, D.; S. Marburg: Influence of different measurement settings and methods on obtained damping values; Proceedings of NOVEM 2018; Ibiza

[2] Geweth, C.A.; Langer, P.; Sepahvand, K.; Marburg, S.: Investigation of various damping measurement techniques. The Journal of the Acoustical Society of America 141 (5), 2017, 3576-3576

[3] Ewins, David J.; Modal Testing: Theory, Practice and Application, 2nd Edition; Wiley, 2000; ISBN: 978-0-863-80218-8

[4] Geweth, C.; Sepahvand, K.; Marburg, S.: Stochastic radiated sound power of structures with uncertain parameters. Proceedings of NOVEM 2015, Dubrovnik

[5] Reich, C., Ritter, R., Thesing, J.: 3-D shape measurement of complex objects by combining photogrammetry and fringe projection. Optical Engineering, 39(1), S. 224-231, 2000

[6] Petz, M.; Tutsch, R.: Optical 3D Measurement of Reflecting Free Formed Surfaces. In: VDI-Berichte 1694, International Symposium on Photonics in Measurement, Aachen, 2002, S. 329-332

[7] Hislop, J.; Krautkrämer, J.; Krautkrämer, H.; Grabendörfer, W.; Frielinghaus, R.; Kaule, W.; Niklas, L.; Opara, U.; Schlengermann, U.; Steiger, H.; et al.; Ultrasonic Testing of Materials. Springer Berlin Heidelberg, 2013.

[8] Achenbach, J.; Wave Propagation in Elastic Solids, ser. North-Holland Series in Applied Mathematics and Mechanics. Elsevier Science, 2012.

[9] Sepahvand, K.; S. Marburg, S.;, Non-sampling inverse stochastic numericalexperimental identification of random elastic material parameters in composite plates, Mechanical Systems and Signal Processing, vol. 54, no. Supplement C, pp. 172-181, 2015.

[10] Stache, M.; Guettler, M.; Marburg, S.; A precise non-destructive damage identification technique of long and slender structures based on modal data, Journal of Sound and Vibration, vol. 365, pp. 89-101, 2016.

[11] Langer, P.; Sepahvand, K.; Guist, C.; Bär, J.; Peplow, A.; Marburg, S.: Matching experimental and three dimensional numerical models for structural vibration problems with uncertainties. Journal of Sound and Vibration 417, pp. 294-305, 2018.

[12] Rayleigh, J. W. S. B. The Theory of Sound, vol. 1. Macmillan, 1877.

[13] Woodhouse, J. Linear damping models for structural vibration. Journal of Sound and Vibration 215, 3 (1998), 547569.

[14] Beerens, C. Zur Modellierung nichtlinearer Dämpfungsphänomene in der Strukturmechanik. PhD thesis, Institut für Mechanik Ruhr-Universität Bochum, 1994.

[15] Adhikari, S. Damping models for structural vibration. PhD thesis, University of Cambridge, 2001.

[16] Adhikari, S., and Woodhouse, J. Quantification of non-viscous damping in discrete linear systems. Journal of Sound and Vibration 260, 3 (2003), 499518.

[17] Yamaguchi, H., and Adhikari, R. Energy-based evaluation of modal damping in structural cables with and without damping treatment. Journal of Sound and Vibration 181, 1 (1995), 7183.

[18] Osinski, Z.: Damping of vibrations. CRC Press, 2018.

[19] Mayer, M. Zum Einfluss von Fgestellen auf das dynamische Verhalten zusammengesetzter Strukturen. PhD thesis, Universitt Stuttgart, 2007.

[20] Barkanov, E.; Skukis, E.; Petitjean, B.: Characterisation of viscoelastic layers in sandwich panels via an inverse technique. Journal of sound and vibration, 327(3-5), pp. 402-412, 2009.

[21] Blaschke, P.; Mallareddy, T.T.; Alarcn, D.J.: Application of a scalable automatic modal hammer and a 3D scanning laser Doppler vibrometer on turbine blades. Proceedings of the 4th VDI conference in vibration analysis and identification, VDI-Berichte 2259, Fulda, p. 87, 2016.

[22] Blaschke, P.; Schneider, S.; Kamenzky, R.; Alarcn, D.J.: Non-linearity Identification of Composite Materials by Scalable Impact Modal Testing. In: Sensors and Instrumentation, Volume 5. Springer, Cham, pp. 7-14. 2017.

[23] Petersen, C.; Werkle, H.: Dynamik der Baukonstruktionen. 2nd Edition; Springer Vieweg; 2017; ISBN: 978-3-8348-1459-3.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

STATISTICAL HOMOGENIZATION OF RANDOM POROUS MEDIA

Marco Pingaro[1], **Emanuele Reccia**[2], **Patrizia Trovalusci**[1] **and Maria Laura De Bellis**[3]

[1] Sapienza University of Rome, Department of Structural and Geotechnical Engineering
Via Gramsci 53, 00197, Rome, Italy
e-mail: {marco.pingaro,patrizia.trovalusci}@uniroma1.it

[2] University of Cagliari, Department of Civil and Environmental Engigneering and Architecture
Via Marengo 2, 09123, Cagliari, Italy
e-mail: emanuele.reccia@unica.it

[3] Gabriele dAnnunzio University, Department of Engineering and Geology
Viale Pindaro 42, 65122, Pescara, Italy
e-mail: marialaura.debellis@unich.it

Keywords: Statistically Homogenization, Porous Media, Virtual Element Method.

Abstract. *In recent times, the scientific community paid great attention to the influence of inherent uncertainties on system behavior and recognize the importance of stochastic and statistical approaches to engineering problems [21]. In particular, statistical computational methods may be useful to the constitutive characterization of complex materials, such as composite materials characterized by non-periodic internal micro-structure. Random porous media exhibit a microstructure made of randomly distributed pores embedded into a continuous matrix. They can be modelled as a bi-material system in which circular soft inclusions (pores) with random distribution and variable diameters are dispersed in a stiffer matrix. A key aspect, recently investigated by many researchers, is the evaluation of appropriate mechanical properties to be adopted for the study of their behaviour. Differently from classical homogenization approaches, in the case of materials with random microstructure it is not possible to 'a-priori' define a Representative Volume Element (RVE), this being an unknown of the problem. Statistical homogenization procedures may be adopted for the definition of equivalent moduli able to take into account at the macroscale the material properties emerging from the internal microstructure with random distribution [26]. Here, a Fast Statistical Homogenization Procedure (FSHP) based on Virtual Element Method (VEM) approach for the numerical solution – previously developed by some of the authors [13] has been adopted for the definition of the Representative Volume Element (RVE) and of the related equivalent elastic moduli of random porous media with different volume fraction, defined as the ratio between mechanical properties of inclusions and matrix. In particular, FSHP with virtual Elements of degree 1 [2] for modelling the inclusions provides reliable results for materials with low contrast.*

1 INTRODUCTION

The characterization of porous materials is increasingly attracting the attention of researchers and engineers due to the widespread use of this class of materials in different fields and industries, including aerospace, automotive, energy, construction, electronics and biomedical. Focusing the attention on porous metals, porous ceramics and polymer foams, the main common features are low density, large specific surface and a range of novel properties ranging from physical, mechanical, thermal, electrical up to acoustical fields [10]. In addition, a distinctive characteristic of such heterogeneous materials is the random distribution of voids within a dense solid. This results in a composite two-phase material whose overall elastic properties depend both on the geometrical nature of the pores - shape and size - and on the value of porosity.

Different approaches have been proposed in literature to handle the study of porous materials, among others we refer to generalized continua in which voids or microcracks are modelled as additional degrees of freedom [22, 28, 27, 20], multifield [14, 6, 7, 8, 19] and/or multiscale descriptions [25, 26].

This paper is devoted to the investigation of porous materials within the framework of linear elasticity. A fast statistical homogenization procedure (FSHP) is adopted to grasp the global elastic behaviour of such a random composite, as in [13]. In FSHP the statistical procedure, proposed in [26], is automatized and integrated in a completely in house specifically developed code implemented to quickly and efficiently perform a high number of parametric analyses.

The material is modelled considering disk shaped soft inclusions randomly distributed within a base stiffer matrix. A first order computational homogenization procedure is exploited within the very recent Virtual Element Method (VEM) [4, 5, 13, 12] that allows us to reliably model porous material and to efficiently solve high number of simulations as required in homogenization techniques applied to random materials [15, 1, 3]. The main point of the procedure is to approach the so–called Representative Volume Element (RVE) using finite–size scaling of Statistical Volume Elements (SVEs). To this end properly defined Dirichlet and Neumann-type boundary value problems are numerically solved on the SVEs defining hierarchies of constitutive bounds.

A set of materials with different porosity is analysed to characterize overall mechanical parameters of porous materials and to investigate their sensitivity to porosity. It emerges that, on the one hand FSHP provides reliable results for the homogenization of porous materials. On the other hand, the choice of virtual elements of degree 1 is perfectly suitable to the case at hand, as shown in [13]. Moreover, both homogenized values of bulk modulus and Poisson coefficient decrease as the porosity increases.

2 FAST STATISTICAL HOMOGENIZATION PROCEDURE

We consider a two–dimensional linear elastic framework and describe, at the microscopic scale, the heterogeneous porous material as a bi–phase system. In two-phase materials it is useful to define the material contrast as the ratio between the elastic moduli of inclusions, E_i, and matrix, E_m, $c = E_i/E_m$. When $0 < c < 1$ ($c = 1$ being the case of a homogeneous material) inclusions are softer than the matrix and we refer to low contrast materials, that are suitable to properly represent porous media [18]. The pores are modelled as circular inclusions of diameter d dispersed into continuous matrix. Furthermore a scale parameter $\delta = L/d$ is introduced, that is the ratio between the side of a square test window L, and the diameter d of the inclusions.

The homogenization procedure for defining the constitutive response of random heterogeneous material requires the definition of the size of a Representative Volume Element (RVE) larger than the microscale characteristic length, corresponding to the diameter of inclusions, d. According to the approach presented in [26], that is based on the approach proposed in [9, 11], the presented procedure requires the statistical definition of a number of realizations, called Statistical Volume Elements (SVEs), of the possible microstructure, sampled in a Monte Carlo sense, which allows determining series of scale–dependent upper and lower bounds for the overall elastic moduli and to approach the RVE size, δ_{RVE}, using a statistical criterion to stop the procedure when the results in terms of average elastic moduli do not change within the selected tolerance interval. All the steps of the homogenization procedure are completely integrated in the so–called Fast Statistical Homogenization Procedure (FSHP), based on the statistical homogenization procedure previously developed in [26]. See [13] for details on FSHP procedure.

The convergence criterion is fulfilled when the values of the homogenized constitutive coefficients are distributed around their averages with a vanishing variation coefficient. This means that the RVE size is achieved. The effective constitutive moduli are consistently estimated as the mean values between the Dirichlet (upper) and Neumann (lower) bounds at the convergence window (RVE). This circumstance also corresponds to reaching the minimum window size δ_{RVE} for which the estimated homogenized moduli remain constant, within a tolerance interval less than 0.5% for both the Dirichlet and Neumann solutions. The minimum number of simulations, N^{lim}, and the tolerance parameter, Tol, are chosen in order to define a narrow confidence interval for the average and to obtain a reliable convergence criterion. The choice is discretionary, values are assumed depending on the data dispersion. All these details are specified in [26].

It is worth noting that FSHP permits the automatic cutting of the inclusions over the limit of the windows' edges, accounting for the presence of inclusions that randomly cross the windows' edges, as required by the randomness of the medium. It is worth noting that to neglect the presence of inclusions that intersect the windows' edges, a less realistic hypothesis widely used in literature, provides results significantly different from the results obtained by taking into account cut inclusions at the windows' edges [25]. Furthermore, the mesh corresponding to each realization of the microstructure has been optimized using VEM methodology, that permits to adopt single virtual element for the inclusions (reduction of degrees of freedom) and triangular virtual elements for the matrix. In order to take into account the stress gradients in the so–called 'hard core regions', the mesh is fine near the inclusions and coarse away from the inclusions. The FSHP allow us to very efficiently solve the series (hundred) of BVPs and to rapidly converge to the RVE solution.

3 VIRTUAL ELEMENT METHOD

In this section we briefly recall the weak formulation of the classical 2D linear elastic problem and describe the related virtual element space [4, 5], as well as the construction of the bilinear form resulting from the weak form. The vectorial notation is adopted in the following, that is suited to the proposed formulation.

We consider a body immersed in the two–dimensional space $\mathbf{R}^2$, where the Cartesian coordinate system (O, x, y) is introduced. The body is subjected to the volume force, represented by the vector $\boldsymbol{f} \in (L^2(\Omega))^2$, $\boldsymbol{f} = \{f_1,\, f_2\}^T$ (within the standard Lebesgue space), and given boundary conditions. In the sake of simplicity we use homogeneous Dirichlet boundary conditions and consider the Sobolev space, $\boldsymbol{V} := \left(H^1_0(\Omega)\right)^2$, of the admissible displacement fields, represented by the vector $\boldsymbol{v}$.

Furthermore we represent the strain, under the hypothesis of small deformations, as the vector $\boldsymbol{\varepsilon} = \{\varepsilon_{11},\, \varepsilon_{22},\, \varepsilon_{12}\}^T$ associated to the displacement field vector $\boldsymbol{u} = \{u_1,\, u_2\}^T$:

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \boldsymbol{S}\,\boldsymbol{u} \quad \text{with } \boldsymbol{S} = \begin{bmatrix} \partial_x & 0 \\ 0 & \partial_y \\ \partial_y & \partial_x \end{bmatrix}\,, \tag{1}$$

where $\partial_{(\cdot)}$ denotes the partial derivative with respect to the $(\cdot)$-coordinate.

The weak form of the linear elastic problem reads:

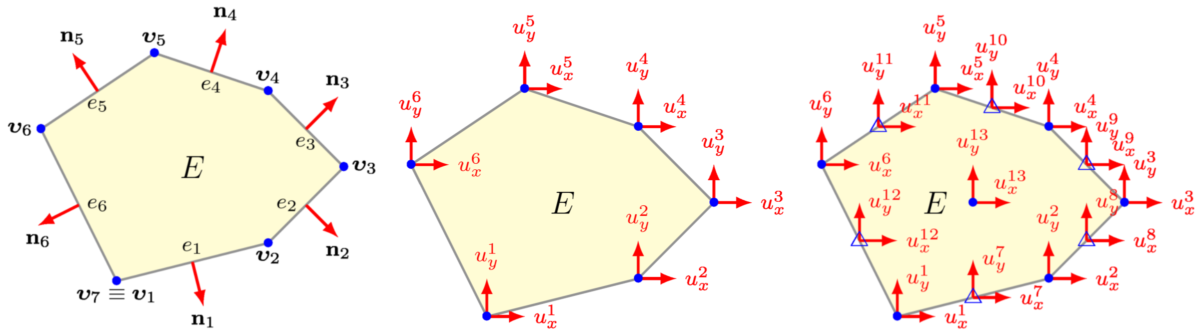
$$\begin{cases} \text{Find } \boldsymbol{u} \in \boldsymbol{V} \text{ such that :} \\ a(\boldsymbol{u}, \boldsymbol{v}) =< \boldsymbol{f}, \boldsymbol{v} > \quad \forall \boldsymbol{v} \in \boldsymbol{V} \end{cases} \tag{2}$$

where:

$$\begin{aligned} & a(\boldsymbol{u}, \boldsymbol{v}) = \textstyle\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{v})^T \boldsymbol{C}\, \boldsymbol{\varepsilon}(\boldsymbol{u})\, d\Omega = \int_\Omega (\boldsymbol{S}\boldsymbol{v})^T\, \boldsymbol{C}\, \boldsymbol{S}\boldsymbol{u}\, d\Omega\,, \\ & < \boldsymbol{f}, \boldsymbol{v} >= \textstyle\int_\Omega \boldsymbol{f}^T\, \boldsymbol{v}\, d\Omega\,, \end{aligned} \tag{3}$$

and $\boldsymbol{C} = \boldsymbol{C}(\boldsymbol{x})$ is the plane elastic tensor (uniformly positive) and possibly depending on the position vector $\boldsymbol{x} = (x, y)^T \in \Omega$.

In order to approximate the solution of the problem (2) we consider a decomposition $\mathcal{T}_h$ of the domain Ω into non overlapping polygonal elements E. In the following, we denote by e the straight edges of the mesh $\mathcal{T}_h$ and, for all $e \in \partial E$, $\boldsymbol{n}_i$ denotes the outward unit normal vector to e_i (Fig. 1(a)). The symbol n_e represents the number of the edges of the polygon E, that coincides with the number of the element vertices.



(a) (b) (c)

Figure 1: Example of virtual element, E, with six edges and seven vertices: relative nodes and edges numeration (a) and degrees of freedom of the virtual element of degree 1 (b) and 2 (c).

Let k be an integer ≥ 1. Let us denote by $P_k(\Omega)$ the space of polynomials, living on the set $\Omega \subseteq \mathbb{R}^2$, of degree less than or equal to k.

By the discretization introduced, it is possible to write the bilinear form (2), as in the finite element methodology, in the following way:

$$a(\boldsymbol{u}, \boldsymbol{v}) = \sum_{E \in \mathcal{T}_h} a^E(\boldsymbol{u}, \boldsymbol{v}) \quad \forall \boldsymbol{v} \in \boldsymbol{V}\,. \tag{4}$$

The discrete virtual element space, $\boldsymbol{V}_h$, is:

$$\boldsymbol{V}_h := \left\{ \boldsymbol{v} \in \boldsymbol{V} \;:\; \boldsymbol{v}\,|_E \in \boldsymbol{V}_{h|E}\, \forall E \in \mathcal{T}_h \right\}\,, \tag{5}$$

where $\boldsymbol{V}_{h|E} := \left(V_{h|E}\right)^2$ and the local space $V_{h|E}$ is defined as

$$V_{h|E} := \left\{\boldsymbol{v}_h \in H^1(E) \cap C^0(E) : \triangle\boldsymbol{v}_h \in P_{k-2}(E),\ \boldsymbol{v}_h \mid_e \in P_k(e)\, \forall e \in \partial E\right\} . \tag{6}$$

By the definition of the local space (6), we can observe that, in contrast to the standard finite element approach, the local space $\boldsymbol{V}_{h|E}$ is not fully explicit, in fact $\boldsymbol{V}_{h|E}$ contain all the polynomials of degree $\leq k$, plus other functions that, in general, will not be polynomials. Moreover $\boldsymbol{v}_h$ is a polynomial of degree k on each edge e of E and globally continuous on ∂E.

The related degrees of freedom for the space $\boldsymbol{V}_{h|E}$ (Figs. 1(b),1(c)) are:

- $2n_e$ point–wise values $\boldsymbol{v}_h(\upsilon_i)$ $i = 1, 2, \ldots, n_e$, where υ_i is the i-th corner of E;
- $2n_e(k-1)$ point–wise values $\boldsymbol{v}_h(y_j^e)$, where $\left\{y_j^e\right\}$, $i = 1, \ldots, k-1$ are the edge internal nodes;
- $k(k-1)$ scalar moments of the unknown field over the element (i.e. $\frac{1}{|E|}\int \boldsymbol{v}_h$, where $\mid E$ is the area of the element), not associated with a specific location over E.

The global dimension of the space $\boldsymbol{V}_{h|E}$ then is $m = \dim(\boldsymbol{V}_{h|E}) = 2n_e k + k(k-1)$

The problem (2) restricted to the discrete space $\boldsymbol{V}_h$ becomes:

$$\begin{cases} \text{Find } \boldsymbol{u}_h \in \boldsymbol{V_h} \text{ such that} \\ a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) =< \boldsymbol{f}, \boldsymbol{v}_h > \quad \forall \boldsymbol{v}_h \in \boldsymbol{V_h}\ , \end{cases} \tag{7}$$

where $a_h(\cdot,\cdot) : \boldsymbol{V}_h \times \boldsymbol{V}_h \to \mathbb{R}$ is the discrete bilinear form approximating the continuous form $a(\cdot,\cdot)$ and, $< \boldsymbol{f}, \boldsymbol{v}_h >$ is the term approximating the virtual work of external load.

The discrete bilinear form is constructed element by element as:

$$a_h(\boldsymbol{u}_h, \boldsymbol{v}_h) = \sum_{E \in \mathcal{T}_h} a_h^E(\boldsymbol{u}_h, \boldsymbol{v}_h) \quad \forall \boldsymbol{u}_h,\ \boldsymbol{v}_h \in \boldsymbol{V}_h\ . \tag{8}$$

The local stiffness matrix can be derived, consistently with [5], after introducing the local projector operator $\Pi_E^\nabla : \boldsymbol{V}_h(E) \to (\mathbb{P}_k(E))^2$. The details on the construction of the local stiffness matrix are presented in [13], while are here omitted for the sake of brevity.

4 NUMERICAL SIMULATION: SENSITIVITY TO POROSITY

In this section, the Fast Statistical Homogenization Procedure (FSHP) based on the low order virtual elements is applied to the analysis of porous materials, modelled as a two–dimensional two–phase material in which circular soft inclusions are randomly embedded in a stiffer continuous matrix.

FSHP with virtual element of degree $k = 1$ is particularly suitable for analysing low contrast materials [13], and in this work a very low value of contrast ($c \to 0$) is adopted, with the purpose of simulating a material with randomly distributed voids. Referring to the properties adopted in [18], an high value of Poisson coefficient has been adopted for the inclusions. All the mechanical properties are reported in Table 1, where E_m and ν_m are Young and Poisson modulus for the matrix, and E_i and ν_i are Young and Poisson modulus for the inclusions, respectively.

By exploiting the opportunities provided by FSHP, several parametric analyses have been performed by varying the porosity, ρ, of random porous media in the range $0\% \div 40\%$ (Fig. 2). The purpose is twofold: on one hand for identifying the RVE size,δ_{RVE}, and its changes in

E_m	E_i	c	ν_m	ν_i
10000	1	10^{-4}	0.30	0.49

Table 1: Mechanical properties adopted

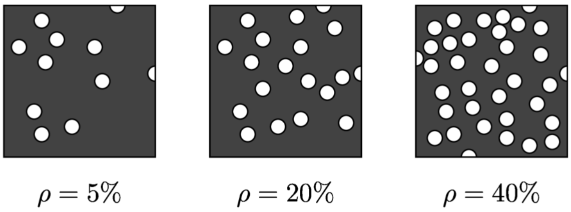


Figure 2: Realizations of micro-structure for different levels of porosity

relation of porosity; on the other hand, to evaluate the sensitivity to porosity of the mechanical properties of homogeneous equivalent continuum materials. The performed analyses allowed us to identify RVE size, δ_{RVE}, and the corresponding material properties. As the homogenized material has found to be essentially isotropic, we focus the attention only on the bulk modulus, as representative material parameter. The results of the parametric analyses performed are reported in the following figures.

Basing on the works [16] and [23] we define, the following constitutive scaling measures are:

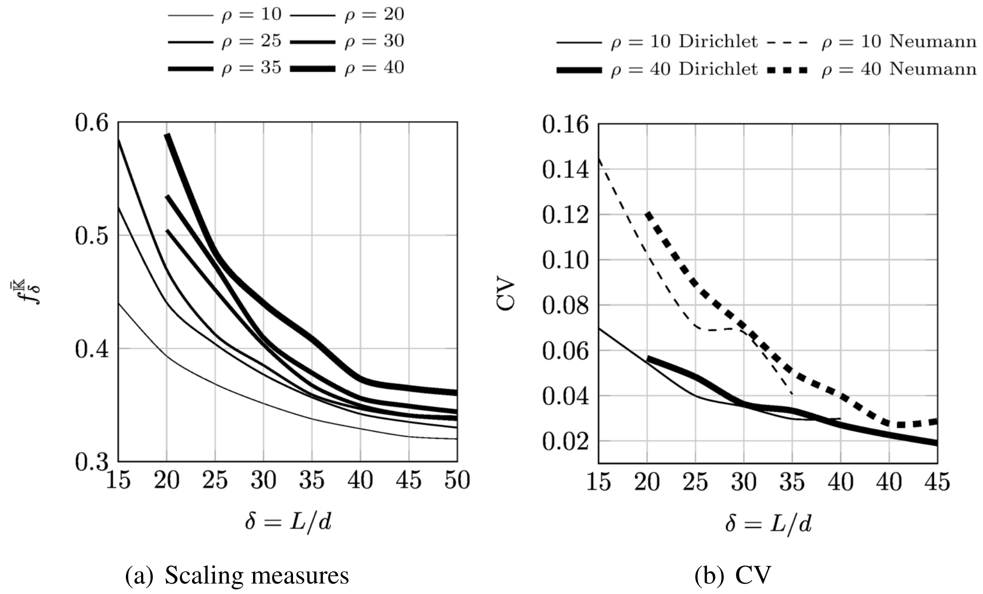
$$f_\delta^{\overline{\mathbb{K}}} = \frac{\overline{\mathbb{K}}_\delta^D}{\overline{\mathbb{K}}_\delta^N} - 1 \, , \tag{9}$$

where $\overline{\mathbb{K}}_\delta^D$ and $\overline{\mathbb{K}}_\delta^N$ are the average of the bulk moduli obtained by solving the boundary value problems (BVPs) by adopting Dirichlet and Neumann–type boundary conditions (BCs), respectively.

Fig.(3(a)), provides a qualitative and quantitative information about the convergence trend of the solution,$f_\delta^{\overline{\mathbb{K}}}$, by varying the window size, δ, and the pore density, ρ. The differences in terms of convergence trend between the two BC solutions depend on the different dispersion of results, as shown in Fig. 3(b), where the Coefficient of Variation, CV, is also plotted for several values of porosity ρ for Dirichlet and Neumann BVPs. Fig. 4(a) summarizes the results of the parametric analyses in term of convergence bulk modulus $\overline{\mathbb{K}}$, both for Dirichlet and Neumann BVPs, versus the porosity ρ.

As expected, the value of $\overline{\mathbb{K}}$ decreases as the porosity increases [15]. Apparently the two bounds tend to become closer as ρ increases, however the percentage difference between the bounds and the mean value remains almost constant.

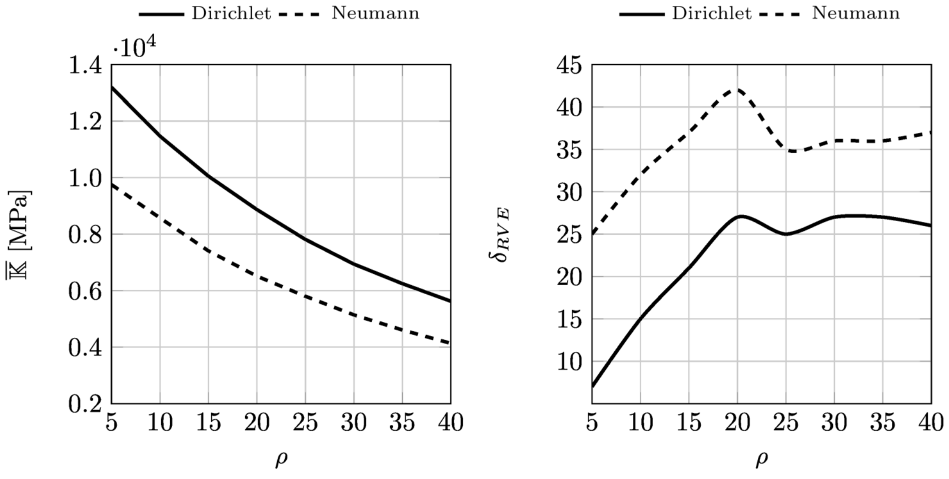
Fig. 4(b) reports the values of RVE size obtained for $\delta = \delta_{RVE}$ plotted versus the different values of porosity ρ. It is worth noting that in the case of $\rho = 0$ the material is homogeneous, starting from $\rho \geq 0$ it is necessary to determine a RVE and, as expected, as porosity increases larger δ_{RVE} are needed. However, it has been found that the δ_{RVE} rapidly increases in the range of porosity $1 \div \rho \div 20$, while slowly decreases up to $\rho = 25$ and then remain almost constant in the range $25 \div \rho \div 40$. The results in terms of RVE size, δ_{RVE}, are synthesized and reported in As previously noticed, Dirichlet BVP need smaller windows with respect to Neumann BVP, that has to be considered to define δ_{RVE} for porous material.



(a) Scaling measures

(b) CV

Figure 3: (a) Convergence trend of the bulk modulus scaling measure, $f_\delta^{\overline{\mathbb{K}}}$, for different levels of porosity, ρ. (b) Coefficient of Variation CV for Dirichlet and Neumann boundary conditions



(a) Bulk modulus $\overline{\mathbb{K}}$

(b) δ_{RVE}

Figure 4: Normalized average bulk modulus $\overline{\mathbb{K}}$ and window size δ_{RVE} for different levels of porosity ρ .

5 CONCLUSIONS

The present work is an application of the Fast Statistical Homogenization Procedure (FSHP) [13] to the homogenization of random porous materials. The model adopted within the FSHP framework is the bi–phase material in which disk shaped soft inclusions are randomly distributed in a stiffer matrix. FSHP uses the Virtual Element technique to model the inclusions with one element, this permit to reduce the number of degrees of freedom with consequently increasing the computational efficiency. FSHP permits to solve high number of simulations as required in homogenization techniques applied to random materials [25, 26, 24, 17]. Furthermore, FSHP permits to analyse a series of materials with different porosity and to rapidly identify the relative RVE size and the overall effective moduli of the equivalent homogeneous material. Virtual Elements of degree one, adopted in this work, well fits the behaviour of porous materials. The approximation of constant stress and strain, using lower virtual element, in fact does not influence the homogenization procedure [13, 12]. For porous materials the Representative Volume Element (RVE) as a function of the porosity has strongly non–linear behaviour and the RVE increases when the porosity increases, but for higher level of porosity the RVE slightly decreases and then remains constant up to the value $\rho = 40\%$, which is the maximum value of porosity. Furthermore, as expected, the homogenized values of the elastic moduli decrease as the porosity increases.

Acknowledgment

This research was supported by Italian Ministry of University and Research: PRIN 2015, project 2015JW9NJT (B86J16002300001); Sapienza Research Grants 'Grandi Progetti' 2016 (B82F16005920005) and 'Progetti Medi' 2017 (B83C17001440005).
Dr. Reccia fully acknowledges the research project funded by P.O.R. SARDEGNA F.S.E. 2014-2020 - Axis III Education and Training, Thematic Objective: 10, Specific Objective: 10.5, Action of the Partnership Agreement: 10.5.12, Call for Funding of Research Projects - Year 2017.

REFERENCES

[1] M. Arnold, A. R. Boccaccini, and G. Ondracek. Prediction of the poisson's ratio of porous materials. *Journal of Materials Science*, 31(6):1643–1646, 1996.

[2] E. Artioli, L. Beiro Da Veiga, C. Lovadina, and E. Sacco. Arbitrary order 2d virtual elements for polygonal meshes: part i, elastic problem. *Computational Mechanics*, 60(3):355–377, 2017.

[3] M. Asmani, C. Kermel, A. Leriche, and M. Ourak. Influence of porosity on youngs modulus and poisson's ratio in alumina ceramics. *Journal of the European Ceramic Society*, 21(8):1081–1086, 2001.

[4] L. Beiro Da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo. Basic principle of virtual element methods. *Mathematical Models and Methods in Applied Sciences*, 23(01):199–214, 2013.

[5] L. Beiro Da Veiga, F. Brezzi, and L. Marini. Virtual elements for linear elasticity problems. *SIAM Journal on Numerical Analysis*, 51(2):794–812, 2013.

[6] M. A. Biot. General theory of three-dimensional consolidation. *Journal of Applied Physics*, 12(2):155–164, 1941.

[7] O. Coussy. *Mcanique des milieux poreux*. Editions Technip, 1991.

[8] O. Coussy. *Mechanics and Physics of Porous Solids*. Mechanics and Physics of Porous Solids. John Wiley & Sons, Ltd, 2010.

[9] X. Du and M. Ostoja-Starzewski. On the size of representative volume element for darcy law in random media. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 462(2074):2949–2963, 2006.

[10] P. Liu and G.-F. Chen. *Porous materials: processing and applications*. Elsevier, 2014.

[11] M. Ostoja-Starzewski. *Microstructural Randomness and Scaling in Mechanics of Materials*. CRC Press, Taylor & Francis Group, 2007.

[12] M. Pingaro, E. Reccia, and P. Trovalusci. Homogenization of random porous materials with low order virtual elements. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, In print.

[13] M. Pingaro, E. Reccia, P. Trovalusci, and R. Masiani. Fast statistical homogenization procedure (fshp) for particle random composites using virtual element method. *Computational Mechanics*, https://doi.org/10.1007/s00466-018-1665-7 2019.

[14] J. Poutet, D. Manzoni, F. Hage-Chehade, C. J. Jacquin, M. J. Boutca, J. . Thovert, and P. M. Adler. The effective mechanical properties of random porous media. *Journal of the Mechanics and Physics of Solids*, 44(10):1587–1620, 1996.

[15] N. Ramakrishnan and V. S. Arunachalam. Effective elastic moduli of porous ceramic materials. *Journal of the American Ceramic Society*, 76(11):2745–2752, 1993.

[16] S. I. Ranganathan and M. Ostoja-Starzewski. Towards scaling laws in random polycrystals. *International Journal of Engineering Science*, 47(11):1322 – 1330, 2009. Mechanics, Mathematics and Materials a Special Issue in memory of A.J.M. Spencer FRS.

[17] E. Reccia, M. L. De Bellis, P. Trovalusci, and R. Masiani. Sensitivity to material contrast in homogenization of random particle composites as micropolar continua. *Composites Part B: Engineering*, 136:39 – 45, 2018.

[18] B. M. Said, M. Salah, T. Kanit, and F. Kamel. On the homogenization of 2d porous material with determination of rve. *International Journal of Mechanical and Mechatronics Engineering*, 16(1):81–86, 2016.

[19] G. Sciarra, F. dell'Isola, and O. Coussy. Second gradient poromechanics. *International Journal of Solids and Structures*, 44(20):6607–6629, 2007.

[20] V. Settimi, P. Trovalusci, and G. Rega. Dynamical properties of a composite microcrackedbar based on a generalized continuum formulation. *Continuum Mechanics and Thermodynamics*, 2019. https://doi.org/10.1007/s00161-019-00761-7.

[21] G. Stefanou. The stochastic finite element method: Past, present and future. *Computer Methods in Applied Mechanics and Engineering*, 198(9-12):1031–1051, 2009.

[22] P. Trovalusci and G. Augusti. A continuum model with microstructure for materials with flaws and inclusions. *Journal de Physique IV*, 8:383–390, 1998.

[23] P. Trovalusci, M. De Bellis, and M. Ostoja-Starzewski. A statistically-based homogenization approach for particle random composites as micropolar continua. In H. Altenbach and S. Forest, editors, *Generalized Continua as Models for Classical and Advanced Materials*, volume 42 of *Advanced Structured Materials*, pages 425–441. Springer International Publishing, 2016.

[24] P. Trovalusci, M. L. De Bellis, and R. Masiani. A multiscale description of particle composites: From lattice microstructures to micropolar continua. *Composites Part B: Engineering*, 128:164 – 173, 2017.

[25] P. Trovalusci, M. L. De Bellis, M. Ostoja-Starzewski, and A. Murrali. Particulate random composites homogenized as micropolar materials. *Meccanica*, 49(11):2719–2727, Nov 2014.

[26] P. Trovalusci, M. Ostoja-Starzewski, M. L. De Bellis, and A. Murrali. Scale-dependent homogenization of random composites as micropolar continua. *European Journal of Mechanics A/Solids*, 49:396–407, 2015.

[27] P. Trovalusci and V. Varano. Multifield continuum simulations for damaged materials: a bar with voids. *International Journal for Multiscale Computational Engineering*, 9:599–608, 2011.

[28] P. Trovalusci, V. Varano, and G. Rega. A generalized continuum formulation for composite microcracked materials and wave propagation in a bar. *Journal of Applied Mechanics, Transactions ASME*, 77(6), 2010.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

RELIABILITY-BASED DESIGN OPTIMISATION OF A DUCTED PROPELLER THROUGH MULTI-FIDELITY LEARNING

Péter Zénó Korondi[1,2,*], **Lucia Parussini**[2], **Mariapia Marchi**[1], **and Carlo Poloni**[1,2]

[1] ESTECO S.p.A
99 Padriciano, Area Science Park, Trieste, Italy 34149
e-mail: {korondi,marchi}@esteco.com

[2] Department of Engineering and Architecture, University of Trieste
Piazzale Europa 1, Trieste, Italy 34127
e-mail: {poloni,lparussini}@units.it

Keywords: Ducted Propeller, Co-Kriging, Reliability-based Design Optimisation, Multi-fidelity Learning, Gaussian Markov Random Fields, Risk Averseness

Abstract. *This paper proposes to apply multi-fidelity learning for reliability-based design optimisation of a ducted propeller. Theoretically, the efficiency of a propeller can be increased by placing the propeller into a duct. The increased efficiency makes the ducted propeller an appealing option for electrical aviation where optimal electricity consumption is vital. The electricity consumption is mainly dictated by the required power to reach the required thrust force. Recent design optimisation techniques such as machine learning can help us to reach high thrust to power ratios. Due to the expensive computational fluid dynamics simulations a multi-fidelity learning algorithm is investigated here for the application of ducted propeller design. The limited number of high-fidelity numerical experiments cannot provide sufficient information about the landscape of the design field and probability field. Therefore, information from lower fidelity simulations is fused into the high-fidelity surrogate using the recently published recursive co-Kriging technique augmented with Gaussian-Markov Random Fields. At each level the uncertainty can be modelled via a polynomial chaos expansion which provides a variable-fidelity quantification technique of the uncertainty. This facilitates the calculation of risk measures, like conditional Value-at-Risk, for reliability-based design optimisation. The multi-fidelity surrogate model can be adaptively refined following a similar strategy to the Efficient Global Optimisation using the expected improvement measure. The proposed combination of techniques provides an efficient manner to conduct reliability-based optimisation on expensive realistic problems using a multi-fidelity learning technique.*

[*] Corresponding Author

1 INTRODUCTION

Ducted Propellers are theoretically operating with higher efficiency than open propellers [1, 2]. This fact makes the ducted propeller a potential candidate for the propulsion of an electrical aircraft where optimal thrust to power ratio is vital. The performance and efficiency of a ducted propeller can be obtained through models and experiments of various fidelity ranging from cheap analytic formulas to expensive Computational Fluid Dynamics (CFD) simulations. Most of the optimisation procedure require a high number of performance analyses to find an optimal design particularly when the uncertain nature of the problem is also considered. This fact makes it difficult to employ high-fidelity performance predictors like CFD simulations throughout the entire optimisation workflow. One traditional way to tackle this difficulty is to use surrogate models [11, 12] which can efficiently replace the expensive CFD simulations. A surrogate model is trained on the available high-fidelity simulations and in the optimisation workflow this surrogate is used instead of the expensive high-fidelity simulation. This surrogate-based optimisation is very efficient; however, it is highly dependant on the quality of the surrogate model. The quality can be increased by increasing the number of training points. Unfortunately, in case of expensive CFD simulations, the increase of the training data-set quickly consumes the computational budget. Therefore, multi-fidelity learning techniques have been invented to fuse information of analyses of different fidelities [8, 9, 10, 13].

The information content of the surrogate at design locations where high-fidelity analyses are not available can be increased by conducting low-fidelity analyses. At locations where both low- and high-fidelity analyses are available the degree of trustfulness of our low-fidelity model can be automatically learned by calculating the cross-correlation of the fidelities.

In this work, the co-Kriging technique [9, 10, 13] is used to construct a multi-fidelity surrogate. The main drawback of Kriging based techniques is that they require to invert the covariance matrix of the observation locations which matrix is dense. This numerical issue is commonly resolved by applying various decomposition techniques [14]. However, this paper investigates an alternative solution to the issue. Namely, Gaussian-Markov Random Fields (GMRF) are applied to construct the inverse of the covariance matrix, the so-called precision matrix, directly [15, 16]. The paper is organised as follows. Section 2 introduces the employed propeller analysis codes: the Blade Element Momentum Theory for low-fidelity calculations and the Ducted Fan Design Code for high-fidelity calculations. Section 3 derives the used multi-fidelity learning technique: GMRF-co-Kriging. Section 4 describes the conditional Value-at-Risk reliability measure and its application in the in the optimisation workflow. A simple training data-set strategy based on the expected improvement is presented in Section 5. Some characteristics of the GMRF-co-Kriging technique is discussed in Section 6; as well, this section presents the performance of the proposed multi-fidelity learning technique on a simple propeller blade optimisation problem. Finally, Section 7 concludes the work conducted in this paper.

2 DUCTED PROPELLER

Ducted propeller is a propulsion unit similar to free propellers, but the propeller is placed inside a duct which increases the mass flow through the propeller. The theoretical calculations credits this increased mass flow to a reduced slipstream contraction [1, 2]. However, for higher Mach numbers, the slipstream contraction decreases anyway and the drag induced by the duct increases. This mitigates the advantages of ducted propellers for high speed aircraft [3].

Remaining in the low speed regime allows to benefit the most from the increased efficiency of a ducted propeller propulsion unit. Therefore, ducted propellers can be applied to small

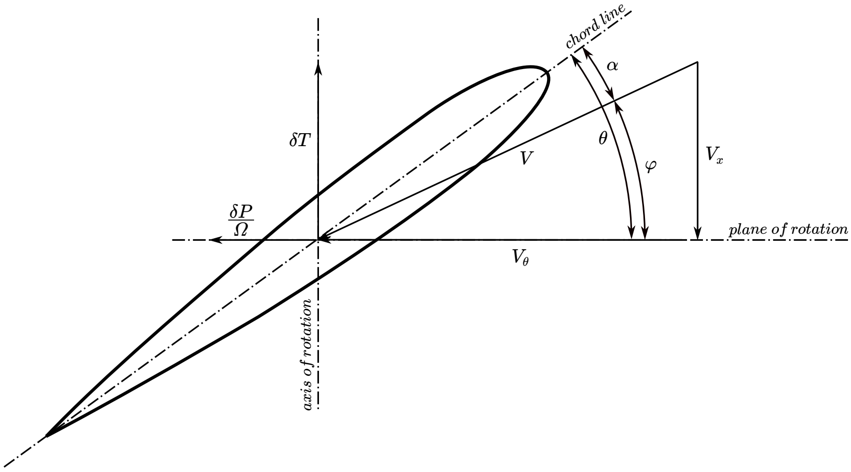


Figure 1: Blade Element velocities and forces

scale aircraft which operate at lower speeds. The increased propulsion efficiency makes ducted propellers promising candidates for electrical aircraft where the ratio of thrust and electricity consumption must be highly optimised.

In this work the performance analysis of the propulsion unit is investigated by two different solvers. Blade Element Momentum Theory (BEMT) [4, 5, 6] is presented in Section 2.1 and a potential flow solver, the Ducted Fan Design Code (DFDC) [7] presented in Section 2.2.

2.1 BLADE ELEMENT MOMENTUM THEORY

Blade Element Momentum Theory (BEMT) combines the Blade Element Theory (BET) and Actuator Disk Theory (ADT) into an iterative solver [4, 5, 6]. In both BET and ADT, the propeller blade is discretised with a given number of annuli. The effect of the actual blade elements are averaged over time. Each annulus is characterised by their local velocities and forces. At each radial station the velocity state is given by Eq.(1):

$$V_x = V_\infty(1+a), \tag{1}$$

$$V_\theta = \omega r(1-b), \tag{2}$$

$$V = \sqrt{V_x^2 + V_\theta^2}, \tag{3}$$

where V_∞ is the free stream velocity, V_θ is the angular velocity and V is the local velocity seen by the blade. r is the radius of the annulus and ω is the angular velocity of the propeller. a and b denote the induced axial and angular inflow factor respectively. The velocity vectors and resulting forces are depicted in Figure 1.

By knowing the induced velocities a and b, BET can determine the thrust and power of each blade element with Eqs. (4), (5):

$$\delta T = \frac{1}{2}\rho V^2 c(C_l\cos(\varphi) - C_d\sin(\varphi))Bdr, \tag{4}$$

$$\delta P = \frac{1}{2}\rho V^2 c(C_d\cos(\varphi) + C_l\sin(\varphi))r\omega Bdr, \tag{5}$$

where the ρ is the fluid density, c is the chord length and B is the number of blades. C_l and C_d are the 2D lift and drag coefficients of the blade element section. The lift $C_l(\alpha)$ and drag

$C_d(\alpha)$ are functions of the angle-of-attack α. Following the angle orientations in Figure 1, the angle-of-attack can be calculated by the following equations:

$$\begin{aligned} \varphi &= \tan \frac{V_x}{V_\theta}, & (6) \\ \alpha &= \theta - \varphi, & (7) \end{aligned}$$

where φ is the relative flow angle seen by the blade and θ is the geometrical twist of the blade element.

The induced velocities, however, are not known and their direct calculation would be a tedious work. Therefore the thrust and power are alternatively calculated according to the ADT:

$$\begin{aligned} \delta T &= \rho 4\pi r V_\infty^2 a(1+a)dr, & (8) \\ \delta P &= \rho 4\pi r^3 V_\infty b(1+a)\omega^2 dr, & (9) \end{aligned}$$

The Eqs. (4), (5) and (8), (9) are equated respectively in BEMT and the a and b induced velocity factors are calculated by iteratively minimising the deviation between the two theory. By considering that $V = \frac{V_x}{\sin\varphi} = \frac{V_\infty(1+a)}{\sin\varphi}$ and the blade solidity is $\sigma_r = \frac{Bc}{2\pi r}$, the problem to be solved iteratively can be reduced to Eqs. (10), (11):

$$\begin{aligned} \frac{a}{1+a} &= \frac{\sigma_r}{4\sin^2(\varphi)}(C_l\cos(\varphi) - C_d\sin(\varphi)), & (10) \\ \frac{b}{1-b} &= \frac{\sigma_r}{4\sin(\varphi)\cos(\varphi)}(C_d\cos(\varphi) + C_l\sin(\varphi)). & (11) \end{aligned}$$

2.2 DUCTED FAN DESIGN CODE

The DFDC software is based on the lifting-line theory of propeller blades and it is tailored to design axisymmetric ducted propellers. The software includes the loss effects due to non-uniform loading. Moreover, the effects of the shrouded tip and presence of centre body are also incorporated in the flow field calculation [7]. The code requires the operational conditions, the geometrical and aerodynamic properties of the blade elements, and the geometry of the centre body and duct as an input. The output of DFDC includes the resulting flow conditions and both the total and spanwise forces acting on the rotor and the duct. The fidelity of the code is higher than classical BEMT but it is still lower than Navier-Stokes solvers.

3 MULTI-FIDELITY MODEL

3.1 Definitions

A *random field* (or stochastic field), $X(\mathbf{s},\omega), \mathbf{s} \in \mathcal{D} \subset \mathbb{R}^d, \omega \in \Omega$ is a random function specified by its finite-dimensional joint distributions

$$F(y_1,\ldots,y_n;\mathbf{s}_1,\ldots,\mathbf{s}_n) = P(X(\mathbf{s}_1) \le y_1,\ldots,X(\mathbf{s}_n) \le y_n)$$

for every finite n and every collection $\mathbf{s}_1,\ldots,\mathbf{s}_n$ of locations in $\mathcal{D}$. To simplify the notation, one often writes $X(\mathbf{s})$, removing the dependency on ω from the notation.

A *Gaussian random field* $X(\mathbf{s})$ is defined by a mean function $\mu(\mathbf{s}) = E(X(\mathbf{s}))$ and a covariance function $\varsigma(\mathbf{s};\mathbf{t}) = Cov(X(\mathbf{s});X(\mathbf{t}))$. It has the property that, for every finite collection of points $\mathbf{s}_1,\ldots,\mathbf{s}_n$,

$$\mathbf{x} \equiv (X(\mathbf{s}_1),\ldots,X(\mathbf{s}_n))^T \sim N(\mu,\Sigma),$$

where $\Sigma_{ij} = \varsigma(\mathrm{s}_i; \mathrm{s}_j)$. For existence of a Gaussian field with a prescribed mean and covariance it is enough to ensure that ς is positive definite. A function $\varsigma(\mathrm{s}; \mathbf{t})$ is positive definite if for any finite set of locations $\mathrm{s}_1, \ldots, \mathrm{s}_n$ in $\mathcal{D}$, the covariance matrix

$$\mathbf{\Sigma} = \begin{pmatrix} \varsigma(\mathrm{s}_1, \mathrm{s}_1) & \varsigma(\mathrm{s}_1, \mathrm{s}_2) & \ldots & \varsigma(\mathrm{s}_1, \mathrm{s}_n) \\ \varsigma(\mathrm{s}_2, \mathrm{s}_1) & \varsigma(\mathrm{s}_2, \mathrm{s}_2) & \ldots & \varsigma(\mathrm{s}_2, \mathrm{s}_n) \\ \vdots & \vdots & \ddots & \vdots \\ \varsigma(\mathrm{s}_n, \mathrm{s}_1) & \varsigma(\mathrm{s}_n, \mathrm{s}_2) & \ldots & \varsigma(\mathrm{s}_n, \mathrm{s}_n) \end{pmatrix}$$

is non-negative definite: $\mathbf{z}^T \mathbf{\Sigma} \mathbf{z} \geq 0$ for all real valued vectors $\mathbf{z}$. The inverse of the covariance matrix $\mathbf{Q} = \mathbf{\Sigma}^{-1}$ is called precision matrix.

A random vector is called a *Gaussian Markov random field* (GMRF) with respect to a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with mean μ and precision matrix $\mathbf{Q} > 0$, if its density has the form

$$\pi(\mathbf{x}) = (2\pi)^{-n/2} |\mathbf{Q}|^{1/2} exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^T \mathbf{Q}(\mathbf{x} - \mu)\right),$$

where $\mathcal{V}$ and $\mathcal{E}$ are the set of nodes in the graph, and the set of edges in the graph, respectively.

3.2 Kriging

Denote a real-valued spatial process in d dimensions by $z(\mathrm{s}) : \mathrm{s} \in \mathcal{D} \subset \mathbb{R}^d$, where s is the location of the process $z(\mathrm{s})$ and s varies over the index set $\mathcal{D}$.

In Kriging theory, the response $z(\mathrm{s})$ is considered as a realisation of a multivariate Gaussian process $Z(\mathrm{s})$. $Z(\mathrm{s})$ is assumed to be the sum of a deterministic regression function $m(\mathrm{s})$, constructed by observed data, and a Gaussian process $Y(\mathrm{s})$, constructed through the residuals:

$$Z(\mathrm{s}) = m(\mathrm{s}) + Y(\mathrm{s}). \tag{12}$$

The trend function $m(\mathrm{s})$ is assumed to be $m(\mathrm{s}) = \mathbf{h}(\mathrm{s})\boldsymbol{\beta}$, where $\mathbf{h}(\mathrm{s})$ is a set of p covariates associated with each site s and $\boldsymbol{\beta}$ is a p-dimensional vector of coefficients. $Y(\mathrm{s})$ is the Gaussian process with zero mean and covariance function $\Sigma_{ij} = \varsigma(\mathrm{s}_i, \mathrm{s}_j) = \sigma^2 c(\mathrm{s}_i, \mathrm{s}_j; \boldsymbol{\theta})$, where σ^2 is a scale parameter, called the process variance, and c is a positive function with parameters $\boldsymbol{\theta}$, called the correlation function. Usual covariance functions are Gaussian, Matérn and exponential (where Gaussian and exponential covariances are particular cases of Matérn family covariance).

Let us suppose that $\mathbf{z}^{(n)}$ are observed values of $z(\mathrm{s})$ at n known locations $\hat{\mathcal{D}} = (\mathrm{s}_1, \ldots, \mathrm{s}_n)^T \subset \mathcal{D}$. For many cases, we do not have direct access to the function to be approximated but only to a noisy version of it. Let us consider this more general noisy case, assuming an independent Gaussian observation noise with zero mean and variance $\sigma_\epsilon^2(\mathrm{s})$. This is usually referred as the nugget effect. So, $\mathbf{z}^{(n)}$ are realisations of the Gaussian vector $\mathbf{Z}^{(n)} = Z(\hat{\mathcal{D}}) + \boldsymbol{\mathcal{E}}^{(n)}$, where $Z(\hat{\mathcal{D}})$ is the random process $Z(\mathrm{s})$ at the points $\hat{\mathcal{D}}$ and $\boldsymbol{\mathcal{E}}^{(n)} = (\sigma_\epsilon(\mathrm{s}_1)\mathcal{E}_1, \ldots, \sigma_\epsilon(\mathrm{s}_n)\mathcal{E}_n)^T$ is the white noise with $\mathcal{E}_{i=1,\ldots,n}$ independent and identically distributed with respect to a Gaussian distribution with zero mean and variance one.

We use the information contained in $\mathbf{Z}^{(n)}$ to predict $Z(\mathrm{s})$ considering the joint distribution of $Z(\mathrm{s})$ and $\mathbf{Z}^{(n)}$:

$$\begin{pmatrix} Z(\mathrm{s}) \\ \mathbf{Z}^{(n)} \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{h}(\mathrm{s})\boldsymbol{\beta} \\ \mathbf{H}\boldsymbol{\beta} \end{pmatrix}, \begin{pmatrix} \varsigma(\mathrm{s}, \mathrm{s}) & \boldsymbol{\varsigma}^T(\mathrm{s}) \\ \boldsymbol{\varsigma}(\mathrm{s}) & \mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I} \end{pmatrix}\right), \tag{13}$$

where $\mathbf{H} = \mathbf{h}\left(\hat{\mathcal{D}}\right)$ is the $n \times p$ model matrix, $\boldsymbol{\Sigma}$ is the $n \times n$ covariance matrix between the observation points $\hat{\mathcal{D}}$, $\boldsymbol{\varsigma}(\mathbf{s})$ is the n-dimensional covariance vector between the prediction point s and the observation points $\hat{\mathcal{D}}$, σ_ϵ^2 is considered constant for simplicity.

Then, the conditional distribution $\left[Z(\mathrm{s}) \left|\mathbf{Z}^{(n)}, \boldsymbol{\beta}, \sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta}\right.\right]$ is Gaussian with mean and variance:

$$\hat{m}_Z(\mathbf{s}) = \mathbf{h}(\mathbf{s})\boldsymbol{\beta} + \boldsymbol{\varsigma}^T(\mathbf{s}) \left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right), \tag{14}$$

$$\hat{s}_Z^2(\mathbf{s}) = \varsigma(\mathbf{s}, \mathbf{s}) - \boldsymbol{\varsigma}^T(\mathbf{s}) \left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \boldsymbol{\varsigma}(\mathbf{s}). \tag{15}$$

In order to estimate the parameters $(\boldsymbol{\beta}, \sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta})$, the Maximum Likelihood Estimation (MLE) is a very popular method. The multivariate normal assumption for $\mathbf{z}^{(n)}$ leads to the following likelihood:

$$f\left(\mathbf{z}^{(n)} \left|\boldsymbol{\beta}, \sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta}\right.\right) = \frac{1}{(2\pi)^{n/2}\sqrt{|\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}|}} \\ exp\left(-\frac{1}{2}\left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right)^T \left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right)\right). \tag{16}$$

Given:

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{H}^T \left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \mathbf{H}\right)^{-1} \mathbf{H}^T \left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \mathbf{z}^{(n)}, \tag{17}$$

which is the MLE of $\boldsymbol{\beta}$ corresponding to its generalised least squares estimate, the MLEs of σ^2, σ_ϵ^2 and hyperparameters $\boldsymbol{\theta}$ are identified by minimising:

$$\mathcal{L}\left(\sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta}\right) = \left(\mathbf{z}^{(n)} - \mathbf{H}\hat{\boldsymbol{\beta}}\right)^T \left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \left(\mathbf{z}^{(n)} - \mathbf{H}\hat{\boldsymbol{\beta}}\right) + log\left(\left|\left(\boldsymbol{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)\right|\right), \tag{18}$$

which is the opposite of the log-likelihood up to a constant.

When there is no measurement error, the observed values $\mathbf{z}^{(n)}$ are free-noise realisations of the Gaussian vector $\mathbf{Z}^{(n)} = Z(\hat{\mathcal{D}})$ and Eqs.(14) and (15) reduce to:

$$\hat{m}_Z(\mathbf{s}) = \mathbf{h}(\mathbf{s})\boldsymbol{\beta} + \mathbf{c}^T(\mathbf{s})\mathbf{C}^{-1}\left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right), \tag{19}$$

$$\hat{s}_Z^2(\mathbf{s}) = \sigma^2\left(1 - \mathbf{c}^T(\mathbf{s})\mathbf{C}^{-1}\mathbf{c}(\mathbf{s})\right), \tag{20}$$

where $\mathbf{C}$ is the $n \times n$ correlation matrix between the observation points $\hat{\mathcal{D}}$ and $c(\mathbf{s})$ is the n-dimensional correlation vector between the prediction point s and the observation points $\hat{\mathcal{D}}$.

For the parameter estimation, the following likelihood:

$$f\left(\mathbf{z}^{(n)} \left|\boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta}\right.\right) = \frac{1}{(2\pi\sigma^2)^{n/2}\sqrt{|\mathbf{C}|}} exp\left(-\frac{1}{2}\frac{\left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right)^T \mathbf{C}^{-1}\left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right)}{\sigma^2}\right) \tag{21}$$

has to be maximised.

Given the MLE of $\boldsymbol{\beta}$, $\hat{\boldsymbol{\beta}} = \left(\mathbf{H}^T\mathbf{C}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{C}^{-1}\mathbf{z}^{(n)}$, in a free-noise case, a closed form expression for the estimate of σ^2 can be derived:

$$\hat{\sigma}^2 = \frac{1}{n}\left(\mathbf{z}^{(n)} - \mathbf{H}\hat{\boldsymbol{\beta}}\right)^T \mathbf{C}^{-1}\left(\mathbf{z}^{(n)} - \mathbf{H}\hat{\boldsymbol{\beta}}\right). \tag{22}$$

The MLE of hyperparameters $\boldsymbol{\theta}$ of the correlation function c are identified by minimising the opposite of the log-likelihood

$$\mathcal{L}\left(\boldsymbol{\theta}\right) = n\, log\left(\hat{\sigma}^2\right) + log\left(|\mathbf{C}|\right). \tag{23}$$

When there is no measurement error, Kriging is an exact interpolator, meaning that if you predict at a location where data has been collected, the predicted value is the same as the measured value. However, when measurement errors exist, you want to predict the filtered value, which does not have the measurement error term. At locations where data has been collected, the filtered value is not the same as the measured value.

3.3 GMRF

With Gaussian models, such as Kriging, the primary difficulty is dimension, which typically scales with the number of observations. The basic complexity of Gaussian processes is $\mathcal{O}(N^3)$ where N is the number of data points, due to the inversion of an $N \times N$ matrix. This is the reason to introduce GMRF models, assuming that a random variable associated with a region depends primarily on its neighbours.

A random field is said to be a *Markov random field* if it satisfies Markov property. A stochastic process has the Markov property if the conditional probability distribution of future states of the process (conditional on both past and present values) depends only on the present state; that is, given the present, the future does not depend on the past. A Markov random field extends this property to two or more dimensions or to random variables defined for an interconnected network of items.

Let the neighbours $\mathcal{N}_i$ of a point $\mathbf{s}_i$ be the points $\{\mathbf{s}_j, j \in \mathcal{N}_i\}$ that are close to $\mathbf{s}_i$. The random field $X(\mathbf{s})$ that satisfies

$$p(X_i|\mathbf{X}_{-i}) = p(X_i|\left\{X_j|j \in \mathcal{N}_i\right\}), \tag{24}$$

where $X_i = X(\mathbf{s}_i)$ and $\mathbf{X}_{-i} = (X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_n)$, is a Markov random field.

A Gaussian random field $X(\mathbf{s}) \sim N(\boldsymbol{\mu}, \mathbf{Q}^{-1})$ that satisfies (24) is a GMRF. In that case the full conditionals are Gaussian with means and precisions:

$$E(X_i|\mathbf{X}_{-i}) = \mu_i - \sum_{j:j\sim i} \beta_{ij}\left(x_j - \mu_j\right), \tag{25}$$

$$Prec(X_i|\mathbf{X}_{-i}) = Var(X_i|\mathbf{X}_{-i})^{-1} = \kappa_i > 0, \tag{26}$$

where β_{ij} and κ_i are parameters satisfying $\beta_{ij}\kappa_i = \beta_{ji}\kappa_j$ for all i and j and with precision matrix $\mathbf{Q}$ positive definite:

$$Q_{ij} = \begin{cases} \kappa_i,\ i = j \\ \kappa_i\beta_{ij},\ i \neq j \end{cases}. \tag{27}$$

The joint density function for $X(\mathbf{s})$ is Gaussian and of the form

$$f(\mathbf{X}) = (2\pi)^{-n/2}\left|\mathbf{Q}\right|^{1/2} exp\left(-\frac{1}{2}\mathbf{X}^T\mathbf{Q}\mathbf{X}\right). \tag{28}$$

In most cases if the total number of neighbours is $\mathcal{O}(n)$, only $\mathcal{O}(n)$ of the $n \times n$ terms in $\mathbf{Q}$ will be non-zero. So numerical algorithms for sparse matrices can be exploited to construct GMRF models.

Given the Gaussian vector $\mathbf{X}^{(n)} = (X_1, \ldots, X_n)^T$ containing the values of the random process $X(\mathbf{s})$ at the points in the experimental design set $\hat{\mathcal{D}} = (\mathbf{s}_1, \ldots, \mathbf{s}_n)^T \subset \mathcal{D}$, considering the joint distribution of:

$$\begin{pmatrix} X(\mathbf{s}) \\ \mathbf{X}^{(n)} \end{pmatrix} \tag{29}$$

with mean:

$$\begin{pmatrix} \mu(\mathbf{s}) \\ \boldsymbol{\mu}^{(n)} \end{pmatrix} \tag{30}$$

and precision:

$$\begin{pmatrix} \mathbf{Q}(\mathbf{s},\mathbf{s}) & \mathbf{Q}(\mathbf{s},\hat{\mathcal{D}}) \\ \mathbf{Q}(\mathbf{s},\hat{\mathcal{D}})^T & \mathbf{Q}(\hat{\mathcal{D}},\hat{\mathcal{D}}) \end{pmatrix}, \tag{31}$$

the conditional expectation is:

$$E(X(\mathbf{s})|\,\mathbf{X}^{(n)}) = \mu(\mathbf{s}) - \mathbf{Q}(\mathbf{s},\mathbf{s})^{-1}\mathbf{Q}(\mathbf{s},\hat{\mathcal{D}})\left(\mathbf{X}^{(n)} - \boldsymbol{\mu}^{(n)}\right) \tag{32}$$

with conditional precision:

$$Prec(X(\mathbf{s})|\,\mathbf{X}^{(n)}) = \mathbf{Q}(\mathbf{s},\mathbf{s}). \tag{33}$$

We are interested in GMRFs where the precision matrix $\mathbf{Q}$ is the numerical discretisation of a diffusion operator. We focus on finite element discretisations.

Gaussian random fields with Matérn covariances

$$C\left(\|\mathbf{u}\|\right) = \sigma^2 \frac{2^{1-\nu}}{\Gamma\left(\nu\right)}\left(\chi\,\|\mathbf{u}\|\right)^{\nu} \mathcal{K}_{\nu}\left(\chi\,\|\mathbf{u}\|\right) \tag{34}$$

with $\|\mathbf{u}\|$ the distance between two points, are solutions to a Stochastic Partial Differential Equation (SPDE) [19, 20]:

$$\left(\chi^2 - \Delta\right)^{\alpha/2} X(\mathbf{s}) = W(\mathbf{s}), \tag{35}$$

where $W(\mathbf{s})$ is white noise, $\Delta = \sum_i \frac{\partial^2}{\partial s_i^2}$ is the Laplacian operator and $\alpha = \nu + d/2$, the parameter ν controls the smoothness and the parameter χ controls the range. So, according to the Whittle characterisation of the Matérn covariance functions, we get a Markovian random field when α is an integer. The solution can be constructed as a finite basis expansion:

$$X(\mathbf{s}) = \sum_k \varphi_k(\mathbf{s}) x_k, \tag{36}$$

with a suitable distribution for the weights $\{x_k\}$. A stochastic weak solution to the SPDE is given by:

$$\left\langle \varphi_j, \left(\chi^2 - \Delta\right)^{\alpha/2} X(\mathbf{s}) \right\rangle = \langle \varphi_j, W \rangle \;\forall j. \tag{37}$$

Replacing $X(\mathbf{s})$ with the finite basis expansion (36) gives:

$$\sum_i \left\langle \varphi_j, \left(\chi^2 - \Delta\right)^{\alpha/2} \varphi_i) \right\rangle x_i = \langle \varphi_j, W \rangle \;\forall j. \tag{38}$$

With the opportune choice of basis functions the Gaussian random field $X(\mathbf{s})$ will result into a GMRF. The piecewise linear basis gives (almost) a GMRF. Indeed, using a piecewise linear basis, only neighbouring basis functions overlap. Increased smoothness of the random field induces a larger neighbourhood in the GMRF representation. The choice of test functions, in relation to the basis functions, governs the approximation properties of the resulting model representation. For $\alpha = 1$ the correct choice is $\phi_k = \left(\chi^2 - \Delta\right)^{1/2} \varphi_k$ which is the least squares finite element approximation, for $\alpha = 2$ the correct choice is $\phi_k = \varphi_k$ which is the Galerkin finite element approximation. For $\alpha \geq 3$, $\phi_k = \varphi_k$ if we let $\alpha = 2$ on the left-hand side of equation and replace the right -hand side with a field generated by $\alpha - 2$. So in practice this generates a recursive Galerkin formulation.

Defining the matrices:

$$M_{ij} = \langle \varphi_i, \varphi_j \rangle, \tag{39}$$

$$S_{ij} = \langle \nabla\varphi_i, \nabla\varphi_j \rangle, \tag{40}$$

$$K_{ij} = \chi^2 M_{ij} + S_{ij}, \tag{41}$$

then the precision matrix for weights $\mathbf{x}$ for $\alpha = 1, 2, \ldots$ is:

$$\mathbf{Q}_1 = \mathbf{K}, \tag{42}$$

$$\mathbf{Q}_2 = \mathbf{K}\mathbf{M}^{-1}\mathbf{K}, \tag{43}$$

$$\mathbf{Q}_\alpha = \mathbf{K}\mathbf{M}^{-1}\mathbf{Q}_{\alpha-2}\mathbf{M}^{-1}\mathbf{K}. \tag{44}$$

$\mathbf{M}$ and $\mathbf{S}$ are both sparse given the choice of piecewise linear basis, so that $\mathbf{K}$ is sparse too. But $\mathbf{M}^{-1}$ is dense, which makes the precision matrix dense as well, losing the Markov property. The matrix $\mathbf{M}$ is replaced by a diagonal matrix $\tilde{\mathbf{M}}$ where $\tilde{M}_{ii} = \langle \varphi_i, 1 \rangle$ which makes the precision matrix sparse with a small approximation error.

Although the approach does give a GMRF representation of the Matérn field on the discretised region, it is an approximation of SPDE solution. Using standard results from the finite element literature, it is also possible to derive rates of convergence results.

3.4 GMRF-Kriging

As in section 3.2, denote a real-valued spatial process in d dimensions by $z(\mathbf{s}) : \mathbf{s} \in \mathcal{D} \subset \mathbb{R}^d$ where $\mathbf{s}$ is the location of the process $z(\mathbf{s})$ and $\mathbf{s}$ varies over the index set $\mathcal{D}$.

The response $z(\mathbf{s})$ is considered as a realisation of a linear latent variable model $Z(\mathbf{s})$:

$$Z(\mathbf{s}) = \boldsymbol{\varphi}^T(\mathbf{s})\mathbf{X} + \mathcal{E}(\mathbf{s}), \tag{45}$$

$$\mathbf{X} \sim N(\boldsymbol{\mu}_x, \mathbf{Q}_x^{-1}), \tag{46}$$

$$\mathcal{E}(\mathbf{s}) \sim N\left(0, \sigma_\epsilon^2(\mathbf{s})\right), \tag{47}$$

where $\boldsymbol{\varphi}^T(\mathbf{s})\mathbf{X}$ is a spatial basis expansion with k basis functions with local (compact) support. The latent variables $\mathbf{X}$ are a GMRF, where $\mathbf{Q}_x$ is derived from an SPDE construction with parameters $\boldsymbol{\theta}$ [15]. $\boldsymbol{\mu}_x$ is usually zero, but for now let us consider the more general case. $\mathcal{E}(\mathbf{s})$ is white noise, with constant variance σ_ϵ^2 for simplicity.

Let us suppose that $\mathbf{z}^{(n)}$ are observed values of $z(\mathbf{s})$ at n known locations $\hat{\mathcal{D}} = (\mathbf{s}_1, \ldots, \mathbf{s}_n)^T \subset \mathcal{D}$. $\mathbf{z}^{(n)}$ are realisations of the random vector $\mathbf{Z}^{(n)} = \mathbf{\Phi}^T\mathbf{X} + \boldsymbol{\mathcal{E}}^{(n)}$, where $\mathbf{\Phi}$ is the $k \times n$ matrix $(\varphi_1(\hat{\mathcal{D}}), \ldots, \varphi_k(\hat{\mathcal{D}}))^T$ containing the values of the basis functions in $\hat{\mathcal{D}}$ and $\boldsymbol{\mathcal{E}}^{(n)}$ is the vector $(\sigma_\epsilon\mathcal{E}_1, \ldots, \sigma_\epsilon\mathcal{E}_n)^T$ with $\mathcal{E}_{i=1,\ldots,n}$ independent and identically distributed with respect to a Gaussian distribution with zero mean and variance one.

We can write the hierarchical model

$$\begin{aligned}\mathbf{X} \sim &\quad N(\boldsymbol{\mu}_x, \mathbf{Q}_x^{-1}), &(48)\\ \left(\mathbf{Z}^{(n)} \middle| \mathbf{X}\right) \sim &\quad N\left(\boldsymbol{\Phi}\mathbf{X}, \mathbf{Q}_\epsilon^{-1}\right), &(49)\end{aligned}$$

where $\mathbf{Q}_\epsilon^{-1} = \sigma_\epsilon^2 \mathbf{I}$ is the $n \times n$ covariance matrix of observations.

The joint distribution for the observations and the latent variables $\mathbf{X}$ is given by:

$$\left(\begin{array}{c}\mathbf{X}\\ \mathbf{Z}^{(n)}\end{array}\right) \sim N\left(\left(\begin{array}{c}\boldsymbol{\mu}_x\\ \boldsymbol{\Phi}\boldsymbol{\mu}_x\end{array}\right), \left[\begin{array}{cc}\mathbf{Q}_x + \boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\boldsymbol{\Phi} & -\boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\\ -\mathbf{Q}_\epsilon\boldsymbol{\Phi} & \mathbf{Q}_\epsilon\end{array}\right]^{-1}\right). \tag{50}$$

The conditional distribution for $\mathbf{X}$ given $\mathbf{Z}^{(n)}$ is $\left(\mathbf{X}\middle| \mathbf{Z}^{(n)}\right) \sim N\left(\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}, \boldsymbol{\Sigma}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right)$, with:

$$\begin{aligned}\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} &= \boldsymbol{\mu}_x + \mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}^{-1}\boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\boldsymbol{\mu}_x\right), &(51)\\ \boldsymbol{\Sigma}_{\mathbf{X}|\mathbf{Z}^{(n)}} &= \mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}^{-1}, &(52)\\ \mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}} &= \mathbf{Q}_x + \boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\boldsymbol{\Phi}. &(53)\end{aligned}$$

The variance can be computed as $\mathrm{s}^2_{\mathbf{X}|\mathbf{Z}^{(n)}} = diag\left(\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}^{-1}\right)$. Note that the elements of $\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}$ are the basis function coefficients and covariate effect estimates in the Kriging predictor:

$$\hat{m}_Z(\mathbf{s}) = \boldsymbol{\varphi}(\mathbf{s})\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} \tag{54}$$

with squared error

$$\hat{s}_Z^2(\mathbf{s}) = diag\left(\boldsymbol{\varphi}(\mathbf{s})\boldsymbol{\Sigma}_{\mathbf{X}|\mathbf{Z}^{(n)}}\boldsymbol{\varphi}^T(\mathbf{s})\right). \tag{55}$$

The method to estimate it hyper-parameter $\boldsymbol{\theta}$ is the MLE.

The likelihood for $\mathbf{X}$ given the parameters $\boldsymbol{\theta}$ is:

$$\pi\left(\mathbf{X}\middle| \boldsymbol{\theta}\right) = \frac{1}{(2\pi)^{\frac{m+p}{2}}\sqrt{|\mathbf{Q}_x|}} exp\left(-\frac{1}{2}\left(\mathbf{X} - \boldsymbol{\mu}_x\right)^T\mathbf{Q}_x\left(\mathbf{X} - \boldsymbol{\mu}_x\right)\right) \tag{56}$$

so that the log-likelihood is:

$$log\,\pi\left(\mathbf{X}\middle| \boldsymbol{\theta}\right) = -\frac{m+p}{2}log\left(2\pi\right) + \frac{1}{2}log\ |\mathbf{Q}_x| - \frac{1}{2}\left(\mathbf{X} - \boldsymbol{\mu}_x\right)^T\mathbf{Q}_x\left(\mathbf{X} - \boldsymbol{\mu}_x\right). \tag{57}$$

For known $\mathbf{X} = \hat{\mathrm{x}}$, the likelihood for $\mathbf{z}^{(n)}$ given the parameters $\boldsymbol{\theta}$ is:

$$\pi\left(\mathbf{z}^{(n)}\middle| \boldsymbol{\theta}\right) = \frac{\pi\left(\boldsymbol{\theta}\middle| \mathbf{z}^{(n)}\right)}{\pi\left(\boldsymbol{\theta}\right)} = \left.\frac{\pi\left(\mathbf{X}\middle| \boldsymbol{\theta}\right)\pi\left(\mathbf{z}^{(n)}\middle| \boldsymbol{\theta}, \mathbf{X}\right)}{\pi\left(\mathbf{X}\middle| \boldsymbol{\theta}, \mathbf{z}^{(n)}\right)}\right|_{\mathbf{X}=\hat{\mathrm{x}}} \tag{58}$$

so that the log-likelihood is:

$$\begin{aligned}log\,\pi\left(\mathbf{z}^{(n)}\middle| \boldsymbol{\theta}\right) = log\,\pi\left(\hat{\mathrm{x}}\middle| \boldsymbol{\theta}\right) + log\,\pi\left(\mathbf{z}^{(n)}\middle| \boldsymbol{\theta}, \hat{\mathrm{x}}\right) - log\,\pi\left(\hat{\mathrm{x}}\middle| \boldsymbol{\theta}, \mathbf{z}^{(n)}\right) = \\ -\frac{m+p}{2}log\left(2\pi\right) + \frac{1}{2}log\ |\mathbf{Q}_x| - \frac{1}{2}\left(\hat{\mathrm{x}} - \boldsymbol{\mu}_x\right)^T\mathbf{Q}_x\left(\hat{\mathrm{x}} - \boldsymbol{\mu}_x\right) \\ -\frac{n}{2}log\left(2\pi\right) + \frac{1}{2}log\ |\mathbf{Q}_\epsilon| - \frac{1}{2}\left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\hat{\mathrm{x}}\right)^T\mathbf{Q}_\epsilon\left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\hat{\mathrm{x}}\right) \\ +\frac{m+p}{2}log\left(2\pi\right) - \frac{1}{2}log\ \left|\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right| + \frac{1}{2}\left(\hat{\mathrm{x}} - \mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right)^T\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}\left(\hat{\mathrm{x}} - \mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right).\end{aligned} \tag{59}$$

In practice the likelihood for $\mathbf{z}^{(n)}$ given the parameters $\boldsymbol{\theta}$ is evaluated for $\hat{\mathrm{x}} = \mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}$, so that:

$$\begin{aligned} log\,\pi\left(\mathbf{z}^{(n)}\middle|\,\boldsymbol{\theta}\right) = -\frac{n}{2}log\,(2\pi) + \frac{1}{2}log\,\left|\mathbf{Q}_x\right| + \frac{1}{2}log\,\left|\mathbf{Q}_\epsilon\right| - \frac{1}{2}log\,\left|\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right| \\ -\frac{1}{2}\left(\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} - \boldsymbol{\mu}_x\right)^T \mathbf{Q}_x \left(\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} - \boldsymbol{\mu}_x\right) \\ -\frac{1}{2}\left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right)^T \mathbf{Q}_\epsilon \left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right). \end{aligned} \tag{60}$$

3.5 RECURSIVE CO-KRIGING

Recursive co-Kriging is a recursive framework which exploits multi-fidelity data coming from sources with different reliability, building l independent Kriging problems [10].

In this case there are l levels of response $(z_t(\mathrm{s}))_{t=1,\ldots,l}$ sorted by increasing order of fidelity and modelled by Gaussian processes $(Z_t(\mathrm{s}))_{t=1,\ldots,l}$, with $\mathrm{s} \in \mathcal{D}$. $z_l(\mathrm{s})$ is the most accurate and costly response and $(z_t(\mathrm{s}))_{t=1,\ldots,l-1}$ are cheaper versions of it, with $z_1(\mathrm{s})$ the least accurate.

An auto-regressive model can be formulated for $t = 2,\ldots,l$:

$$\left\{ \begin{array}{c} Z_t(\mathrm{s}) = \rho_{t-1}(\mathrm{s})Z_{t-1}(\mathrm{s}) + \delta_t(\mathrm{s}), \\ Z_{t-1}(\mathrm{s})\perp\delta_t(\mathrm{s}), \\ \rho_{t-1}(\mathrm{s}) = \mathbf{g}_{t-1}^T(\mathrm{s})\boldsymbol{\beta}_{\rho_{t-1}}, \end{array} \right. \tag{61}$$

where $\delta_t(\mathrm{s})$ is a Gaussian process, with mean $\mathbf{f}_t^T(\mathrm{s})\boldsymbol{\beta}_t$ and covariance function $\sigma_t^2 c_t(\mathrm{s},\mathrm{s}')$, independent of $Z_{t-1}(\mathrm{s}),\ldots,Z_1(\mathrm{s})$ and $\rho_{t-1}(\mathrm{s})$ represents a scale factor between $Z_t(\mathrm{s})$ and $Z_{t-1}(\mathrm{s})$. $\mathrm{g}_{t-1}(\mathrm{s})$ and $\mathbf{f}_t(\mathrm{s})$ are vectors of polynomial basis functions and $\boldsymbol{\beta}_{\rho_{t-1}}$ and $\boldsymbol{\beta}_t$ are the vectors of coefficients.

The Gaussian process $Z_t(\mathrm{s})$ modelling the response at level t is expressed as a function of the Gaussian process $Z_{t-1}(\mathrm{s})$ conditioned by the values $\mathbf{z}^{(t-1)} = (\mathbf{z}_1,\ldots,\mathbf{z}_{t-1})$ at points in the experimental design sets $(\mathcal{D}_i)_{i=1,\ldots,t-1}$.

Considering the joint distribution of $\delta_t(\mathrm{s}) = Z_t(\mathrm{s}) - \rho_{t-1}(\mathrm{s})Z_{t-1}(\mathrm{s})$ and $\delta_t(\mathcal{D}_t) = \mathbf{Z}^{(t)} - \rho_{t-1}(\mathcal{D}_t) \odot \mathbf{z}_{t-1}(\mathcal{D}_t)$:

$$\left(\begin{array}{c} Z_t(\mathrm{s}) - \rho_{t-1}(\mathrm{s})Z_{t-1}(\mathrm{s}) \\ \mathbf{Z}^{(t)} - \rho_{t-1}(\mathcal{D}_t) \odot \mathbf{z}_{t-1}(\mathcal{D}_t) \end{array} \right) \sim N\left(\left(\begin{array}{c} \mathbf{f}_t(\mathrm{s})\boldsymbol{\beta}_t \\ \mathbf{F}_t\boldsymbol{\beta}_t \end{array} \right), \left(\begin{array}{cc} \mathbf{c}_t(\mathrm{s},\mathrm{s}) & \mathbf{c}_t^T(\mathrm{s}) \\ \mathbf{c}_t(\mathrm{s}) & \mathbf{C}_t \end{array} \right) \right), \tag{62}$$

we have for $t = 2,\ldots,l$ and for $\mathrm{s} \in \mathcal{D}$:

$$\left[Z_t(\mathrm{s})\,\middle|\mathbf{Z}^{(t)} = \mathbf{z}^{(t)}, \boldsymbol{\beta}_t, \boldsymbol{\beta}_{\rho_{t-1}}, \sigma_t^2\right] \sim \mathcal{N}\left(\hat{m}_{Z_t}(\mathrm{s}), \hat{s}_{Z_t}^2(\mathrm{s})\right), \tag{63}$$

where:

$$\hat{m}_{Z_t}(\mathrm{s}) = \rho_{t-1}(\mathrm{s})\hat{m}_{Z_{t-1}}(\mathrm{s}) + \mathbf{f}_t^T(\mathrm{s})\boldsymbol{\beta}_t + \mathbf{c}_t^T(\mathrm{s})\mathbf{C}_t^{-1}(\mathbf{z}_t - \rho_{t-1}(\mathcal{D}_t) \odot \mathbf{z}_{t-1}(\mathcal{D}_t) - \mathbf{F}_t\boldsymbol{\beta}_t) \tag{64}$$

and:

$$\hat{s}_{Z_t}^2(\mathrm{s}) = \rho_{t-1}^2(\mathrm{s})\hat{s}_{Z_{t-1}}^2(\mathrm{s}) + \sigma_t^2\left(1 - \mathbf{c}_t^T(\mathrm{s})\mathbf{C}_t^{-1}\mathbf{c}_t(\mathrm{s})\right). \tag{65}$$

The notation $\odot$ represents the Hadamard product. $\mathbf{C}_t$ is the correlation matrix and $\mathbf{c}_t^T(\mathrm{s})$ is the correlation vector. We denote by $\rho_{t-1}(\mathcal{D}_t)$ the vector containing the values of $\rho_{t-1}(\mathrm{s})$ for $\mathrm{s} \in \mathcal{D}_t$, $\mathbf{z}_{t-1}(\mathcal{D}_t)$ the vector containing the known values of $Z_t(\mathrm{s})$ at points in $\mathcal{D}_t$ and $\mathbf{F}_t$ is the experience matrix containing the values of $\mathbf{f}_t(\mathrm{s})^T$ on $\mathcal{D}_t$.

The recursive framework of co-Kriging is clearly visible in Eqs.(64,65), where the mean and the variance of the Gaussian process $Z_t(\mathrm{s})$ are functions of mean and variance of the Gaussian process $Z_{t-1}(\mathrm{s})$.

The mean $\hat{\mu}_{Z_t}(\mathrm{s})$ is the surrogate model of the response at level t, $1 \leq t \leq l$, taking into account the known values of the t first levels of responses $(\mathbf{z}_i)_{i=1,\dots,l}$. The variance $\hat{s}^2_{Z_t}(\mathrm{s})$ represents the mean squared error of the surrogate model of the response at level t. The variance will be zero at known values of the first t levels of responses.

The parameters $(\boldsymbol{\theta}_t)$ are estimated by minimising the opposite of the concentrated restricted log-likelihoods at each level t:

$$log(|det(\mathbf{C}_t)|) + (n_t - p_t - q_{t-1})log(\hat{\sigma}^2_t) \tag{66}$$

for $t = 1, \dots, l$.

3.6 RECURSIVE GMRF-CO-KRIGING

Similarly to the classical recursive co-Kriging there are l levels of response $(z_t(\mathrm{s}))_{t=1,\dots,l}$ sorted by increasing order of fidelity.

An auto-regressive model using GMRF can be formulated for $t = 2, \dots, l$:

$$\begin{cases} Z_t(\mathrm{s}) = \boldsymbol{\varphi}^T(\mathrm{s})\mathbf{X}_t + \mathcal{E}_t(\mathrm{s}), \\ \mathbf{X}_t = \boldsymbol{\rho}^T_{t-1}\mathbf{X}_{t-1} + \boldsymbol{\delta}_t, \\ \mathbf{X}_{t-1} \perp \boldsymbol{\delta}_t, \end{cases} \tag{67}$$

where $\boldsymbol{\delta}_t$ is a a GMRF with mean $\boldsymbol{\mu}_{x_t}$ and precision matrix $\mathbf{Q}_{x_t}$ derived from an SPDE construction with parameters $\boldsymbol{\theta}_t$.

Let us suppose that $\mathbf{z}_t^{(n_t)}$ are observed values of $z_t(\mathrm{s})$ at n_t known locations $\hat{\mathcal{D}}_t \subset \mathcal{D}$. $\mathbf{z}_t^{(n_t)}$ are realisations of the random vector $\mathbf{Z}_t^{(n_t)}$.

We can write the hierarchical model

$$\boldsymbol{\delta}_t \sim N(\boldsymbol{\mu}_{x_t}, \mathbf{Q}^{-1}_{x_t}), \tag{68}$$

$$\left(\mathbf{Z}_t^{(n_t)} \middle| \mathbf{X}_t\right) - \boldsymbol{\rho}^T_{t-1} \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{X}_{t-1} \sim N\left(\boldsymbol{\Phi}_t\boldsymbol{\delta}_t, \mathbf{Q}^{-1}_{\epsilon_t}\right), \tag{69}$$

where $\mathbf{Q}^{-1}_{\epsilon_t} = \sigma^2_{\epsilon_t}\mathbf{I}$ is the $n_t \times n_t$ covariance matrix of observations.

The joint distribution for the observations and the latent variables $\mathbf{X}_t$ is given by:

$$\begin{pmatrix} \mathbf{X}_t - \boldsymbol{\rho}^T_{t-1}\mathbf{X}_{t-1} \\ \left(\mathbf{Z}_t^{(n_t)} \middle| \mathbf{X}_t\right) - \boldsymbol{\rho}^T_{t-1} \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{X}_{t-1} \end{pmatrix} \sim \\ N\left(\begin{pmatrix} \boldsymbol{\mu}_{x_t} \\ \boldsymbol{\Phi}_t\boldsymbol{\mu}_{x_t} \end{pmatrix}, \begin{bmatrix} \mathbf{Q}_{x_t} + \boldsymbol{\Phi}^T_t\mathbf{Q}_{\epsilon_t}\boldsymbol{\Phi}_t & -\boldsymbol{\Phi}^T_t\mathbf{Q}_{\epsilon_t} \\ -\mathbf{Q}_{\epsilon_t}\boldsymbol{\Phi}_t & \mathbf{Q}_{\epsilon_t} \end{bmatrix}^{-1} \right). \tag{70}$$

The conditional distribution for $\mathbf{X}_t$ given $\mathbf{Z}_t^{(n_t)}$ is $\left(\mathbf{X}_t \middle| \mathbf{Z}_t^{(n_t)}\right) \sim N\left(\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}, \boldsymbol{\Sigma}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right)$, with:

$$\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \boldsymbol{\rho}^T_{t-1}\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{t-1}^{(\mathbf{n_{t-1}})}} + \boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}, \tag{71}$$

$$\boldsymbol{\Sigma}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \boldsymbol{\rho}^T_{t-1}\boldsymbol{\Sigma}_{\mathbf{X}_{t-1}|\mathbf{Z}_{t-1}^{(\mathbf{n_{t-1}})}}\boldsymbol{\rho}_{t-1} + \mathbf{Q}^{-1}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}, \tag{72}$$

$$\boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \boldsymbol{\mu}_{x_t} + \mathbf{Q}^{-1}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\boldsymbol{\Phi}^T_t\mathbf{Q}_{\epsilon_t}\left(\mathbf{Z}_t^{(n_t)} - \boldsymbol{\rho}^T_{t-1} \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{t-1}^{(\mathbf{n_{t-1}})}} - \boldsymbol{\Phi}_t\boldsymbol{\mu}_{x_t}\right), \tag{73}$$

$$\mathbf{Q}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \mathbf{Q}_{x_t} + \boldsymbol{\Phi}^T_t\mathbf{Q}_{\epsilon_t}\boldsymbol{\Phi}_t. \tag{74}$$

Note that the elements of $\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}$ are the basis function coefficients and covariate effect estimates in the co-Kriging predictor at t level:

$$\hat{m}_{Z_t}(\mathbf{s}) = \boldsymbol{\varphi}(\mathbf{s})\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}, \tag{75}$$

with squared error:

$$\hat{s}^2_{Z_t}(\mathbf{s}) = diag\left(\boldsymbol{\varphi}(\mathbf{s})\boldsymbol{\Sigma}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\boldsymbol{\varphi}^T(\mathbf{s})\right). \tag{76}$$

The method to estimate the hyper-parameter $\boldsymbol{\theta}_t$ as is the MLE. In practice the likelihood for $\mathbf{z}_t^{(n_t)}$ given the parameters $\boldsymbol{\theta}_t$ is:

$$\begin{aligned} log\,\pi\left(\mathbf{z}_t^{(n_t)}\middle|\,\boldsymbol{\theta}_t\right) = -\frac{n}{2}log\,(2\pi) + \frac{1}{2}log\ |\mathbf{Q}_{x_t}| + \frac{1}{2}log\ |\mathbf{Q}_{\epsilon_t}| - \frac{1}{2}log\ \left|\mathbf{Q}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right| \\ -\frac{1}{2}\left(\boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} - \boldsymbol{\mu}_{x_t}\right)^T \mathbf{Q}_{x_t}\left(\boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} - \boldsymbol{\mu}_{x_t}\right) \\ -\frac{1}{2}\left(\mathbf{Z}_t^{(n_t)} - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathfrak{D}}_t)\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_{t-1}})}} - \boldsymbol{\Phi}_t^T\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right)^T \mathbf{Q}_{\epsilon_t} \\ \left(\mathbf{Z}_t^{(n_t)} - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathfrak{D}}_t)\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_{t-1}})}} - \boldsymbol{\Phi}_t^T\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right). \end{aligned} \tag{77}$$

4 RELIABILITY MEASURE FOR DESIGN OPTIMISATION

The design process of a ducted propeller aims to estimate the performance of the propulsion system in various conditions. During the operation the loading of the blades can vary depending on the environmental conditions. Stemming from the manufacturing process, material and geometrical imperfections can cause performance disturbances.

Generally, the uncertainties can be classified into two categories: aleatory and epistemic [21]. Aleatory uncertainty is an inherent property of a natural process. Epistemic uncertainty is the impreciseness of our models stemming from the lack of knowledge. The latter type of uncertainty is not considered in this work. The aleatory uncertainty is modelled with random variables characterised by probability distributions. In the design optimisation context, the uncertainty on system responses due to input random variables and parameters is not known. In this work it is quantified with the Polynomial Chaos Expansion (PCE) which provides a sound mathematical tool to efficiently quantify probabilistic uncertainty. The probability space is spanned by a set of polynomials where the polynomial family depends on the probability distribution of the random variables [22].

Modelling of the probability space with PCE, it makes computationally affordable to calculate a risk measure for reliability-based optimisation using Monte Carlo sampling techniques. It is desirable to use risk measures that possess the properties of coherence and regularity to avoid the dependency on scaling and paradoxes. [23, 24]. Therefore, the conditional Value-at-Risk is employed here which is indeed a coherent and regular risk measure.

4.1 CONDITIONAL VALUE-AT-RISK

The conditional Value-at-Risk (cVaR) is also called superquantile and given by the Eq. (78):

$$\bar{q}_\alpha(Y) = \frac{1}{1-\alpha}\int_\alpha^1 q_\beta(Y)d\beta, \tag{78}$$

where Y is the random response and $q_\alpha(Y) = F^{-1}(Y)$ is the inverse cumulative distribution function of Y. The parameter α is the degree of risk averseness and is set to zero when the risks are indifferent and expected performance is seeked, while $\alpha = 1$ measures the worst-case scenario. The calculation of the cVaR can be generalised as a convex minimisation problem [23]:

$$\bar{q}_\alpha(Y) = \min_c c + \frac{1}{1-\alpha} E\left[\max(0, Y - c)\right]. \tag{79}$$

5 TRAINING DATA-SET UPDATE STRATEGY

The optimisation workflow is constructed similarly to the Efficient Global Optimisation strategy [25]. The Expected Improvement (EI) is calculated for the highest fidelity and new designs are calculated with the high-fidelity solver at locations where the maximal improvement is expected.

5.1 EXPECTED IMPROVEMENT

The EI of a location $\mathbf{x}$ measures how much improvement can be achieved by evaluating a new design at that location [26]. The formal representation assumes a minimisation problem of a function f:

$$\min f(\mathbf{x}), \tag{80}$$

where $\mathbf{x}\epsilon\mathbb{R}^n$. The unknown function f is modelled by a Gaussian Process and the prediction at $\mathbf{x}$ location is denoted $Y(\mathbf{x})$. The current minimum of the function is y_{min}. An improvement function can be defined as:

$$I(\mathbf{x}) = \max(y_{min} - Y(\mathbf{x}), 0) \tag{81}$$

The expected value of the improvement is:

$$EI(\mathbf{x}) = E\left[\max(y_{min} - Y(\mathbf{x}), 0)\right], \tag{82}$$

which can be reformulated into its closed form:

$$EI(\mathbf{x}) = (y_{min} - \mu(\mathbf{x}))\Phi\left(\frac{y_{min} - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right) + \sigma(\mathbf{x})\phi\left(\frac{y_{min} - \mu(\mathbf{x})}{\sigma(\mathbf{x})}\right), \tag{83}$$

where Φ is the cumulative distribution function, ϕ is the probability density function and erf is the error function:

$$\Phi(z) = \frac{1}{2}\left[1 + \mathrm{erf}(\frac{z}{\sqrt{2}})\right] \tag{84}$$

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}}\int_0^z e^{-t^2}dt \tag{85}$$

$$\phi(z) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{z^2}{2}\right) \tag{86}$$

6 RESULTS

6.1 ONE-DIMENSIONAL TEST CASE

A simple one-dimensional problem is investigated in this section. The test function for multi-fidelity surrogates were presented in [13]. The high- and low-fidelity functions are the following:

$$f_{high} = (6x-2)^2\sin(12x-4), \tag{87}$$

$$f_{low} = \frac{1}{2}f_{high} + 10(x-0.5) - 5. \tag{88}$$

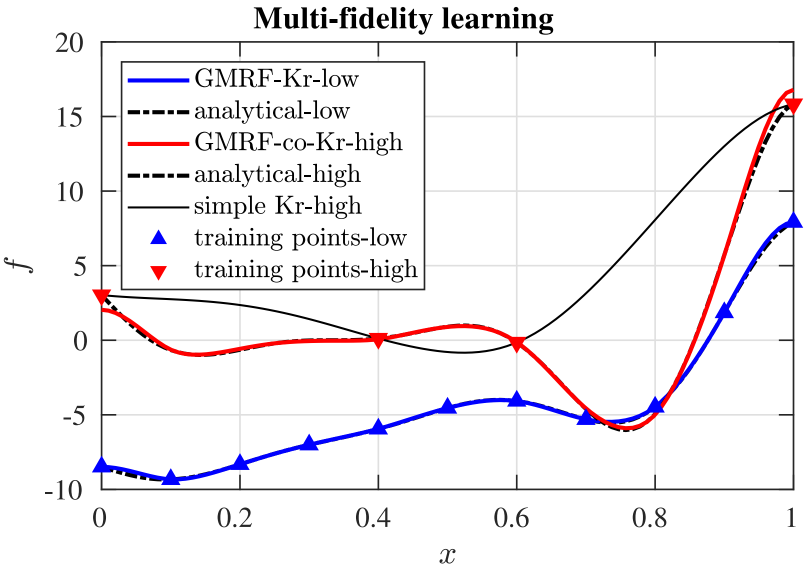


Figure 2: Multi-fidelity learning compared to single fidelity surrogate

In this case four observation are available at the high-fidelity level $X_{high} = \{0, 0.4, 0.6, 1\}$ and eleven at the low-fidelity level $X_{low} = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$. The surrogate built-on variable fidelity data is depicted in Figure 2. The result clearly shows that single-fidelity learning technique is not able to capture correctly the function landscape due to the limited number of observation points. The multi-fidelity learning technique is able to fuse the information from the low fidelity function into the high fidelity approximation and thus provides an adequate approximation of the true function. The multi-fidelity learning technique with GMRF is not able to properly learn the function landscape at the domain boundaries because Neumann boundary conditions with value zero are assumed. This results in a slightly higher approximation error compared to standard co-Kriging as it can be seen in Table 1.

	co-Kr-low	GMRF-co-Kr-low	co-Kr-high	GMRF-co-Kr-high
Mean Absolute Error	0.0389	0.0459	0.0852	0.1255

Table 1: Comparison of co-Kriging and GMRF-co-Kriging

6.2 SIMPLE DUCTED PROPELLER CASE

In this case study a design optimisation of a ducted propeller is considered. The problem is highly simplified and only two design parameters are considered: namely, the twist at the root and at the tip, see Figure 3. The geometry of the centre body and the duct is considered to be constant.The chord length is considered to be constant along the blade but with a zero mean Gaussian error. Also, the inflow velocity is loaded with a zero mean Gaussian error. These two uncertainties are considered to represent the manufacturing and environmental uncertainties respectively. The objective of the design problem is to maximise the expected efficiency (to get

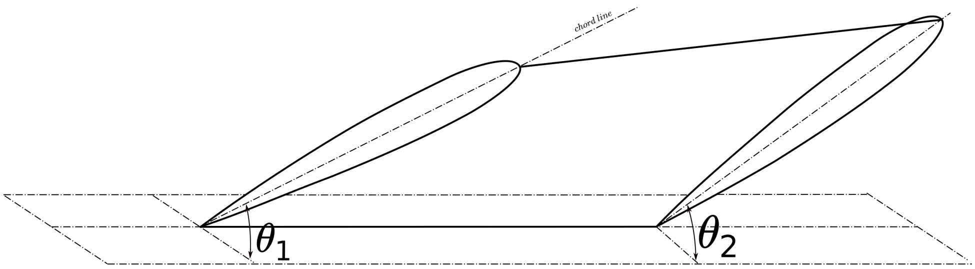


Figure 3: Design parameters of the propeller: twist at the root and at the tip.

the expected value the α parameter of the cVaR risk measure is set to zero):

$$\max_{\theta_{root},\theta_{tip}} \quad E[\eta], \tag{89}$$

where η is the total efficiency of the propeller and it is calculated as follows:

$$\eta = \frac{TV_\infty}{P}, \tag{90}$$

where TV_∞ is the useful power and P is the power absorbed.

For the low-fidelity calculations the Blade Element Momentum Theory (BEMT) is used and the high-fidelity analyses are conducted with the Ducted Fan Design Code (DFDC). Due to the inexpensiveness of the low-fidelity a full factorial data-set with 121 design are considered at low-level. Each design is evaluated 10 times and a second order full PCE is built to model the local probability space of the design. Clearly, DFDC is also an inexpensive solver compared to CFD but in this simple design scenario the available high-fidelity observation data is assumed to be limited. Only 4 design point are considered at the high-fidelity level. Similarly to the low-fidelity, each design point is evaluated 10 times to build a PCE to model the local probability space.

In this simple scenario the expected value is seeked which is exactly given by the first coefficient of the PCE. The GMRF-co-Kriging model learns from both the low- and high-fidelity data-set and constructs a surrogate model combining the information from both fidelities. From the Gaussian Process variance of the GMRF-co-Kriging model the EI can be calculated for the entire design space.

At the location of the maximal EI a new design point is evaluated and the GMRF-co-Kriging model is re-trained. This procedure is repeated until the maximal EI arrives below a threshold value ϵ. The optimisation workflow is depicted in Figure 4 and the learning history of the landscape of the objective space of the optimisation problem is shown in Figure 5.

7 CONCLUSION

Multi-fidelity learning can provide more accurate surrogate models than their single-fidelity counterparts. It is important to note that multi-fidelity learning is applicable only when the low-fidelity models carry sufficient information to enhance the model on the highest fidelity. In the field of aerospace engineering it is evident that many well-calibrated formula are available for low-fidelity evaluations since aircraft were designed even before the spread of sophisticated CFD techniques.

Kriging based multi-fidelity learning techniques are suffering from the fact that they require to invert large ill-conditioned covariance matrices. This drawback can be overcame by exploiting the link between Gaussian fields and Gaussian Markov random fields. This link allow us to

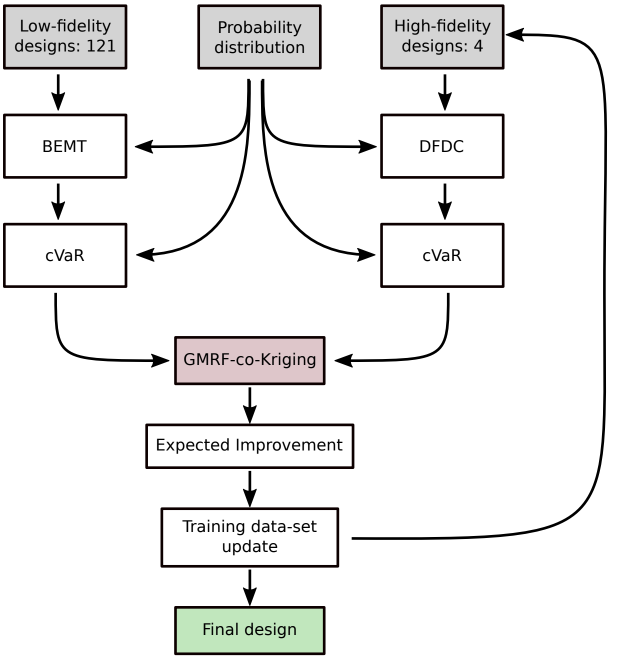


Figure 4: The optimisation workflow of the ducted propeller design optimisation

approximate the inverse of the covariance matrix with a sparse precision matrix and the advantages of finite element methods can be leveraged.
Currently, the authors are working on to include high-fidelity CFD simulations into the chain of fidelity hierarchy and to explore how much computational saving can be realised through multi-fidelity learning when real-world design problems are considered.

Acknowledgement

The work in this paper was partly supported by the H2020-MSCA-ITN-2016 UTOPIAE, grant agreement 722734.

REFERENCES

[1] R.J. Weir. Ducted propeller design and analysis. *Sandia National Laboratories*, 1987.

[2] D. Black, C. Rohrbach, Shrouded propellers-a comprehensive performance study. *5th Annual Meeting and Technical Display*, 1968.

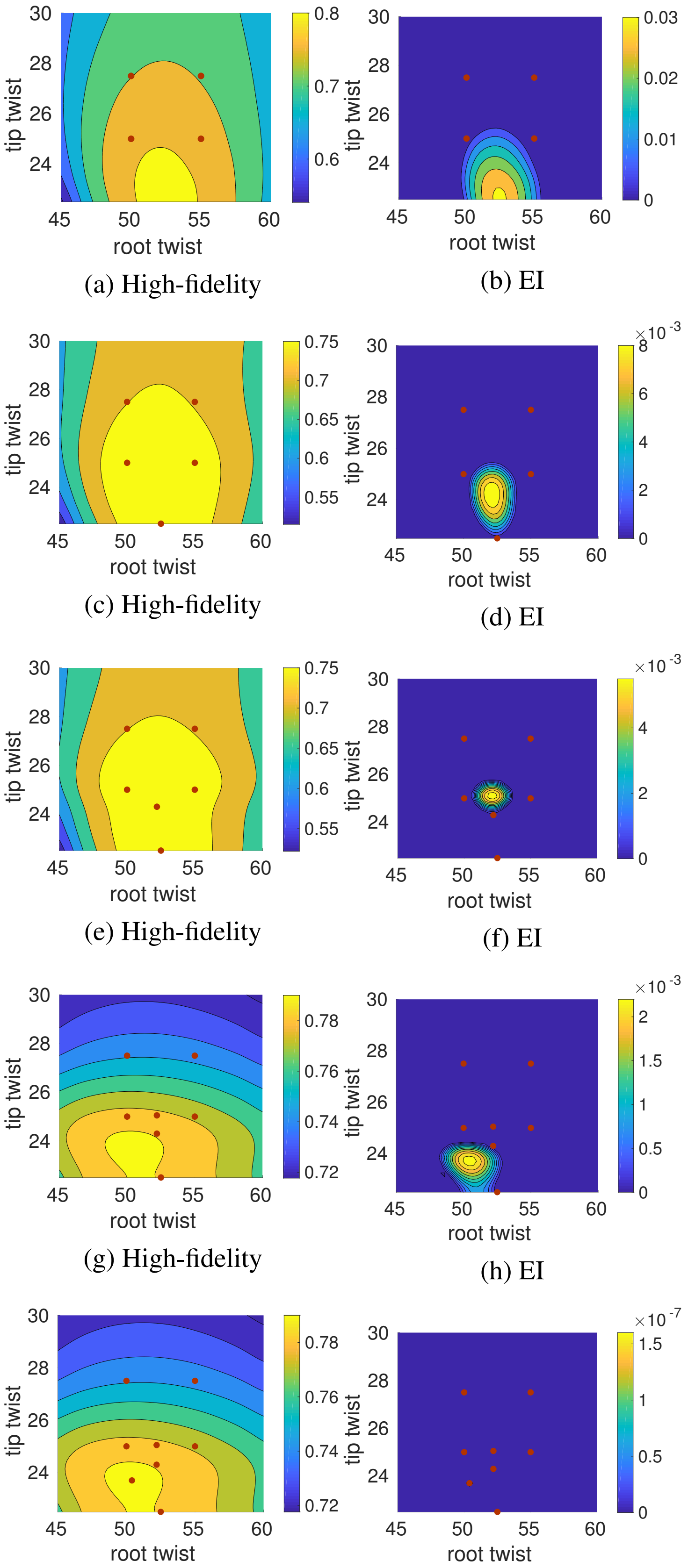
[3] A.F. El-Sayed. Aircraft propulsion and gas turbine engines. *CRC press*, 2017.

[4] H. Glauert. The elements of aerofoil and airscrew theory, *Cambridge University Press*, 1983.

[5] L.L.M.Veldhuis. Propeller wing aerodynamic interference, *PhD thesis*, 2005.

[6] M.O.L. Hansen. Aerodynamics of wind turbines. *Routledge*, 2015.

[7] M. Drela, H. Youngren, Ducted Fan Design Code (DFDC) - Axisymmetric Analysis and Design of Ducted Rotors, *Tech. Rep*, 2005



(a) High-fidelity

(b) EI

(c) High-fidelity

(d) EI

(e) High-fidelity

(f) EI

(g) High-fidelity

(h) EI

(i) High-fidelity

(j) EI

Figure 5: Learning history of the landscape of the objective space of the optimisation problem.

[8] N. Alexandrov, et al. Optimization with variable-fidelity models applied to wing design. *38th aerospace sciences meeting and exhibit*, 2000.

[9] M.C. Kennedy, A. O'Hagan. Predicting the output from a complex computer code when fast approximations are available. *Biometrika 87.1* 2000.

[10] L. Le Gratiet. Multi-fidelity Gaussian process regression for computer experiments. *Diss. Universit Paris-Diderot-Paris VII*, 2013.

[11] C.K.I. Williams, C.E. Rasmussen. Gaussian processes for machine learning. *Vol. 2. No. 3. Cambridge, MA: MIT Press*, 2006.

[12] N. Cressie. The origins of kriging, *Mathematical geology 22.3*, 1990.

[13] A.I.J. Forrester, A. Sóbester, and A.J. Keane. Multi-fidelity optimization via surrogate modelling. *Proceedings of the royal society a: mathematical, physical and engineering sciences*, 2007.

[14] L.N. Trefethen, D. Bau. Numerical linear algebra. *Vol. 50. Siam*, 1997.

[15] F. Lindgren, H. Rue, and J. Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2011.

[16] P. Perdikaris, et al. Multi-fidelity modelling via recursive co-kriging and GaussianMarkov random fields. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 2015.

[17] D. Xiu, G.E. Karniadakis. The Wiener–Askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing 24.2* , 2002.

[18] G. Blatman, B. Sudret. Adaptive sparse polynomial chaos expansion based on least angle regression. *Journal of Computational Physics*, 2011.

[19] P. Whittle. On stationary processes in the plane. *Biometrika, 434-449*, 1954.

[20] P. Whittle. Stochastic-processes in several dimensions. *Bulletin of the International Statistical Institute*, 1963.

[21] J.C. Helton, et al. Representation of analysis results involving aleatory and epistemic uncertainty. *International Journal of General Systems*, 2010

[22] A. Clarich, et al. Reliability-based design optimization applying polynomial chaos expansion: theory and applications. *10th World Congress on Structural and Multidisciplinary Optimization, Orlando, Florida, USA*, 2013.

[23] T. Rockafellar, J. ROYSET. Engineering decisions under risk averseness. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, 2015.

[24] D. Quagliarella, G. Petrone and G. Iaccarino. Optimization under uncertainty using the generalized inverse distribution function. *Modeling, Simulation and Optimization for Science and Technology (pp. 171-190), Springer, Dordrecht*, 2014.

[25] D.R. Jones, M. Schonlau, W.J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 1998.

[26] E. Rigoni, T. Montrone. Technical Report 2018-001, EGO Algorithm: General Description, *modeFRONTIER user guide*, 2018.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

UNCERTAINTY QUANTIFICATION OF OPTIMAL THRESHOLD FAILURE PROBABILITY FOR PREDICTIVE MAINTENANCE USING CONFIDENCE STRUCTURES

Adolphus Lye[1], Alice Cicirello[2], and Edoardo Patelli[1]

[1] Institute for Risk and Uncertainty, University of Liverpool
University of Liverpool, Chadwick Building, Liverpool L69 7ZF
e-mail: {adolphus.lye, epatelli}@liverpool.ac.uk

[2] Dynamics, Vibration and Uncertainty (DVU) Lab, University of Oxford
University of Oxford, Parks Road, Oxford OX1 3PJ
e-mail: alice.cicirello@eng.ox.ac.uk

Keywords: Uncertainty Quantification, Confidence Structures, Predictive Maintenance, Optimization, Maintenance Cost, Negative Binomial Distribution, Plasma Etching.

Abstract. *This paper seeks to analyze the imprecision associated with the statistical modelling method employed in devising a predictive maintenance framework on a plasma etching chamber. During operations, the plasma etching chamber may fail due to contamination as a result of a high number of particles that is present. Based on a study done, the particle count is observed to follow a Negative Binomial distribution model and it is also used to model the probability of failure of the chamber. Using this model, an optimum threshold failure probability is determined in which maintenance is scheduled once this value is reached during the operation of the chamber and that the maintenance cost incurred is the lowest. One problem however is that the parameter(s) used to define the Negative Binomial distribution may have uncertainties associated with it in reality and this eventually gives rise to uncertainty in deciding the optimum threshold failure probability. To address this, the paper adopts the use of Confidence structures (or C-boxes) in quantifying the uncertainty of the optimum threshold failure probability. This is achieved by introducing some variations in the p-parameter of the Negative Binomial distribution and then plotting a series of Cost-rate vs threshold failure probability curves. Using the information provided in these curves, empirical cumulative distribution functions are constructed for the possible upper and lower bounds of the threshold failure probability and from there, the confidence interval for the aforementioned quantity will be determined at 50%, 80%, and 95% confidence level.*

1 INTRODUCTION

Predictive maintenance (PdM) is the technique which seeks to predict the time in which a maintenance of an equipment is to be carried out through the monitoring of its operating conditions in real-time. A key advantage of PdM over the conventional practice of preventive maintenance (PM) is that PdM allows for the lowering of maintenance costs owing to the fact that maintenance is conducted only when necessary instead of on a routine basis as observed in the case of PM [1]. This is the basis on which the literature by Duc *et al.* [2] is written.

In this conference paper, a case-study based on this literature will be presented in section 2.1 with a study of the statistical modelling method employed in devising a PdM framework on a plasma etching chamber. This devised framework seeks to determine an optimized threshold failure probability (PrF) beyond which, maintenance is performed on the chamber. The statistical model adopted, however, contains a parameter which was determined with a significant degree of uncertainty. As such, the research methodology proposed in this literature aims to investigate the effect of this uncertainty in determining the optimum PrF and the quantification of its associated uncertainty. This is achieved using Confidence boxes (or C-boxes) to which details will be provided in section 3.1. Using this tool, the 2-sided confidence interval of the optimum threshold PrF can be determined and it will be attained at 50%, 80%, and 95% confidence level.

2 CASE-STUDY: PLASMA ETCHING CHAMBER

2.1 Background

In his paper, Duc starts off by highlighting the key reasoning behind the decrease in the production yield of the plasma etching chamber, and therefore its reliability, being the presence of particles on the wafer [3]. These particles are generated as by-product of the plasma etching process [4] and its amount can be monitored via the Particle per Wafer Pass (PWP) method [5]. However, only particles exceeding a specified size are recorded in the total particle count.

Next, Duc identifies the stochastic nature associated with the total particle count based on data obtained over 8-months of chamber operation and models it to follow a Negative Binomial distribution [6] as shown below:

$$P(Y = y) = \frac{\Gamma(y + r)}{y! \cdot \Gamma(r)} \cdot p^{r} \cdot (1 - p)^{y} \tag{1}$$

In Equation (1), y indicates the random variable for particle count, r indicates the number of runs by the chamber in which the particle count is zero, p indicates the probability of the particle count being zero within a single run, and $\Gamma()$ represents the Gamma function [7]. According to the literature, the value of r was determined to be 2.2608 (rounded down to 2) while that of p was determined to be 0.039 ($\pm 30\%$) [2]. Based on these information, he then derives a cumulative distribution function to model the PrF of the chamber. Here, failure is defined as the event in which the particle count reaches or exceeds a certain threshold value, k_t. As such, the mathematical expression for the PrF is as follows:

$$P(Y \geq k_t) = \sum_{k=k_t}^{\infty} P(Y = k) \tag{2}$$

To ensure cost-effectiveness, a threshold PrF has to be set such that upon reaching this value, the plasma etching chamber undergoes a scheduled maintenance. This threshold value of PrF cannot take values which are either too small or too large. A small threshold value would mean that maintenance would now have to be performed more frequently resulting in high maintenance costs. A large threshold value, on the other hand, would imply that the chamber is allowed to operate for a longer duration without maintenance but this increases the occurrence of failure due to high particle count which leads to a loss in production yield and earnings. As such, Duc proceeds to devise a method to determine the optimum value of threshold PrF analytically.

2.2 Methodology

In the literature, he introduces a new quantity called the Cost-rate, g, which is defined to be the mean total costs associated with maintenance and particle count failures, C, divided by the total time between two successive scheduled maintenance, T. This value of T would depend on the value of PrF that is set. Further details to the calculation of C and T can be found in reference [2]. From there, Duc proceeds to tabulate the values of g for the respective values of threshold PrF and plots a graph to illustrate the relationship between these two quantities. The result is as shown in the next section.

2.3 Results

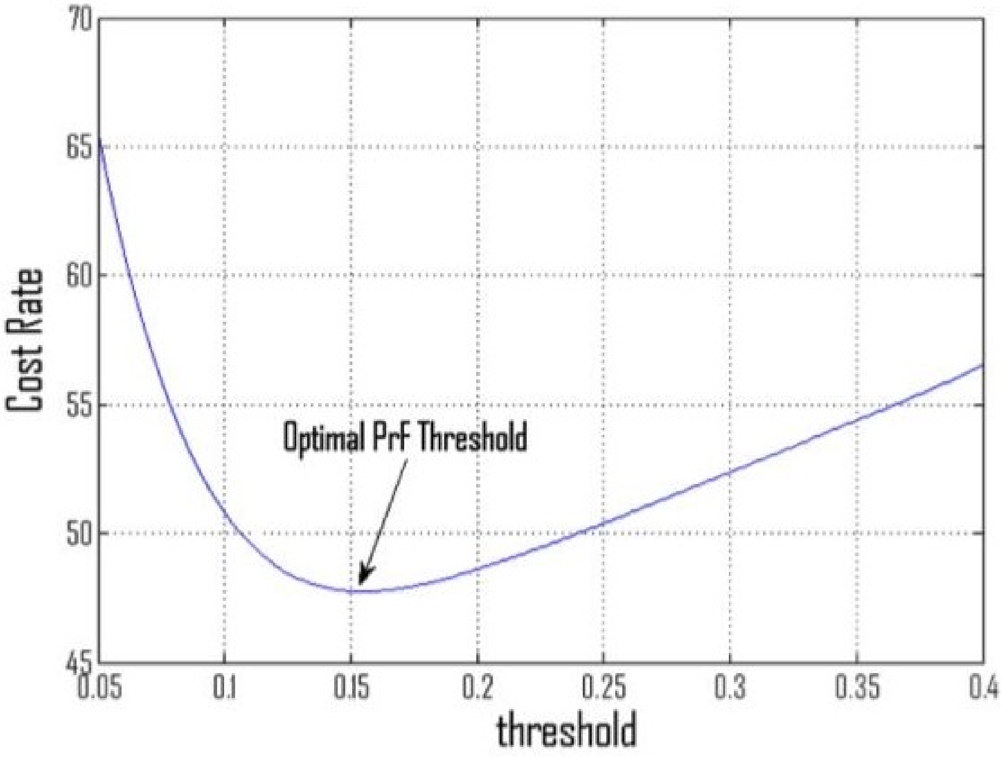


Figure 1: Graph of Cost-rate, g, against threshold PrF. Image obtained from [2].

Based on the results illustrated in Figure 1, the optimum threshold PrF corresponds to the value at which the C is at a minimum. From this, the optimum threshold PrF is determined to be 0.16 [2]. This implies that a scheduled maintenance is performed once the PrF of the plasma etching chamber reaches 16% during its operation.

3 PROPOSED FRAMEWORK

3.1 Concept of C-boxes

C-boxes are structures which serve to provide a generalized approach in producing confidence distributions. They can be used as a tool to estimate values of fixed, real-valued quantities obtained through random sampling and contains information of its Neyman-Pearson confidence at every level of confidence [9]. Unlike traditional confidence intervals, C-boxes can be propagated via mathematical calculations and can be used in calculations to produce results with identical confidence interval interpretation. From there, it is able to reflect both the uncertainty associated with the sampled quantity which stems from the process of inferring observations as well as the effects of imprecision in the data and demographic uncertainty which comes from the process of characterizing a continuous parameter based on discrete observations. One significant advantage of C-boxes is that it can be constructed even if the distribution of the sampled quantity is unknown [10] which makes it the favoured method to perform uncertainty quantification for the purpose of this research. More details to the theory of C-boxes can be found in reference [9].

A C-box can be constructed using one of the two forms of cumulative distribution functions (CDFs): Continuous or Empirical. In this paper, the latter is used given that the distribution to which the parameter p follows is unknown. Empirical CDFs can be described as a step-function which jumps up by $\frac{1}{n}$ unit of probability at each of the n data points where n is the total number of data available [11]. Like continuous CDFs, the value of the cumulative probability increases with the value of the data point. A simple illustration of an empirical CDF is provided below as an example:

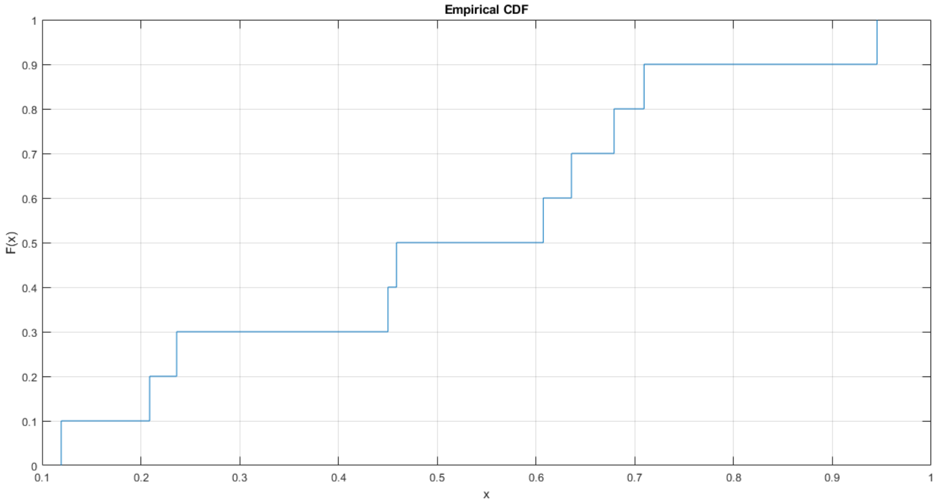


Figure 2: Empirical CDF curve obtained for $x \in \{0.1994, 0.2089, 0.2362, 0.4501, 0.4587, 0.6073, 0.6358, 0.6790, 0.7093, 0.9452\}$. As seen in the image, the cumulative probability, $F(x)$, increases by $\frac{1}{10}$ with each value of x.

For a set of data x with its associated uncertainty, each data consists of a lower bound, $\underline{x}$, and an upper bound value, $\overline{x}$. Based on these information, the empirical CDF for the lower bound values (the Belief function), $\underline{F}(\underline{x})$, and the upper bound values (the Plausibility function), $\overline{F}(\overline{x})$, can be plotted simultaneously. This gives rise to an interval which exists between $\underline{F}(\underline{x})$ and $\overline{F}(\overline{x})$, within which the empirical CDF of the actual value of x could possibly exist. This resulting plot is a C-box structure. To determine a two-sided confidence interval of a parameter θ at $1-\alpha$ confidence level, $\underline{\theta}$ is such that $\underline{F}(\underline{\theta}) = \frac{\alpha}{2}$ while $\overline{\theta}$ is such that $\overline{F}(\overline{\theta}) = 1 - \frac{\alpha}{2}$.

3.2 Methodology

The proposed methodology in this conference paper seeks to supplement the work by Duc *et al.* by introducing some "noise" into a key parameter within the Negative Binomial distribution model and then observing its effects in the determination of the optimum threshold PrF and its associated uncertainty. As seen in Equation (1), the Negative Binomial distribution is defined by two parameters: r and p, whose respective values can be found in section 2.1. Given that it was mentioned that the value of p has an upper and lower bound of 30%, this provides a degree of uncertainty in p which can be adopted to realize the outcome of this research.

To perform the investigation, 20 sets of values of threshold PrF and Cost-rate will first be extracted from the Cost-rate vs threshold PrF curve in Figure 1 using the Getdata Graph Digitizer software [8]. The obtained values of threshold PrF will serve as references to obtain the approximate values of k_t from the PrF function shown in Equation 2 with parameters $r = 2$ and $p = 0.039$. This process is done using Wolfram Mathematica and the results are presented in Table 1 of section 3.3.

Next, an assumption will be made that the Cost-rate, g, is only affected by k_t. This is because physically, it is the amount of particles which will determine the machine down-time due to the duration of cleaning. In essence, the more particles there are, the longer time it takes to clean, the longer the down-time and the higher the costs incurred due to failure (excluding the costs from scheduled maintenance), thus increasing g. With this in mind, new threshold PrF values will be obtained for the respective values of k_t via Equation (2) for different values of p while r is kept at 2. The values of p are chosen such that they are within the aforementioned upper and lower bounds with respect to the derived value of 0.039. As such, p will take values ranging between 0.028 to 0.050 in steps of 0.002. For each value of p, a graph of Cost-rate, g, vs threshold PrF will be plotted in similar fashion to Figure 1. This would yield a family of curves for all 13 values of p as shown in Figure 3 of section 3.3. From there, the optimum threshold PrF for each of these curves and its range of values of the threshold PrF will be determined graphically. The results are summarized in Table 2 of section 3.3.

Finally, to perform the necessary uncertainty quantification associated with the optimum threshold PrF values, C-boxes will be constructed with empirical CDFs of the lower bounds, upper bounds, and the optimum threshold PrF using the data presented in Table 2. The resulting C-box diagram is illustrated in Figure 4 of section 3.3.

3.3 Results

Threshold PrF	k_t	Threshold PrF	k_t
0.05642	115	0.18401	78
0.06426	111	0.20825	73
0.07424	107	0.22607	71
0.08422	103	0.24817	67
0.09420	99	0.26955	65
0.10346	96	0.29735	61
0.11986	91	0.31731	59
0.13055	89	0.33299	57
0.14908	84	0.36365	54
0.16548	81	0.39644	51

Table 1: Results for k_t and its respective values of threshold PrF for the default parameter values of $r = 2$ and $p = 0.039$.

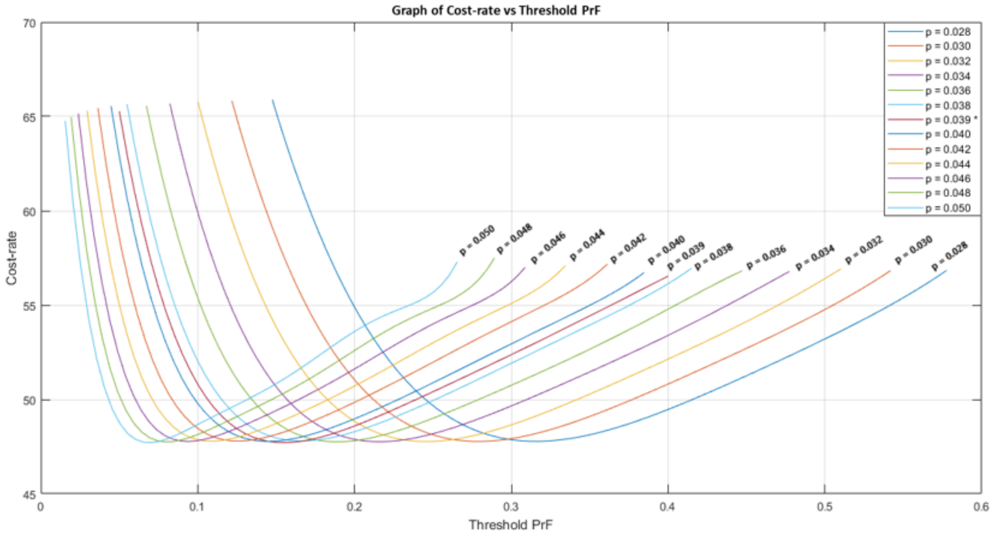


Figure 3: Graph of Cost-rate, g, against threshold PrF for different values of p. r is kept constant at 2.

Value of p	Optimum Threshold PrF	Threshold PrF range
0.028	0.31552	[0.14756 0.58013]
0.030	0.27888	[0.12183 0.54516]
0.032	0.24572	[0.10026 0.51138]
0.034	0.21584	[0.08227 0.47887]
0.036	0.18879	[0.06732 0.44772]
0.038	0.16605	[0.05495 0.41798]
0.039	**0.15484**	**[0.05000 0.40000]**
0.040	0.14490	[0.04474 0.38966]
0.042	0.12559	[0.03635 0.36278]
0.044	0.10863	[0.02947 0.33732]
0.046	0.09361	[0.02385 0.31328]
0.048	0.08042	[0.01926 0.29062]
0.050	0.06898	[0.01553 0.26931]

Table 2: Results for optimum threshold PrF and the range of threshold PrF for the respective values of p.

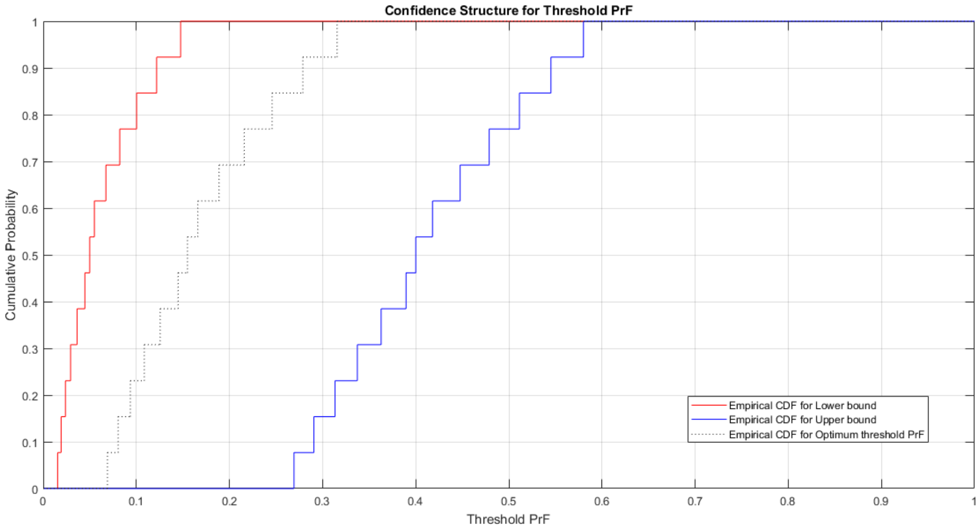


Figure 4: Confidence structure (C-boxes) summarizing the data in Table 2.

3.4 Discussion

Based on the C-box constructed, the confidence interval for the value of threshold PrF can be determined at 50%, 80%, and 95% confidence level and the results are summarized in Table 3 below:

Confidence level	Confidence interval of threshold PrF
50%	[0.02947 0.47887]
80%	[0.01926 0.54516]
95%	[0.01553 0.58013]

Table 3: Confidence interval of threshold PrF for the respective level of confidence.

The results above could serve as a guide for the industry in the decision–making of suitable value of threshold PrF under uncertainty and from there, devise and compare the numerous PdM plan for the plasma etching chamber based on the range of threshold PdF chosen as well as the respective maintenance costs associated with the respective PdM plan.

4 CONCLUSION

This paper has addressed the problem of quantifying the uncertainty associated with the optimum threshold PrF as a result of the uncertainty in the determination of p which is a key parameter of the Negative Binomial distribution model as seen in Equation (1). In summary, the threshold PrF values are calculated for the respective values of k_t using different values of p whilst assuming that the Cost–rate is only affected by k_t. From there, the Cost–rate vs threshold PrF curves are plotted for the different values of p and the information illustrated in the family of curves is then used to construct the C-box structure. Using the C-box structure, the confidence interval of the optimum threshold PrF value is obtained at 50%, 80%, and 95% confidence level.

ACKNOWLEDGEMENTS

This research is made possible with the guidance and insights to the concepts from Dr Alice Cicirello and Dr Edoardo Patelli who are also co-authors of this paper. In addition, we would also like to express our sincere gratitude to the Institute for Risk and Uncertainty for providing the facilities and resources necessary for this research to take place. Finally, we would also like to acknowledge the Singapore Nuclear Research and Safety Initiatives (SNRSI) for the sponsorship and the financial support without which the undertaking of this PhD programme would not have been possible.

REFERENCES

[1] R. K. Mobley, *An Introduction to Predictive Maintenance, Second Edition*. Amsterdam: Butterworth-Heinemann, 2002. ISBN: 9780750675314.

[2] L. M. Duc, C. M. Tan, M. Luo, & I. C. Leng, Maintenance Scheduling of Plasma Etching Chamber in Wafer Fabrication for High-Yield Etching Process. *IEEE Transactions on Semiconductor Manufacturing*, **27(2)**, 204–211, 2014. doi: 10.1109/tsm.2014.2304461

[3] T. Moriya, H. Nakayama, H. Nagaike, Y. Kobayashi, M. Shimada, & K. Okuyama, Particle Reduction and Control in Plasma Etching Equipment. *IEEE Transactions on Semiconductor Manufacturing*, **18(4)**, 477–486, 2005. doi: 10.1109/tsm.2005.858464

[4] H. Jansen, H. Gardeniers, M. D. Boer, M. Elwenspoek, & J. Fluitman, A Survey on the Reactive Ion Etching of Silicon in Microtechnology. *Journal of Micromechanics and Microengineering*, **6(1)**, 14–28, 1996. doi: 10.1088/0960-1317/6/1/002

[5] K. Reinhardt, & W. Kern, *Handbook of Silicon Wafer Cleaning Technology, Third Edition*. William Andrew, 2018. ISBN: 9780323510844.

[6] R. A. Fisher, The Negative Binomial Distribution. *Annals of Human Genetics*, **11(1)**, 182–187, 1941. doi: 10.1111/j.1469-1809.1941.tb02284.x

[7] P. J. Davis, Leonhard Eulers Integral: A Historical Profile of the Gamma Function. *The American Mathematical Monthly*, **66(10)**, 849–869, 1959. doi: 10.1080/00029890.1959.11989422

[8] H. Zein, V. L. H. Tran, A. Azmy, A. T. Mohammed, A. M. Ahmed, A. Iraqi, & N. T. Huy, How to Extract Data from Graphs using Plot Digitizer or Getdata Graph Digitizer. 2015. doi: 10.13140/RG.2.2.17070.72002

[9] M. S. Balch, Mathematical Foundations for a Theory of Confidence Structures. *International Journal of Approximate Reasoning*, **53(7)**, 1003-1019, 2012. doi: 10.1016/j.ijar.2012.05.006

[10] S. Ferson, M. Balch, K. Sentz, & J. Siegrist, Computing with Confidence: Imprecise Posteriors and Predictive Distributions. *Sixth International Symposium on Uncertainty, Modeling, and Analysis (ISUMA)*, 895–904, 2014. doi: 10.1061/9780784413609.091

[11] A. W. Van der Vaart, *Cambridge Series in Statistical and Probabilistic Mathematics: Asymptotic Statistics*. Cambridge University Press, 1998. ISBN: 0521496039.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

IMPROVED FLOW PREDICTION IN INTRACRANIAL ANEURYSMS USING DATA ASSIMILATION

F. Schulz[1], C. Roloff[1], D. Stucht[2], D. Thévenin[1], O. Speck[2] and G. Janiga[1]

[1]Department of Fluid Dynamics and Technical Flows, Otto-von-Guericke University Magdeburg,
Magdeburg, Germany
e-mail: {franziska.schulz,christoph.roloff,janiga,thevenin}@ovgu.de

[2] Department of Biomedical Magnetic Resonance, Otto-von-Guericke University Magdeburg,
Magdeburg, Germany
e-mail: {oliver.speck,daniel.stucht}@ovgu.de

Keywords: CFD, Data Assimilation, Hemodynamics, Intracranial Aneurysm, PC-MRI

Abstract. *Rupture of intracranial aneurysms often leads to irreversible disabilities or even death. The investigation of hemodynamics increases the understanding of cardiovascular diseases, this gain of knowledge can support physicians in outcome prediction and therapy planning. Hemodynamic simulations are restricted by modeling assumptions and uncertain initial conditions, whereas PC-MRI data is affected by measurement noise and artifacts. To overcome the limitations of both techniques, the current study uses a Localization Ensemble Transform Kalman Filter (LETKF) to incorporate uncertain Phase-Contrast MRI data into an ensemble of numerical simulations. The analysis output provides an improved state estimate of the three-dimensional blood flow field. Benchmark measurements are carried out in a silicone phantom model of an idealized aneurysm under user-specific inflow conditions. Validation is ensured with high-resolution Particle Imaging Velocimetry (PIV) obtained from a vertical slice in the center of the same geometry. Results show that even velocity peaks smaller than the PC-MRI resolution can be reconstructed using the employed approach. The root mean square error (RMSE) of the analysis state estimate is reduced by 27 % to 89 % in comparison to interpolation of the PC-MRI data onto the PIV grid resolution.*

1 INTRODUCTION

Computational Fluid Dynamics (CFD) has been frequently used in past studies for the investigation of hemodynamics in intracranial aneurysms. Flow-dependent parameters, such as wall shear stress or the oscillary shear index, are meant to have an influence on the growth and rupture probability of aneurysms [1–3]. However, such simulations require accurate initial and boundary conditions, and depend on the assumptions used in the numerical model [4–6]. Although high temporal and spatial resolution can be reached, uncertainties lead to a limited clinical acceptance of the simulation results [7–9]. Phase-Contrast Magnetic Resonance Imaging (PC-MRI) *in-vivo* measures blood flow by encoding the velocities in the phase of the acquired MR signal [10–12]. In addition to measurement noise and artefacts, the limited temporal and spatial resolution impact clinically-relevant flow features in the measurement data. The present study incorporates uncertain measurement data into numerical simulations by using data assimilation to improve the accuracy and physical correctness of the measured velocity fields. An Ensemble Kalman Filter technique samples the system state and covariance matrices by an ensemble of model states. Thus, the covariance matrices are not calculated directly, but estimated through an ensemble and replaced by the sample covariance. The background uncertainty is estimated and the Ensemble Kalman Filter (EnKF) can be seen as a Monte-Carlo approximation of the original Kalman Filter [13].

Several attempts have been made to improve the accuracy of intracranial velocity fields acquired from PC-MRI data. Whereas de-noising techniques, as well as divergence-free filtering approaches, improve the physical correctness of the velocity field, spatial and temporal resolution remains low e.g. [14, 15]. Hence, data assimilation seems to be a promising remedy to improve resolution while keeping constraints, such as incompressibility and conservation laws. Variational data assimilation approaches in intracranial anerysms have been applied by D'Elia et al. [16–18] and Funke et al. [19]. As an alternative to the Ensemble Kalman Filters they minimize the error between observations of a reference flow and a numerical estimation in terms of a cost function. The need for linear and adjoint models increases computational complexity by a factor of 50 to 100 in comparison to one simple model simulation. As a consequence, most numerical studies on variational data assimilation in intracranial aneurysms currently addresses steady-state flow and/or 2D geometries. Funke et al. [19] investigates transient 3D flow fields, but to keep computational costs in an acceptable range, spatial resolution is decreased. Although promising results have been achieved for other fluid dynamical applications e.g. [20, 21], little attention has been payed to the sequential Kalman Filters in intracranial aneurysm modeling. Bakhshinejad et al. implemented an Extended Kalman Filter for pulsatile cardiac flow [22]. Nevertheless, the lack of localization requires a large amount of ensemble members to ensure filter convergence. The resulting high computational costs illustrate the need for a sequential data assimilation technique that can gain convergence with a limited amount of ensemble simulations.

Although the current study deals with steady-state 3D flow, this is the first approach using a Localization Ensemble Transform Kalman Filter to assimilate CFD and PC-MRI data for improved flow prediction in intracranial aneurysm. The study comes along with a systematic analysis of parameters inside the algorithm. Additional uniqueness is ensured with a high quality PIV measurement of the same geometry which enables proper quantitative validation of the assimilation step.

2 MATERIAL and METHODS

Underlying CFD and measurement data for the assimilation step are obtained from the same silicon phantom model of an idealized intracranial aneurysm (figure 1a). Afferent and efferent vessels have a diameter of 4 mm. The phantom model, consisting of two-component silicon (Wacker RT 601, Burghausen, Germany), is well suited because it allows blood flow measurements with the MR device, as well as optical based PIV measurements. A blood substitute, which was required to match both the fluid-dynamical properties of real blood as well as the refractive index of the silicone block (=1.4122 at 22°C) for optimal optical conditions within the validation PIV measurements, was formulated.

2.1 Experimental Set-Up

The flow data was acquired on a 7 Tesla whole-body MRI system (Siemens Healthineers, Forchheim, Germany) in a 32-channel head coil (Nova Medical, Wilmington, MA) using 4D phase-contrast magnetic resonance imaging (PC-MRI). Hereby, the acquisition sequence is based on a rf-spoiled gradient echo with quantitative flow encoding in all three spatial dimensions [23, 24]. A micro-gear pump (HNP Mikrosysteme, Schwerin, Germany), placed in the control room of the MRI scanner, delivered a constant flow rate of Q=227 mL/min throughout the measurements (figure 1b). This relatively low flow rate was chosen to ensure laminar flow inside the aneurysm which was needed for accurate validation of the data assimilation parameters. A total scan time of approx. 9 minutes achieves a resolution of 0.57x0.57x0.57 mm in the resulting phase difference images. The same measurement but without activated pump and thus without flow inside the aneurysm-phantom was acquired for reference. The reference data was subtracted from the flow data to obtain purely flow related phase differences. As the flow information is encoded in the phase of the complex MR-Signal, a velocity encode parameter (venc) is necessary to specify the highest velocitiy, encoded in one complete phase. For the aquired data, this parameter was set to 0.6 m/s. As a consequence, the signal-to-noise SNR of the acquired images was calculated using the mean of the signal inside the aneurysm and the signal density of the background noise and was found to be SNR ≈ 55. The data was post processed using MeVisLab 2.3.1 and the automated tool described in [25]. This includes noise masking, antialiasing and conversion to the format of the commercial software package EnSight (ANSYS Inc., Canonsburg, PA, USA).

2.2 Numerical Background

The ensemble boundary conditions for the CFD simulations are obtained from the MRI blood flow measurements with a specific mean (228 ml/min) and variance (10 ml/min). A structured hexaedral mesh is created using ANSYS IcemCFD (ANSYS Inc., Canonsburg, PA, USA) resulting in approx. 171.000 cells. Ensemble simulations are carried out using the open source software OpenFOAM 5.0 (OpenCFD Ltd., Bracknell, UK). Blood is treated as an isothermal, incompressible fluid ($1222\ \mathrm{kg/m^3}$) and Newtonian behavior with a constant dynamic viscosity (4.03 mPa s) is assumed. The vessel walls are assumed to be rigid and no-slip boundary conditions and a zero pressure outlet are implemented. Convergence was obtained when the scaled residuals of pressure and momentum decreased below a value of 10^{-6}.

2.3 Data Assimilation Algorithm

The Local Ensemble Transform Kalman Filter applied in this paper was originally introduced by Harlim and Hunt [27, 28] in the field of meteorology and combines the localization method

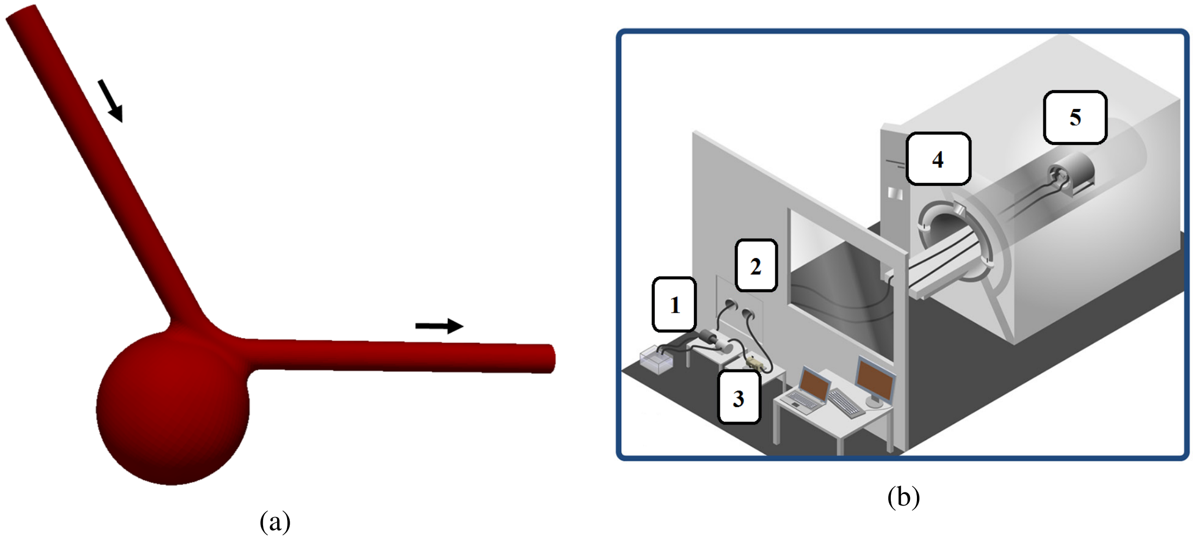


Figure 1: (a) Surface model of the idealized aneurysm used for phantom manufacturing and CFD discretization; (b) MRI setup with the gear pump (1), wave guide through rf-shield (2), flow meter (3), MR-scanner (4) and the 32-channel head coil with the phantom (5) [26].

of the Local Ensemble Kalman Filter (LEKF) of Ott et al. [29] and the Ensemble Transform Kalman Filter (ETKF) of Bishop et al. [30]. The analysis ensemble is formed as a weighted average of the background ensemble mean and the observations. Using background and observation uncertainties, the weights are determined in a way, such that the analysis ensemble mean best fits the given background and observation probability distributions.

With the implementation of the localization, local analyses at each model grid point are obtained. Only observations within a local region surrounding the grid point are accounted for the desired local analysis. The localization scheme enables efficient parallel computation of the analysis model state and limits the number of needed ensemble members. For the current data assimilation experiment a localization radius of 3 mm was chosen.

The following section contains a short summary of the LETKF algorithm. The inputs of the steps below are:

- m-dimensional velocity vectors describing the backgorund ensemble $\{\mathbf{x}^{b(i)} : i = 1, 2, ..., k\}$ at m grid points for k ensemble members. The ensembles are calculated by k different CFD simulations.
- The s-dimensional observation vector $\mathbf{y}^o$ in the form of a velocity vector obtained from the PC-MRI measurement.
- An observational operator H to map the state variables from the m-dimensional simulation space to the s-dimensional observation space. In the current data assimilation experiment observations, as well as the background ensemble are velocity values. Therefore, the current observation operator H is a spatial binning operator which downsamples the ensemble velocity vectors to the MRI grid resolution.
- An $s \times s$ dimensional observation error covariance matrix $\mathbf{R}$ based on the noise characteristics of the measurement data.

In a first step global transformations are performed with the background ensemble members. Form $\{\mathbf{x}^{b(i)}\}$ into an $m \times k$ dimensional matrix $\mathbf{X}$ and average the columns to get an

m-dimensional vector $\bar{\mathbf{x}}^b$. Substract this vector from each column of $\mathbf{X}$ to get $\mathbf{X}^b$.

$$\begin{aligned}\bar{\mathbf{x}}^b &= k^{-1}\sum_{i=1}^{k}\mathbf{X}^{(i)} \\ \mathbf{X}^{b(i)} &= \mathbf{X}^{(i)} - \bar{\mathbf{x}}^b\end{aligned} \tag{1}$$

By applying the spatial binning observational operator H to each column of $\mathbf{X}$, the state vector in the model space is transfered to the observational space, followed by a repetition of the previous transformations (equation (1)) with the resulting matrix.

$$\begin{aligned}\mathbf{Y} &= H_l(\mathbf{X}) \\ \bar{\mathbf{y}}^b &= k^{-1}\sum_{i=1}^{k}\mathbf{Y}^{(i)} \\ \mathbf{Y}^{b(i)} &= \mathbf{Y}^{(i)} - \bar{\mathbf{y}}^b\end{aligned} \tag{2}$$

From this point on local calculations at each model grid point *j* can be performed, which results in faster convergence of the algorithm due to parallel computations. For each *j* observations are chosen to be used in the local analysis of a certain grid point. The analysis error covariance matrix $\tilde{\mathbf{P}}^a(j)$ is calculated and used to compute the weight vector $\mathbf{w}^a(j)$.

$$\begin{aligned}\tilde{\mathbf{P}}^a(j) &= \left[(k-1)\mathbf{I}/(1+r) + \left[\mathbf{Y}^b(j)\right]^T \mathbf{R}^{-1}(j)\mathbf{Y}^b(j)\right]^{-1} \\ \mathbf{w}^a(j) &= \tilde{\mathbf{P}}^a(j)\left\{\left[\mathbf{Y}^b(j)\right]^T \mathbf{R}(j)^{-1}\left[\mathbf{y}^o(j) - H(\bar{\mathbf{x}}^b)(j)\right]\right\}\end{aligned} \tag{3}$$

The desired amount of multiplicative covariance inflation *r* = 1.05 is added to increase the background error. This avoids underestimating the background uncertainty with a small ensemble size.

$$\begin{aligned}\mathbf{W}^a(j) &= [(k-1)\tilde{\mathbf{P}}^a(j)]^{1/2} \\ \mathbf{W}(j) &= \mathbf{W}^a(j) + \mathbf{w}^a(j)\end{aligned} \tag{4}$$

With the weight vector and perturbations, the analysis mean state can be calculated. In the current study it provides an improved state estimate for the velocity field inside the idealized aneurysm geometry. In addition to that, the analysis ensemble members are formatted, which can be used as initial conditions for ensemble simulations in the subsequent step of a transient data assimilation experiment.

$$\begin{aligned}\bar{\mathbf{x}}_l^a(j) &= \bar{\mathbf{x}}_l^b(j) + \mathbf{X}_l^b(j)\mathbf{w}^a(j) \\ \{\mathbf{x}_l^{a(i)}(j)\} &= \mathbf{X}_n^b(j)\mathbf{X}_l^b(j)\mathbf{W}(j) + \bar{\mathbf{x}}_l^b(j)\end{aligned} \tag{5}$$

2.4 Validation

The generation of a quantitative gold standard is one of the main challenges in the formulation of a suitable data assimilation experiment. High-resolution stereoscopic PIV measurement [26] obtained from a vertical slice in the center of the idealized aneurysm geometry provide a unique possibility for validation. The resulting PIV based velocity fields are sufficiently accurate to validate the data assimilation procedure. The Root-Mean Square Error (RMSE) between PIV and MRI or analysis, respectively is calculated. To further validate the calculated analysis with respect to physical accuracy, the divergence of the velocity field is calculated. The incompressible flow field inside the aneurysm should fullfill $\mathrm{div}(\overrightarrow{v}) = 0$.

To enable reasonable comparisons between the different modalities, the resulting data were registered with the implementation of an Iterative Closest Point (ICP) algorithm. Difficulties in the registration process rises due to geometric distortions in the PC-MRI data. These artefacts increase with the distance to the measurement center, which was chosen to be in the center of the aneurysm sack. For registration purposes, the distorted parts of afferent and efferent vessels are cut, which results in an improved registration of the volume of interest.

3 RESULTS

3.1 PC-MRI data

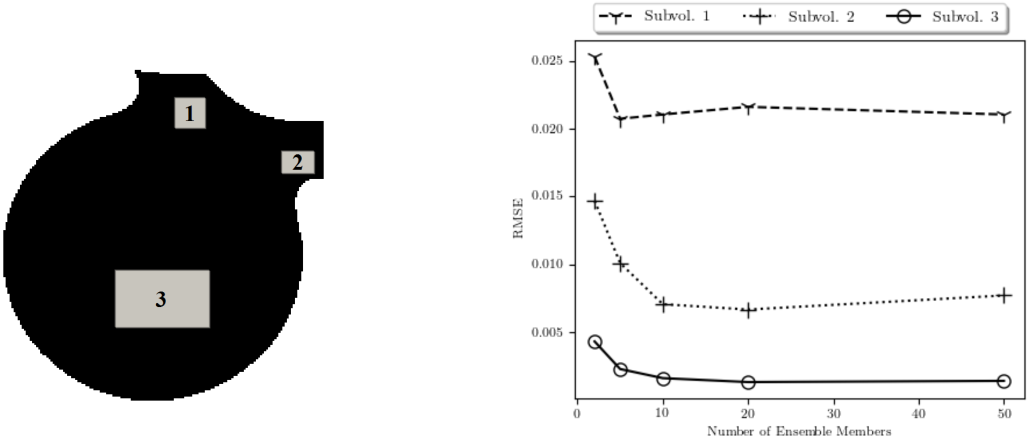
Figure 3a presents the flow field as well as divergence distribution at different slices acquired from the PC-MRI data. Due to the chosen small flow rate (Q=227 mL/min) laminar flow is ensured inside the aneurysm. The velocity fields suffer from acquisition noise and low spatial resolution, both make an accurate definition of the geometric boundaries difficult. In a laminar, incompressible flow field, the divergence should be zero. Data acquired by Phase-Contrast MRI measurements does not automatically fullfill this constraint. Divergence calculated at different slices in the aneurysm geometry highly differs from $\mathrm{div}(\overrightarrow{v}) = 0$.

3.2 Data Assimilation

To ensure that the number of ensembles used to calculate the analysis is statistically representative, the RMSE is calculated in dependency of different amounts of ensemble members used in the data assimilation experiment. To suppress the influence of outliers at the geometry edges, different subvolumina are defined in which the RMSE values are compared (figure 2a). Outliers occur due to geometric misfits between MRI, CFD and PIV data, mainly caused by acquisition based distortions of the MRI data. For the current assimilation experiment an ensemble of 10 is chosen to be statistically representative.

Figure 3b represents the velocity fields and divergence of the analysis, the outcome of the data assimilation step. A qualitative comparison between the calculated analysis and the MRI based data reveals a reduction of noise and improvement of image resolution. The latter was achieved by using the CFD grid resolution in the data assimilation algorithm to calculate the analysis. The Navier-Stokes equations as the basis for the ensemble simulations fullfill conservations laws, which results in physically accurate calculated analyses. Divergence in the velocity fields was significantly reduced after the assimilation step.

Figure 4 compares the MRI data and analysis with a vertical slice in the center of the idealized aneurysm. MRI data and analysis are interpolated onto the PIV grid for the calculation of the RMSE in the predefined subvolumina. The small velocity peak at the transition from the aneurysm sack to the efferent vessel completely vanishes in the PC-MRI data. Although, the velocity values are still underestimated by the calculated analysis, the assimilation step was able to reconstruct the flow characteristics at the outlet more accurately. The qualitative investigation is supported by the calculation of the RMSE in all three subvolumina. In two cases the RMSE is significantly reduced by 38 % and 89 %, respectively, whereas for subvolume 2 the RMSE only decreases by 27 %.



(a) Subvolumina for RMSE

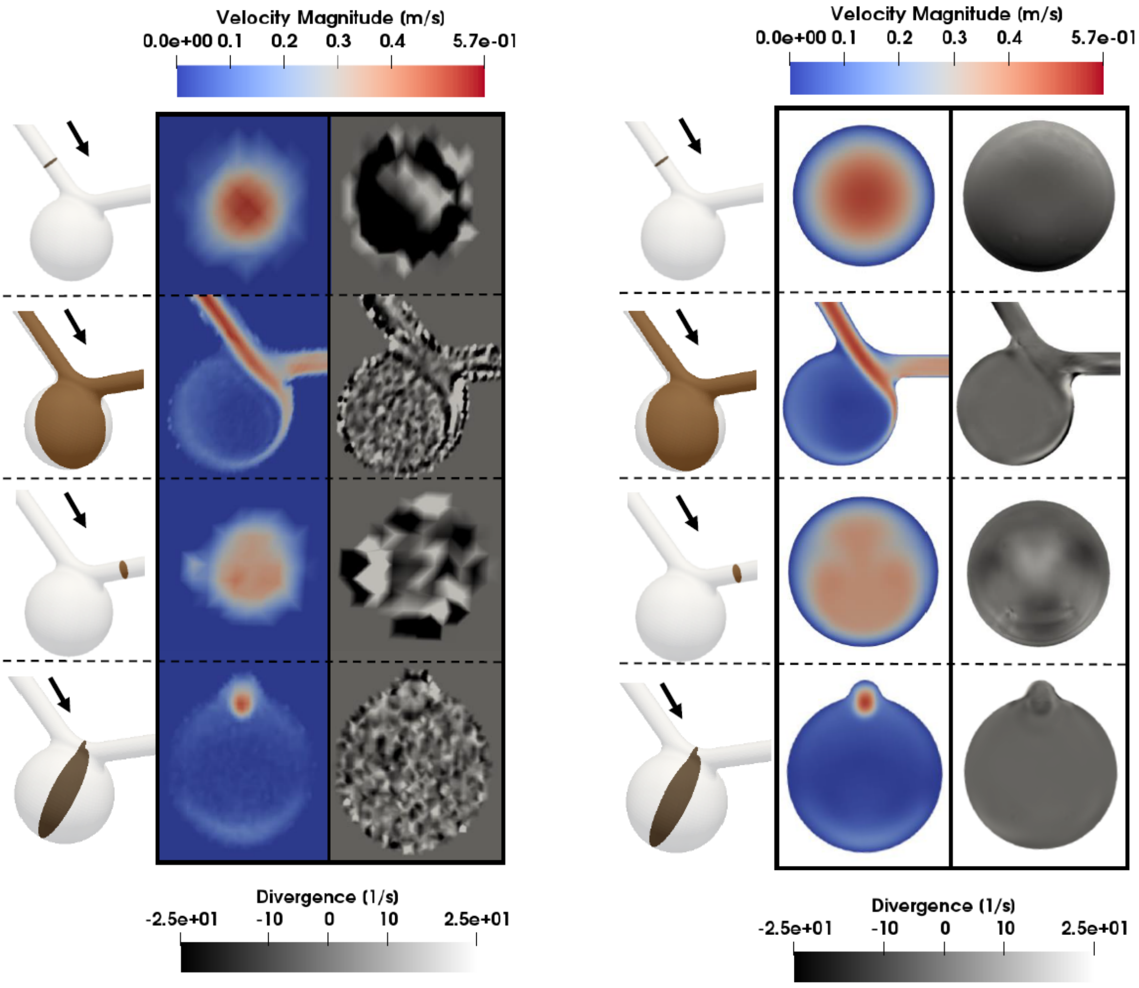
(b) RMSE in dependence of used ensemble members

Figure 2: The RMSE in different areas of the aneurysm (a) is calculated in dependency of the number of used ensemble members in the data assimilation algorithm. For the systematic variation of ensemble members, the entries of the observation error covariance matrix are increased to better depict the influence of ensemble simulations

4 DISCUSSION

Flow investigation using Phase-Contrast MRI results in low resolution, noisy images. Optical-based stereoscopic PIV measurements provide high-resolution velocity fields, but can not be used for *in-vivo* applications. The current study uses PIV as a validation criteria for the introduced data assimilation algorithm. An Ensemble Transform Kalman Filter improves the flow prediction in the geometry of an idealized intracranial aneurysm. Although, measured and assimilated flow fields qualitatively predict similar flow characteristics, acquisition noise and artifacts disturb the MRI based velocity fields. Resulting flow fields are not divergence-free, which reduces the physical correctness of the measured data. With the use of the assimilation algorithm conservation laws are introduced into the measurement data, which moves the divergence field closer to zero. By increasing the resolution of the velocity in the analysis, small velocity peaks can be reconstructed that are low-pass filtered in the original measurement data.

In addition to the improvement of physical correctness, the assimilation step moves the velocity field closer to the generated ground truth. The RMSE for the pre-defined subvolumina is reduced by the data assimilation step. Nevertheless, MRI as well as analysis based velocity fields seem to underestimate the general velocity values in comparison to the PIV data. This phenomena also occurs in areas with a uniform velocity distribution, hence downsampling by velocity averaging can not play a major role. These findings lead to the assumption, that it is not only stochastic errors resulting in measurement noise that play a role in the PC-MRI acquisition sequence, rather that systematic deviations also have an influence. Investigations in previous studies, in which PC-MRI data generally underestimates the velocity values, support this fact [26]. Possible factors could be a distortion of the images caused by gradient inhomogenities in the acquisition sequence. To further reduce the RMSE in the assimilated velocity fields a correct quantification of the systematic error sources is essential, this can be incorporated into the data assimilation algorithm. In a next step, the described data assimila-


Velocity Magnitude (m/s)
0.0e+00 0.1 0.2 0.3 0.4 5.7e-01
Divergence (1/s)
-2.5e+01 -10 0 10 2.5e+01
Velocity Magnitude (m/s)
0.0e+00 0.1 0.2 0.3 0.4 5.7e-01
Divergence (1/s)
-2.5e+01 -10 0 10 2.5e+01

(a) PC-MRI data

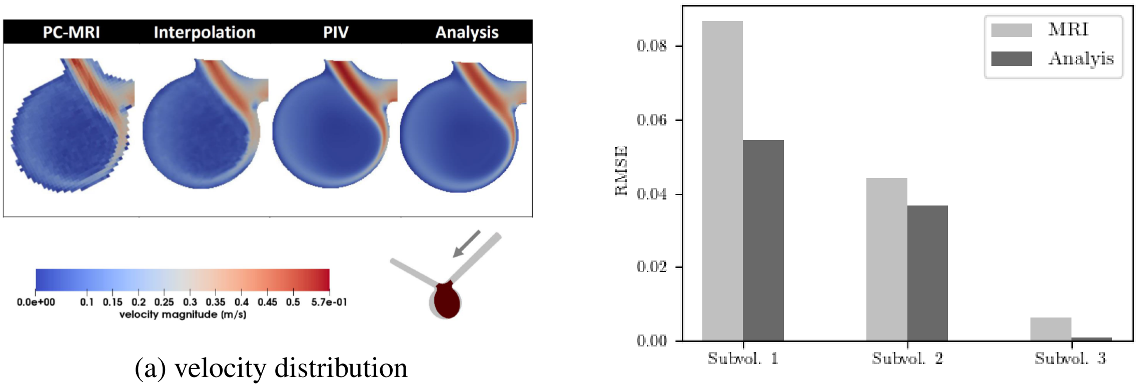
(b) Analyis

Figure 3: Velocity magnitude and divergence for PC-MRI data (a) and analyis (b) at different cut planes through the underlying geometry.

tion approach is applied to a 3D pulsatile cardiac cycle. Here, specific focus will be payed to the amount of ensemble members needed, which can hopefully be reduced in comparison to [22] with the introduced localization procedure. For patient-specific considerations one of the main difficulties is the accurate definition of geometric boundaries, which will also be a main component in further studies. Incorporating uncertain boundaries of the geometry into the assimilation algorithm could make it suitable for the clinical routine. Further ideas include the projection onto a divergence-free subspace previous to the assimilation step or direct assimilation of phase-difference data by mapping the simulated variables onto the observation space by an inverse observation operator.

5 CONCLUSION

The current study assimilates the flow field in an idealized aneurysm by combining data from numerical simulations together with measured PC-MRI velocity fields. The introduction of PIV measurements originating from the same geometry ensures proper quantitative validation. For



(a) velocity distribution

(b) RMSE

Figure 4: (a) Qualitative comparison of the velocity distribution between all three modalities in PIV grid resolution; (b) RMSE calculation for measured MRI data and calculated analysis in the predefined subvolumina.

the first time the LETKF was used for hemodynamic investigations, which enables the calculation of local analyses. The data assimilation algorithm successfully calculates a high resolution divergence-free velocity field inside the aneurysm geometry. It was able to reconstruct small velocity peaks that have been filtered out by the MRI measurement and reduced the RMSE of the analysis state estimate in comparison to the PC-MRI data.

REFERENCES

[1] J. R. Cebral, M. Vazquez, D. M. Sforza, G. Houzeaux, S. Tateshima, E. Scrivano, C. Bleise, P. Lylyk, and C. M. Putman, "Analysis of hemodynamics and wall mechanics at sites of cerebral aneurysm rupture," *Journal of neurointerventional surgery*, vol. 7, no. 7, pp. 530–536, 2015.

[2] G. Janiga, P. Berg, S. Sugiyama, K. Kono, and D. A. Steinman, "The Computational Fluid Dynamics Rupture Challenge 2013—Phase I: prediction of rupture status in intracranial aneurysms," *American Journal of Neuroradiology*, vol. 36, no. 3, pp. 530–536, 2015.

[3] J. Liu, J. Xiang, Y. Zhang, Y. Wang, H. Li, H. Meng, and X. Yang, "Morphologic and hemodynamic analysis of paraclinoid aneurysms: ruptured versus unruptured," *Journal of neurointerventional surgery*, vol. 6, no. 9, pp. 658–663, 2014.

[4] I. G. H. Jansen, J. J. Schneiders, W. V. Potters, P. van Ooij, R. van den Berg, E. van Bavel, H. A. Marquering, and C. B. L. M. Majoie, "Generalized versus patient-specific inflow boundary conditions in computational fluid dynamics simulations of cerebral aneurysmal hemodynamics," *AJNR. American journal of neuroradiology*, vol. 35, no. 8, pp. 1543–1548, 2014.

[5] C. Karmonik, C. Yen, O. Diaz, R. Klucznik, R. G. Grossman, and G. Benndorf, "Temporal variations of wall shear stress parameters in intracranial aneurysms–importance of patient-specific inflow waveforms for CFD calculations," *Acta neurochirurgica*, vol. 152, no. 8, pp. 1391–8; discussion 1398, 2010.

[6] J. Jiang and C. Strother, “Computational fluid dynamics simulations of intracranial aneurysms at varying heart rates: a patient-specific study,” *Journal of biomechanical engineering*, vol. 131, no. 9, p. 091001, 2009.

[7] J. R. Cebral and H. Meng, “Counterpoint: Realizing the Clinical Utility of Computational Fluid Dynamics—Closing the Gap,” *American Journal of Neuroradiology*, vol. 33, no. 3, p. 396, 2012.

[8] D. F. Kallmes, “Point: CFD–computational fluid dynamics or confounding factor dissemination,” *AJNR. American journal of neuroradiology*, vol. 33, no. 3, pp. 395–396, 2012.

[9] C. M. Strother and J. Jiang, “Intracranial Aneurysms, Cancer, X-Rays, and Computational Fluid Dynamics,” *American Journal of Neuroradiology*, vol. 33, no. 6, p. 991, 2012.

[10] M. Markl, A. Frydrychowicz, S. Kozerke, M. Hope, and O. Wieben, “4D flow MRI,” *Journal of Magnetic Resonance Imaging*, vol. 36, no. 5, pp. 1015–1036, 2012.

[11] J. Lotz, C. Meier, A. Leppert, and M. Galanski, “Cardiovascular flow measurement with phase-contrast MR imaging: basic facts and implementation,” *Radiographics : a review publication of the Radiological Society of North America, Inc*, vol. 22, no. 3, pp. 651–671, 2002.

[12] N. Pelc, R. Herfkens, A. Shimakawa, *et al.*, “Phase Contrast Cine Magnetic Resonance Imaging,” *Magnetic resonance quarterly*, vol. 7, 1991.

[13] G. Evensen, *Data Assimilation - The Ensemble Kalman Filter*. Springer International Publishing, 2nd ed., 2009.

[14] J. Busch, D. Giese, L. Wissmann, and S. Kozerk, “Reconstruction of divergence-free velocity fields from cine 3d phase-contrast flow measurements,” *Journal of Magnetic Resonance in Medicine*, vol. 69, pp. 200–210.

[15] M. F. Sereno, B. Köhler, and B. Preim, “Comparison of divergence-free filters for cardiac 4d pc-mri data,” in *Bildverarbeitung für die Medizin 2018*, pp. 139–144, Springer Berlin Heidelberg, 2018.

[16] M. D’Elia, L. Mirabella, T. Passerini, M. Peregor, M. Piccinelli, C. Vergara, and A. Veneziani, “Applications of Variational Data Assimilation in Computational Hemodynamics.” 2011.

[17] M. D’Elia, M. Perego, and A. Veneziani, “A Variational Data Assimilation Procedure for the Incompressible Navier-Stokes Equations in Hemodynamics,” *Journal of Scientific Computing*, vol. 52, no. 2, pp. 340–359, 2012.

[18] M. D’Elia and A. Veneziani, “Uncertainty quantification for data assimilation in a steady incompressible Navier-Stokes problem,” *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 47, no. 4, pp. 1037–1057, 2013.

[19] S. W. Funke, M. Nordaas, Ø. Evju, M. S. Alnæs, and K. A. Mardal, “Variational data assimilation for transient blood flow simulations.”

[20] X. Gao, Y. Wang, N. Overton, M. Zupanski, and X. Tu, “Data-assimilated computational fluid dynamics modeling of convection-diffusion-reaction problems,” *Journal of Computational Science*, vol. 21, pp. 38–59, 2017.

[21] M. C. Rochoux, B. Cuenot, S. Ricci, A. Trouv, B. Delmotte, S. Massart, R. Paoli, and R. Paugam, “Data assimilation applied to combustion,” *Comptes Rendus Mcanique*, vol. 341, no. 1, pp. 266 – 276, 2013. Combustion, spray and flow dynamics for aerospace propulsion.

[22] A. Bakhshinejad, V. L. Rayz, and R. M. D’Souza, “Reconstructing Blood Velocity Profiles from Noisy 4D-PCMR Data using Ensemble Kalman Filtering,” in *Biomedical Engineering Society (BMES) - Annual Meeting, Minneapolis, Minnesota,*, 2016.

[23] M. Markl, F. Chan, M. Alley, K. Wedding, M. Draney, C. Elkins, D. Parker, R. Wicker, C. A. Taylor, and R. J. Herfkens, “Time-resolved three-dimensional phase-contrast MRI,” *Journal of Magnetic Resonance Imaging*, vol. 17, no. 4, pp. 499–506, 2003.

[24] M. Markl, A. Harloff, T. Bley, M. Zaitsev, B. Jung, E. Weigang, M. Langer, J. Hennig, and A. Frydrychowicz, “Time-resolved 3D MR velocity mapping at 3T: Improved navigator-gated assessment of vascular anatomy and blood flow.,” *Journal of Magnetic Resonance Imaging*, vol. 25, no. 4, pp. 824–831, 2007.

[25] J. Bock, B. Kreher, J. Hennig, and M. Markl, “Optimized pre-processing of time-resolved 2D and 3D phase contrast MRI data.,” in *Proceedings of the 15th Annual Meeting of ISMRM, Berlin, Germany,* , p. 3135, 2007.

[26] C. Roloff, D. Stucht, O. Beuing, and P. Berg, “Comparison of intracranial aneurysm flow quantification techniques: standard PIV vs stereoscopic PIV vs tomographic PIV vs phase-contrast MRI vs CFD,” *Journal of neurointerventional surgery*, July 2018.

[27] B. R. Hunt, E. J. Kostelich, and I. Szunyogh, “Efficient data assimilation for spatiotemporal chaos: A local ensemble transform kalman filter,” *Physica D: Nonlinear Phenomena*, vol. 230, no. 1, pp. 112 – 126, 2007. Data Assimilation.

[28] J. Harlim and B. R. Hunt, “Four-dimensional local ensemble transform Kalman filter: numerical experiments with a global circulation model,” *Tellus A: Dynamic Meteorology and Oceanography*, vol. 59, no. 5, pp. 731–748, 2007.

[29] E. Ott, B. R. Hunt, I. Szunyogh, A. V. Zimin, E. J. Kostelich, M. Corazza, E. Kalnay, D. J. Patil, and J. A. Yorke, “A local ensemble Kalman filter for atmospheric data assimilation,” *Tellus A: Dynamic Meteorology and Oceanography*, vol. 56, no. 5, pp. 415–428, 2004.

[30] C. H. Bishop, B. J. Etherton, and S. J. Majumdar, “Adaptive Sampling with the Ensemble Transform Kalman Filter. Part I: Theoretical Aspects,” *Monthly Weather Review*, vol. 129, no. 3, pp. 420–436, 2001.

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

UNCERTAINTY ASSESSMENT OF THE BLOOD DAMAGE IN A FDA BLOOD PUMP

Chen Song[1,2*], **Vincent Heuveline**[1,2]

[1] 1Heidelberg Institute for Theoretical Studies (HITS)
Schloss-Wolfsbrunnenweg 35, 69118 Heidelberg, Germany
e-mail: {chen.song, vincent.heuveline}@h-its.org

[2] Engineering Mathematics and Computing Lab (EMCL)
Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University
Im Neuenheimer Feld 205, 69120 Heidelberg, Germany
e-mail: {chen.song, vincent.heuveline}@iwr.uni-heidelberg.de

Keywords: Uncertainty Quantifiaction, FDA Blood Pump, Hemolysis, Navier-Stokes, Variational Multiscale Method, Biomedical Engineering.

Abstract. *Heart failure (HF) is a severe cardiovascular disease, millions people are suffered from HF worldwide. The ventricular assist device (VAD) can replace the function of failing hearts, when there are no heart donations available, it is becoming a common daily practice for the heart failure patients. U.S. Food and Drug Administration (FDA) Critical Path Initiative (CPI) announced a benchmark study of a centrifugal blood pump few years ago in order to improve current practices of applying computational fluid dynamics (CFD) to medical devices.*

In our previous works, we developed our numerical model for the blood pump simulation with the consideration of Uncertainty Quantification (UQ). We introduced a shear layer update approach in order to facilitate and accelerate the moving mesh process in the framework of High Performance Computing (HPC). The uncertainties in the parametric data and geometric information are quantified with the Polynomial Chaos (PC) method, a Multilevel preconditioning technique is therefore proposed for expediting the linear solvers.

In this work, we show an instationary blood flow through a FDA blood pump configuration with Galerkin Projection method, which is realized in our open source Finite Element library HiFlow3. We consider the stress-based hemolysis model to demonstrate the blood damage during the operation of the blood pump. Three uncertainty sources are considered: inflow boundary condition, rotor angular speed and dynamic viscosity, the numerical results are demonstrated with more than 45 Million degree of freedoms by using supercomputer.

1 INTRODUCTION

There are more than 40 Million people worldwide suffering from the heart failure, the number of patients increases also every year due to the population ageing. There exist currently two standard treatments for severe heart failure patients: the heart transplant and the heart implant. The heart transplant is until now the golden standard for critical patients in regard to the life quality and the survival rate. But, in general the patients have to wait for long time till one heart donor is available, and the donor quality cannot be guaranteed. On the other hand, the heart implant is available even for the emergency cases, the performance is rapidly improving since last decade. Therefore, the ventricular assist devices (VADs), or blood pumps, play a more and more important role in the field of health care [8, 22]. However, mechanical design has limits due to the lack of knowledge about input parametric data, specially for numerical modeling. Hence, the Uncertainty Quantification can be very useful for improving the performance of such devices.

In our previous work in 2015 [24], we studied the steady state of incompressible Navier-Stokes equations in the laminar regime within the blood pump chamber by using the Multiple Reference Frame (MRF) method [12, 10, 18, 20]. The Multiple Reference Frame method suggests to divide the computational domain into two adjacent parts: rotating and stationary domains. In the stationary domain, the velocity and pressure fields are defined in the absolute frame, it implies that the steady incompressible Navier-Stokes equations are applied, yet in the rotating domain, the velocity field has to be considered in a relative frame. Therefore, two additional forces – Corilios force and centrifugal force – are introduced in the momentum equation because of the transformation between the absolute frame and the relative frame. However, the MRF model provides only the information in the steady state, the unsteady flow simulation is required for obtaining further insight of the performance of blood pump.

The last publication [27] in 2017 presented our first unsteady blood flow simulation on a full blood pump based on the residual-based Variational Multiscale method (VMS) [16, 5, 4, 17]. The blood pump geometry is provided by the U.S. Food and Drug Administration (FDA) under the framework of Critical Path Initiative (CPI). This project intends to assess the accuracy of the computational fluid dynamics (CFD) in biomedical devices [14]. In this work, we proposed the shear layer update approach in order to realize the moving mesh procedure. This method is designed based on the Shear-Slip Mesh Update Method (SSMUM) [6, 7], and it proposes to generate two regular identical shaped layers in stead of only one, such that the solution update step, once the mesh re-generation is required, can be accelerated, especially for the parallel computing. The blood flow in the pump casing is subject to a strong external force from the pump impeller. Due to the high rotation speed of the device, the fluid flow is impossible to be maintained in the laminar regime. Therefore, we applied the residual-based Variational Multiscale method for simulating the turbulent flow. In addition to that, we employed the intrusive Polynomial Chaos Expansion (PCE) to study the propagation of uncertainties caused by three input parameters, i.e. inflow boundary, dynamic viscosity and rotating speed.

We here continue to apply the stochastic finite element method (SFEM) [15] for the blood pump modelization. The shear layer update approach, which is introduced in [27], is employed in this work for acquiring the moving mesh. The intrusive stochastic Galerkin approach is implemented for quantifying the uncertainty introduced by three uncertain parameters, by means of inflow velocity, rotation speed and dynamic viscosity. Our greater interest for this work is put in the velocity and pressure fields in the chamber of the pump device. Furthermore, the shear stress created by the high rotation speed can cause severe blood damage by means of destroying the red blood cells, such that the oxygen can not be transported into human body sufficiently. Therefore, a study concerning the hemolysis is also presented by using the index of hemolysis. The mean value and the standard deviation of the index hemolysis are showed in different locations in the pump chamber.

The rest of this paper is organized as follow. In Section 2, the mathematical modeling is introduced. It starts with a general formulation of the incompressible Navier-Stokes equations for a rotating system, then the Variational Multiscale method is introduced in order to cope with the turbulent flow. In Section 3, we discuss about the three sources of uncertainty in the computation, as well as the stochastic Galerkin projection method. Section 4 is the numerical results, the velocity and pressure fields within the pump chamber are presented with the mean value and the standard deviation. In addition, the index of hemolysis is also calculated in order to quantity the blood damage caused by the high shear rate of the rotor. We conclude our contribution in Section 5.

2 MATHEMATICAL MODELING

2.1 Incompressible Navier-Stokes equations on the stationary and rotating domain

We consider the incompressible Navier-Stokes equations for modeling the blood flow within the device. Let $\Omega \in \mathbb{R}^3$ to be the spatial domain, $\Omega = \Omega_{stat} \cup \Omega_{rot}$, $\Omega_{stat} \cap \Omega_{rot} = \emptyset$. The spatial domain Ω consists of two subdomains: the rotating stationary domain Ω_{stat} and the rotating domain Ω_{rot}. The incompressible Navier-Stokes equations on the stationary and rotating domain are stated as follow:

$$
\begin{aligned}
\frac{\partial \boldsymbol{u}}{\partial t} + ((\boldsymbol{u} - \boldsymbol{u}^r) \cdot \nabla)\boldsymbol{u} - \frac{\mu}{\rho}\Delta \boldsymbol{u} + \frac{1}{\rho}\nabla p = 0 \ , \quad & \text{in } \Omega \ , && \text{(1a)}\\
\nabla \cdot \boldsymbol{u} = 0 \ , \quad & \text{in } \Omega \ , && \text{(1b)}\\
\boldsymbol{u}^r = \boldsymbol{d} \times \boldsymbol{\omega} \ , \quad & \text{in } \Omega_{rot} \ , && \text{(1c)}\\
\boldsymbol{u}^r = 0 \ , \quad & \text{in } \Omega_{stat} \ , && \text{(1d)}\\
\boldsymbol{u} = \boldsymbol{g} \ , \quad & \text{on } \Gamma_{in} \ , && \text{(1e)}\\
(\frac{\mu}{\rho}\nabla \boldsymbol{u} - p\mathbb{1}) = 0 \ , \quad & \text{on } \Gamma_{out} \ , && \text{(1f)}\\
\boldsymbol{u}^r = \boldsymbol{d} \times \boldsymbol{\omega} \ , \quad & \text{on } \Gamma_{rotor} \ , && \text{(1g)}\\
\boldsymbol{u} = 0 \ , \quad & \text{on } \partial\Omega \backslash (\Gamma_{in} \cup \Gamma_{out} \cup \Gamma_{rotor}) \ . && \text{(1h)}
\end{aligned}
$$

Here, $\boldsymbol{u}$ and p are the velocity field and the pressure field respectively. $\boldsymbol{u}^r$ is the rotating velocity of the rotor, it is also the revolution velocity of the rotating domain Ω_{rot} (Equation (1c)). Hence $\boldsymbol{u}^r$ is only defined on Ω_{rot} (Equation (1c), Equation (1d)). $\boldsymbol{\omega}$ describes the angular speed $(\mathrm{rad/s})$ of the impeller, $\boldsymbol{d}$ is the distance to the rotating axis. Furthermore, ρ is the density of the blood, and μ is the dynamic viscosity.

Figure 1 shows the geometry of the pump device. The rotor consists of four blades and one rub, and it is embedded in the pump casing. The rotor operates with a constant angular speed $\boldsymbol{\omega}$, this rotation is conducted by an external motor, which is omitted in Figure 1. The flow is induced by the inlet tube, which is located on the top of pump chamber. After entering the pump chamber, the fluid is accelerated owning to the absorption of the rotational kinetic energy from the rotor. The blood is further directed into the outlet, which connects to the aorta of the patient.

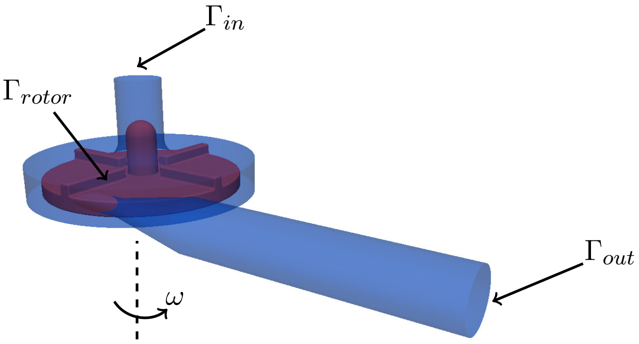


Figure 1: Illustration of the boundaries and the axis of rotation on the blood pump geometry.

On the inflow boundary Γ_{in}, we apply a Dirichlet boundary condition by using the Poiseuill profile, which is defined as:

$$\boldsymbol{g} = \begin{bmatrix} 0 \\ 0 \\ -U_{max}(1 - l^2/L^2) \end{bmatrix} , \tag{2}$$

where L is the radius of circular geometry of Γ_{in}, l is the distance to the center point. U_{max} is the maximum inflow velocity, which is a scalar value and positive, the inflow direction in our figure is in $-\boldsymbol{e}_3 = [0, 0, -1]^T$. Meanwhile, the angular velocity is defined as:

$$\boldsymbol{\omega} = \begin{bmatrix} 0 \\ 0 \\ \omega \end{bmatrix} , \tag{3}$$

the rotation axis is also $\boldsymbol{e}_3$. In addition, the boundary condition on the rotor's surface Γ_{rotor} is also described with $\boldsymbol{\omega}$ as in Equation (1g). The outflow boundary condition is prescribed with ”do-nothing” boundary. The rest of rigid walls is governed by the ”no-slip” condition [23].

2.2 Variational Multiscale Method

As described in the introduction, we want to model the fluid flow in blood pump device. For the centrifugal pumps, the Reynolds number is defined as [9]:

$$Re = \frac{\rho \omega D^2}{\mu}. \tag{4}$$

Here, ρ is the density, ω is the angular speed of the impeller, D is the diameter of the rotor, and μ the dynamic viscosity. According to the simulation information provided by the Critical Path Initiative [14], we have:

ρ	$1035\,\mathrm{kg/m^3}$	ω	$261.667\,\mathrm{rad/s}$
D	$52\,\mathrm{mm}$	μ	$0.0035\,\mathrm{kg/m/s}$

Table 1: Simulation information.

The Reynolds number for the blood pump simulation is approximately $210,000$. Therefore, in order to overcome the difficulty of computing high Reynolds number flow, the Variational

Multiscale Method comes into play. VMS provides a general framework of treating the subgrid phenomena while dealing with the incompressible Navier-Stokes equations numerically, such that the scale spectra can be handled adequately. The fundamental concept of VMS is to separate the complete scale range into different scale ranges in order to treat them individually. For demonstrating the our Variational Multiscale model, we illustrate first the discrete variational formulation of the incompressible Navier-Stokes equations (Equation (1)):

Taking $\boldsymbol{u} \in V, p \in Q$, such that:

$$\int_\Omega (\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} - \boldsymbol{u}^r) \cdot \nabla \boldsymbol{u}) \cdot \boldsymbol{v} \, d\boldsymbol{x} + \int_\Omega \frac{\mu}{\rho} \nabla \boldsymbol{u} : \nabla \boldsymbol{v} \, d\boldsymbol{x} - \int_\Omega \frac{1}{\rho} p \nabla \cdot \boldsymbol{v} \, d\boldsymbol{x} = \boldsymbol{0} \ ,$$

$$\int_\Omega q \nabla \boldsymbol{v} d\boldsymbol{x} = 0 \ ,$$

$\forall \boldsymbol{v} \in V, \forall q \in Q$.

Here, V and Q are appropriate solution spaces for the velocity $\boldsymbol{u}$ and the pressure p on Ω. $\boldsymbol{v}$ and q are the test functions for the velocity and pressure respectively. We omit the weak form for the boundary conditions and the definition of $\boldsymbol{u}^r$ for the sake of simplicity.

In this work, we consider two-scale residual-based Variational Multiscale model, it implies that the velocity and pressure variables can be decomposed like:

$$\boldsymbol{u} = \boldsymbol{u}_h + \hat{\boldsymbol{u}} \ , \tag{6a}$$

$$p = p_h + \hat{p} \ , \tag{6b}$$

where $\boldsymbol{u}_h$ and p_h are resolvable solutions, $\hat{\boldsymbol{u}}$ and $\hat{p}$ are unresolvable solutions. According to the decomposition of solution functions, the underlying function spaces V and Q can also be separated by using the direct sump decomposition [1, 26], i.e.:

$$V = V^h \oplus \hat{V} \ , \tag{7a}$$

$$Q = Q^h \oplus \hat{Q} \ , \tag{7b}$$

V^h and Q^h represent the finite-dimensional subspace of resolved scales, $\hat{V}$ and $\hat{Q}$ represent the infinite-dimensional subspace of unresolved scales. Under the Finite Element framework, the resolvable space is chosen as the finite element space [3]. By this decomposition, the variational formulation of the Navier-Stokes equations (Equation (5)) is decoupled into a resolved-scale equation and an unresolved-scale equation:

$$A(\boldsymbol{u}; (\boldsymbol{u}_h, p_h), (\boldsymbol{v}_h, q_h)) + A(\boldsymbol{u}; (\hat{\boldsymbol{u}}, \hat{p}), (\boldsymbol{u}_h, q_h)) = \boldsymbol{0} \ , \qquad \forall (\boldsymbol{v}_h, q_h) \in V^h \times Q^h \ , \tag{8a}$$

$$A(\boldsymbol{u}; (\boldsymbol{u}_h, p_h), (\hat{\boldsymbol{v}}, \hat{q})) + A(\boldsymbol{u}; (\hat{\boldsymbol{u}}, \hat{p}), (\hat{\boldsymbol{v}}, \hat{q}) = \boldsymbol{0} \ , \qquad \forall (\hat{\boldsymbol{v}}, \hat{q}) \in \hat{V} \times \hat{Q} \ . \tag{8b}$$

If we write Equation (5) in the following fashion:

Find $\boldsymbol{u} \in V, p \in Q$, such that:

$$A(\boldsymbol{u}; (\boldsymbol{u}, p), (\boldsymbol{v}, q)) = \boldsymbol{0} \ , \quad \forall (\boldsymbol{v}, q) \in V \times Q \ . \tag{9}$$

Therefore, the resolved-scale equation is not closed, the unresolved-scale contributions have to be appropriately modeled. We apply the residual-based stabilization method in order to model the velocity and pressure on the subgrid scale. The two-scale residual based Variational Multiscale formulation is stated as below:

$$\begin{aligned}
&(\frac{\partial \boldsymbol{u}_h}{\partial t}, \boldsymbol{v}_h) + ((\boldsymbol{u}_h - \boldsymbol{u}^r) \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h) && \text{(10a)}\\
&+\frac{\mu}{\rho}(\nabla \boldsymbol{u}_h, \nabla \boldsymbol{v}_h) - \frac{1}{\rho}(p_h, \nabla \cdot \boldsymbol{v}_h) &&\\
&+(\boldsymbol{\tau}_M \boldsymbol{r}_M, (\boldsymbol{u}_h - \boldsymbol{u}^r) \cdot \nabla \boldsymbol{v}_h) + (\tau_C r_C, \nabla \cdot \boldsymbol{v}_h) &&\\
&-(\boldsymbol{\tau}_M \boldsymbol{r}_M \cdot \nabla \boldsymbol{u}_h, \boldsymbol{v}_h) - (\boldsymbol{\tau}_M \boldsymbol{r}_M, \boldsymbol{\tau}_M \boldsymbol{r}_M \cdot \nabla \boldsymbol{v}_h) = \boldsymbol{0} \ , \qquad \text{in } [0,T] \times \Omega \ , &&\\
&(\nabla \cdot \boldsymbol{u}_h, q_h) + (\boldsymbol{\tau}_M \boldsymbol{r}_M, \nabla q_h) = 0 \ , \qquad \text{in } [0,T] \times \Omega \ . && \text{(10b)}
\end{aligned}$$

Here, the spatial domain Ω is divided into n_{el} finite element subdomains. V^h and Q^h are the finite dimensional spaces for the discrete solutions $\boldsymbol{u}_h$ and p_h, respectively. $\boldsymbol{v}_h$ and q_h are the test functions for the velocity and pressure. $\boldsymbol{r}_M$ and $\boldsymbol{r}_C$ are the residual of momentum and continuity equation:

$$\boldsymbol{r}_M = \frac{\partial \boldsymbol{u}_h}{\partial t} + \boldsymbol{u}_h \cdot \nabla \boldsymbol{u}_h - \frac{\mu}{\rho} \Delta \boldsymbol{u}_h + \frac{1}{\rho} \nabla p_h \ , \qquad \text{(11a)}$$

$$r_C = \nabla \cdot \boldsymbol{u}_h \ . \qquad \text{(11b)}$$

$\boldsymbol{\tau}_M$, $\boldsymbol{\tau}_C$ are the stabilization parameters, which are defined in [26]

3 UNCERTAINTY MODEL

3.1 Uncertain inputs

According to [27], we consider three different sources of input uncertainty in our model: the inflow boundary condition $\boldsymbol{g}$, the angular speed $\boldsymbol{\omega}$ and the dynamic viscosity μ. We employ the generalized Polynomial Chaos Expansion (gPCE) [28, 25, 2] technique to model the uncertainty propagation in the input parameters. We chose the ignorance mode, a Uniform distribution, to model the input uncertainty, i.e. $\xi_i \sim U(-1, 1), i = 1, 2, 3$. They read:

$$\boldsymbol{g} = \boldsymbol{g}_0 + \boldsymbol{g}_1 \xi_1 \ , \qquad \text{(12a)}$$

$$\boldsymbol{\omega} = \boldsymbol{\omega}_0 + \boldsymbol{\omega}_2 \xi_2 \ , \qquad \text{(12b)}$$

$$\mu = \mu_0 + \mu_3 \xi_3 \ , \qquad \text{(12c)}$$

where $\boldsymbol{g}_0$, $\boldsymbol{\omega}_0$ and μ_0 are the mean values for each random input. $\boldsymbol{g}_1$, $\boldsymbol{\omega}_2$ and μ_3 are the maximum deviation with respect to the mean their value, which are defined as: $\boldsymbol{g}_1 = \sigma_1 \boldsymbol{g}_0$, $\boldsymbol{\omega}_2 = \sigma_2 \boldsymbol{\omega}_0$, $\mu_3 = \sigma_3 \mu_0$ respectively. $\sigma_i, i = 1, 2, 3$ are the deviation factor, hence $0 < \sigma_i < 1$. We also define the multivariate random variable $\boldsymbol{\xi} := (\xi_1, \xi_2, \xi_3)$. Accordingly, $\boldsymbol{\xi}$ allows us directly to map the outcomes of an abstract probability space $(\Omega, \mathcal{A}, \mathbb{P})$ to a subset T of $\mathbb{R}^3$. Afterwards, we can express our stochastic solution immediately with the aid of $\boldsymbol{\xi}$.

3.2 Stochastic Galerkin projection

Two primitive variables velocity $\boldsymbol{u}$ and pressure p are expressed with Polynomial Chaos Expansion technique [19]:

$$\boldsymbol{u}(\boldsymbol{x},\boldsymbol{\xi}) = \sum_{i=0}^{\infty} \boldsymbol{u}_i(\boldsymbol{x})\psi_i(\boldsymbol{\xi}) \ , \tag{13a}$$

$$p(\boldsymbol{x},\boldsymbol{\xi}) = \sum_{i=0}^{\infty} p_i(\boldsymbol{x})\psi_i(\boldsymbol{\xi}) \ . \tag{13b}$$

Here $\psi_i(\boldsymbol{\xi})$ the orthogonal Chaos Polynomials, as we model the uncertain input parameters with the standard Uniform distribution, ψ_i are practically the Legendre Polynomials. The orthogonality of ψ_i with respect to the probability density function of $\boldsymbol{\xi}$ in this work can be written as:

$$\int_{[-11]^3} \psi_i(\boldsymbol{\xi})\psi_j(\boldsymbol{\xi})\frac{1}{2^3}d\boldsymbol{\xi} = \delta_{ij} \ , \tag{14}$$

where δ_{ij} is the Kronecker delta function. However, working with a infinite series (Equation (13)) is numerically unpractical. We have to approximate $\boldsymbol{u}$ and p by truncating the infinite sequence up to certain polynomial order N_0, it gives:

$$\boldsymbol{u}(\boldsymbol{x},\boldsymbol{\xi}) \approx \sum_{i=0}^{P} \boldsymbol{u}_i(\boldsymbol{x})\psi_i(\boldsymbol{\xi}) \ , \tag{15a}$$

$$p(\boldsymbol{x},\boldsymbol{\xi}) \approx \sum_{i=0}^{P} p_i(\boldsymbol{x})\psi_i(\boldsymbol{\xi}) \ , \tag{15b}$$

where $P+1 = (M+N_0)!\,/(M!\,N_0!\,)$ is the total number of Polynomial Chaos modes, M is the number of random variables, in this study $M=3$ [26].

Equation (15) are also valid for the discrete solutions $\boldsymbol{u}_h$ and p_h. We replace at first the velocity and pressure by the Chaos Expansion (Equation (15)), then we multiply one Chaos polynomial $\psi_k, k=0,...,P$ on both side the equations. After that, we take the L^2 inner product on $L^2(T)$. The procedure we describe above is called as the stochastic Galerkin projection. Equation (10) can be rewritten as:

$$\begin{aligned}\frac{\partial \boldsymbol{u}_k}{\partial t}\boldsymbol{v}_k + \sum_{i=0}^{P}\sum_{j=0}^{P}((\boldsymbol{u}_i - \boldsymbol{u}_i^r)\cdot\nabla)\boldsymbol{u}_j\boldsymbol{v}_k c_{ijk} + \sum_{i=0}^{P}\sum_{j=0}^{P}\frac{\mu_i}{\rho}\nabla\boldsymbol{u}_i : \nabla\boldsymbol{v}_j c_{ijk} + \frac{1}{\rho}p_i\nabla\cdot\boldsymbol{v}_k \\ +\tau_M((\boldsymbol{u}_k - \boldsymbol{u}_k^r)\cdot\nabla\boldsymbol{v}_k)[\frac{\partial \boldsymbol{u}_k}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P}((\boldsymbol{u}_i - \boldsymbol{u}_j^r)\cdot\nabla)\boldsymbol{u}_j c_{ijk} \\ -\sum_{i=0}^{P}\sum_{j=0}^{P}\frac{\mu_i}{\rho}\Delta\boldsymbol{u}_j c_{ijk} + \frac{1}{\rho}\nabla p_k] + (\nabla\cdot\boldsymbol{u}_k)\tau_C(\nabla\cdot\boldsymbol{v}_k) \ , \quad \text{in}\,\Omega \ ,\end{aligned} \tag{16a}$$

$$q_k\nabla\cdot\boldsymbol{u}_k \tag{16b}$$

$$\tau_M \nabla q_k [\frac{\partial \boldsymbol{u}_k}{\partial t} + \sum_{i=0}^{P}\sum_{j=0}^{P}((\boldsymbol{u}_i - \boldsymbol{u}_i^r)\cdot\nabla)\boldsymbol{u}_j c_{ijk} \\ - \sum_{i=0}^{P}\sum_{j=0}^{P}\frac{\mu_i}{\rho}\Delta\boldsymbol{u}_j c_{ijk} + \frac{1}{\rho}\nabla p_k] \ , \quad \text{in}\,\Omega \ .$$

for $k = 0, ..., P$, and $c_{ijk} :=< \psi_i\psi_j, \psi_k >$.

The Variational Multiscale formulation of the incompressible Navier-Stokes is showed under the framework of moving mesh computation, especially for the blood pump modelization. The three uncertain input parameters are modeled with a first-order expansion, the generalized Polynomial Chaos Expansion is to use to quantify the uncertainty propagation.

4 NUMERICAL RESULTS

The nonlinear system (Equation (16)) is solved by the inexact Newton scheme, we applied the Crank-Nicolson time stepping scheme with the strategy choice 1 of Eisenstat and Walker in [13] with an initial forcing term equals to 0.5.

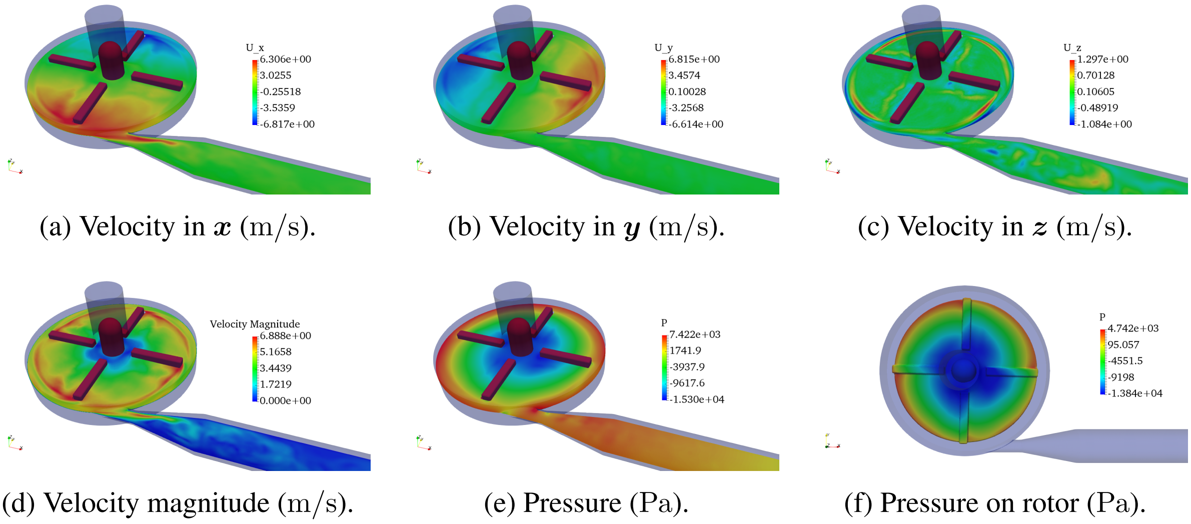
Inflow maximal speed (m/s)	0.5	Inflow speed variation (σ_1)	0.1
Angular speed (rad/s)	261.8	Angular speed variation (σ_2)	0.1
Dynamic viscosity ($N \cdot s/m^2$)	0.0035	Viscosity variation (σ_3)	0.1
RPM	2500	Density (Kg/m^3)	1035

Table 2: Model parameter values.

The spatial domain (Figure 1) is discretized with $2,984,859$ elements, it gives $2,274,904$ degrees of freedom (DOFs) for the deterministic problem. We set the polynomial degree to 3, it results in total 20 Polynomial Chaos modes, it implies that the global stochastic system has around 45.5 Millions DOFs. The full simulation is computed until the flow becomes stable, in our case, we proceed the simulation till 5 rotations. Table 2 illustrates the model parameters are used in the simulation, we set $\sigma_i, i = 1, 2, 3$ to 0.1, it means that the uncertainty in the three input parameters are taken as 10% of their mean value. The full simulation is computed with 2048 processors, the total computational time is around 100 hours.

Figure 2 demonstrates the mean value of the velocity field and the pressure field on a cross-section ($6.5mm$ from the bottom) of blood pump at the 5th rotation, and Figure 3 shows the standard deviation of the velocity and the pressure at the same location. In the pump chamber, vortexes can be observed due to the strong rotation, and more larger vortexes occur in the outlet (Figure 2d). For the velocity, only the velocity in $y-$axis direction contributes the most uncertainties in the pump chamber (Figure 3b), but the uncertainty in the outlet is mainly dominated (Figure 3d). However, the pressure in the chamber is almost asymmetrically distributed (Figure 2e), strong uncertainties are located in the chamber and the outlet (Figure 3e). We also observe that the pressure on the center of rotor is negative (Figure 2f) due to the suction effect, the amount of uncertainty of pressure is relatively 10% of the mean value (Figure 3f), it could be caused by the choice of the input uncertainty (Table 2).

Furthermore, the centrifugal pump can cause a non-negligible amount of hemolysis based on the shear stress and the exposure time. Therefore, it is very important to assess the quantity of the hemolysis under sever conditions in blood pump, such that there is still enough red blood cells entering the body.



(a) Velocity in $\boldsymbol{x}$ $(\mathrm{m/s})$.

(b) Velocity in $\boldsymbol{y}$ $(\mathrm{m/s})$.

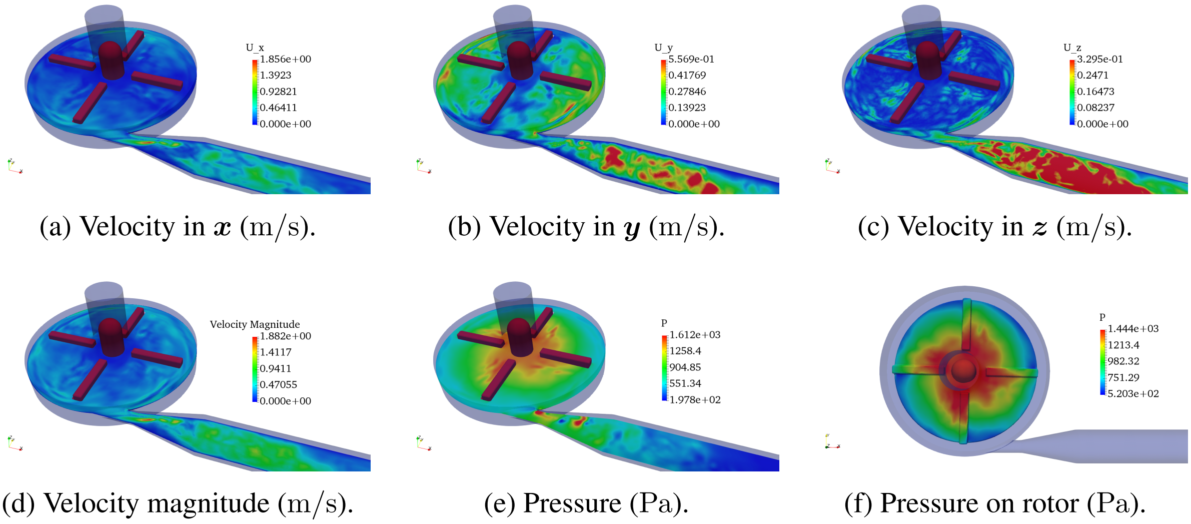
(c) Velocity in $\boldsymbol{z}$ $(\mathrm{m/s})$.

(d) Velocity magnitude $(\mathrm{m/s})$.

(e) Pressure (Pa).

(f) Pressure on rotor (Pa).

Figure 2: Mean value of velocity and pressure at the 5th rotation.



(a) Velocity in $\boldsymbol{x}$ $(\mathrm{m/s})$.

(b) Velocity in $\boldsymbol{y}$ $(\mathrm{m/s})$.

(c) Velocity in $\boldsymbol{z}$ $(\mathrm{m/s})$.

(d) Velocity magnitude $(\mathrm{m/s})$.

(e) Pressure (Pa).

(f) Pressure on rotor (Pa).

Figure 3: Standard deviation of velocity and pressure at the 5th rotation.

One of widely used estimators for quantifying the amount of hemolysis is the index of hemolysis (IH) [11, 21], which is defined as:

$$IH := (1 - \frac{Hct}{100})\frac{\Delta Hb}{Hb} \times 100 \ . \tag{17}$$

Here, Hct is the hematocrit, which signifies the amount red blood cells (RBCs) in blood. Hb is hemoglobin, and ΔHb is the quantity of hemoglobin, which is released into blood plasma after the shear stress and the exposure time both attain the critical magnitudes. Nevertheless, the ratio between the released hemoglobin and the hemoglobin ($\Delta Hb/Hb$) has to be obtained by the experiments *in vivo*, thus an empirical model is suggested to replace this part in Equation (17) by using the power law:

$$\frac{\Delta Hb}{Hb} = A_{Hb}\sigma_s^{\alpha_{Hb}}\tilde{t}^{\beta_{Hb}} \ . \tag{18}$$

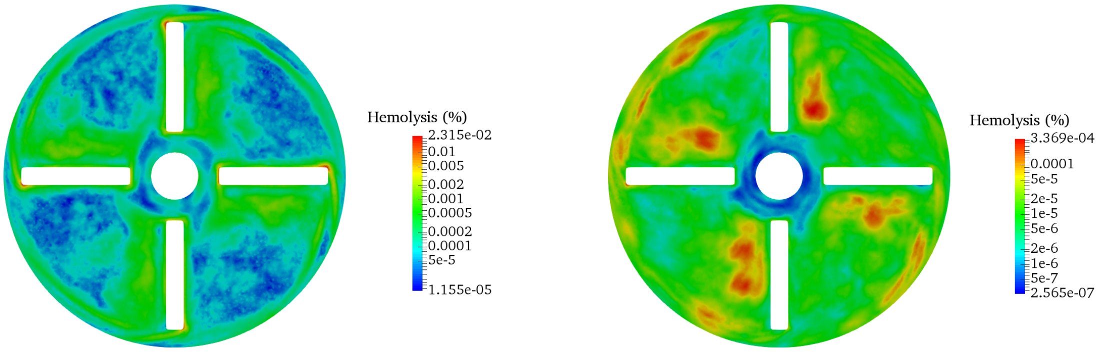
A_{Hb} is the plasma-free hemoglobin, σ_s is the scalar shear stress, and $\tilde{t}$ is the exposure time [26]. In our model, the scalar shear stress is defined by:

$$\sigma_s := \mu G_f = \mu\sqrt{2\boldsymbol{\varepsilon} : \boldsymbol{\varepsilon}} \ , \tag{19}$$

where G_f is the shear rate of blood flow, $\boldsymbol{\varepsilon}$ is the strain rate tensor:

$$\boldsymbol{\varepsilon} := \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T) \ , \tag{20}$$

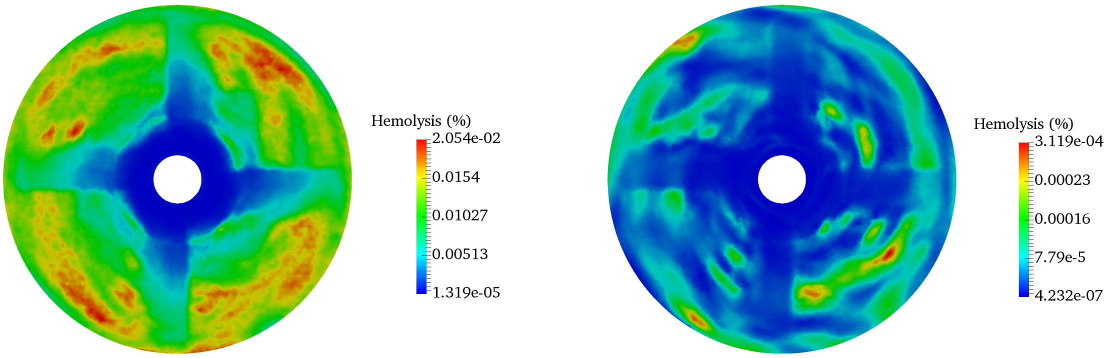
$\boldsymbol{u}$ is the velocity of the fluid.



(a) Mean value.

(b) Standard deviation.

Figure 4: Index of hemolysis (IH) distribution (at 5th rotation).



(a) Mean value.

(b) Standard deviation.

Figure 5: Index of hemolysis (IH) distribution (at 5th rotation).

Figures 4 and 5 demonstrates the index of hemolysis in the blood chamber. Figure 4 shows the IH on a cross-section $6.5mm$ from the bottom. The index of hemolysis is very high next to the outer part of the leading edge, it is caused by high shear stresses (Figure 4a). However, the uncertainty of IH is high between the blades next to the trailing edge (Figure 4b). Figure 5 presents the IH on a cross-section close to the upper wall of the pump. In contrast to Figure 4a, the index of hemolysis is very high in the passages, it can be caused by the narrow space between the upper wall and the blades, the blood is accelerated after going through this area (Figure 5a). The uncertainty of IH is basically important at the locations, where the mean value is high (Figure 5b).

5 CONCLUSION

This work is a following work of [27], which was focused on the solving techniques, i.e. the Multilevel preconditioner, for dealing with the coupled large stochastic system. The numerical algorithms have been showed the efficiency on the stochastic Galerkin system especially for high performance computing. In this work, we apply the two-scale residual-based Variational Multiscale method enables the possibility of computing high Reynolds number flow in the blood pump. The intrusive stochastic Galerkin approach can be constructed systematically.

Three sources of uncertainty are considered in this application, i.e. the inflow boundary condition, the rotational speed and the dynamic viscosity. We place our interest in the velocity and pressure fields, quantitative comparison and analysis is showed in Section 4. Moreover, quantifying the amount of blood damage induced by the blood handling device is also very important, because it influences directly the quality of blood, which is inducted into the body. Therefore, the index of hemolysis on two different locations are also presented, we observe the distribution of hemolysis is remarkably different. By providing the access of the standard deviation of the IH, we can access the confidence in the evaluation of uncertain input parameters on the blood damage.

6 ACKNOWLEDGMENT

The authors acknowledge support by the Heidelberg Institute for Theoretical Studies, HITS gGmbH. The numerical results of this work are obtain by using bwForCluster MLS&WISO (production and development part) at Heidelberg University. The authors acknowledge support by the state of Baden-Wrttemberg through bwHPC and the German Research Foundation (DFG) through grant INST 35/1134-1 FUGG.

REFERENCES

[1] Naveed Ahmed, Tomás Chacón Rebollo, Volker John, and Samuele Rubino. A review of variational multiscale methods for the simulation of turbulent incompressible flows. *Archives of Computational Methods in Engineering*, 24(1):115–164, Jan 2017.

[2] I. Babuska, R. Tempone, and G. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM Journal on Numerical Analysis*, 42(2):800–825, 2004.

[3] Y. Bazilevs, V.M. Calo, J.A. Cottrell, T.J.R. Hughes, A. Reali, and G. Scovazzi. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 197(1):173 – 201, 2007.

[4] Y. Bazilevs, J. Yan, X. Deng, and A. Korobenko. *Simulating Free-Surface FSI and Fatigue Damage in Wind-Turbine Structural Systems*, pages 1–28. Springer International Publishing, Cham, 2018.

[5] Yuri Bazilevs, Kenji Takizawa, Tayfun E. Tezduyar, Ming-Chen Hsu, Nikolay Kostov, and Spenser McIntyre. *Computational Wind-Turbine Analysis with the ALE-VMS and ST-VMS Methods*, pages 355–386. Springer International Publishing, Cham, 2014.

[6] M. Behr and T. Tezduyar. The shear-slip mesh update method. *Computer Methods in Applied Mechanics and Engineering*, 174(34):261 – 274, 1999.

[7] Marek Behr and Dhruv Arora. Shear-slip mesh update method: Implementation and applications. *Computer Methods in Biomechanics and Biomedical Engineering*, 6(2):113–123, 2003. PMID: 12745425.

[8] Emma J. Birks, Patrick D. Tansley, James Hardy, Robert S. George, Christopher T. Bowles, Margaret Burke, Nicholas R. Banner, Asghar Khaghani, and Magdi H. Yacoub. Left ventricular assist device and drug therapy for the reversal of heart failure. *New England Journal of Medicine*, 355(18):1873–1884, 2006. PMID: 17079761.

[9] Christopher E. Brennen. *Hydrodynamics of Pumps*. Cambridge University Press, 2011.

[10] W. Bujalski, Z. Jaworski, and A.W. Nienow. CFD study of homogenization with dual rushton turbinescomparison with experimental results: Part ii: The multiple reference frame. *Chemical Engineering Research and Design*, 80(1):97 – 104, 2002. Process and Product Development.

[11] ASTM committee et al. Standard practice for assessment of hemolysis in continuous flow blood pumps. *Annual Book of ASTM Standards, F1844-97*, 13:1–5, 1998.

[12] S.M.A. Cruz, A.J.M. Cardoso, and H.A. Toliyat. Diagnosis of stator, rotor and airgap eccentricity faults in three-phase induction motors based on the multiple reference frames theory. In *Industry Applications Conference, 2003. 38th IAS Annual Meeting. Conference Record of the*, volume 2, pages 1340–1346 vol.2, Oct 2003.

[13] S. Eisenstat and H. Walker. Choosing the forcing terms in an inexact newton method. *SIAM Journal on Scientific Computing*, 17(1):16–32, 1996.

[14] U.S. Food and Drug Administration (FDA). Fdas critical path computational fluid dynamics (cfd)/blood damage project, 2013.

[15] R.G. Ghanem and P.D. Spanos. *Stochastic Finite Elements: A Spectral Approach*. Civil, Mechanical and Other Engineering Series. Dover Publications, 2003.

[16] Thomas J.R. Hughes, Gonzalo R. Feijo, Luca Mazzei, and Jean-Baptiste Quincy. The variational multiscale methoda paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166(1):3 – 24, 1998. Advances in Stabilized Methods in Computational Mechanics.

[17] Volker John and Songül Kaya. A finite element variational multiscale method for the navier-stokes equations. *SIAM J. Scientific Computing*, 26:1485–1503, 2005.

[18] P.C. Krause. Method of multiple reference frames applied to the analysis of symmetrical induction machinery. *Power Apparatus and Systems, IEEE Transactions on*, PAS-87(1):218–227, Jan 1968.

[19] Olivier Le Maître and Omar M Knio. *Spectral methods for uncertainty quantification: with applications to computational fluid dynamics*. Springer Science & Business Media, 2010.

[20] JY Luo and AD Gosman. Prediction of impeller-induced flow in mixing vessels using multiple frames of reference. INSTITUTE OF CHEMICAL ENGINEERS SYMPOSIUM SERIES, 1994.

[21] Kozo Naito, Kazumi Mizuguchi, and Yukihiko Nos. The need for standardizing the index of hemolysis. *Artificial Organs*, 18(1):7–10, 1994.

[22] Francis D. Pagani, Leslie W. Miller, Stuart D. Russell, Keith D. Aaronson, Ranjit John, Andrew J. Boyle, John V. Conte, Roberta C. Bogaev, Thomas E. MacGillivray, Yoshifumi Naka, Donna Mancini, H. Todd Massey, Leway Chen, Charles T. Klodell, Juan M. Aranda, Nader Moazami, Gregory A. Ewald, David J. Farrar, and O. Howard Frazier. Extended mechanical circulatory support with a continuous-flow rotary left ventricular assist device. *Journal of the American College of Cardiology*, 54(4):312 – 321, 2009.

[23] M. Schäfer, S. Turek, F. Durst, E. Krause, and R. Rannacher. *Benchmark Computations of Laminar Flow Around a Cylinder*, pages 547–566. Vieweg+Teubner Verlag, Wiesbaden, 1996.

[24] Micheal Schick, Chen Song, and Vincent Heuveline. A polynomial chaos method for uncertainty quantification in blood pump simulation. In *International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP), Greece, 2015*. Scopus, Elsevier, 2015.

[25] Christoph Schwab and Radu Alexandru Todor. Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. *IMA Journal of Numerical Analysis*, 27(2):232–261, 04 2007.

[26] Chen Song. *Uncertainty quantification for a blood pump device with generalized polynomial chaos expansion*. University library of Ruprecht-Karls-Universität Heidelberg, Heidelberg, 2018. Dissertation.

[27] Chen Song and Vincent Heuveline. Multilevel preconditioner of polynomial chaos method for quantifying uncertainties in a blood pump. In *International Conference on Uncertainty Quantification in Computational Sciences and Engineering (UNCECOMP), Greece, 2017*. Scopus, Elsevier, 2017.

[28] D. Xiu and G. Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM Journal on Scientific Computing*, 24(2):619–644, 2002.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

A HEALTH MONITORING FRAMEWORK FOR OPTIMAL SERVICE LIFE PREDICTIONS OF STEEL STRUCTURES UNDER FATIGUE LOADING

Nour A. Wehbi[1], **Wael G. Slika**[1]

[1] Beirut Arab University
Beirut-Lebanon
{n.wehbi,w.slika}@bau.edu.lb

Abstract

The accumulated damage in aging Steel Structures, especially due to fatigue, is considered as a critical phenomenon that affects safety and serviceability of civil engineering structures. Although, fatigue damage is influenced by various parameters such as the frequency of loading, sequence of load application, material properties, geometry, etc, in practice simplified S-N curve is typically used for condition assessment. In order to mitigate risks of catastrophic failure resulting from fatigue brittle nature, even in normally ductile materials, researchers have generated several non-linear damage models to predict the remaining service life of the structure considered. These models are mainly based on the S-N curve, material dependent parameters and loading conditions. However, due to the complexity of the fatigue phenomenon and expensive-long term full scale experimental testing, the models presented in literature have shown high degree of uncertainty due to simplifications of mathematical models, parametric uncertainties and varying loading conditions. Furthermore, the usage of S-N curve generated from experimental work is limited to identical loading mechanism and constant boundary conditions. Therefore, this study presents a structural health monitoring approach to overcome the limitation and inaccurate estimation of damage quantification models. The suggested framework relies on fatigue damage prediction models incorporated with real time damage records. All sources of uncertainty are incorporated in the health monitoring scheme to guarantee an optimal statistical identification of the state damage. The accuracy and robustness of the presented scheme will be assessed through a set of controlled experiments and numerical simulation of real case scenario.

Keywords: Data assimilation**,** Ensemble Kalman filter, Fatigue damage, Steel structures, Uncertainty quantification.

1 INTRODUCTION

When designing steel structures, a significant attention must be paid for a critical phenomena known as “material fatigue”. Fatigue means that the material can fail below its monotonic strength when its subjected to repeated loading due to accumulated deterioration in its stiffness. In order to mitigate catastrophic structural failure due to brittle fatigue failure, accurate prediction of the remaining service life is essential. Therefore, several nonlinear modeling attempts were made in [1 to 4], based on different well established fatigue fundamentals, yet their application requires modification of S-N curve parameters or the determination of material properties. Recently, a new damage model is generated on the basis of S-N curve parameters without requiring neither additional parameters nor curve modifications [5]. The validity of the later model was checked for two steel materials of grades C45 and 16Mn and found to be satisfactory. However, in practice, several sources of uncertainty are identified once developing any damage detecting model, such as mathematical simplifications, experimental errors and variability in loading conditions, that can yields non-satisfactory prediction accuracy [6].

In the light of what is presented, the aim of this research is to present a structural health monitoring framework to calibrate the predictive response of the fatigue damage model to mitigate brittle failure. The presented framework is based on Ensemble Kalman filter and real time damage which updates the statistical characteristics of the selected model parameters. Therefore, to guarantee accurate statistical representation of the output response, all statistical input errors, such as initial guess errors, model errors and measurement errors are quantified and incorporated in the developed data assimilation technique.

The accuracy of the presented scheme is assessed based on numerical and experimental verification. The importance of this developed strategy relies in its ability to significantly enhance the predicted model parameters to provide accurate determination of fatigue damage prior to any critical damage or sudden failure.

2 FATIGUE DAMAGE MODELS: COMMON LIMITATIONS

The history of Cumulative fatigue damage started more than eighty years ago when Miner [7] suggested the concept known as “The Linear Rule”. In 1945, Miner displayed this concept in a mathematical form based on the summation of the ratio of number of cycles to the total number of cycles at failure under constant load amplitude. Due to the simplicity and easy application of this rule, it has been adopted by several researchers and recommended by design codes such as Euro code [8]. However, it was obtained by several researchers that Miner’s rule may lead to unrealistic life estimation since it doesn’t consider the damage due to load sequence and interaction. As a result, improvements were done by Palmgren in the form of nonlinear Palmgren-Miner rule, yet the modified rule depends on fatigue testing to determine its parameters. Many other nonlinear models, such as [9], were developed based on material parameters that can be obtained only through extensive testing. Though these models have proven to provide satisfying agreement with experimental data, yet the necessity of material testing hindered their engineering applications.

In the 1999, efforts were paid to generate damage evolution curves for several materials that are based on experimental damage records versus the number of cycles to failure. As a result, several models were proposed to provide agreement with the tested experimental data of the damage curves, but it’s still limited to specific materials only. To overcome this issue, a sequence law was generated and then applied in steel bridges [10] requiring only full range of S-N curve and provided excellent results. Similar to the case of previous concepts and models, this sequence law didn't have wide implementation in engineering problems since it needs the availability of full range S-N curve, as well as, it can't be applied for bilinear and trilinear S-N curves found in codes.

Therefore, although several fatigue prediction models are available nowadays, design codes and standards are still recommending Miner's Rule due to its simplicity and independency from extensive material testing.

In this study, the fatigue prediction model presented in [5], and summarized below, will be adopted to simulate the material damage in the EnKF propagation. This model serves as a convenient candidate for its simplicity and significant prediction accuracy. Moreover, this model is based solely on one parameter which makes it an attractive option for data filtering techniques as it reduces the possibility of over fitting and inefficient filtering. The details of the adopted model are discussed in the next section.

3 SELECTED FATIGUE DAMAGE MODEL

In data assimilation setting, the main concern is to rely on a forward predictive model that can be both accurate and guarantees convergence to optimal solution. Based on these conditions, a fatigue damage model relying only on existing S-N curve parameters was selected. For instant, the accuracy of this model was verified for several applications and it possesses a simple parametric structure that ensures convergence in a data filtering setting.

3.1 S-N Curve Overview

The S-N diagram is the plot of nominal stress amplitude S versus the number of cycles to failure N. There are numerous testing procedures to generate the required data for a proper S-N diagram. The S-N test data is often presented as log-log plot, with the actual S-N line denoting the mean of the data from several experimental tests.

In fact, the material response to applied stress is considered complicated since there are several factors that can alter the endurance limit which is displayed in figure 1. These factors include: surface finish, size, type of loading, temperature, corrosive, and other aggressive environments, mean stresses, residual stresses, and stress concentrations. Also, fatigue data must be obtained from specimens and used in design for structural safety as mentioned in design codes. However, this information is not often available, so approximations of the S-N curve must be made. To overcome the stated obstacle in the S-N curve, the Basquin model is selected to be an approximation of the curve found in standards; besides endurance limit corrections. What makes Basquin a proper selection is related to its simplicity and ability in providing accurate results as mentioned in the literature.

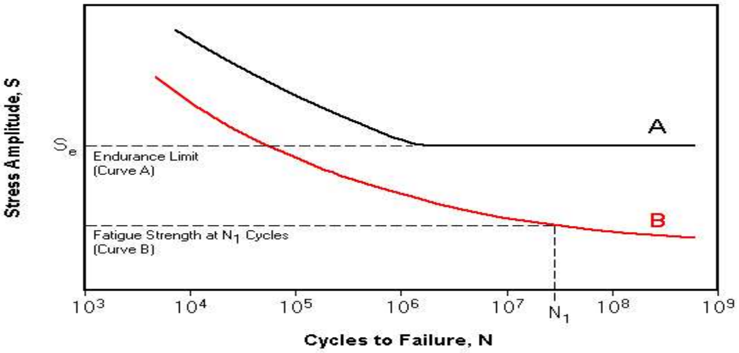


Figure 1: Typical S-N Curve of two Different Materials

Basquin representation of S-N curve is demonstrated by Eq.1

$$S=aN^{b} \qquad (1)$$

Where, S is the applied stress ,N is the number of cycles at failure, a and b are coefficients related to the tested material.

3.2 Damage Index

The chosen damage model [5] shown in Eq.2, is based only on one modal parameter that can be applicable to any design category including material discontinuity and stress concentrations. Consequently, this parameter overcomes the need of material testing and S-N curve modifications as it was the case in previous nonlinear models [9,11].

$$Di=1-[1-n_i/N_i]^{\delta i} \qquad (2)$$

Where, ni is the number of cycles at specific stress amplitude, Ni is the total number of cycles at failure, and δi is the modal parameter calculated based on Eq.3.

$$\delta i =-1.25/\ln(N_i) \qquad (3)$$

3.3 Applicability of the Selected Model

The presented damage model is extended under the concept of damage transfer to introduce a new parameter, denoted by load interaction factor, to account for variable amplitude loading and load sequence along with the interaction between them. This new concept was verified with experimental results to give satisfactory estimation of the remaining fatigue life. Furthermore, unlike the case of most nonlinear models, the current model presented an application section on butt and fillet welded joints and showed better prediction of fatigue damage compared to other models including Miner rule [5].

4 SEQUENTIAL DATA ASSIMILATION AND CASE STUDY

Sequential data assimilation is based on estimating unknown state variables based on the dynamic response of the structure besides the available observation data. Recently, sequential data assimilation has been popular in many engineering fields especially with the advances in monitoring technique and computer-based simulations [12,13]. A widely used sequential assimilation filters are the Kalman filter family. The standard Kalman filter was initially derived based on minimum variance error for the system with linear dynamics and Gaussian errors.

However, several extensions were suggested later to overcome its limitation and to make it applicable on systems with nonlinear dynamics, leading to the formulation of the Ensemble Kalman Filter [14].

The Ensemble Kalman filter was generated by Evensen 1994 [14] as an alternative of the extended Kalman Filter which is limited by statistical linearization and closure approximation. The EnKF propagates the state vectors of various samples forward in time so that they can be updated at each time increment of available measurements.

In this study, the state vector is composed of S-N curve model parameters, maximum number of cycles N, damage index δ and the predicted damage level using damage model elaborated in section 3. However, the observed quantity in this study is the material damage. In practice, material damage can be monitored through several destructive and non-destructive techniques, such as uniaxial or multi axial testing, vibration analysis, ultrasonic wave analysis, deflection response etc…

5 UNCERTAINTY QUANTIFICATION

Upon developing a mathematical model to simulate the fatigue damage of steel members, many sources of uncertainty are identified. First, the model input parameters have a wide range of variability. For instant, the endurance strength displayed by the S-N curve is influenced by several parameters as previously discussed in section 3.1. Furthermore, additional uncertainty arises by the simplification of mathematical models adopted, as well as, field measurement errors and varying application conditions. Consequently, to accurately detect the fatigue damage in steel members, researchers are interested by Structural Health Monitoring (SHM) along with Ensemble Kalman Filter to accurately identify and minimize these uncertainties.

In this research, all contributing sources of uncertainty are identified and incorporated in the EnKF framework. The adopted damage prediction model is verified on several experimental data and the average error was estimated to be around 9% and the maximum error is found to be around 20%. The statistical distribution of the error in the suggested model, based on all the presented experimental results in [5], can be statistically verified to be modeled as a normal distribution with zero mean and a standard deviation of 8% of the predicted value. The measurement noise is also considered in this study. Since measurement error depends on several factors such as, human errors, type of measurement and scale of the project, in this exercise it will be assumed a white noise with a standard deviation of 1.5% of the measured state. The initial parameters statistics of S-N curve are commonly represented by a lognormal distribution [6]. Therefore, a lognormal statistical distribution will be used to simulate the initial error statistics.

6 NUMERICAL AND EXPERIMENTAL VERIFICATION

In the following section robustness and accuracy of the presented scheme will be assessed based on a simulated numerical example and experimental data.

6.1 Numerical verification

In order to assess convergence and stability of the presented scheme, a set of simulated measurements based on the damage prediction model, presented before, are employed to serve as the real time measurements for the calibration of the EnKF. True state S-N curve model parameters, a,b and N, are used to simulate the True state damage data. Moreover, to simulate a real case scenario, an 8% normal error perturbation is added to the simulated damage data. Finally, for better resemblance of the measured data, this exercise also accounts for measurement errors by adding a white noise with 1.5% standard deviation of the measured response. Therefore, a measurement error will be added to the perturbed true damage data and will be considered as the measured data or measured state. A summary of the measured data simulation is presented in table 1.

Variable	Value
aTrue	2738
bTrue	-0.241
Stress	200 Mpa
N (Maximum Cycles at stress=200MPa)	$5.4\text{x}10^4$ Cycles
Perturbed True state	True state+8% error
Measure data	Perturbed True state+1.5% error

Table 1 : Input for simulation of measured data

To initiate the EnKF framework, an initial set of parameters, model errors and measurement were selected as discussed in the uncertainty quantification section. A large ensemble sample size equals to 20,000 sample was selected to accurately represent and integrate the time varying statistics of the state vector. The update frequency is 1350 cycles (2.5% of True life time cycles). Table 2 summarizes the initialization of the EnKF framework.

Parameter	Mean	Standard deviation	Distribution
a initial	3300	15% of mean	Lognormal
b initial	0.21	15% of mean	Lognormal
Model error	0	8% of predicted state	Normal
Measurement error	0	1.5% of measured state	Normal

Table 2: EnKF framework statistical input

The simulated measurements are incorporated in the EnKF and serve as the real time data that calibrate the state vector. The measurements data, the EnKF mean prediction (mean prediction with real time updates) and the initial mean prediction (mean prediction without EnKF updates) are presented in figure 2. It's evident from figure 2 that the EnKF significantly improves the prediction of the steel damage with time. It's worth noting here that even less than 20% error in the initial parameter estimation can drastically deteriorate the prediction potential of the damage model as compared to measured damage. The notable error in the initial prediction is attributed to the exponential nature of the damage prediction. The presented results serve as evidence on the power of the presented framework and emphasize the need to incorporate real time data with damage prediction models to render accurate and useful results.

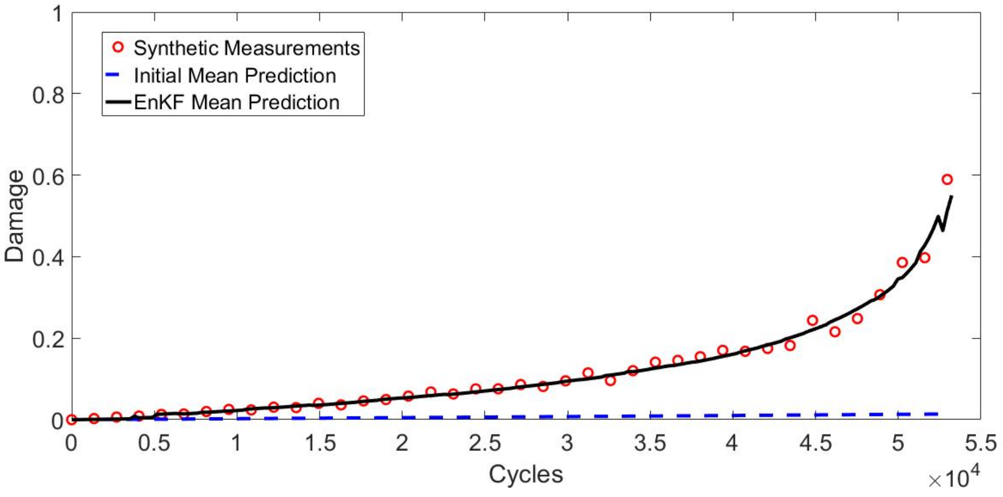


Figure 2: EnKF mean damage prediction versus initial mean prediction versus synthetic data

The prediction capabilities of the suggested framework are further investigated in this study by predicting the damage using mean state vector estimates after different updates. Figure 3 shows the prediction of fatigue after 0, 1.08x10^4, 2.7x10^4, 3.78 x10^4 cycles that corresponds to 0%, 20%, 50% and 70% of the life time of the structure. Figure 3 clearly shows the convergence of EnKF mean prediction to the true state as more measurements become available. This figure confirms that EnKF significantly increases the accuracy of damage prediction starting from early life stage of the structure (after 20% of structure life) when compared to true state. Therefore, the presented framework can guide a well informed decision making analysis to early detect risks and to efficiently update maintenance schedules.

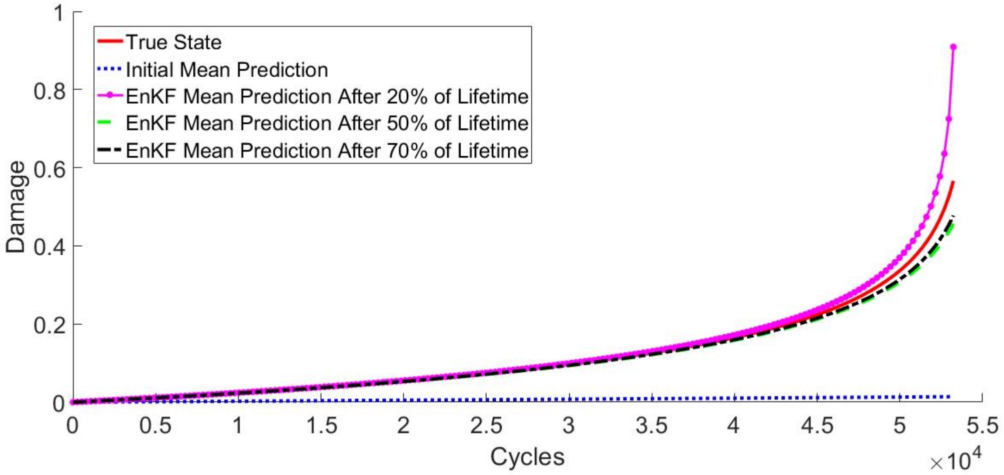


Figure 3: EnKF mean damage prediction versus updates of number of cycles

An important variable of interest, in addition to damage factor, is the maximum number of cycles N, at a specified stress level, that can be safely carried by structure before failure. Therefore, the ability of the presented framework to predict N is also investigated. Table 3 summarizes the statistical prediction of N starting from different EnKF updates or after different cycles. The presented results in table 3 reassure the ability of the framework to predict the damage starting from early life of the structure especially that the true state lies within one standard deviation from the mean estimate for all presented EnKF simulations. It's also worth noting that the standard deviation decreases as more cycles or updates become available reflecting a more confident prediction especially after 50% of the structure life time (less than 2% of predicted mean).

Available Measurements	**EnKF Predicted Mean lifetime (Cycles)**	**Standard deviation (% of Predicted Mean)**
Up to 1.08x10^4 Cycles (20% of N)	5.24 x10^4	8.86%
Up to 2.7 x10^4 Cycles (50% of N)	5.53 x10^4	1.91%
Up to 3.78x10^4 Cycles (70% of N)	5.50 x10^4	1.31%
True value of lifetime cycles	5.4 x10^4	-

Table 3: EnKF statistical lifetime prediction versus updates or number of cycles

6.2 Experimental verification

To further investigate the accuracy of the suggested framework, an experimental data set presented in [5] is utilized for real time calibration of the employed damage model. A random initial distribution data was used as initial prediction of the damage. Figure 4 presents the experimental data versus initial mean prediction and the EnKF mean prediction. The presented results emphasize the robustness of the EnKF framework as all the experimental data points were predicted with less than 15% error, even when the initial prediction error exceeded 100% of the measured value.

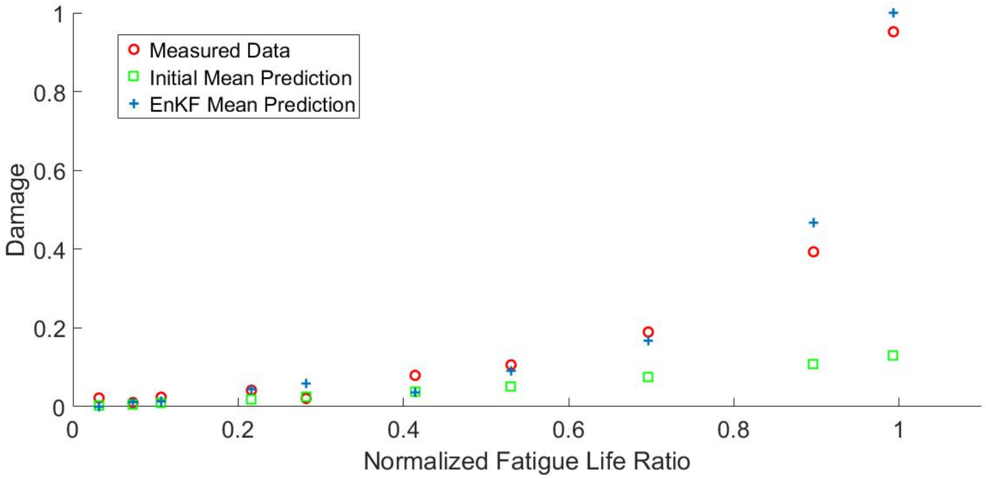


Figure 4: EnKF mean damage prediction versus initial mean damage prediction versus experimental data

7 CONCLUSIONS AND FUTURE WORK

This study presents an EnKF framework for real time update of predicted damage and lifetime of steel structures in a statistical setting. The accuracy of the suggested framework was verified on numerical example and experimental data. The presented results showed significant improvement over initial prediction whch motivates further investigation and testing of the framework on large scale and complicated steel projects.

REFERENCES

[1] Rege, Kristen, and Dimitrios G. Pavlou. "A one-parameter nonlinear fatigue damage accumulation model." *International Journal of Fatigue* 98 (2017): 234-246.

[2] Fatemi, A., and Lianxiang Yang. "Cumulative fatigue damage and life prediction theories: a survey of the state of the art for homogeneous materials." *International journal of fatigue* 20, no. 1 (1998): 9-34.

[3] Richart, F. E., and N. M. Newmark. "An hypothesis for the determination of cumulative damage in fatigue." In *Selected Papers By Nathan M. Newmark: Civil Engineering Classics*, pp. 279-312. ASCE, 1948.

[4] Marco, S. M., and W. L. Starkey. "A concept of fatigue damage." *Trans. Asme* 76, no. 4 (1954): 627-632.

[5] Aeran, Ashish, Sudath C. Siriwardane, Ove Mikkelsen, and Ivar Langen. "A new nonlinear fatigue damage model based only on SN curve parameters." *International Journal of Fatigue*103 (2017): 327-341.

[6] Sudret, Bruno, P. Hornet, J-M. Stephan, Z. Guede, and Maurice Lemaire. "Probabilistic assessment of fatigue life including statistical uncertainties in the SN curve." (2003).

[7] Miner, M. A. "Cumulative fatigue damage." *Journal of applied mechanics* 12, no. 3 (1945): A159-A164.

[8] Euro code 3: *design of steel structures –part 1-9: fatigue*. NS-EN 1993-1-9: 2005+NA: 2010.

[9] Lemaitre, J., and A. Plumtree. "Application of damage concepts to predict creep-fatigue failures." *Journal of Engineering Materials and Technology* 101, no. 3 (1979): 284-292..

[10] Siriwardane, Sudath, Mitao Ohga, Ranjith Dissanayake, and Kazuhiro Taniwaki. "Application of new damage indicator-based sequential law for remaining fatigue life estimation of railway bridges." *Journal of Constructional Steel Research* 64, no. 2 (2008): 228-237.

[11] Shang, De-Guang, and Wei-Xing Yao. "A nonlinear damage cumulative model for uniaxial fatigue." International Journal of Fatigue 21, no. 2 (1999): 187-194.

[12] Slika, Wael, and George Saad. "A practical polynomial chaos Kalman filter implementation using nonlinear error projection on a reduced polynomial chaos expansion." International Journal for Numerical Methods in Engineering 112, no. 12 (2017): 1869-1885.

[13] Slika, Wael, and George Saad. "An Ensemble Kalman Filter approach for service life prediction of reinforced concrete structures subject to chloride-induced corrosion." Construction and Building Materials 115 (2016): 132-142.

[14] Evensen, Geir. "Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics." Journal of Geophysical Research: Oceans 99, no. C5 (1994): 10143-10162.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

EVALUATING SYSTEM RELIABILITY OF CABLE-STAYED BRIDGES CONSIDERING CABLE CORROSION

N. Lu[1], and Y. Liu[2]

[1] Changsha University of Science and Technology
Changsha Hunan 410114, China
e-mail: lunaiweide@163.com

[2] Changsha University of Science and Technology
Changsha Hunan 410114, China
e-mail: liuyangbridge@163.com

Abstract

The cables in a cable-stayed bridge are prone to fatigue damage and atmospheric corrosion, which directly affect the bridge safety. This study presents a framework for system reliability evaluation of in-service cable-stayed bridges subjected to cable degradation. The effect of cable strength degradation on the bridge reliability is demonstrated through simulation on a parallel-series system representation. Machine learning techniques are utilized to approximate the nonlinear and dynamic response surfaces of critical components due to cable rupture, and the system reliability is finally evaluated from the event tree established by the β-unzipping method. Both short-span and long-span cable-stayed bridges are selected as prototypes to investigate the influence of cable degradation on the structural system reliability. System reliability of the bridge under ultimate limit state was analyzed. Numerical results show that: the intelligent algorithm is applicable in system reliability assessment of cable-stayed bridges; the main failure sequence of the cable-stayed bridges is strength failure of cables in side-span followed by bending failure of towers in the cross section of tower and girder, and the second failure sequences is strength failure of cables in mid-span followed by bending failure of girders in root section; Degradation of cables is the main factor to influence on system reliability of cable-stayed bridges. Compared with fatigue damage of cables, the corrosion of cables leads to a larger decline of system reliability indices. It is important for managers to provide maintenance and prompt replacement measures in order to insure the system reliability of cable-stayed bridges in operational period.

Keywords: bridge engineering, system reliability, corrosion, fatigue, failure sequence.

1 INTRODUCTION

Cable-stayed bridges are widely used to cross canyons and rivers because of their long-span capacity and economic property. The long-span capacity of the cable-stayed bridge originates from the elastic support of the stay cables. However, the stay cables are vulnerable caused by the corrosion and the fatigue damage accumulation (Deeble et al. 2012; Yan, et al. 2012). The failure of a stay cable can lead to the failure of another stay cables or girder, and then the propagation will lead to the collapse of the entire structure. Such collapses were generally summarized as the term of progressive collapse. Mehrabi et al. (2010) indicated that 39 out of 72 cables of the Hale Bogges Bridge critically needed repair or replacement after 25 years of service. In practice, the corrosion and fatigue damage will make a great contribution to the degradation of the cable strength resistance. This phenomenon results in a continuous declination of the strength resistance of the stay cable. Even through recommendations for robustness were provided in the design codes of cable-stayed bridges, the impact of degradation or loss of cables on the structural safety are still not clear. Thus, it’s extraordinarily necessary and urgent to evaluate the safety of cable-stayed bridges with degenerated stay cables.

Lots of achievements have been obtained on Mechanical property and dynamic property of cable-stayed bridges under the case of losing cables. Mozos and Aparicio (2011) studied the structural dynamic behavior of a cable-stayed bridge during the rupture of a stay cable. Wolff and Starossek (2010) studied the collapse resistance and collapse behavior of a cable-stayed bridge caused by loss of cables and recommended a robust design for avoiding the propagation of the cable loss. Wolff and Starossek (2009) examined the structural non-linear dynamic responses of a cable-stayed subject to the loss of a stay cable. However, most of the existing studies focused on the deterministic analysis without considering the structural uncertainties or utilizing a probability approach to assess the structural safety.

Taking account of the structural uncertainties and the random loads, the failure of a stay cable is a probability and the propagation route is random. Thus, a reliability-based approach is needed for uncertainty induced safety assessment of in-service cable-stayed bridges. In particular, system reliability theory provides an appropriate solution for searching the failure sequence and evaluating the failure probability of the entire system. In the present, most research efforts were concentrated on developing an efficient algorithm for estimating the structural system reliability. The time demanding Monte Carlo simulation was proved to be not suitable for calculating the system reliability of bridges with an extremely low failure probability. The famous non-sampling method is theβ-bound method (Thoft Christensen, 2012) and the branch-and-bound method (Lee and Song, 2011). In addition, the advanced response surface method, article neural network, and other type of meta-model were presented to develop the efficiency or accuracy of the β-bound method. However, the application of the system reliability to the safety assessment of long-span bridges, such as cable-stayed bridges, is still insufficient. Bruneau (1992) utilized system reliability method to analysis the ultimate global behavior of a cable-stayed bridge and discovered 9 potential failure patterns for the cable-stayed bridge. Estes and Frangopol (2001) presented a system model by combining ultimate and serviceability limit state of a highway bridge. Cheng and Xiao (2005) studied the serviceability reliability of cable-stayed bridges by utilizing the response surface method and finite element method and indicated that the cable sag had a major effect on the structural reliability assessment. Liu et al. (2016) developed an adaptive support vector regression approach to assess structural system reliability. However, the degeneration of the cable strength was not considered in the above mentioned works. Li et al. (2012) utilized structural health monitoring data to evaluate the reliability of a long-span bridge.

In view of the large complex structure, especially the research and application of reliability evaluation of structural system of cable bridge load-bearing structure c is relatively small. The main reason are: First of all, with the increase of bridge spans, the nonlinear increase of structure leads to the complex of limit state surface, leads to that the conventional first order two moment method (FOSM) and the MCS are no longer suitable for solving reliability index of the complicated structure directly; The second, the complex structure results in the arge increase of failure mode and failure path. It makes the construction of structure system failure tree and the calculation system reliability index difficulties. So, there is an urgent need to develop study on the reliability of the bridge structure system from the efficient intelligent algorithm.

This study aims at evaluating the system reliability of cable-stayed bridges with corroded stay cables. First, the mathematical modeling of potential failure pattern of the cable-stayed bridge and strength degradation of the stay cables was conducted. Subsequently, a computational framework integrating the intelligent learning machine technology and efficient searching technology of failure sequences was proposed. Finally, a prestressed concrete cable-stayed bridge was selected as a prototype to conduct the evaluation analysis. Main failure sequences of the cable-stayed bridge are identified. Influence of the strength degradation on the structural system reliability is studied.

2 THEORETICAL BASIS

2.1 Failure modes of cable-stayed bridges

In order to introduce the general failure modes of cable-stayed bridges, consider a sample single tower cable-stayed bridge provided by Bromn (1992) shown in Fig. 1. The distances between the cable anchors in the girders or in the towers are 30m. More details regarding the material and sectional properties and performance functions can be found by Bruneau (1992). In general, the cables are considered as brittle since the rupture of a stay cable is momentary. The concrete girders and towers for long-span bridges are considered as ductile since the prstressed structures are allowed to be large deformation. The structural system failure is defined by a plastic collapse mechanism. The plastic failure mechanism is identified by the plastic-hinge locations and plastic capacities. The potential failure locations are shown in Fig. 1. Points A-E is defined as bending failure of girders due to negative moment plastic hinges. Points I-L is strength failure of brittle cables. Points M-R is the bending failure of girders due to positive bending moment.

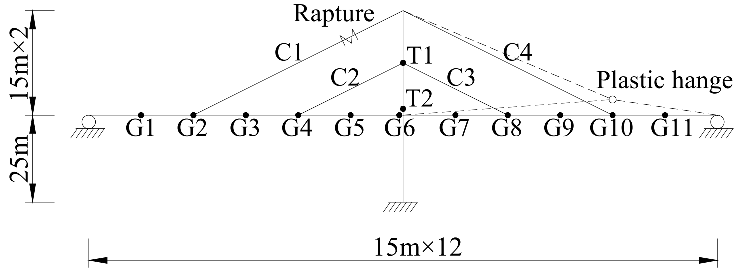


Fig. 1 Dimensions and component number of a cable-stayed bridge

In this case, the structural mechanical behavior is assumpted as linear and elastic for a more simple illustration. A component reliaiblity analysis was performed to evaluate the failure probability of each identified strcutral candidate element. When the cable rupture occures, delete the cable directly and continue the next analysis. When the the bending failure occures in the girders, add a plastic hinge in the location and continue the next analysis. It is acknologed that the structural stiffness and resistance are changing at any modified step. This means the remaining structural elements reform a new structural system and need reanalysis at the next step. This process will be repeated for updating the structural behavior. The new performance functions and new failure probability will be obtained in each process. As the end of the processes, the progression along the failure sequrnces will be stopped in case of the failure propbability of the final component is expected to be extremely high. Note that the process should be stopped for saving computational effort in case that the structure still have the load carrying capability but the system is very instability. The falut tree of the cable-stayed bridge are shown in Fig. 2. It is observed that the oritinal strcutural system reliaiblity index is beta=4.54, while with the consideration of the strengthend degradation of stay cables, the beta reduce to 3.5. Furthermore, the main failure sequence changed from the initial negative bending failure of girders to positive failure bending failure of girders followed by the failure of stay cables

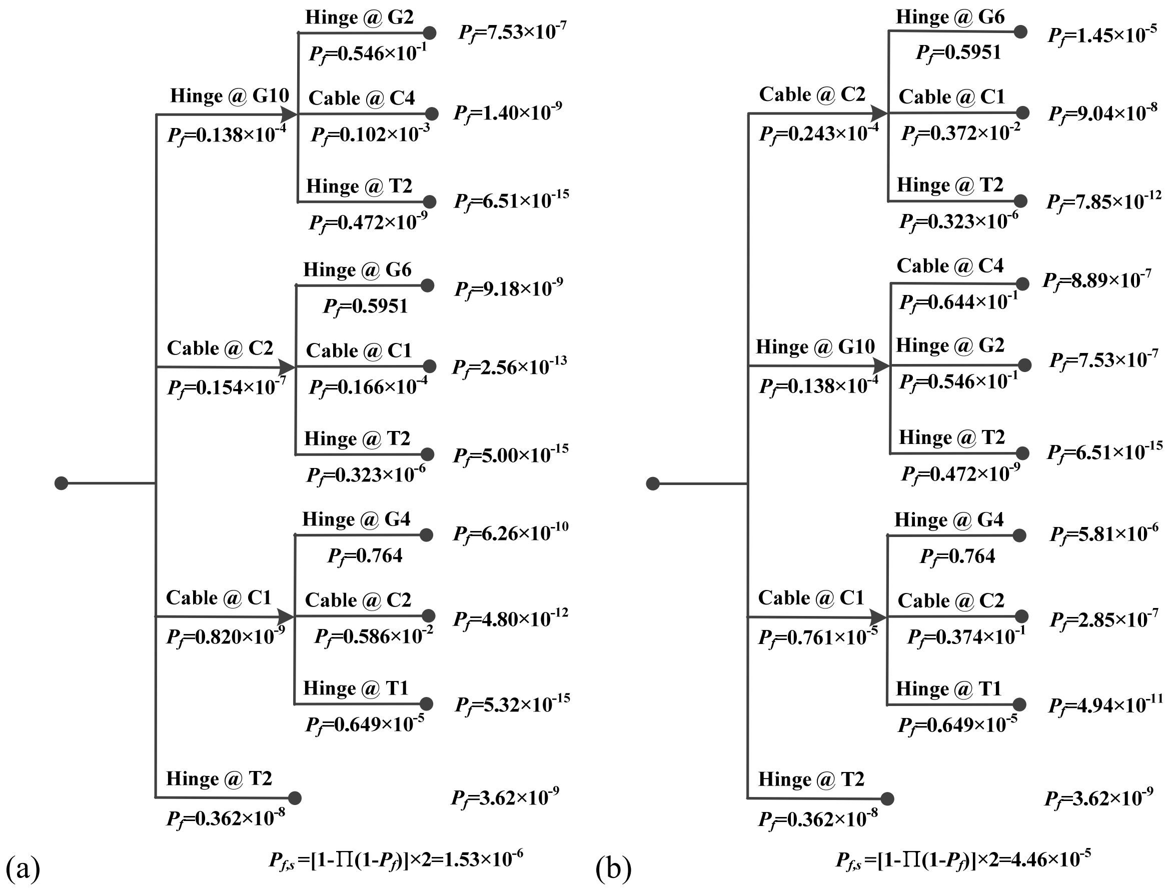


Fig. 2 Event tree of the cable-stayed bridge: (a) original model; (b) with consideration of strength degradation rate of 20% for the cables

As observed from Fig. 2, the following conclusions can be driven. Firstly, the failure probabilities of the cables are increased. For instance, the failure probability of the C2 decrease from 0.154×10^{-4} to 0.243×10^{-4}. Secondly, the dominant failure mode changed, where the initial domain failure mode is Hinges at G10 and G2, while the new domain failure mode is the rapture of cable at C2 followed by the Hinge at G6. Finally, the failure probability of the structural system increase from 1.53×10^{-6} to 4.46×10^{-5}. As elaborated above, the strength degradation of the stay cable not only impact the reliability of the stay cables, but also has a significant impact on the structural system reliability and the dominant failure mode.

2.2 Geometrically nonlinear effects

For the aforementioned simplified cable-stayed bridge, the structural nonlinear properties were not considered. However, as the span of modern cable-stayed bridges increases, the geometrically nonlinearity property become more obvious for the long-span cable-stayed bridge. Such nonlinearity mostly includes the cable slag effect, the beam-column effect and the large displacements effect. The cable sag effect is the most important one and the common solution is to use the Ernst method or modified elastic modulus for the cables. Considering a parabolic instead of a catenary shape for the cable, the modified modulus E_{eq} are written as (Freire et al. 2006)

$$E_{eq} = E\frac{1}{1+\frac{q^2L_h^2}{12T^3}EA} \tag{1}$$

where, E is the Young modulus of the material, T is the cable tension, q is the unit self-weight per length, L_h is the horizontal component, A is the cable cross section.

The beam-column effect is another feature for a typical cable-stayed bridge, because the prestress stay cables provide large axial force for the girders and towers. It is acknowledged that the beam-column interaction is a second-order effect and can be conveniently considered by utilizing stability functions. The interaction between the bending moments and axial force will modify the component stiffness coefficient and the internal forces. Assume a hollow rectangular section, where the neutral axis in the ultimate stays within the webs, from the sample plastic analysis, the axial bending interaction curve can be defined as (Yoo et al., 2012)

$$\frac{M}{M_P} = 1-\left(\frac{P}{P_P}\right)^2\frac{A^2}{4wZ_x} \tag{2}$$

where, M is the applied moment, M_P is the plastic moment capacity in the absence of axial loads, P is the applied axial force, P_P is the plastic axial force capacity in the absence of applied moment, w is the web thicknesses, and Zx is the bending plastic modulus.

2.3 Cable degradation modeling

Cable degradation is a common phenomenon in existing cable-supported bridges. This phenomenon is mostly caused by lacks of construction quality and regular maintenance. Therefore, under the long-term affection of corrosion and cyclic stresses, the stay cables became rusty and then fractured. In general, stay cables are supported by the steel strand or parallel wires. Liu et al. (2004) utilized a parallel-series model to establish the probability model of the steel wire cables during construction written as:

$$F_X(x) = 1-\exp[-\exp(x-1915)] \tag{3}$$

where, x is the strength (MPa) of a stay cable. It is observed that x follows an Extreme value distribution. With respect to the cable degradation, this study considers corrosion and fatigue effect.

First, Taking into account the cable degradation due to corrosion and fatigue, Tang () established the time varying model for cables of a cable-stayed bridge at Chongqing, China. The mean value and standard deviation of the degradation coefficient of the cables in 20 years are written as:

$$\begin{cases} \mu(t) = 0.97 + 0.0112t - 3.6\times10^{-3}t^2 + 8\times10^{-5}t^3 \\ \sigma(t) = 0.98 + 0.095t - 3.1\times10^{-3}t^2 + 7\times10^{-5}t^3 \end{cases} \tag{4}$$

where, t is the service period (year) of a bridge, $g(t)$ is the mean value of the degradation coefficient for the cable, g'(t) is the standard deviation of the degradation coefficient.

For fatigue-induced cable degradation, fretting-fatigue induced cable degradation is proposed by Wang et al. (2010), and is written as:

$$\begin{cases} \mu(t) = 1 - 3.2\times10^{-4}t & 0 < t < 13.8 \\ \mu(t) = 0.950 + 0.0121t - 6.4\times10^{-4}t^2 & 13.8 \le t < 20 \end{cases} \tag{5}$$

With the aforementioned cable degradation equations, the probability model can be then derived.

3 COMPUTATIONAL FRAMEWORK FOR STRUCTURAL SYSTEM RELIABILITY EVALUATION

Due to the high order statically indeterminate of the long-span cable-stayed bridge, the structural system is complex and time-variant during the long-term service period. First, since the component failure probability is extremely small, the rough Monte Carlo Simulation (MCS) is time-consuming. The multi failure sequences of the bridge lead to national computational efforts. The popular reliability evaluation approaches, such as, First Order Second Moment (FOSM) and Response Surface Method, are not excellent for solving aforementioned issues. Therefore, based on the above formulations, this study utilizes an intelligent combined computational framework to carry out analysis. Fig. 3 plots the flowchart of the framework. The main procedures of the framework are estimating component reliability by RBF neural network approach, and the searching failure sequences based on beta-bound theory. Details of the main procedures are discussed below.

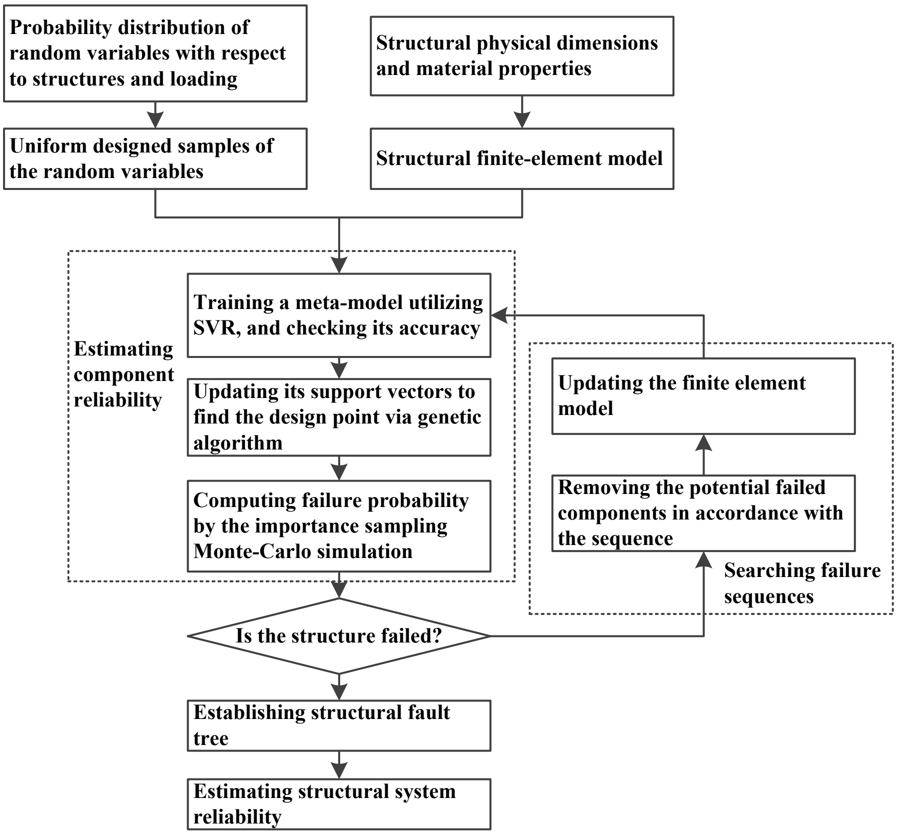

Probability distribution of random variables with respect to structures and loading
Structural physical dimensions and material properties
Uniform designed samples of the random variables
Structural finite-element model
Estimating component reliability
Training a meta-model utilizing SVR, and checking its accuracy
Updating its support vectors to find the design point via genetic algorithm
Computing failure probability by the importance sampling Monte-Carlo simulation
Updating the finite element model
Removing the potential failed components in accordance with the sequence
Searching failure sequences
Is the structure failed?
Establishing structural fault tree
Estimating structural system reliability

Fig. 3 Flowchart of the proposed computational framework

Support vector regression (SVR) is a machine learning approach that is more effective and accurate in comparison to the response surface method and the neural networks. The application of SVR in reliability evaluation is initially proposed. Subsequently, numerous advanced SVR approaches have been developed, such as least squares-SVR, particle filter-SVR, and Genetic algorithm-SVR (Dai et al. 2012; Zhang et al. 2012). This study utilized an adaptive support vector regression (ASVR) approach proposed by Liu et al. (2016) to estimate structural component reliability. Since the formulations can be found by Liu et al. (2016), the mainly steps are descripted below.

The ASVR utilizes two updating procedures to calculate the system reliability. The first updating procedure is to update the design point via the genetic algorithm. With such design point, the reliability index can be calculated by an optimization equation. The second updating procedure is to update the support vectors under the condition that the potential failure component is removed from the original structural finite element model. After the second updating procedure, a failure sequence is obtained. Continue to redo the above updating procedures, and all of the failure sequences will be found. Finally, the system reliability can be calculated by the parallel-serial approach with the established fault tree.

A GUI-based program is developed based on the ASVR for an efficient calculation. In this program, there are mainly two commercial finite-element programs including Matlab and Ansys. With respect to the cable-stayed bridge in the present study, there are some special steps for the system reliably estimation utilizing the ASVR approach. First, since the probability of failure of each component is extremely small, ranges of the variables in the first sampling design stage should be within μ-3σ and μ+3σ. After the first updating of support vectors, the sampling range can be reduced to μ-σ and μ+σ. Second, ANSYS is recommended as the commercial FE program, because there is connection between ANSYS and MATLAB, and the entire computational framework will be more convenient. Third, at the second updating procedure, the stay cables can be removed directly when the stay cables are considered as a potential failure component. The final failure mode of the cable stayed bridge is considered as the bending failure of the concrete girders and pylons components. Thus, the branching point of the flowchart in Fig. 4 is to check if the concrete girders and pylons is failed or close to failed.

4 CASE STUDY

4.1 Bridge details

Hejiang bridge is a cable-stayed bridge corssing Yangzi River at Sichuang, China. The layout and dimensions are shown in Fig. 4. According to the design documents, the material of girders and pylons is concrete, and the material of cables is steel strand. There are 4 traffic lanes in the opposite travelling directions. In Fig. 4, CBA34 denotes the serial number of the 34th Cable in the Mountain-side of the North-pylon, GNR34 denotes the serial number of the 34th Girder in the River-side of the South-pylon, T1 denote the first section of the pylon.

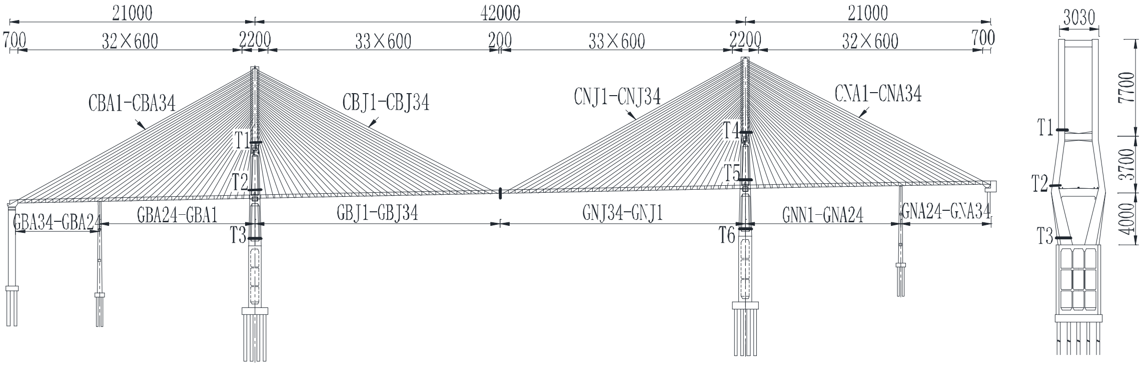


Fig. 4. Layout and series number of a cable-stayed bridge(unit: cm)

This bridge is select herein as a prototype to evaluate the system reliably with consideration of cable degradation utilizing the computational framework. The self-weight of initial cable force are considered. The live load is simplified as uniformly distributed load in the mid-span. The statistics of the random variables are shown in Table 1.

Table 1. Statistics of the random variables

Variable	Distribution	Mean value	Standard deviation	Remarks
E_1	Normal	3.64×10^4	3.64×10^3	Elastic modulus of girders
E_2	Normal	3.52×10^4	3.52×10^3	Elastic modulus of pylons
E_3	Normal	1.95×10^5	1.95×10^4	Elastic modulus of cables
A_1	Lognormal	20.846	1.042	Cross sectional area of girders
A_2	Lognormal	24.694	1.235	Cross sectional area of pylons
A_5	Lognormal	1.4×10-4	7.0×10-6	Cross sectional area of cables
γ_1	Normal	26.56	1.33	Equivalent unit weight
γ_2	Normal	26.24	1.31	Equivalent unit weight
γ_3	Normal	78.5	3.93	Equivalent unit weight
I_1	Lognormal	18.598	0.930	Moment of inertia of girders
I_3	Lognormal	118.412	5.921	Moment of inertia of pylons
q_1	Normal	132	6.6	Secondary deck load
q_2	Extreme value	63.5	6.35	Live load

4.2 Results and discussion

In the platform of ANSYS, establishing the parametric finite element model of the bridge through APDL language, considering the geometric nonlinear of cable element through equivalent elastic modulus. A total of 470 BEAM44 elements make up of the girder and tower, along with the stay-cables consisting of 270 LINK10 elements. Assuming the final hanger force as the real force regardless of the shrinkage and creep effect.

The appropriate simplified process is necessary when screening failure paths, with various failure paths and high correlation of the failure mode betwen the same components. The sytem reliability analysis was introduced from the angle of cable failure this paper. Condering Firstly the influence of bending moment, with respect to the key section of tower and beam, caused by successive failure of the north tower side span or medium span. Influence of parameters on the the bending moment of mid-span-point girder is shown in Fig. 5.

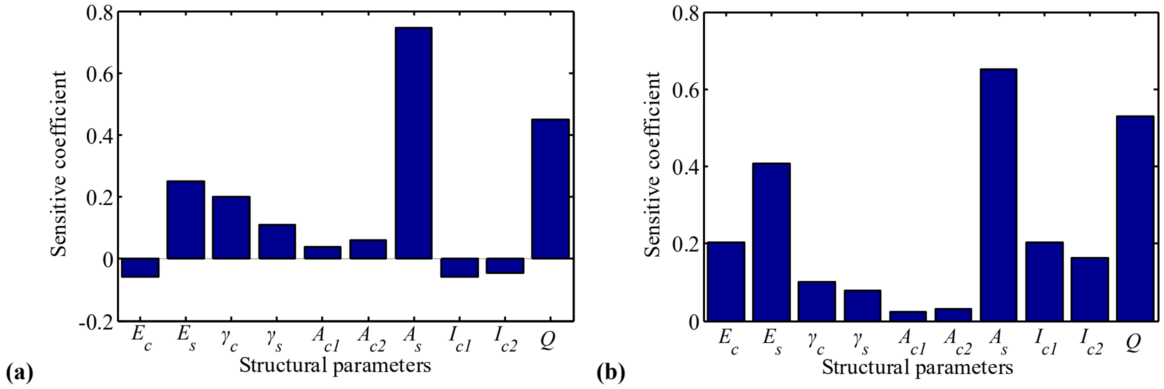


Fig. 5. Influence of structural parameters on the mid-span-point bending moment

As illustrated before, the failure of the side-span cable with a relatively large effect for bending moment on tower T2 section, while there appears big influence to the bending moment on girder caused by the failure of midspan cable. Therefore it is feasible to ignore branch events of beam bending failure and tower failure, caused respectively by the cable failure of side-span and midspan. Three layers fault tree with four failure paths of the cable-stayed bridge is as shown in Figure 6, calculated in accordance with the flow chart of combined intelligent algorithm. Figure 6 illustrates that, the range of system reliability index for cable-stayed bridge is from 7.24 to 7.38. It is the bending failure of tower resulting from side-span cable failure that affects greatly the system reliability index.

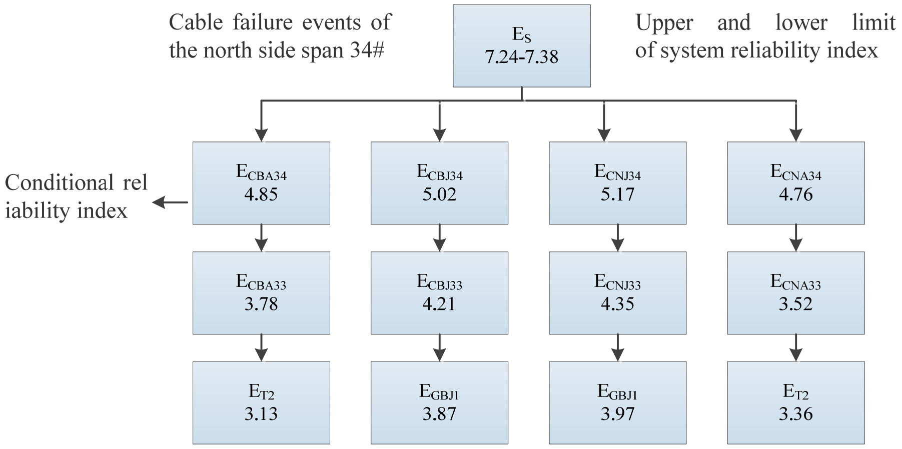


Fig. 6 Three-layer failure tree of the cable-stayed bridge

There is great influence on the structure system reliability index for the reliability index of the cable, with the failure events of the cable-stayed bridge system all resulting from failure of two pairs of cable. The variation tendency of reliability index, as shown in Figure 7, with respect to the two pairs of stay cables has been educed in 1~20 years, using function of force degradation resulting from the cable corrosion and fatigue.

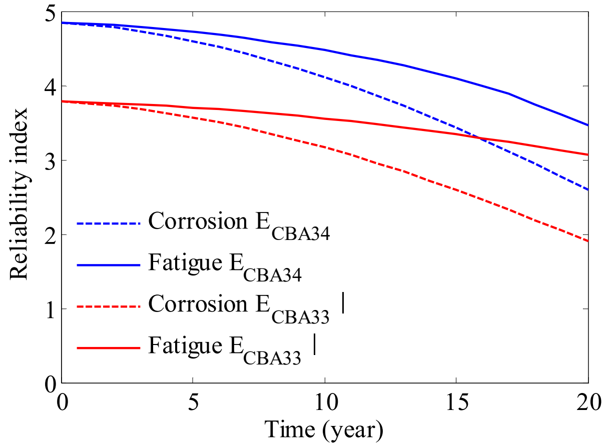


Fig. 7. Influence of corrosion and fatigue damage of cables to reliability indices

In Fig. 7, E_{CBA34} represents the failure event of the No. 34 cable on the north side, $E_{CBA33}|$ is the conditional probability event of the failure event of No. 33 cable, which follows the failure of No. 34 cable. Figure 7 illustrates that cable resistance deterioration caused by cable corrosion makes greater effect on the reliability index of the stay cables over time, compared with fatigue effects. The reliability index of 34#cable and 33# cable would respectively be 2.6 and 1.9 in 20 years, owing to mean resistance coefficient 0.42 of cable in 20th years according to the test data.

With the cable corrosion and fatigue damage, system reliability index downward trend of the cable-stayed bridge is as shown in Fig. 8 within 20 years.

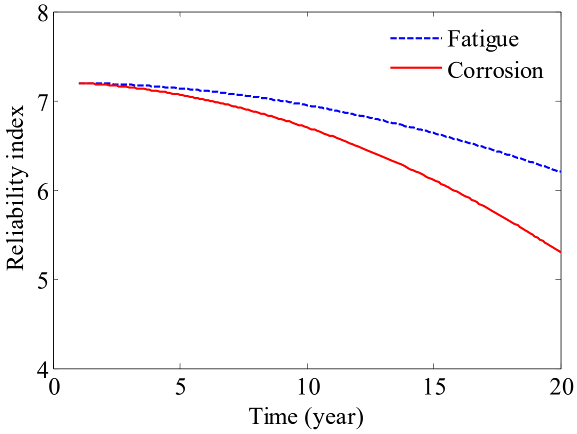


Fig. 8 Influence of corrosion and fatigue damage of cables to system reliability

In Fig. 8, the two ends of vertical lines at each time point denotes the upper and lower limit of system reliability index. Downward trend curve of system reliability index could be obtained connecting the mid-point of vertical lines. As Fig. 8 shows, it is the cable resistance degradation caused by the cable corrosion and fatigue damage that prompts the accelerated declination of system reliability index, considering resistance deterioration of the stay cables. The bound of mean reliability index respectively are 5.22 and 6.20 under the influence of these two factors in 20 years. Considering the great effect that cable corrosion makes on system reliability index, the important factor causing the decline of the system reliability for cable-stayed bridge is resistance degradation of cable due to the rustiness. Cable maintenance and timely replacement has an important significance on guaranteeing system security level of cable-stayed bridge in operation period.

5 CONCLUSIONS

Based on the failure characteristics of long-span cable-stayed bridge, this paper proposed the combined intelligent algorithm for reliability evaluation of structure system. The main failure paths for the cable-stayed bridge were investigated. The effect of cable degradation caused by corrosion and fatigue damage on the structural system reliability was investigated. The conclusions obtained are shown as follows:

(1) It is feasible to evaluate the system reliability of cable-stayed bridge accurately and efficiently by utilizing the combination of intelligent algorithm process.

(2) With respect to the concrete cable-stayed bridge of rigid frame system, the beam bending failure of cable tower, at the junction of tower and beam, caused by strength failure of the side-span cable is the main failure path. Secondly it is about the bending failure at the root section of girder caused by strength failure of the mid-span cable.

(3)The important factor with respect to the system reliability for cable-stayed bridge is the cable resistance degradation. There appears a larger decline about the system reliability in operation period caused by the cable corrosion, compared with the cable fatigue damage. Cable maintenance and timely replacement have significance on ensuring the security level of system reliability for cable-stayed bridge in operation period.

6 ACKNOWLEDGEMENT

The research was supported by the National Basic Research Program (973 program) of China (Grant No. 2015CB057705), the National Science Foundation of China (Grant No. 51378081), the China Postdoctoral Science Foundation (Grant No. 2015M580383), and the

Jiangsu Postdoctoral Science foundation (Grant No. 1501045B). Opinions, findings, and conclusions expressed are those of the authors and do not necessarily represent the views of the sponsors.

REFERENCES

[1] O. Yang, H. Li, J. Ou, et al. Failure patterns and ultimate load-carrying capacity evolution of a prestressed concrete cable-stayed bridge: case study. *Advances in Structural Engineering*, **16**(7), 1283-1296, 2013.

[2] S. Jang, H. Jo, S. Cho. Structural health monitoring of a cable-stayed bridge using smart sensor technology: deployment and evaluation. *Smart Structures and Systems*, **6**(5-6), 439-459, 2010.

[3] A. Mehrabi, C. Ligozio, A. Ciolko and S. Wyatt. Evaluation rehabilitation planning, and stay-cable replacement design for the Hale Boggs Bridge in Luling, Louisiana. *Journal of Bridge Engineering*, **15**(4), 364-372, 2010.

[4] C.M. Mozos, A.C. Aparicio. Numerical and experimental study on the interaction cable structure during the failure of a stay in a cable stayed bridge. *Engineering Structures*, **33**(8), 2330-2341, 2011.

[5] A.C. Estes, D.M. Frangopol. Bridge lifetime system reliability under multiple limit states. *Journal of Bridge Engineering*, **6**(6), 523-528, 2001.

[6] P. Thoft-Christensen, Y. Murotsu. *Application of structural systems reliability theory.* Springer Science & Business Media, 2012.

[7] S.M.J. Deeble, R. Betti, G. Marconi. Experimental analysis of a nondestructive corrosion monitoring system for main cables of suspension bridges. *Journal of Bridge Engineering*, **18**(7):653-662, 2013.

[8] S.M. Deeble, R. Betti, G. Marconi, A. Hong, D. Khazem. Experimental Analysis of a Nondestructive Corrosion Monitoring System for Main Cables of Suspension Bridges. *Journal of Bridge Engineering*, 10.1061/(ASCE)BE.1943-5592.0000399, 653-662, 2012.

[9] W. Yang, P. Yang, X. Li, W. Feng. Influence of tensile stress on corrosion behavior of high-strength galvanized steel bridge wires in simulated acid rain. *Materials and Corrosion*, 401-407, 2012.

[10] J. Cheng, R. Xiao. Serviceability reliability analysis of cable-stayed bridges. *Structural Engineering and Mechanics*, **20**(6), 609-630, 2005.

[11] M. Wolff, U. Starossek. Cable loss and progressive collapse in cable-stayed bridges. *Bridge structures*, **5**(1), 17-28, 2009.

[12] H. Dai, H. Zhang, W. Wang. A support vector density-based importance sampling for reliability assessment. *Reliability Engineering & System Safety*, **106**, 86-93, 2012.

[13] M. Bruneau. Evaluation of system-reliability methods for cable-stayed bridge design. *Journal of Structural Engineering*, **118**(4), 1106-1120, 1992.

[14] H.H. Choi. Safety Assessment Using Imprecise Reliability for Corrosion ‐ Damaged Structures. Computer-Aided Civil and Infrastructure Engineering, **24**(2), 293-301, 2010.

[15] A.M.S. Freire, J.H.O. Negrao, A.V. Lopes. Geometrical nonlinearities on the static analysis of highly flexible steel cable-stayed bridges. *Computers & Structures*, **84**(31): 2128-2140, 2006.

[16] V.K. Papanikolaou. Analysis of arbitrary composite sections in biaxial bending and axial load. *Computers & Structures*, **98**, 33-54, 2012.

[17] H. Yoo, H.S. Na, D.H. Choi. Approximate method for estimation of collapse loads of steel cable-stayed bridges. *Journal of Constructional Steel Research*, **72**, 143-154, 2012.

[18] H. Li, S. Li, J. Ou, et al. Reliability assessment of cable-stayed bridges based on structural health monitoring techniques. *Structure and Infrastructure Engineering*, **8**(9), 829-845, 2012.

[19] H. Dai, H. Zhang, W. Wang, G. Xue. Structural reliability assessment by local approximation of limit state functions using adaptive Markov chain simulation and support vector regression. *Computer-Aided Civil and Infrastructure Engineering*, **27**(9), 676-686, 2012.

[20] X. Zhang, D. Liang, J. Zeng, A. Asundi. Genetic algorithm-support vector regression for high reliability SHM system based on FBG sensor network. *Optics and Lasers in Engineering*, **50**(2), 148-153, 2012.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

PRINCIPLES FOR UNCERTAINTY ASSESSMENT IN KERNEL SMOOTHING ESTIMATIONS

David Vališ[1] **and Kamila Hasilová**[2]

[1]Dept. of Combat and Special Vehicles, University of Defence
Kounicova 65, Brno, 66210 Czech Republic
e-mail: david.valis@unob.cz

[2]Dept. of Quantitative Methods, University of Defence
Kounicova 65, Brno, 66210 Czech Republic
e-mail: kamila.hasilova@unob.cz

Keywords: Kernel smoothing, uncertainty, square error, bootstrap, probability density function.

Abstract. *In this article, we present application of kernel smoothing estimation and bootstrap on real data. We possess statistically significant data set from experiments performed on composite materials. These data form a random sample of observed variable. Probability distribution function (pdf) of such observed variable is estimated using kernel smoothing approach and bootstrap. This estimation depends on a bandwidth of kernel smoother which is defined using both reference density method and our empirical data. Parameters of typical parametric distributions are also estimated from the same empirical data set. We apply methods such as mean square error (MSE) and integrated square error (ISE) to address the uncertainty and vagueness in pdf estimated by kernel smoothing.*

1 INTRODUCTION

In this paper, we present mathematical approaches in statistical comparison of various tools. Theoretical background form kernel smoothing and bootstrap. Practical comparison is performed on data which represent specific climatic tests of composite materials. The climatic test means that the composites were exposed to thermal effects. In our case, we present outcomes where freezing temperature and normal ambient temperature were applied. The composite materials used for comparison had either two, four or six layers while the layers are natural fibres such as jute and linen.

Some attempts to study composites reliability can be found e.g. in [1, 2], where authors deal with selected aspects of safety and reliability features of polymer composites with natural fibres. These kinds of composites have been starting to be used quite widely in the technical industry like automotive, aerospace, military etc. Authors present approaches for selection of those properties and features of polymer composites which are to be used for an event cause description. Selected mechanical properties and qualities which usually create the fundamental point of view in terms of decision about the applicability of polymer composites are being studied. The materials measures like strength, hardness and elasticity play vital role in terms of physical in-situ applicability. Selection of some other attempts to evaluate composite reliability and failure occurrence can be found e.g. in [3, 4, 5, 6].

2 THEORETICAL FRAMEWORK

A practical approach to deal with data uncertainties coming from inadequate information and incomplete knowledge should be robust and statistically consistent across different scales (global, local). Also it should be flexible enough to deal with the variety of data and obtain the maximum information from the sample [7]. Most parametric methods do not meet all these requirements. Therefore, we turn our attention to the nonparametric methods, namely the kernel estimation and bootstrap.

The kernel density estimation is one of the nonparametric approaches to reconstruct the underlying probability density function (pdf) from a given sample. Let a univariate random sample $X_1, \dots, X_n$ come from a distribution with a continuous probability distribution with a density $f(x)$. The kernel density estimator f_{est} is defined as a weighted average of observations at a point x

$$f_{\text{est}}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right), \tag{1}$$

where K is a univariate function called a kernel, h is a smoothing parameter ($h > 0$) called a bandwidth. Kernel K is usually taken to be a symmetric probability density function, which ensures that the estimate itself is also a pdf [8].

The problem of choosing how much to smooth, i.e. what value of the bandwidth should be used, is a crucial element in the kernel smoothing. The quality of the smoothing, i.e. the closeness of the estimate to the true density, can be measured locally or globally [9]. A useful local criterion is the mean square error (MSE) defined by

$$\text{MSE}(f_{\text{est}}(x,h)) = E[f_{\text{est}}(x,h) - f(x)]^2. \tag{2}$$

As a global criterion, we consider the integrated square error (ISE), which is given by the formula

$$\text{ISE}(f_{\text{est}}(\cdot,h)) = \int [f_{\text{est}}(x,h) - f(x)]^2 \,\mathrm{d}x. \tag{3}$$

The bootstrap is a computer-based method for assigning measures of accuracy to statistical estimates – either parameters [10] or functions [11]. Bootstrap is viewed as a general tool for confidence intervals and assessment of uncertainty. Like other nonparametric approaches, bootstrap does not presume any specifications about the distribution of the sample. The only major assumption behind the bootstrap is that the sample distribution is a good approximation of the population distribution.

3 PRINCIPLES OF MATHEMATICAL EXPERIMENT

Into the mathematical experiment, we included parametric models to assess the performance of the nonparametric estimate of the density. The parametric models for the simulated data are as follows:

- Weibull distribution with the scale parameter $b = 3$ and the shape parameter $k = 6$, i.e. $W(3, 6)$. The Weibull distribution is used in reliability and lifetime modeling, and to model the breaking strength of materials.
- Inverse Gaussian distribution with the scale parameter $\mu = 1$ and the shape parameter $\lambda = 5$, i.e. $IG(1, 5)$. The inverse Gaussian is used to model nonnegative positively skewed data.
- Normal (Gaussian) distribution with the mean value equal to five and the variance equal to one, i.e. $N(5, 1)$. Although normal distribution can take negative values, we decided to include it into the experiment as well.

For the real data, we assumed that all models, i.e. nonparametric, Weibull, inverse Gaussian and normal are possible. Parameters of the parametric models were calculated using Matlab function fitdist, which uses maximum likelihood estimates of the parameters. For the kernel estimate, we employed the standard Gaussian kernel and the bandwidth calculated according to the so called normal reference rule [12].

The size of the simulated sample was inspired by the real data – we started with sample of size 30. To show the performance of the used methods, we also included into the study the samples of sizes 300 and 3000.

Bootstrap procedure was carried out according to the following steps [13, 14, 15]:

1. From the sample $X = \{X_1, \ldots, X_n\}$ calculate its kernel density estimate f_{est}.
2. Draw the resample $X^* = \{X_1^*, \ldots, X_n^*\}$ from the sample X (with replacement).
3. Calculate the kernel density estimate f^*_{est} of the resample X^*.
4. Repeat the steps 2 and 3 $n_B = 400$ times.
5. Find 2.5% and 97.5% quantiles of the estimated densities f^*_{est} and construct a pointwise confidence band.

For the simulated data, the errors MSE and ISE were calculated using the repeatedly chosen random samples from the simulated distributions. Number of repetition was set to be $n_{rep} = 100$.

4 RESULTS AND DISCUSSION

Firstly, we tested the proposed procedure on the real data set. Then, we supported the rightness of this idea by a simulation study.

4.1 Real data

For the graphical example of the real data, we randomly selected probes denoted as J4F. It means that the set of probes is composed of four jute layers and was exposed to the freezing effects. The data set and its density estimates – both nonparametric and parametric – are displayed in Figure 1.

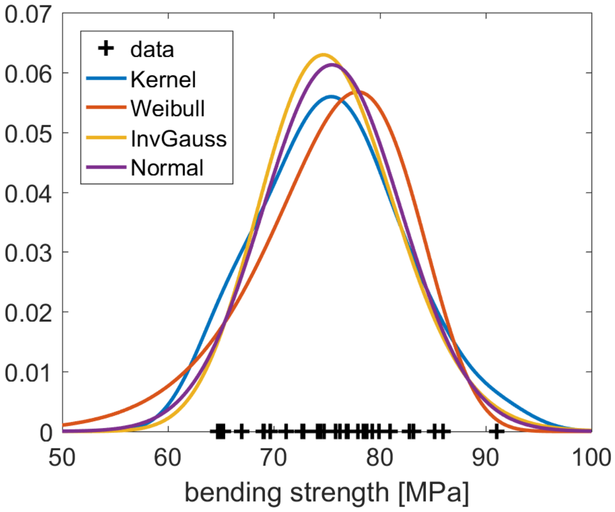


Figure 1: Real data (+) with estimated densities – kernel estimate (blue), Weibull (orange), inverse Gaussian (yellow) and normal (purple).

In Figure 2, there are shown all considered models applied to the set of real data. Each panel consists the estimate itself, either parametric or nonparametric, and the pointwise confidence band calculated using bootstrap.

The standard Kolmogorov-Smirnov test did not reject the hypothesis that the data come from the respective distributions. Therefore, one can choose any model we propose. However, using the ISE, we can decide which one of the parametric models is the most likely original density. The density which is believed to be the root form shall indicate the lowest values of ISE when compared to kernel estimate, see Tables 1 and 2.

Data	ISE		
Jute	Weibull	InvGauss	Normal
2 layers	0.002165	0.001460	0.001714
4 layers	0.001389	0.000693	0.000484
6 layers	0.003183	0.000913	0.001202

Table 1: Values of ISE when comparing parametric models to the kernel estimate for the set of probes with jute fibres (with indicated number of layers) and freezing ambient temperature.

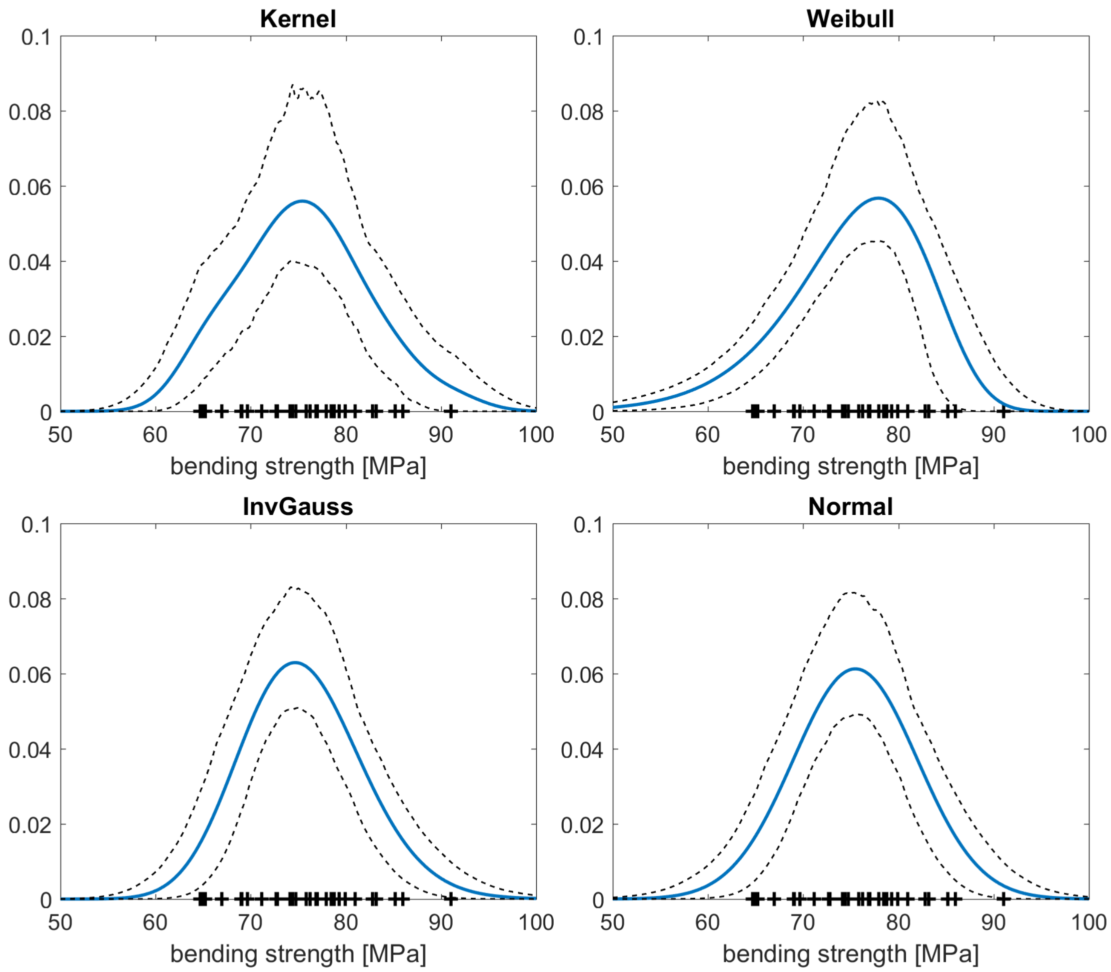


Figure 2: Real data (+) with estimated densities and confidence bands – kernel estimate (left upper panel), Weibull (right upper panel), inverse Gaussian (left lower panel) and normal (right lower panel).

As we can see from the Tables 1 and 2, the inverse Gaussian model would be the best from parametric models regarding the probes with jute fibres. On the other hand, for probes with linen fibres, the Weibull model seems to be the best parametric option. However, as we can conclude from Figure 1, the decision is up to the user, since the models are close one to another.

4.2 Simulated data

As a typical instance of the data describing the bending strength, we select a data set coming from the Weibull distribution and show graphical results of the study on this data set. Other considered models are summarized in a text form later (see Table 3).

From the graphs in Figure 3, we can see that with the growing size of the sample, the uncertainty of the functional shape of the density is smaller. Also the confidence bands are thinner; however, we have to keep in mind that the confidence bands are constructed around the estimated density f_{est}. They may not completely cover the true density function f as we can see on the middle graph in Figure 3.

In Figure 4, there are summarized mean square errors of all considered models. As we can expect, the Weibull model has the smallest error, because the data came from the Weibull distribution. The kernel estimate, as we can see, is a very good approximation of the data.

Data	ISE		
Linen	Weibull	InvGauss	Normal
2 layers	0.003191	0.001354	0.002057
4 layers	0.001485	0.000579	0.000466
6 layers	0.000810	0.003569	0.002032

Table 2: Values of ISE when comparing parametric models to the kernel estimate for the set of probes with linen fibres (with indicated number of layers) and normal ambient temperature.

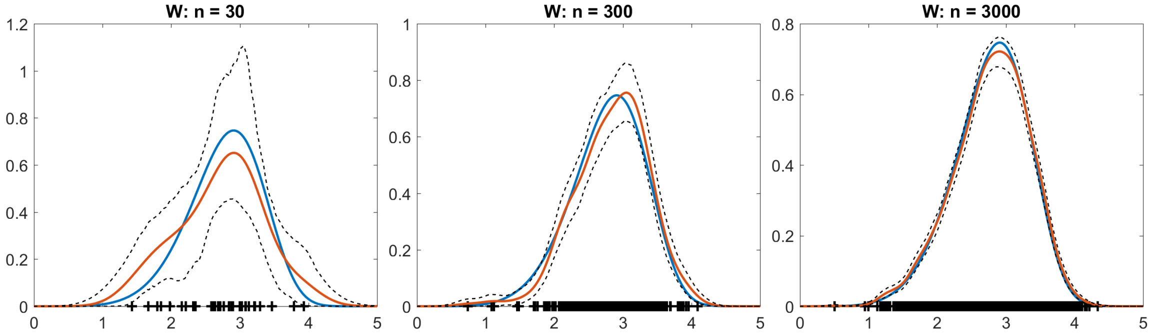


Figure 3: Data (+) coming from the Weibull distribution with density f (blue) and kernel estimate f_{est} (orange) accompanied by the bootstrap confidence interval (black).

The inverse Gaussian and normal models give the biggest error, which support the idea that assuming the wrong parametric shape of the distribution behind the data can lead to results burdened with uncertainty.

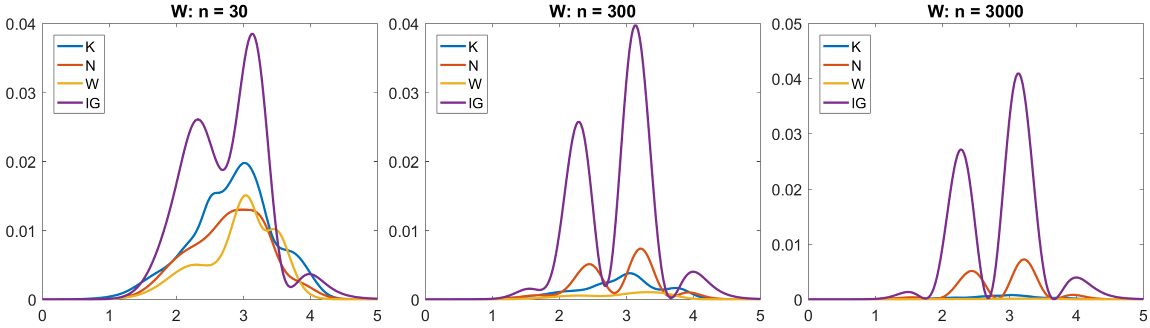


Figure 4: Mean square error (MSE) for the kernel estimate (K, blue), parametric estimates – normal (N, orange), Weibull (W, yellow) and inverse Gaussian (IG, purple) based on the simulated data coming from the Weibull distribution.

Similarly, in Figure 5, we can see that with growing size of the sample the integrated square error is getting smaller. Also, we can see that the kernel density estimator is a good choice to determine the shape of the data distribution while not assuming its shape beforehand.

Results of the rest of the simulation study are in Table 3, where the values of ISE, namely the medians are summarized. Again, we can conclude that nonparametric approach is an excellent complementary method to the parametric ones.

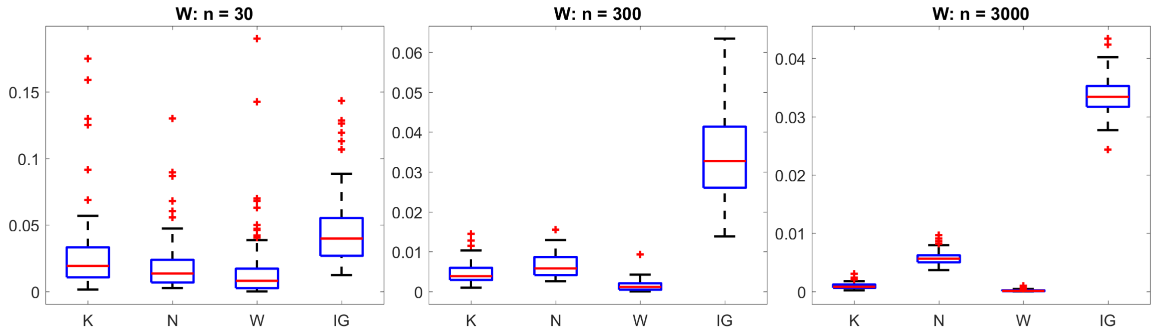


Figure 5: Integrated square error (ISE) of the four used models – kernel estimate (K), normal (N), Weibull (W) and inverse Gaussian (IG) – based on the simulated data coming from the Weibull distribution.

		Model			
Data	Size	Kernel	Weibull	InvGauss	Normal
	30	0.0193	0.0081	0.0398	0.0136
$W(3,6)$	300	0.0039	0.0012	0.0327	0.0058
	3000	0.0008	0.0001	0.0334	0.0056
	30	0.0485	0.0579	0.0199	0.0797
$IG(1,5)$	300	0.0095	0.0485	0.0014	0.0722
	3000	0.0021	0.0480	0.0002	0.0722
	30	0.0119	0.0083	0.0113	0.0060
$N(5,1)$	300	0.0021	0.0026	0.0068	0.0006
	3000	0.0004	0.0027	0.0066	0.0001

Table 3: Medians of ISE values of the respective simulated data sets and model used.

5 CONCLUSIONS

In this paper, we studied uncertainty evaluation which can be related to various mathematical approaches. Nonparametric kernel smoothing and bootstrapping approaches have been studied and later compared. Data from specific climatic tests of composite materials reinforced by natural fibres were chosen as an application example.

REFERENCES

[1] A. Krzyżak, D. Vališ, Selected reliability measures of composites with natural fibres tested in climatic environment. *INTERNATIONAL CONFERENCE ON MILITARY TECHNOLOGIES (ICMT 2015)*, 81–87, 2015.

[2] A. Krzyżak, D. Vališ, Selected safety aspects of polymer composites with natural fibres. *Safety and Reliability: Methodology and Applications*, 903–909, 2015.

[3] X. Li, Z. Lv, Z.P., Qiu, A novel univariate method for mixed reliability evaluation of composite laminate with random and interval parameters. *Composite Structures*, **203**, 153–163, 2018.

[4] S.L. Omairey, P.D. Dunning, S. Sriramula, Influence of micro-scale uncertainties on the reliability of fibre-matrix composites. *Composite Structures*, **203**, 204–216, 2018.

[5] V.P. Berardi, L. Feo, G. Mancusi, M. De Piano, Influence of reinforcement viscous properties on reliability of existing structures strengthened with externally bonded composites. *Composite Structures*, **200**, 532–539, 2018.

[6] R. Prikryl, P. Otrisal, V. Obsel, L. Svorc, R. Karkalic, J. Buk, Protective properties of a microstructure composed of barrier nanostructured organics and SiOx layers deposited on a polymer matrix. *Nanomaterials*, **8**, 679, 2018.

[7] T.A. Solaiman, S.P. Simonovic, D.H. Burn, Quantifying uncertainties in the modelled estimates of extreme precipitation events at upper Thames river basin. *British Journal of Environment and Climate Change* **2**, 180–215, 2012.

[8] M.P. Wand, M.C. Jones, *Kernel Smoothing*. Chapman and Hall, 1995.

[9] I. Horová, J. Koláček, J. Zelinka, *Kernel Smoothing in Matlab. Theory and Practice of Kernel Smoothing*. World Scientific, 2012.

[10] B. Efron, R. Tibshirani, *An introduction to the bootstrap*. Chapman and Hall, 1993.

[11] A. Cuevas, R. Fraiman, On the bootstrap methodology for functional data. *COMPSTAT 2004 – Proceedings in Computational Statistics*, 127–135, 2013.

[12] J.E. Chacón, T. Duong, M.P. Wand, Asymptotics for general multivariate kernel density derivative estimators. *Statistica Sinica*, **21**, 807–840, 2011.

[13] B.W. Silverman, *Density estimation for statistics and data analysis*. Chapman and Hall, 1986.

[14] R. Davidson, J. MacKinnon, Bootstrap tests: How many bootstraps? *Queens Economics Department Working Paper*, no. 1036, 2001.

[15] P. Hall, J. Horowitz, A simple bootstrap method for constructing non-parametric confidence bands for functions. *The Annals of Statistics*, **41**, 1892–1921, 2013.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
International Conference on Uncertainty Quantification in
Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

UNCERTAINTY QUANTIFICATION ON AERODYANAMIC CHARACTERISTICS OF FLOW AROUND SQUARE AND CORNER-ROUNDED CYLINDER WITH GLANCING ANGLE

Tsubasa Hamada[1], Tetsuro Tamura[2]

[1] Tokyo Institute of Technology, 4259 Nagatsuda, Midori-ku, Yokohama, Japan
e-mail: hamada.t.aj@m.titech .ac.jp

[2] Tokyo Institute of Technology, 4259 Nagatsuda, Midori-ku, Yokohama, Japan
e-mail: tamura.t.ab@ m.titech .ac.jp

Abstract

Flow around bluff body largely depends on many uncertainties: inflow condition, boundary condition and numerical method. Particularly, in certain angle of inflow such as 14°, pressure distribution on cylinder surface dramatically changes due to change of flow, and thus understanding stochastic behavior around certain angle causing drastic changes of pressure plays an essential role in designing civil structure. In the present paper, therefore, Uncertainty Quantification (UQ) based on Non-Intrusive Polynomial Chaos (NIPC) has been carried out for flow around square cylinder and corner-rounded cylinder by Large Eddy Simulation (LES) by assuming that angle of attack and Reynolds number are uncertain parameters on both cylinders. These uncertainties follow uniform distribution, and angle of attack α [12°-14°] and Reynolds number [1000-10000] are uncertain range on square cylinder, and α [5°-7°] and Reynolds number [1000-10000] on corner-rounded cylinder with curvature r/D=2/15. Time-statistics of aerodynamic forces on the surface are evaluated in this study. Moreover, Sensitivity analysis based on Sobol index is also performed to evaluate impact of uncertain parameters on uncertain output. As a result, both uncertainties considered in this study have a significant impact on dispersion of aerodynamic forces. On the windward side, angle of attack results in large impact on time-averaged pressure coefficient, but in the case of corner-rounded cylinder, Reynolds number has more impact than angle of attack. On the side surface, Reynolds number has large impact on both cylinders.

Keywords: Uncertainty quantification, Square cylinder, Corner-rounded cylinder, Non-intrusive polynomial chaos, Large eddy simulation

1 INTRODUCTION

Bluff body such as a square cylinder and corner-rounded cylinder is typical shape of civil structure. Even though those geometries are relatively simple, flow around cylinder is complex. Flow around cylinder is separated at corner of cylinder, and it cause recirculation in wake and reattachment on side. Figure 1 illustrates Computational Fluid Dynamics (CFD) results of flow pattern visualized by vorticity contours around square cylinder at different conditions. As a result of these flow phenomena, local severe suction on the surface can be appeared, and it cause strong wind forces to the structure. Therefore, it is important for civil structure to clarify the relationship between flow around cylinder and aerodynamic forces.

So far, many numerical simulations and wind tunnel tests have been carried out to flow around bluff body, and those make a substantial contribute to develop wind resistant design and to understand physical meaning of flow structure. Especially, over the last several decades, CFD has become powerful tool to investigate flow because it is superior in terms of visualizing whole and detail flows. However, in numerical simulation, there are many uncertainty and errors due to numerical scheme, turbulence modeling, limitation of grid number and geometry of structure, and so on. In this context, considering those uncertainty and errors in CFD is essential to assess the reliability and validation of CFD as a next step.

A. Mariotti et al. [1] studied impacts of uncertain parameters: grid resolution in spanwise direction and the weight of the explicit filter on aerodynamic characteristics of 5:1 cylinder surface by LES. It was shown that flow characteristics on side surface are largely impacted by the spanwise grid resolution and that the grid resolution of other direction and the amount of SGS dispersion also play an important role in catching dynamic small vortex. Moreover, A. Mariotti et al. [2] studied effects of inflow uncertainties on aerodynamic characteristics of 5:1 cylinder surface by URANS. The angle of attack, the longitudinal turbulence intensity and the turbulence length scale were assumed to be uncertain parameters. As a result of UQ study, the dispersion of average of time-average pressure coefficients is highly lower than that of BARC contributions, and it means the longitudinal turbulence intensity and the turbulence length scale do not affect in dispersion of pressure on cylinder.

Above UQ studies in the literature mostly have been studied by assuming uncertain range that does not result in drastic change on flow, but from the engineering view point, it is important for design to clarify how much flow and pressure on the surface change, such as that rapid change of flow pattern and of surface pressure occur at angle of attack 13°~15° [3].

To the author’s knowledge, there is no UQ analysis that subjects to the drastic range causing drastic changes on bluff body. Assuming drastic range of uncertainty, it should be kept in mind that convergent ratio is decreased, so assuming appropriate uncertain range is essential to obtain exact result. In the present study, thereafter, stochastic aerodynamic forces on two-dimensional (2D) square cylinder and on 2D corner-rounded square cylinder are evaluated by NIPC. Angle of attack and Reynolds number are considered as uncertainty parameters with range of uniform distribution. Here, it is focused on time-statistics of aerodynamic forces.

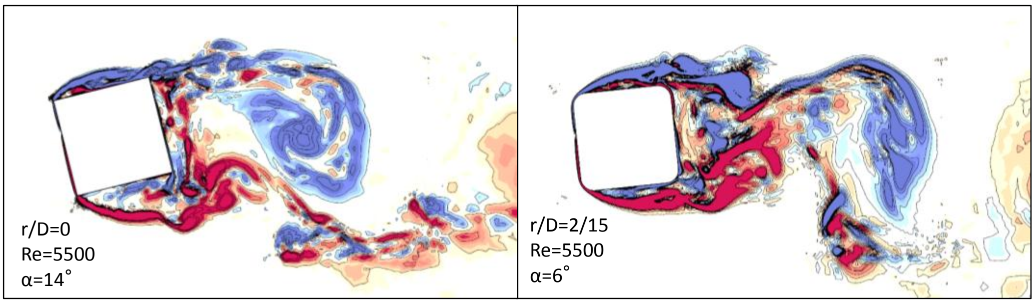


Figure 1. CFD of the flow pattern around square and rounded cylinder at a critical angle of attach

2 COMPUTATIONAL SET-UP

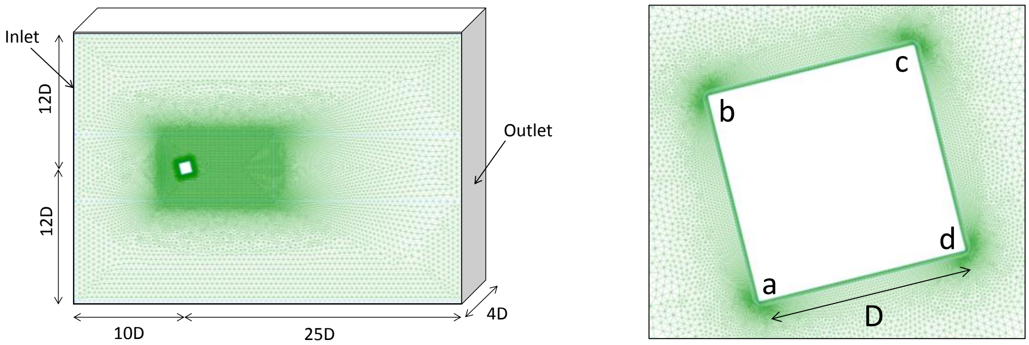
2.1 Computational model and numerical validation

Simulation is carried out for incompressible Navier-Stokes and continuity equation for flow around 2D square cylinder and 2D corner-rounded cylinder. Solver is finite-volume CFD code FrontFlow/Red HPC, and unstructured grid is employed.

For calculation condition of LES, the total cell number is 6.5 million. The domain is shown in Fig 2. The height of the cell nearest to cylinder surface is determined by $0.1D/\sqrt{Re} = 1.3 \times 10^{-3}$ according to Tamura and Ono [4]. Grid resolution of surface of cylinder is B/100 approximately. The inlet condition is uniform flow inflow, and no-slip condition is used on cylinder surface. The spanwise end boundary condition is periodic. Top and bottom surface condition are free-slip.

Spatial discretization is generally treated by the second order central difference. For discretization of convective term, 5% first-order upwind scheme is blended in order to eliminate the numerical oscillation in the region around cylinder. SMAC algorithm is used for the pressure-velocity coupling. The implicit Euler method is applied for the time integration. The turbulence model is dynamic Smagorinsky model [5], [6] and the ratio of test-filter scale and grid-filter scale is set to 2. The computational domain size and grid system are shown in Fig 2. Time increment is $dtU_\infty/D = 1.0 \times 1.0^{-4} {\sim} 2.6 \times 1.0^{-4}$. Simulation-time for statistical analysis is at least 10 vortex-shedding cycles after flow becomes statistically stationary.

Before conducting UQ study, validation study is performed to investigate the grid resolution is sufficient for drastic region such as angle of attack 14° by comparison with wind experiment date [7], which is in range of uncertain parameters considered in this paper. Three LES simulations are carried out with different Reynolds number 5300, 8985, 20000. Figure 3 shows average pressure coefficient. Difference between LES and EXP might be caused by Reynolds effect. Thus, 6.5 million cell is sufficient to predict pressure correctly.



(a) Whole region

(b) Local region

Figure 2. Grid of computational domain

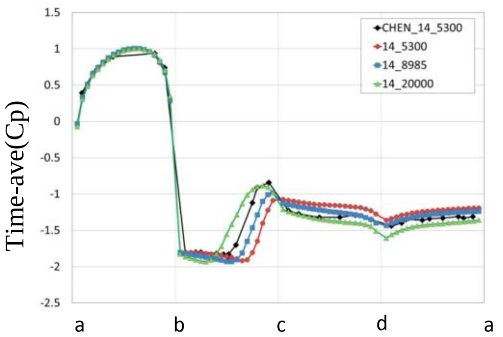
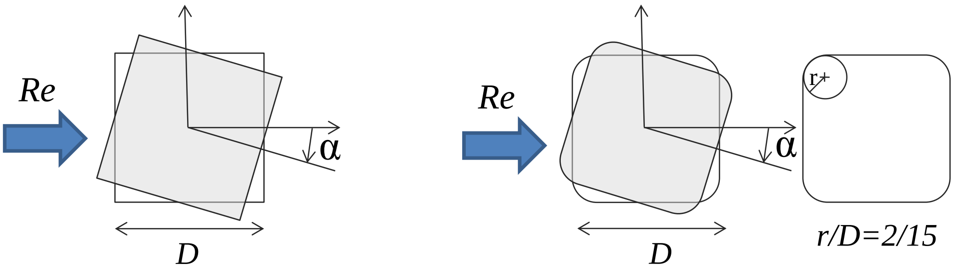


Figure 3. Average pressure coefficient

3 UNCERTAINTY QUANTIFICATION METHODOLOGY

3.1 Uncertainty parameter

In the present study, three uncertain variables for UQ are assumed: angle of attack α, Reynolds number (Fig 4). These uncertain parameters largely impact flow pattern and pressure distribution, so the range of uncertain parameters should be carefully chosen to obtain fast convergent. There are two kind of uncertain ranges for both cylinders: in the case of square cylinder, uncertain ranges are angle of attack α [*12°*, *14°*] and Reynolds number *Re*[*1000, 10000*], and in the latter case, angle of attack α [*5°*, *7°*], Reynolds number *Re* [*1000, 10000*]. The reason to choice the range of angle of attack α is based on experimental study [3]. According to [3], there is a discontinuity of aerodynamic forces by changing angle of attack α. In the case of square cylinder, discontinuity exists at 13°, *and with the rounded corner r/D=2/15* discontinuity *exists at 6°*. Therefore, in the present study, to investigate stochastic behavior in range of drastic region, above uncertain range is used. By considering range of $\pm 1°$, it represents difficulty that it is difficult to reproduce perfect angle of attack in wind tunnel test, and this slight different angle of attack largely impacts on aerodynamic forces, e.g. Lift coefficient. For the range of Reynolds number, to the author's knowledge, there is few study considering Reynolds number as uncertain parameters. According to [8], recommended range of Reynolds number [$20{,}000 \leq Re \leq 60{,}000$] does not have large effect on dispersion of BARC contribution. In this study, therefore, sensitive region of Reynolds number is purposely chosen to obtain large impacts as well as angle of attack $\pm 1°$ on aerodynamic forces. Other parameters are fixed. Since there is no prior knowledge about probabilistic distribution of above uncertain parameters, uniform distribution is considered to assume quite large range of variation.



(a) Square cylinder (b) Corner-rounded cylinder

Figure 4. Uncertain set-up

3.2 Non-Intrusive Polynomial Chaos

Over the last decades, many UQ studies have been carried out, and one of the effective approaches to quantify uncertainty is Polynomial Chaos Expansion (PC) [9]. In the PC, stochastic output is represented by

$$\mathrm{y}(\boldsymbol{x}, \boldsymbol{\xi}) = \sum_{i=0}^{Np-1} a_i(\boldsymbol{x}) \cdot \Psi_i(\boldsymbol{\xi}) \tag{1}$$

where $a_i(\boldsymbol{x})$ is deterministic coefficients, and $\Psi_i(\boldsymbol{\xi})$ is the multidimensional orthogonal polynomials, which is determined by Wiener-Askey Scheme [9]. In practical ways, Equation 1 is truncated to N_p terms which is determined by

$$N_p = \frac{(nv + d)!}{nv!\, d!} \tag{2}$$

where *nv* is number of random variables (uncertain parameters) and *d* is polynomial order. PC is classified to two types: Intrusive PC (IPC) and Non-Intrusive PC (NIPC). IPC requires modification of code to conduct UQ study, so it is tough and time-consuming. On the other hand, NIPC is not required to modify code. In NIPC, simulation code is treated as black box, and thus it is easy to apply PC to any numerical study. In this study, therefore, NIPC is used to UQ study.

The important things in NIPC is to calculate deterministic coefficients $a_i(\boldsymbol{x})$. To obtain $a_i(\boldsymbol{x})$ in NIPC, it is required to do N_p deterministic simulations. In this study, Regression method is used to obtain $a_i(\boldsymbol{x})$ as follows:

$$\begin{bmatrix} \Psi_0(\xi_0) & \Psi_1(\xi_0) & \dots & \Psi_{N_p}(\xi_0) \\ \Psi_0(\xi_1) & \Psi_1(\xi_1) & \dots & \Psi_{N_p}(\xi_1) \\ \vdots & \vdots & \dots & \vdots \\ \Psi_0(\xi_N) & \Psi_1(\xi_N) & \cdots & \Psi_{N_p}(\xi_N) \end{bmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{pmatrix} = \begin{pmatrix} y(\xi_0) \\ y(\xi_1) \\ \vdots \\ y(\xi_N) \end{pmatrix} \tag{3}$$

After calculating $a_i(\boldsymbol{x})$, mean and variance are easily evaluated by

$$\mu = a_0 \tag{4}$$

$$\sigma^2 = \sum_{i=1}^{Np-1} (a_i^2 \Psi_i^2) \tag{5}$$

In the present paper, random variable *nv* is two and polynomial order *d* is two, so the number of polynomial terms is six. These uncertain parameters are assumed to follow uniform distribution, and Legendre polynomial is used as multidimensional orthogonal polynomials. Since order of polynomial *d* is two, the number of deterministic simulation is determined by zeros of third order-Legendre polynomials. Theoretically, to solve Equation 3 at least N_p deterministic simulations are sufficient. In this study, drastic region is chosen as uncertain range and parameters, so there is a possibility that some numerical error occur. To avoid influence of these error, Hosder et al. [10] suggest that many more points than minimum number of deterministic simulation to solve Equation 3 can make surrogate model robust, and about more than $N/N_p=2$ is desirable. Table 1 and Table 2 show sampling points for UQ and aerodynamic forces obtained by deterministic simulations.

3.3 PC based Sobol indies

It is easy to calculate Sobol index by using PC decomposition [11]. Direct and combined Sobol indices mean how much each input parameters contribute to the total uncertainty. As mentioned above, total variance is evaluated and decomposed by Equation 6:

$$D = \sum_{i=1} D_i + \sum_{i<j} D_{ij} + \sum_{i<j<l} D_{ijl} + \cdots + D_{1,2\cdots k} \tag{6}$$

Where

$$D_{i_1,i_2\cdots i_s} = \int f_{i_1,i_2\cdots i_s}^2 (x_{i_i}, \cdots, x_{i_s}) dx_{i_1} \cdots dx_{i_s} \tag{7}$$

corresponds partial variance. Then, the Sobol indices are defined as:

$$S_{i_1,i_2\cdots i_s} = \frac{D_{i_1,i_2\cdots i_s}}{D} \tag{8}$$

These Sobol indices allow to measure sensitivity of the contribution of single uncertain input (S_i) and interactive contribution ($S_{i,j}, S_{i,j,k}, \cdots$).

4 UQ RESULTS AND DISCUSSIO

Table 3 and Table 4 show mean and standard deviation of time-average and time-rms of Drag coefficient: *t-ave(C_D)* and *t-std(C_D)* and of Lift coefficient: *t-ave(C_L)* and *t-std(C_L).* As can be seen, with rounded corner, mean of *t-ave(C_D)* and *t-ave(C_L)* become small, and standard deviation become larger than square cylinder. For standard deviation of *t-ave(C_L)*, square cylinder has large dispersions. This simply attributes to change of angle of attack, while on corner-rounded cylinder the influence of change of angle of attack can be reduced. To evaluate these stochastic values, it is necessary to compare with other numerical simulation and wind tunnel study, and this should be done as a future work.

Figure 5 and Figure 6 show mean and standard deviation of distribution of time-average pressure coefficient and of time-rms pressure coefficient respectively. As can be seen from Figure 5 (a), there is dispersion at side and leeward side on square cylinder. Especially at CD side, dispersion of mean is larger than other side. At this side separation

Table 1 Sampling points of square cylinder based on Legendre polyno-

case	α	*Re*	*t-ave(C_D)*	*t-std(C_D)*	*t-ave(C_L)*	*t-std(C_L)*
1	12.225	2014.315	1.625	0.151	-0.610	0.442
2	12.225	5500.000	1.702	0.150	-0.805	0.561
3	12.225	8985.685	1.626	0.130	-0.975	0.454
4	13.000	2014.315	1.696	0.198	-0.722	0.418
5	13.000	5500.000	1.638	0.138	-0.826	0.421
6	13.000	8985.685	1.636	0.155	-0.920	0.487
7	13.775	2014.315	1.694	0.293	-0.772	0.469
8	13.775	5500.000	1.666	0.181	-0.923	0.547
9	13.775	8985.685	1.812	0.233	-0.774	0.658

Table 2 Sampling points of corner-rounded cylinder based on Legendre polyno-

case	α	*Re*	*t-ave(C_D)*	*t-std(C_D)*	*t-ave(C_L)*	*t-std(C_L)*
1	5.225	2014.315	1.554	0.298	-0.206	0.816
2	5.225	5500.000	1.504	0.159	-0.307	0.636
3	5.225	8985.685	1.459	0.144	-0.392	0.633
4	6.000	2014.315	1.564	0.366	-0.418	0.816
5	6.000	5500.000	1.462	0.165	-0.381	0.608
6	6.000	8985.685	1.407	0.122	-0.417	0.487
7	6.774	2014.315	1.462	0.291	-0.329	0.651
8	6.774	5500.000	1.466	0.176	-0.345	0.579
9	6.774	8985.685	1.414	0.144	-0.445	0.549

Table 3 Uncertain mean of forces

	t-ave(C_D)	*t-std(C_D)*	*t-ave(C_L)*	*t-std(C_L)*
Square	1.672	0.174	-0.822	0.489
Corner-round	1.477	0.202	-0.365	0.635

Table 4 Uncertain standard deviation of forces

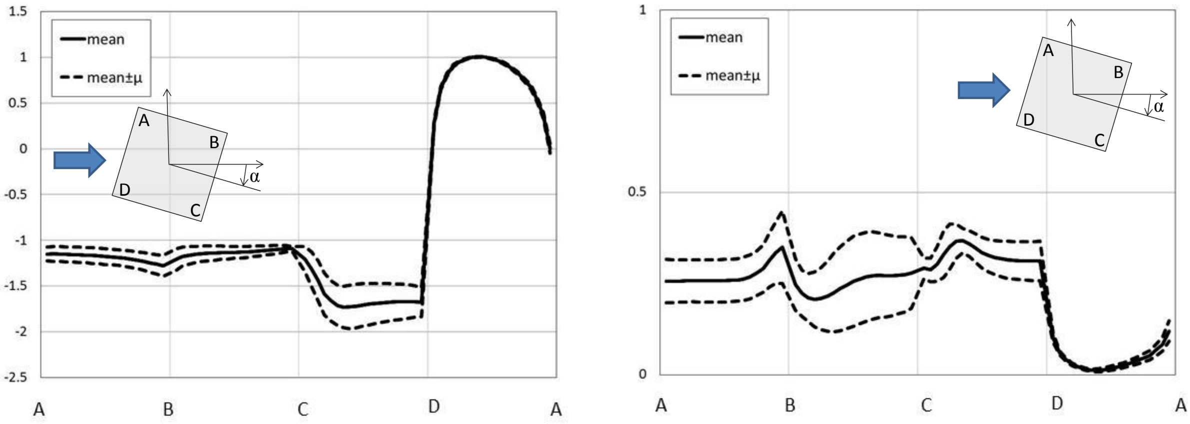
	t-ave(C_D)	*t-std(C_D)*	*t-ave(C_L)*	*t-std(C_L)*
Square	0.003	0.004	0.022	0.009
Corner-round	0.004	0.011	0.007	0.016

bubble, which is separated at corer of D and reattach on around corner C, occurs. This phenomenon is extremely sensitive to change of angle of attack and Reynolds number, so large dispersion is appeared on CD side. However, for rms pressure coefficient from Figure 5 (b), dispersion is appeared on all side except wind ward side. Figure 6 shows mean and standard deviation of time-average pressure coefficient and time-rms pressure coefficient distribution on corner-rounded cylinder. For time-average pressure coefficient, large dispersion is not appeared, while for rms pressure coefficient, significant dispersion is appeared.

Figure 7 shows Sobol index of time-average pressure coefficient. As can be seen, Reynolds number plays an important role in the dispersion on CD side where reattachment occurs of square cylinder and of corner-rounded cylinder. On the other hand, on the AB and BC side, angle of attack is dominant factor on square cylinder, but on corner-rounded cylinder Reynolds number has large impact on those sides.

For Sobol index of time-rms pressure coefficient, on AB side and BC side, angle of attack on square cylinder and Reynolds number on corner-rounded cylinder are dominant factor for uncertain output, but on CD side, dominant Sobol index is largely fluctuated. This means dispersion of CD side is sensitive to uncertainties considered in this study and its combination.

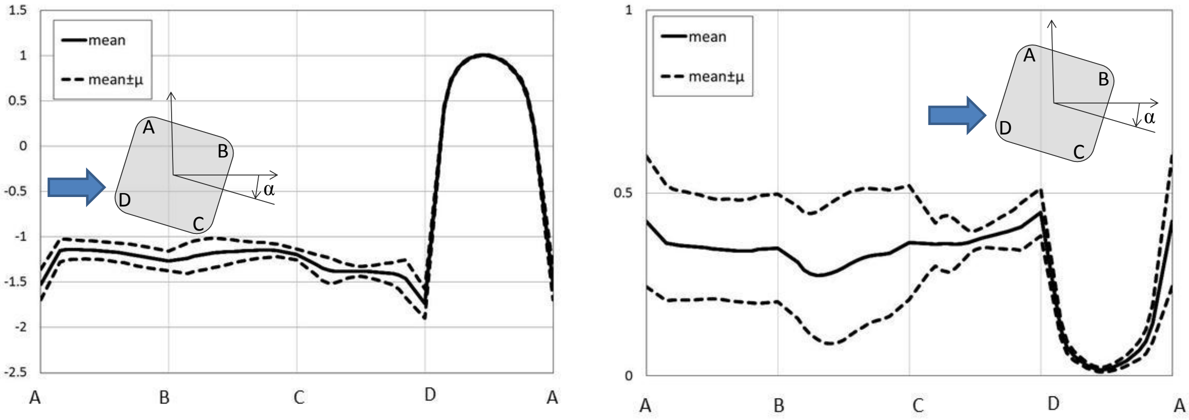
At first glance, on AD side, fluctuation of Sobol index can be seen. However, these values do not have physical meaning because on AD side, stochastic dispersion cannot be seen in Figure 5 and Figure 6.



(a) Time-averaged pressure coefficient

(b) Time-rms of pressure coefficient

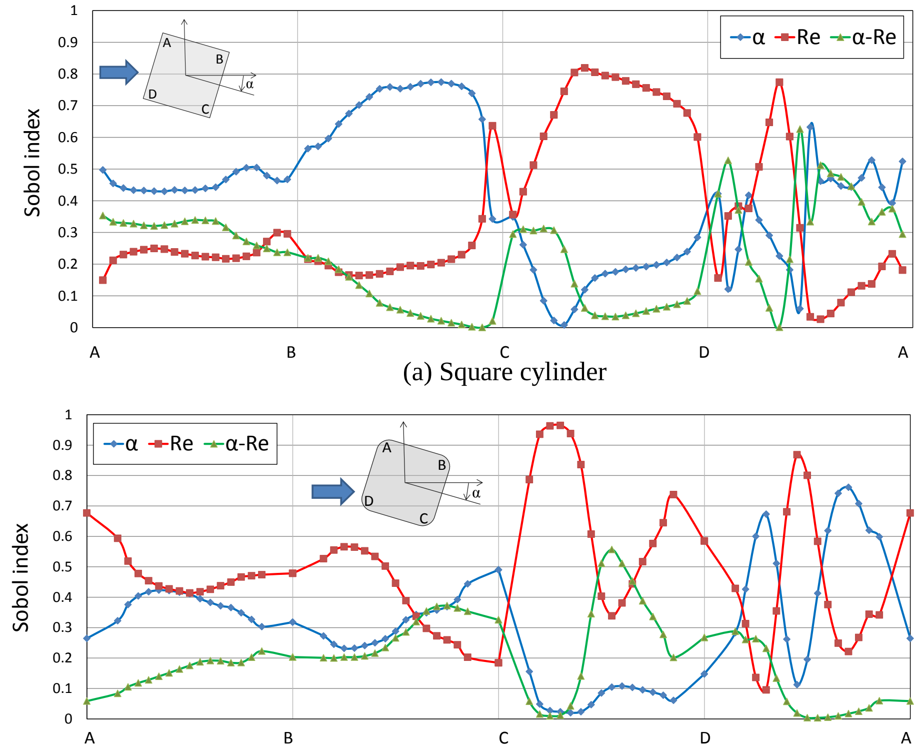
Figure 5. Stochastic mean ± standard deviation of pressure coefficient of square cylinder



(a) Time-averaged pressure coefficient

(b) Time-rms of pressure coefficient

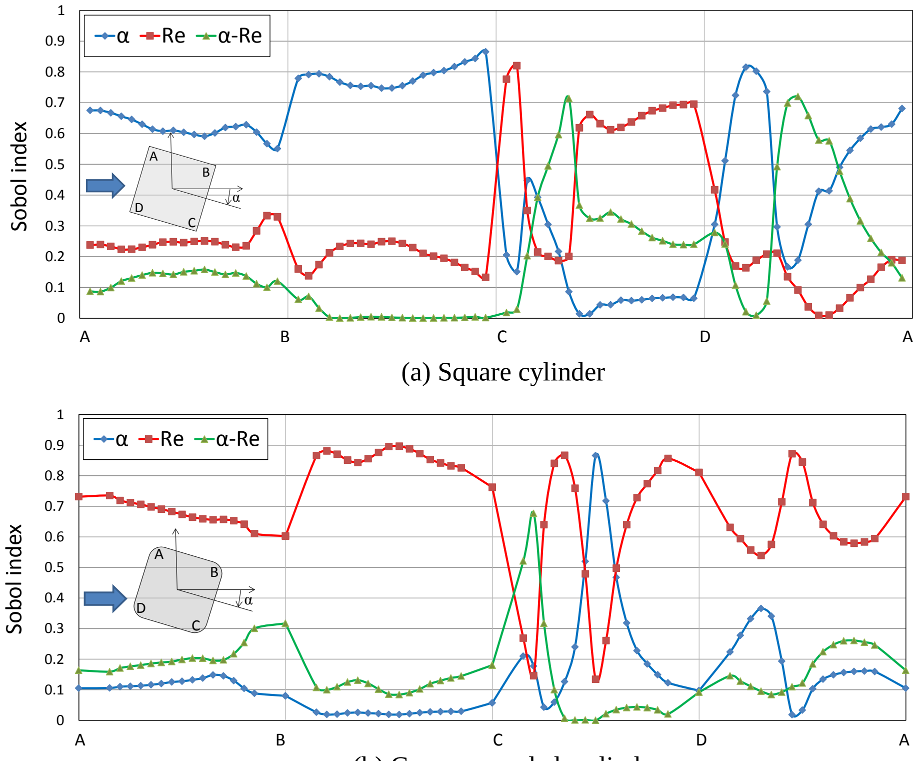
Figure 6. Stochastic mean ± standard deviation of pressure coefficient of corner-rounded cylinder



(a) Square cylinder

(b) Corner-rounded cylinder

Figure 7. Sobol decomposition of time-averaged pressure coefficient in time



(a) Square cylinder

(b) Corner-rounded cylinder

Figure8. Sobol decomposition of rms pressure coefficient in time

5 CONCLUTION AND FUTURE WORK

In this study, quantifying uncertain angle of attack and Reynolds number on aerodynamic forces of bluff body has been carried out by LES and NIPC, and it can be summarized as follows:

1. On the region where flow is separated at corner and reattach on side surface, there are large dispersion of mean of averaged pressure coefficient on square cylinder, while in the case of corner-rounded cylinder, dispersion of mean of average pressure coefficient is smaller than that of square cylinder.
2. For rms pressure coefficient, corner-rounded cylinder has larger dispersion than that of square cylinder.
3. On the square cylinder, Reynolds number has significant impact on dispersion of average pressure coefficient of side surface where reattachment occurs, while angle of attack affects largely at dispersion of average pressure coefficient on another side and windward surface.
4. On the corner-rounded cylinder, Reynolds number mostly has dominant influence on time average and rms pressure distribution on all surfaces.

6 ACKNOWLEDGEMENT

This research used computational resources of K computer provided by RIKEN Advanced Institute for Computational Science through HPCI System Research project (Project ID: hp180056). Also, we are grateful to Dr. Yang Cao for helpful discussions and supports.

REFERENCES

[1] A. Mariotti, L. Siconolfi, M. V. Salvetti, Stochastic sensitivity analysis of large-eddy simulation prediction of the flow around a 5:1 rectangular cylinder, European Journal of Mechanics B/Fluids 62 (2017), 149-165.

[2] A. Mariotti, M.V. Salvetti, P. Shoeibi Omrani, J.A.S. Witterveen, Stochastic analysis of the impact of freestream conditions on the aerodynamics of a rectangular 5:1 cylinder, Computers and Fluids 136 (2016), 170-192.

[3] Luigi Carassale, Andre Freda, Michela Marre-Brunenghi, Experimental investigation on the aerodynamic behavior of square cylinders with corner-rounded, Journal of Fluids and Structures 44 (2014), 195-204.

[4] Tamura T, Ono Y, LES analysis on aeroelastic instability of prisms in turbulence flow, J Wind Eng Ind Aerodyn (2003), 91 (12) 1827-46.

[5] Germano, Massimo, et al. A dynamic subgrid scale eddy viscosity model, Physics of Fluids A: Fluid Dynamics 3.7 (1991): 1760-1765.

[6] Lilly, Douglas K, A proposed modification of the Germano subgrid scale closure method, Physics of Fluids A: Fluid Dynamics 4.3 (1991): 633-635.

[7] Jerry. M. Chen-Hung Liu, Vortex shedding and surface pressure on a square cylinder at incidence to a uniform air stream, International Journal of heat and Fluid Flow (1999), 20(6): 592-597.

[8] Bruno L, Salvetti MV, Ricciarelli F, Benchmark on the Aerodynamics of a Rectanfular 5:1 Cylinder: an over view after the first four years of activity, J Wind Eng Ind Aerod 2014; 126;87-106.

[9] D.B. Xiu, Numerical Methods for Stochastic Computations: a Spectral Method Approach, Princeton University press. 2010.

[10] S. Hosder, R. Walters, M. Balch, Efficient sampling for non-intrusive polynomial chaos application with multiple input uncertain variables, 48th AIAA/ASME/ASCE/AHS/ASC, Structure Dynamics and Materials Conference, Honolulu, Hawaii, 2007, p.1939.

[11] Bruno Sudret, Global sensitivity analysis using polynomial chaos expansion, Reliability Engineering, System Safety, July 2008.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

MODEL VALIDATION USING BAYESIAN OPTIMAL EXPERIMENTAL DESIGN IN URBAN MECHANISED TUNNELLING

Raoul Hölter[1], **Maximilian Schoen**[1], **Arash A. Lavasan**[1], **and Elham Mahmoudi**[1]

[1]Ruhr-Universität Bochum
Department of Civil and Environmental Engineering
Universitätsstraße 150
44801 Bochum, Germany
e-mail: {raoul.hoelter,maximilian.schoen,arash.alimardanilavasan,elham.mahmoudi}@rub.de

Keywords: Mechanised Tunnelling, Finite Element Modelling, Surrogate Modelling,Optimal Experimental Design, Bayesian Updating.

Abstract. *Tunnel construction in urban areas neccessitates not only a safe construction of the tunnel itself, but also minimising the impacts on the environing infrastructure. Constructing a tunnel always causes changes in the state of stress and deformations in the surrounding soil area that can induce settlements, cracks, or even collapse in buildings at the ground surface. To plan effective countermeasures, an adequate model is necessary that considers all relevant details of the system. However, the main component of this system is the adjacent soil that cannot be modified and which properties are highly uncertain. To reduce the high level of uncertainty, in-situ measurements are preformed and used in a back-analysis to validate the existing model. Reducing the uncertainty of the soil parameters allows more reliable predictions of the system behaviour. However, initially only assumptions can be made what type of measurement might be most suitable to reduce the parameter uncertainty most efficiently. Using the approach of Bayesian optimal experimental design, the present study enables to find an arrangement of sensors that provides data which is most likely to enable an accurate model validation. The application of this approach is performed using a Finite-Element model of a residence building that is underpassed by twin-tubed tunnel.*

1 INTRODUCTION

To meet the infrastructure requirements of modern societies, tunnel construction, especially in urban areas, is essential. However, even after decades of experience an efficient and safe construction is still challenging. One main issue is that the tunnel construction induces a settlement trough at the ground surface which often causes severe damage at surface structures which are located in the range of this trough. This is still the case when Tunnel Boring Machines (TBM) are employed that build the tunnel of prefabricated concrete segments what is nowadays the mostly used method. When the interaction between tunnel construction process and ground behaviour is well known, effective countermeasures can be initiated such as compensation grouting or an adaptation of the face and grouting pressure of the TBM. However, this requires accurate simulation models and precise knowledge of the surrounding ground properties. Nowadays, simulation of TBM advancement can be performed highly accurate using the Finite-Element (FE) method with the possible drawback of long calculation time, as shown e.g. in [4][8]. To increase the efficiency of this method, hybrid models, as introduced in [20], can be employed to improve the accuracy while reducing the computational effort.

However, the most sophisticated model does not help if its employed parameters do not reflect in-situ conditions. To receive such an overall understanding of the soil properties, laboratory experiments are usually performed on samples taken in-situ. These tests provide accurate results, but one should be aware that they cannot be representative for the whole considered domain, as natural soils are often heterogeneous [16]. Besides, the in-situ conditions such as stress level or degree of saturation might not be exactly reproducible in the laboratory. To overcome this problem, the concept of the observational method, introduced in [14] can be efficient as in-situ data is employed to validate an existing model. Thereby, a cost function is formulated that aims to reduce the discrepancy between in-situ measurements and model response by varying the properties of the soil as shown for applications from geotechnical engineering e.g. in [18], [9].

A usually neglected aspect is the question which type of measurement data should be employed to allow an efficient and reliable parameter identification. This means among others what type of measurements such as vertical displacements or excess pore water pressure, but also where a measurement device should be placed and at what time and time frequency the data should be obtained. Unlike geotechnical engineering, several other research field have addressed this type of questions by the concept of the so-called Optimal Experimental Design (OED). In the field of system biology e.g [17] investigated how an experiment should be designed to identify growth parameters in a bioreactor, while in [1] an example from chemical engineering, namely defining sampling intervals to determine reaction rates, is shown.

In geotechnical Engineering, recent works have shown the applicability of the concept to examples of a dike subjected to a rapid water drawdown [5] and a tunnel passing by an existing building [6]. In these cases, the methods of global sensitivity analysis and Bootstrap resampling, respectively, are employed to identify where and when sensors should be placed to record e.g. displacements in order to identify the underlying soil properties. However, both approaches have disadvantages as they were either inaccurate or highly time consuming. Therefore, in the present study the concept of Bayesian OED is employed that refers to the basic Bayes' theorem, introduced in [2], that describes how conditional probabilities of events are combined. As the objective of OED for parameter identification is in general to reduce parameter uncertainties based on observed values, the Bayes' theorem is a powerful tool to improve this process. Therefore, the Bayes' theorem is employed as shown in Eq. 1 where the uncertainty of the parameters of interest $\boldsymbol{\theta}$ are considered in the form of probability distributions $P(\theta)$. Measurement data $\tilde{y}$

contains the "true" system response y, but is subjected to some measurement error e that corresponds to a known probability distribution. Therefore, obtained data is regarded as event of probability $P(\tilde{y})$. The conditional probability $P(\tilde{y}|\theta)$ describes the probability that a certain output results from a parameter combination, what is assumed to be known, while $P(\theta|\tilde{y})$ is the value of interest that describes the probability of a set of parameters given a measured output value.

$$P(\theta|\tilde{y}) = \frac{P(\tilde{y}|\theta) \cdot P(\theta)}{P(\tilde{y})} \tag{1}$$

Within the process of Bayesian OED, it is investigated which measurement data has the highest probability of reducing the initial uncertainty of the model parameters.

2 APPLICATION TO TUNNEL CONSTRUCTION EXAMPLE

The methodology described in the previous section shall be applied using an FE-model that simulates the construction of a twin tunnel below an existing eleven storey building. A detailed description of the model and to the simulated real construction site can be found in [4] including detailed soil and process data. This model has been generated using the FE-code Plaxis 3D (Version 2016 [15]) and it is shown without the soil clusters in Fig. 1. The two tunnel tubes with diameter of 6.7 m are constructed one after another in a mean distance of 16.7 m in between and a depth of 15 m below ground surface. The soil below the building consists mostly of gravelly sand with a band of sandy loam in-between 20 and 25 m depth. Both soils' constitutive behaviour is simulated using the Hardening Soil model with small strain stiffness (HSsmall) [3]. The tunnel tubes are subdivided into segments of 1.4 m in longitudinal direction, corresponding to the employed concrete linings. To enable a realistic simulation of the stepwise construction, the TBM is simulated consisting of the first seven segments, that advance in each phase of the simulation by one segment. Between the TBM and the following concrete linings, one segment is left out, only supported by a surface load to simulate the fresh grout mortar.

As the tunnel construction underpasses the building, a settlement trough develops which exact shape can only be roughly estimated in advance, especially due to the offset position and load distribution of the building, and the complex soil-building interaction. Preliminary to the present work, detailed statistical evaluations, including global sensitivity analyses, a method successfully applied in geotechnical applications e.g. in [11], have been performed that show that the parameters that influence mostly these settlements and the tilting behaviour of the building are the secant stiffness E_{50}^{ref} (that is correlated with the tangent stiffness E_{ur}^{ref} and unloading-reloading stiffness E_{ur}^{ref}), the small strain stiffness modulus G_0, and the volume loss factor V_L. Therefore, it is of highest interest to know the exact values of these parameters and it is most promising to do this using back-analysis of measured data. Different types of sensors are possible in case of such applications, e.g. inclinometer or pore water pressure, horizontal, and vertical settlement transducers. As on the real construction site, vertical displacement monitoring has been performed, in this study these are considered as model response. Consequently, the question posed is where to perform measurements of vertical displacements to validate the existing numerical model.

3 EMPLOYED CONCEPT OF BAYESIAN OED

To apply the intended concepts of OED and identify a Bayesian design, first of all a fast model is needed that enables frequent calls to generate reliable distributions of model responses.

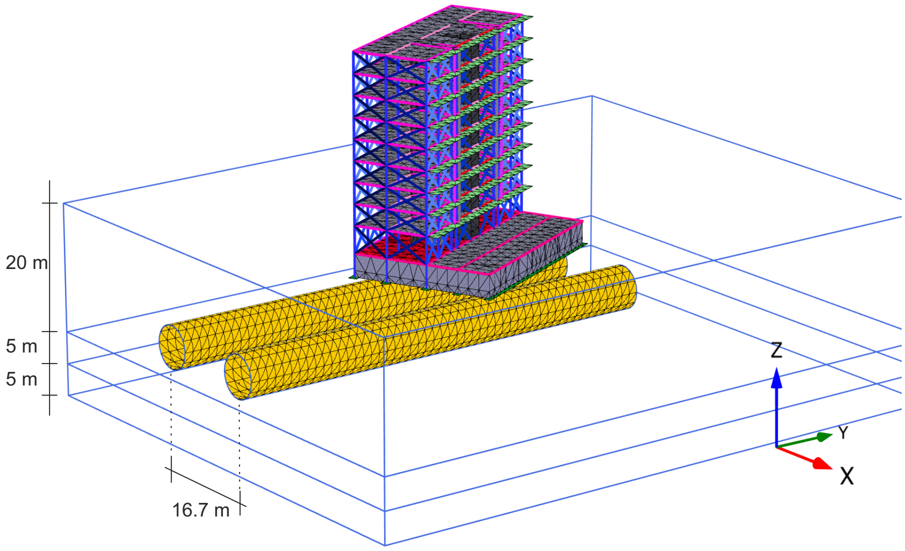


Figure 1: View on the FE-model with removed soil elements

This is obviously not the case for the employed FE-model that needs more than eight hours for one calculation run. For substitution, 100 samples of the input parameter vector $\boldsymbol{\theta}$ (containing the aforementioned E_{50}^{ref}, E_{ur}^{ref}, and E_{ur}^{ref}) are generated using Latin Hypercube sampling (LHS) [13] and run in the FE-model. As results, the vertical settlements are obtained in an evenly spaced grid at the surface of the model for each of the 100 runs. For each point of the grid, the relationship between input parameter $\boldsymbol{\theta}$ and model response y is approximated by a second-order polynomial function. To test the accuracy of this approximation function, a second set of 20 LHS-generated samples is run in the FE-model and the results are compared with each other, having coefficient of determination 0.99. As the second step, to be able to access all coordinates in between the grid points, a cubic interpolation is performed that does not use discrete model outputs as input data, but the function defined to approximate $y(\boldsymbol{\theta})$. Using both steps allows to obtain the settlement for any position and any combination of soil parameters. In the present study, exemplary, only data of the 44th excavation step is employed that corresponds to the end of the construction of the first tunnel tube. As afterwards the uncertainties in the numerical model are reduced, the soil-building interaction can be predicted much more accurately and the construction of the second tube can be executed in a safer manner. In practice, the approach would be performed continuously, providing constantly updated soil parameters or new suggestions for an optimal sensor arrangement.

As described in Sec. 1, the objective of OED is to reduce the uncertainty of the employed model parameters. This uncertainty is described by the standard deviation $\sigma(\boldsymbol{\theta})$. The utility $U(\delta)$ defines how a certain design δ reduces the uncertainty and is to be maximised. The identification of the optimal design $\delta^{\dagger}$ is performed in this study by using artificially generated outputs $\tilde{y}$ that are back-calculated using the aforementioned surrogate model, while validation of the final results can be done by comparison with in-situ measurement data. In this process, one should be aware that besides the soil inhomogeneity, model uncertainty and measurement errors can never be avoided and always influence the process of measurement based parameter identification, making it an ill-posed problem. To take into account measurement uncertainty, artificial noise is added to generated model outputs before back-calculation while in this study the model uncertainty is neglected. Therefore, random output samples y are generated according to the

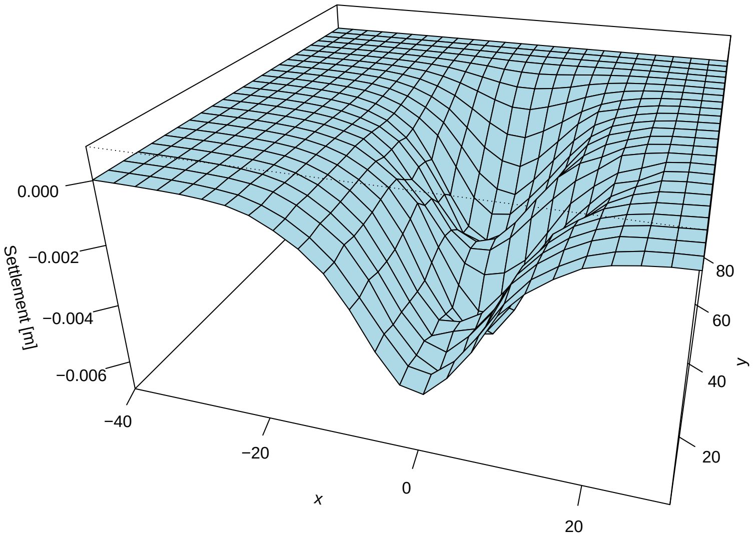


Figure 2: Visualisation of the settlement shape of the surrogate model of one random combination of soil parameters

defined initial distribution of the input parameter and the considered ”experimental design”, δ:

$$y = f(\boldsymbol{\theta}, \delta) \tag{2}$$

To account for the realistic case of measurement uncertainty, artificial Gaussian white noise is added to the randomly generated output data (Eq. (3)), consisting of a random part and a systematic part that is related to the value of y. By varying the ratio and values of e_1 and e_2, sensors with different types of accuracy can be considered.

$$\tilde{y}(\boldsymbol{\theta}) = y(\boldsymbol{\theta}) + y(\boldsymbol{\theta}) \cdot \omega_{sys} \cdot e_1 + \omega_{ran} \cdot e_2,\ \omega \sim \mathcal{N}[0, 1] \tag{3}$$

During back-calculation of this noisy data $\tilde{y}$, the identified parameters $\tilde{\boldsymbol{\theta}}$ will most probably deviate from the ”true” solution. Therefore, this parameter identification is repeated B times such that it converges to mean value $\bar{\boldsymbol{\theta}}$ that for good experimental designs will be equal to the true value $\boldsymbol{\theta}$. The difference between different designs is the standard deviation of $\tilde{\boldsymbol{\theta}}$ and the best design is consequently that where it is smallest.

In the framework of Bayesian OED this means we want to find that design that provides measurements that allows to reduce most the probability range of the considered soil parameters $\boldsymbol{\theta}$. The design variables that are considered in this study are the amount of measurement points and the x- and y- coordinate of each of these points. The number of placed sensors is assumed to be equal to six, but will be the subject of a subsequent study. As each new parameter increases the dimensionality of the problem by two dimensions, identifying the OED becomes computationally too expensive, wherefore the so-called Approximate Coordinate Exchange (ACE) algorithm is employed that was introduced in [12]. Later on, in [10] this algorithm was adapted for Bayesian OED and implemented as a package in the software for statistical computing R. Hereby, each dimension of the problem is considered individually in the optimisation process, while the other parameters (here the coordinates of the sensors) keep fixed. For a random (or predefined) initial design δ^0 the utility $u(\boldsymbol{\delta}, \boldsymbol{y}, \boldsymbol{\theta})$ is calculated for a number B of parameter samples that is generated according to their initial distribution. The mean of these B evaluations gives the expected utility $\tilde{U}(\delta)$ of this design (Eq. 4).

$$\tilde{U}(\delta) = \sum_{l=1}^{B} u(\delta, y_l, \theta_l)/B \tag{4}$$

By varying parameters of the current design δ^C (i.e. the coordinates of the sensors), a series of utility values is obtained as:

$$\tilde{U}(\delta^1|\delta^C_{\mathbf{i}(i)}), ..., \tilde{U}(\delta^m|\delta^C_{\mathbf{i}(i)}), \tag{5}$$

where m referes to the dimension of the design (i.e. number of sensors times two) and i is the control variable of the variations of the current design. The design $\delta_i^\dagger$ that enables the maximum value of the obtained utilities is replacing the current optimal design δ_i^C (Eq. 6) and the underlying probability distribution of θ is obtained according to Bayesian principle

$$\delta_i^\dagger = \arg max_{\delta_i \in \mathfrak{D}_i} \hat{U}(\delta|\delta^C_{\mathbf{i}(i)}) \tag{6}$$

With the updated probability distribution and the current optimal design $\delta_i^\dagger$ as start values the previous steps are repeated until the algorithm converges.

4 OUTPUTS

The start design δ^0 for the introduced problem in Sec. 3 is generated as LHS design of six points on the ground surface of the numerical model, while the area where the building is located is excluded of the design area. The utility function is defined in way that for each of the three parameters of interest, two sensors are assigned to find the position where data can be obtained to identify this parameter most accurately. The identified parameters are grouped together in the vector of parameters $\tilde{\theta}$ and are evaluated for each design according to Eq. 7.

$$\tilde{U}(\delta) = \sum_{l=1}^{B} \left(- \sum_{i=1}^{m} |\tilde{\theta}_l(\delta_i) - \bar{\theta}| + \frac{1}{\det C_\theta(\delta_i)} \right) /B \tag{7}$$

Using the deviation from the parameter mean $|\tilde{\theta}_i(\delta) - \bar{\theta}|$ and the inverse of the parameters' covariance matrix C_θ enables the utility function to find designs that provide the correct parameter values with least variation at the same time. By choosing a sufficiently large value of B, $\tilde{U}(\delta)$ becomes a distribution smooth enough to be handled in the context of Eq. 1.

The algorithm converges within few steps to an optimal arrangement that is shown in Fig.3. There, the tunnel excavation proceeds from bottom to top of the figure (in y-direction). Larger settlements are indicated by red coloured areas. The bypassed building is symbolised by the black block and the sensor arrangement by the crossed circles. The area of the building itself is excluded from the search space as it would be technically not possible to place any sensors below it. It is suggested to position the sensors not right above the tunnel centreline where the settlements are largest but on the edge of the settlement trough. This corresponds to the further works in this field, e.g. [6] and [19], that most suitable placements for measurements are not where the maximum value of a system output occurs, but where the changes in the parameters provoke the largest relative changes of the settlements, i.e. where the output is most sensitive to a certain parameter. However, it should be indicated that this is highly influenced by the amount and type of error, defined by e_1 and e_2 in Eq. 3, that is added. Regarding the parameters for which the differently located sensors are employed, the two sensors at the lowest positions are both designated for E_{50}^{ref} what shows that this parameter needs locations with higher gradient.

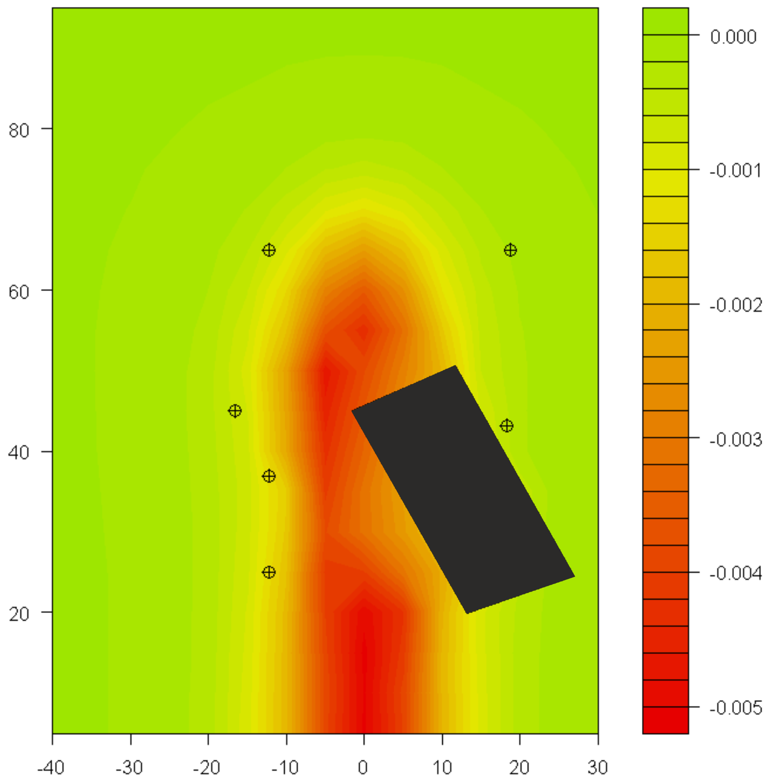


Figure 3: Positions of the identified design, mapped on the ground surface in top view, all dimensions in [m]

5 CONCLUSIONS

In this study, an approach is introduced to identify an optimal experimental design for model validation of mechanised tunnelling using settlement data. To find this design, that consists of the positions in which sensors should be placed to measure vertical settlements, the concept of Bayesian OED is employed using the ACE algorithm to reduce the calculation time. The suitability is shown using a FE-model of a tunnel underpassing a high-rise building. The identified results are comprehensible and comparable with results of previous studies that employed different methods. However, the presented case is a simplified example and many further aspects are investigated in ongoing research. Details of the noisy data should be considered. The impact of different values of e_1 and e_2, but also their ratio and the type of error distribution will be varied systematically. Besides, varying the amount of sensors with different ”inaccuracies” e gives additional insight to a cost-efficiency understanding as high accuracy of sensors is usually connected to higher costs. Further research is intended to address the model state depending on the advancement of the TBM: as the relative position of building an TBM is different in every excavation step, the resulting OED is always different. Therefore, it can be useful to change the arrangement of measurement positions or their weighting in the utility function. Besides, it is of interest to consider further types of measurement data such as excess pore water pressures or horizontal deformations.

ACKNOWLEDGEMENT

The authors acknowledge financial support provided by the German Science Foundation (DFG) under the framework of subproject C2 in the Collaborative Research Center SFB 837.

REFERENCES

[1] A. Bardow, Optimal experimental design of ill-posed problems: The METER approach", *Computers and Chemical Engineering*, **32**, 115 – 124, 2008.

[2] T. Bayes and R. Price, An Essay towards solving a Problem in the Doctrine of Chances, *Philosophical Transactions of the Royal Society of London*, **53**, 370–418, 1763.

[3] T. Benz, *Small-Strain Stiffness of Soils and its Numerical Consequences*, Doctoral Dissertation, University of Stuttgart, 2006.

[4] V. Fargnoli and C.G Gragnano and D. Boldini and A. Amorosi, 3D numerical modelling of soil-structure interaction during EPB tunnelling, *Géotechnique*, **65** **(1)**, 23 – 37,2015.

[5] R. Hölter, C. Zhao, E. Mahmoudi, A. A. Lavasan, M. Datcheva, M. König and T. Schanz. Optimal measurement design for parameter identification in mechanized tunneling, *Underground Space*, **3**, 34-44, 2018.

[6] R. Hölter, E. Mahmoudi, S. Rose, M. König, M. Datcheva and T. Schanz. Employment of the bootstrap method for optimal sensor location considering uncertainties in a coupled hydro-mechanical application, *Applied Soft Computing Journal*, **75**, 298-309, 2019.

[7] X. Huan and Y. M. Marzouk, Simulation-based optimal Bayesian experimental design for nonlinear systems, *Journal of Computational Physics*, **232** **(1)**, 288–317, 2013.

[8] T. Kasper, G. Meschke, A 3D finite element simulation model for TBM tunnelling in soft ground, *International Journal for Numerical and Analytical Methods in Geomechanics*, **28** **(14)**, 1441-1460, 2006.

[9] X. Li, C. Zhao, R. Hölter, M. Datcheva and A.A. Lavasan, Modelling of a Large Landslide Problem under Water Level FluctuationModel Calibration and Verification, *Geosciences*, **9** **(2)**, 89, 2019.

[10] A. M. Overstall and D.C. Woods, Bayesian Design of Experiments Using Approximate Coordinate Exchange, *Technometrics*, **59** **(4)**, 458–470, 2017.

[11] E. Mahmoudi, R. Hölter, R. Georgieva, M. König and T. Schanz, On the Global Sensitivity Analysis Methods in Geotechnical Engineering: A Comparative Study on a Rock Salt Energy Storage, *International Journal of Civil Engineering*, **17**, 131–143, 2019.

[12] R. K. Meyer and C.J. Nachtsheim, The Coordinate-Exchange Algorithm for Constructing Exact Optimal Experimental Designs, *Technometrics*, **37** **(1)**, 60–69, 1995.

[13] M.D. McKay, R.J. Beckman and W.J. Conover, A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code, *Technometrics*, **21** **(2)**, 239-245, 1979.

[14] R. B. Peck, Advantages and Limitations of the Observational Method in Applied Soil Mechanics, *Géotechnique*, **19 (2)**, 171–187, 1969.

[15] Brinkgreve, R. B. J., Kumarswamy, S. and Swolfs, W. M. , *PLAXIS 3D 2016: Reference Manual*, PLAXIS BV, Delft, The Netherlands, 2016.

[16] K.K. Phoon, F.H. Kulhawy, Characterization of geotechnical variability, *Canadian Geotechnical Journal*, **36**, 612–624, 1999.

[17] R. Schenkendorf, A. Kremling and M. Mangold, Optimal Experimental Design with the sigma point method, *IET Systems Biology*, **3 (1)**, 10–23, 2009.

[18] C. Zhao, A. A. Lavasan, T. Barciaga, V. Zarev, M. Datcheva and T. Schanz, Model validation and calibration via back analysis for mechanized tunnel simulations - The Western Scheldt tunnel case, *Computers and Geotechnics*, **69**, 601–614, 2015.

[19] C. Zhao, A. A. Lavasan, R. Hölter, and T. Schanz, Mechanized tunneling induced building settlements and design of optimal monitoring strategies based on sensitivity field, *Computers and Geotechnics*, **97**, 246–260, 2018.

[20] C. Zhao, R. Hölter, M. König and A. A. Lavasan, A hybrid model for estimation of ground movements due to mechanized tunnel excavation, *Computers-Aided Civil and Infrastructure Engineering*, https://doi.org/10.1111/mice.12438, 2019.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

RELIABILITY-BASED DESIGN OPTIMIZATION BY USING ENSEMBLES OF METAMODELS

N. Strömberg

Department of Mechanical Engineering
Örebro University
SE-701 82 Örebro, Sweden
e-mail: niclas.stromberg@oru.se

Keywords: RBDO, Metamodels, Ensemble, Convex combination

Abstract. *The standard approach when establishing optimal ensembles of metamodels (OEM) is to apply affine combinations of metamodels. However, we have found when performing reliability-based design optimization (RBDO) that this choice might sometimes result in poor representation of the limit state surfaces. Therefore, in this work, we suggest to use convex combinations instead of affine combinations in order to get robust RBDO by using OEM. Optimal convex combinations of metamodels are established by minimizing the taxicab, Euclidean or infinity norm of the PRESS vector. The PRESS vector is defined by the leave-one-out cross-validation errors of a linear combination of ten metamodels, which constitute different settings of quadratic regression, Kriging, radial basis functions, polynomial chaos and support vector regression. Thus, the minimization of the norms are constrained such that the sum of weights of the linear combination equals one and only non-negative weights are allowed. We have found that the latter constraints might be extremely important when the ensembles of metamodels represent the limit surfaces. The most probable point (MPP) of the OEM is established in the physical space where the distance is minimized in the metric of Hasofer-Lind. The solution to the corresponding Karush-Kuhn-Tucker conditions is obtained by using Newton's method with an inexact Jacobian and a line-search of Armijo type. At the MPP, we perform Taylor expansions of the OEM using intermediate variables defined by the iso-probabilistic transformation. In such manner, we derive a quadratic programming (QP) problem which is solved in the standard normal space. This is done for several probability distributions such as e.g. lognormal, Gumbel, gamma and Weibull. The optimal solution to the QP problem is mapped back to the physical space and new Taylor expansions of the OEM are derived and a new QP problem is formulated and solved. This procedure continues in sequence until we obtain convergence of our RBDO problem. The steps presented above constitute our proposed FORM-based sequential QP approach for RBDO by using OEM. The implementation of the approach in an in-house toolbox is very robust and efficient.*

1 INTRODUCTION

Optimization of machine components and systems of machine components by using non-linear finite element analysis are typically performed for deterministic variables and parameters. This is indeed a draw-back because it is not trivial to choose a corresponding optimal safety factor. Examples of uncertainties influencing this choice of safety factor are uncertainties in loading conditions, material parameters and geometry. An established approach to include uncertainties like these in the optimization is to apply reliability-based design optimization [1]. The safety factor is then included in the optimization by setting a target on the reliability that the design criteria are satisfied. However, the RBDO problem is not easy to solve and it is indeed a challenging task to incorporate the RBDO formulation explicitly in a non-linear finite element code. A way forward to handle this latter difficulty is to treat the non-linear finite element model as a black-box by setting up design of experiments (DoE) and training metamodels for these DoEs. Popular metamodels for this task are Kriging, radial basis function networks, polynomial chaos expansion and support vector regression. For instance, in Strömberg [2], RBDO was performed by using radial basis function networks with a priori bias [3]. Most recently, RBDO with support vector machines was studied in [4], and a comparison of RBDO with several different established metamodels was conducted in [5].

A question discussed and investigated over the years is which one of all different types of metamodels is the best. In my opinion, looking for the answer to this question is like looking for the holy grail. A better way is to set up a linear combination of your metamodels and then find the optimal weights for you particular DoE. The standard approach is to use affine combinations. An early work on ensemble of metamodels can be found in [6], where a weighted average of the metamodels was adopted. Examples of other works on ensemble of metamodels are e.g. [7, 8, 9, 10, 11]. Acar [7] studied various approaches for constructing affine combinations of metamodels using local measures. Zhou et al. [8] established affine combinations of metamodels with recursive arithmetic average. Lee and Choi [9] proposed a new pointwise affine combination of metamodels by using a nearest points cross-validation approach. Shi et al. [10] proposed efficient affine combinations of radial basis function networks. Most recently Song et al. [11] suggested an advanced and robust affine combination of metamodels by using extended adaptive hybrid functions. But, in this work, we do not adapt the approach of affine combinations of metamodels, instead we suggest to use convex combinations of metamodels for robust treatment of the limit state surface. The optimal ensembles of metamodels are then used to set up RBDO problems which we solve by using the FORM-based sequential quadratic programming (SQP) approach presented by Strömberg [12].

The outline of the paper is as follows: in the next section the basic equations of polynomial regression models, Kriging, radial basis function networks, polynomial chaos expansion and support vector regression are reviewed, in section 3 we present our new approach for convex combinations of our metamodels by minimizing the PRESS vector for three different norms, section 4 reviews the SQP-based RBDO approach by using FORM and SORM, and then we study a RBDO benchmark as a blackbox by performing DoE and adopting our new convex combinations of metamodels. It is demonstrated that a most accurate solution is obtained most efficiently. In this section, we also motivate our choice of convex combinations instead of affine combinations by studying the Hosaki test function. Finally, some concluding remarks are given.

2 METAMODELS

Let us assume that we have a set of sampling data $\{\hat{\boldsymbol{x}}^i, \hat{f}^i\}$ obtained from design of experiments. We would like to represent this set of data with a function, which we call a response surface, a surrogate model or a metamodel. One choice of such a function is the regression model given by

$$f = f(\boldsymbol{x}) = \boldsymbol{\xi}(\boldsymbol{x})^T\boldsymbol{\beta}, \tag{1}$$

where $\boldsymbol{\xi} = \boldsymbol{\xi}(\boldsymbol{x})$ is a vector of polynomials of $\boldsymbol{x}$ and $\boldsymbol{\beta}$ contains regression coefficients. By minimizing the sum of squared errors, i.e.

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{N} \left(X_{ij}\beta_j - \hat{f}^i \right)^2, \tag{2}$$

where $X_{ij} = \xi_j(\hat{\boldsymbol{x}}^i)$ and N is the number of sampling points, then we obtain optimal regression coefficients from the normal equation according to

$$\boldsymbol{\beta}^* = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\hat{\boldsymbol{f}}. \tag{3}$$

Examples of other useful metamodels are Kriging, radial basis functions, polynomial chaos expansion and support vector regression. The basic equations of these models are presented in the following.

2.1 Kriging

The Kriging model is given by

$$f(\boldsymbol{x}) = \boldsymbol{\xi}(\boldsymbol{x})^T\boldsymbol{\beta}^* + \boldsymbol{r}(\boldsymbol{x})^T\boldsymbol{R}^{-1}(\boldsymbol{\theta}^*)\left(\hat{\boldsymbol{f}} - \boldsymbol{X}\boldsymbol{\beta}^*\right), \tag{4}$$

where the first term represents the global behavior by a linear or quadratic regression model and the second term ensures that the sample data is fitted exactly. $\boldsymbol{R} = \boldsymbol{R}(\boldsymbol{\theta}) = [R_{ij}]$, where

$$R_{ij} = R_{ij}(\boldsymbol{\theta}, \hat{\boldsymbol{x}}^i, \hat{\boldsymbol{x}}^j) = \exp\left(-\sum_{k=1}^{N}\theta_k(\hat{x}_k^i - \hat{x}_k^i)^2\right). \tag{5}$$

Furthermore, $\boldsymbol{\theta}^*$ is obtained by maximizing the following likelihood function:

$$\frac{1}{\sigma^N\sqrt{\det(\boldsymbol{R})(2\pi)^N}}\exp\left(-\frac{(\boldsymbol{X}\boldsymbol{\beta} - \hat{\boldsymbol{f}})^T\boldsymbol{R}^{-1}(\boldsymbol{X}\boldsymbol{\beta} - \hat{\boldsymbol{f}})}{2\sigma^2}\right) \tag{6}$$

and

$$\boldsymbol{\beta}^* = \left(\boldsymbol{X}^T\boldsymbol{R}^{-1}(\boldsymbol{\theta}^*)\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{R}^{-1}(\boldsymbol{\theta}^*)\hat{\boldsymbol{f}}. \tag{7}$$

2.2 Radial basis function networks

For a particular input $\hat{\boldsymbol{x}}^k$ the outcome of the radial basis function network can be written as

$$f^k = f(\hat{\boldsymbol{x}}^k) = \sum_{i=1}^{N_\Phi} A_{ki}\alpha_i + \sum_{i=1}^{N_\beta} B_{ki}\beta_i, \tag{8}$$

where N_Φ is the number of radial basis functions, N_β is the number of regression coefficients in the bias,

$$A_{ki} = \Phi_i(\hat{\boldsymbol{x}}^k) \quad \text{and} \quad B_{ki} = \xi_i(\hat{\boldsymbol{x}}^k). \tag{9}$$

Both linear and quadratic regression models are used as bias. Furthermore, $\Phi_i = \Phi_i(\hat{\boldsymbol{x}}^k)$ represents the radial basis function.

Thus, for a set of inputs, the corresponding outgoing responses $\boldsymbol{f} = \{f^i\}$ of the network can be formulated compactly as

$$\boldsymbol{f} = \boldsymbol{A}\boldsymbol{\alpha} + \boldsymbol{B}\boldsymbol{\beta}, \tag{10}$$

where $\boldsymbol{\alpha} = \{\alpha_i\}$, $\boldsymbol{\beta} = \{\beta_i\}$, $\boldsymbol{A} = [A_{ij}]$ and $\boldsymbol{B} = [B_{ij}]$. If we let $\boldsymbol{\beta}$ be given a priori by the normal equation as

$$\boldsymbol{\beta} = \left(\boldsymbol{B}^T\boldsymbol{B}\right)^{-1}\boldsymbol{B}^T\hat{\boldsymbol{f}}, \tag{11}$$

then

$$\boldsymbol{\alpha} = \boldsymbol{A}^{-1}\left(\hat{\boldsymbol{f}} - \boldsymbol{B}\hat{\boldsymbol{\beta}}\right). \tag{12}$$

Otherwise, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are established by solving

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{0} \end{bmatrix} \left\{ \begin{array}{c} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{array} \right\} = \left\{ \begin{array}{c} \hat{\boldsymbol{f}} \\ \boldsymbol{0} \end{array} \right\}. \tag{13}$$

2.3 Polynomial chaos expansion

Polynomial chaos expansion by using the Hermite polynomials $\varphi_n = \varphi_n(y)$ can be written as

$$f(\boldsymbol{x}) = \sum_{i=0}^{M} c_i \prod_{j=1}^{N_{\text{VAR}}} \varphi_i(x_j), \tag{14}$$

where $M+1$ is the number of terms and constant coefficients c_i, and N_{VAR} is the number of variables x_i. The Hermite polynomials are defined by

$$\varphi_n = \varphi_n(y) = (-1)^n \exp\left(\frac{x^2}{2}\right)\frac{\mathrm{d}^n}{\mathrm{d}x^n}\left(\exp\left(-\frac{x^2}{2}\right)\right). \tag{15}$$

For instance, one has

$$\varphi_0 = 1, \tag{16a}$$
$$\varphi_1 = y, \tag{16b}$$
$$\varphi_2 = y^2 - 1, \tag{16c}$$
$$\varphi_3 = y^3 - 3y, \tag{16d}$$
$$\varphi_4 = y^4 - 6y2 + 3, \tag{16e}$$
$$\varphi_5 = y^5 - 10y^3 + 15y, \tag{16f}$$
$$\varphi_6 = y^6 - 15y^4 + 45y^2 - 15, \tag{16g}$$
$$\varphi_7 = y^7 - 21y^5 + 105y^3 - 105y. \tag{16h}$$

The unknown constants c_i are then established by using the normal equation. A nice feature of the polynomial chaos expansion is that the mean of $f(\boldsymbol{X})$ in (14) for uncorrelated standard normal distributed variables X_i is simply given by

$$\mathrm{E}[f(\boldsymbol{X})] = c_0. \tag{17}$$

2.4 Support vector regression

The soft non-linear support vector regression model reads

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \lambda^i k(\boldsymbol{x}^i, \boldsymbol{x}) - \sum_{i=1}^{N} \hat{\lambda}^i k(\boldsymbol{x}^i, \boldsymbol{x}) + b^*, \tag{18}$$

where λ^i, $\hat{\lambda}^i$ and b^* are established by solving

$$\begin{cases} \displaystyle\min_{(\boldsymbol{\lambda}, \hat{\boldsymbol{\lambda}})} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\lambda^i - \hat{\lambda}^i)(\lambda^j - \hat{\lambda}^j) k(\boldsymbol{x}^i, \boldsymbol{x}^j) + \sum_{j=1}^{N} (\hat{\lambda}^i - \lambda^i) \hat{f}^i + \delta \sum_{j=1}^{N} (\lambda^i + \hat{\lambda}^i) \\ \text{s.t.} \begin{cases} \displaystyle\sum_{j=1}^{N} (\lambda^i - \hat{\lambda}^i) = 0, \\ 0 \leq \lambda^i, \hat{\lambda}^i \leq C, \quad i = 1, \ldots, N. \end{cases} \end{cases} \tag{19}$$

Finally, the corresponding least square support vector regression model is established by solving

$$\begin{bmatrix} 0 & -\boldsymbol{1}^T & \boldsymbol{1} \\ \boldsymbol{1} & \boldsymbol{B} + \gamma \boldsymbol{I} & -\boldsymbol{B} \\ -\boldsymbol{1} & -\boldsymbol{B} & \boldsymbol{B} + \gamma \boldsymbol{I} \end{bmatrix} \begin{Bmatrix} b \\ \boldsymbol{\lambda} \\ \hat{\boldsymbol{\lambda}} \end{Bmatrix} = \begin{Bmatrix} 0 \\ \hat{\boldsymbol{f}} - \delta \boldsymbol{1} \\ -\hat{\boldsymbol{f}} - \delta \boldsymbol{1} \end{Bmatrix}, \tag{20}$$

where $\gamma = 1/C$, $\boldsymbol{1} = \{1; \ldots; 1\}$ and

$$\boldsymbol{B} = [B_{ij}], \quad B_{i,j} = \varphi(\boldsymbol{x}^i)\varphi(\boldsymbol{x}^j) = k(\boldsymbol{x}^i, \boldsymbol{x}^j) \tag{21}$$

is a matrix containing kernel values.

3 ENSEMBLE OF METAMODELS

Let us now define a new metamodel $f_{\text{en}} = f_{\text{en}}(\boldsymbol{x})$ as a convex combination of the metamodels presented in the previous section, i.e.

$$f_{\text{en}} = f_{\text{en}}(\boldsymbol{x}) = \sum_{i=1}^{M} w_i f_i(\boldsymbol{x}), \tag{22}$$

where M is the total number of metamodels in the ensemble, $w_i \geq 0$ are weights satisfying $w_1 + w_2 + \ldots + w_M = 1$ and $f_i = f_i(\boldsymbol{x})$ represents any particular metamodel of the ones presented above. In this work, we let $M = 10$ and let the ensemble be given by the following metamodels:

1. Quadratic regression model,
2. Kriging with linear bias,
3. Kriging with quadratic bias,
4. RBFN with linear a priori bias,
5. RBFN with quadratic a priori bias,
6. RBFN with linear a posteriori bias,
7. RBFN with quadratic a posteriori bias,

8. Polynomial chaos expansion,
9. Support vector regression,
10. Least-square support vector regression.

The leave-one-out cross-validation error at a point $\hat{\boldsymbol{x}}^k$ of y_{en} is given by

$$e_k = e(\hat{\boldsymbol{x}}^k) = \hat{f}^k - f_{\text{en}}^{(-k)}(\boldsymbol{x}^k) = \hat{f}^k - \sum_{i=1}^{M} w_i f_i^{(-k)}(\hat{\boldsymbol{x}}^k), \tag{23}$$

where $y_i^{(-k)}(\boldsymbol{x})$ represents the metamodel with the k-*th* data point excluded from the sampling set $\{\hat{\boldsymbol{x}}^i, \hat{f}^i\}$. If we now perform the leave-one-out cross-validation in (23) for every data point, then we establish the vector of PRESS residuals:

$$\boldsymbol{e} = \{e_i\} = \hat{\boldsymbol{f}} - \boldsymbol{Y}\boldsymbol{w}, \tag{24}$$

where $\hat{\boldsymbol{f}}$ contains $\hat{f}^i$, $\boldsymbol{w}$ is a vector of weights w_i and

$$[\boldsymbol{Y}]_{ij} = f_j^{(-i)}(\hat{\boldsymbol{x}}^i). \tag{25}$$

Let us now minimize the norm of the PRESS vector $\boldsymbol{e}$ subjected to $\boldsymbol{w}^T\mathbf{1} = 1$ and $w_i \geq 0$, i.e.

$$\begin{cases} \min\limits_{\boldsymbol{w}} \; ||\boldsymbol{e}||_x \\ \text{s.t.} \begin{cases} \boldsymbol{w}^T\mathbf{1} = 1, \\ w_i \geq 0, \quad i = 1, \ldots, M, \end{cases} \end{cases} \tag{26}$$

where $||\boldsymbol{e}||_x$ represents the taxicab norm $||\boldsymbol{e}||_1$, the Euclidean norm $||\boldsymbol{e}||_2$ or the infinity norm $||\boldsymbol{e}||_\infty$ according to

$$\begin{gathered} ||\boldsymbol{e}||_1 = \sum_{i=1}^{N} |e_i|, \\ ||\boldsymbol{e}||_2 = \sqrt{\boldsymbol{e}^T\boldsymbol{e}}, \\ ||\boldsymbol{e}||_\infty = \max(e_1, \ldots, e_N), \end{gathered} \tag{27}$$

where N is the number of sampling points.

The problem in (26) with the taxicab norm corresponds to the following LP-problem:

$$\begin{cases} \min\limits_{(\boldsymbol{w},\boldsymbol{p},\boldsymbol{q})} \; \sum_{i=1}^{N} p_i + q_i \\ \text{s.t.} \begin{cases} \boldsymbol{Y}\boldsymbol{w} - \hat{\boldsymbol{f}} = \boldsymbol{p} - \boldsymbol{q}, \\ \boldsymbol{w}^T\mathbf{1} = 1, \\ w_i, p_j, q_j \geq 0, \quad i = 1, \ldots, M, \quad j = 1, \ldots, N. \end{cases} \end{cases} \tag{28}$$

By taking the square of the Euclidean norm, (26) becomes of course a QP-problem. Finally, using the infinity norm, (26) can be rewritten as the following LP-problem:

$$\begin{cases} \min\limits_{(\boldsymbol{w},t)} \; t \\ \text{s.t.} \begin{cases} \boldsymbol{Y}\boldsymbol{w} - \hat{\boldsymbol{f}} \leq t\mathbf{1}, \\ -\boldsymbol{Y}\boldsymbol{w} + \hat{\boldsymbol{f}} \leq t\mathbf{1}, \\ \boldsymbol{w}^T\mathbf{1} = 1, \\ w_i, t \geq 0, \quad i = 1, \ldots, N. \end{cases} \end{cases} \tag{29}$$

Here, and above, $\mathbf{1}$ represents a column vector of ones of proper size.

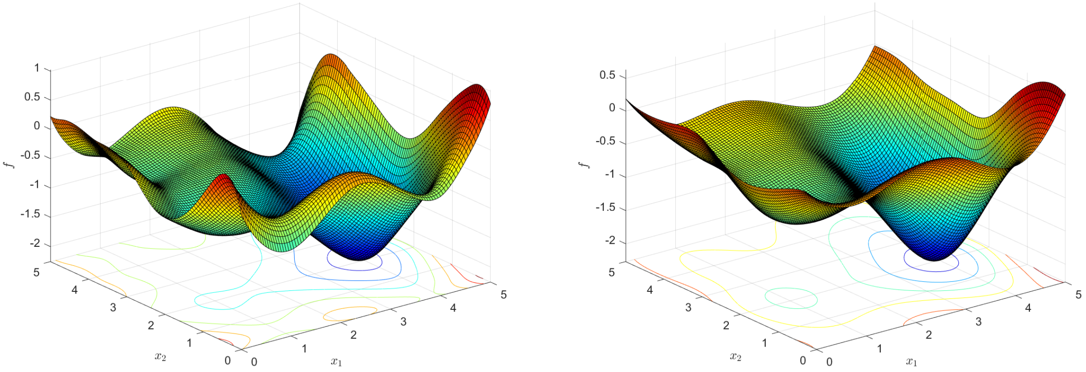


Figure 1: Ensemble of metamodels of the Hosaki test function in (37). Left: affine combination, right: convex combination.

4 RELIABILITY-BASED DESIGN OPTIMIZATION

By using the convex combinations of metamodels presented in the previous section it is straight-forward to set up any design optimization problem as

$$\begin{cases} \min\limits_{\boldsymbol{x}} f_{\text{en}}(\boldsymbol{x}) \\ \text{s.t. } g_{\text{en}}(\boldsymbol{x}) \leq 0. \end{cases} \tag{30}$$

For instance the OEM $f_{\text{en}} = f_{\text{en}}(\boldsymbol{x})$ might represent the mass of a design and $g_{\text{en}} = g_{\text{en}}(\boldsymbol{x})$ is a OEM-based limit surface for the stresses obtained by finite element analysis.

A possible draw-back with the formulation in (30) is that it is not obvious how to include a margin of safety. For instance, what is the optimal safety factor to be included in g_{en}? An alternative formulation that includes a margin of safety is

$$\begin{cases} \min\limits_{\boldsymbol{\mu}} & \mathrm{E}[f_{\text{en}}(\boldsymbol{X})] \\ \text{s.t.} & \Pr[g_{\text{en}}(\boldsymbol{X}) \leq 0] \geq P_s, \end{cases} \tag{31}$$

where $\boldsymbol{X}$ now is treated as a random variable, $\mathrm{E}[\cdot]$ designates the expected value of the function f_{en}, and $\Pr[\cdot]$ is the probability that the constraint $g_{\text{en}} \leq 0$ being true. P_s is the target of reliability that must be satisfied.

4.1 FORM

An established invariant approach for estimating the reliability is the first order reliability method (FORM) suggested by Hasofer and Lind [13]. The basic idea is to transform the reliability constraint from the physical space to a space of uncorrelated standard Gaussian variables and then find the closest point to the limit surface from the origin. This point is known as the most probable point (MPP) of failure. The distance from the origin to the MPP defines the Hasofer-Lind reliability index β_{HL}, which in turn is used to approximate the probability of failure as

$$\Pr[g_{\text{en}} \leq 0] \approx \Phi(-\beta_{\text{HL}}). \tag{32}$$

Assuming that $\boldsymbol{X}$ is normal distributed with means collected in $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ containing standard deviations, the MPP is obtained by solving

$$\begin{cases} \min\limits_{\boldsymbol{x}} \beta_{\text{HL}} = \sqrt{\sum\limits_{i=1}^{N} \left(\dfrac{x_i - \mu_i}{\sigma_i} \right)^2} \\ \text{s.t. } g_{\text{en}}(\boldsymbol{x}) = 0. \end{cases} \tag{33}$$

4.2 SORM

The approximation in (32) is derived by performing a first order Taylor expansion at the MPP and then evaluating the probability. Second order reliability methods (SORM) is obtained by also including the second order terms in the Taylor expansion. Based on these higher order terms the FORM approximation of the reliability is corrected.

For instance, by letting λ_i denoting the principle curvatures of a second order Taylor expansion of g, we can correct (32) by using e.g. Tvedt's formula [14], i.e.

$$\begin{aligned} \Pr[g_{\text{en}} \leq 0] &\approx P_1 + P_2 + P_3, \\ P_1 &= \Phi(-\beta_{\text{HL}}) \prod_{i=1}^{N-1} \frac{1}{\sqrt{1 + 2\beta_{\text{HL}}\lambda_i}}, \\ P_2 &= (\beta_{\text{HL}}\Phi(-\beta_{\text{HL}}) - \phi(-\beta_{\text{HL}})) \left(\prod_{i=1}^{N-1} \frac{1}{\sqrt{1 + 2\beta_{\text{HL}}\lambda_i}} \right. \\ &\qquad \left. - \prod_{i=1}^{N-1} \frac{1}{\sqrt{1 + 2(\beta_{\text{HL}} + 1)\lambda_i}} \right), \\ P_2 &= (\beta_{\text{HL}} + 1)(\beta_{\text{HL}}\Phi(-\beta_{\text{HL}}) - \phi(-\beta_{\text{HL}})) \left(\prod_{i=1}^{N-1} \frac{1}{\sqrt{1 + 2\beta_{\text{HL}}\lambda_i}} \right. \\ &\qquad \left. - \text{Re} \left[\prod_{i=1}^{N-1} \frac{1}{\sqrt{1 + 2(\beta_{\text{HL}} + i)\lambda_i}} \right] \right). \end{aligned} \tag{34}$$

4.3 SQP-based RBDO approach

Recently, a FORM-based SQP approach for RBDO with SORM and MC corrections was proposed in [12]. For non-Gaussian variables, we derive the following FORM-based QP-problem in the standard normal space:

$$\begin{cases} \min\limits_{\eta_i} & f_{\text{en}}(\boldsymbol{\eta}) \\ \text{s.t.} & \begin{cases} \mu_g \leq -\beta_t \sigma_g, \\ -\epsilon \leq \eta_i \leq \epsilon, \end{cases} \end{cases} \tag{35}$$

where

$$
\begin{aligned}
f_{\text{en}}(\boldsymbol{\eta}) &= \sum_{i=1}^{N_{\text{VAR}}} \left.\frac{\partial f_{\text{en}}}{\partial X_i}\right|_{\boldsymbol{X}=\boldsymbol{\mu}^k} \frac{\phi(Y_i^k)}{\rho_i(\mu_i^k;\boldsymbol{\theta}_i^k)}\eta_i + \frac{1}{2}\sum_{i=1}^{N_{\text{VAR}}}\sum_{j=1}^{N_{\text{VAR}}} \tilde{H}_{ij}\eta_i\eta_j, \\
\tilde{H}_{ij} &= \left.\frac{\partial^2 f_{\text{en}}}{\partial X_i \partial X_j}\right|_{\boldsymbol{X}=\boldsymbol{\mu}^k} \frac{\phi(Y_i^k)}{\rho_i(\mu_i^k;\boldsymbol{\theta}_i^k)}\frac{\phi(Y_j^k)}{\rho_j(\mu_j^k;\boldsymbol{\theta}_j^k)}, \\
\mu_g &= \sum_{i=1}^{N_{\text{VAR}}} \left.\frac{\partial g_{\text{en}}}{\partial X_i}\right|_{\boldsymbol{X}=\boldsymbol{x}^{\text{MPP}}} \frac{\phi(y_i^{\text{MPP}})}{\rho_i(x_i^{\text{MPP}};\boldsymbol{\theta}_i^k)}\left(\eta_i - y_i^{\text{MPP}}\right), \\
\sigma_g &= \sqrt{\sum_{i=1}^{N_{\text{VAR}}} \left(\left.\frac{\partial g_{\text{en}}}{\partial X_i}\right|_{\boldsymbol{X}=\boldsymbol{x}^{\text{MPP}}} \frac{\phi(y_i^{\text{MPP}})}{\rho_i(x_i^{\text{MPP}};\boldsymbol{\theta}_i^k)}\right)^2}.
\end{aligned}
\qquad (36)
$$

Here, $\beta_t = \Phi^{-1}(P_s)$ is the target reliability index which can be corrected by a SORM approach as presented above or any Monte Carlo (MC) approach. The optimal solution to (35), denoted η_i^*, is mapped back from the standard normal space to the physical space using

$$
\mu_i^{k+1} \approx \mu_i^k + \frac{\Phi(Y_i^k)}{\rho_i(\mu_i^k;\boldsymbol{\theta}_i^k)}\eta_i^*.
$$

Then, a new QP-problem is generated around $\boldsymbol{\mu}^{k+1}$ and this procedure continues in sequence until convergence is obtained. The QP-problem in (35) is solved using `quadprog.m` in Matlab.

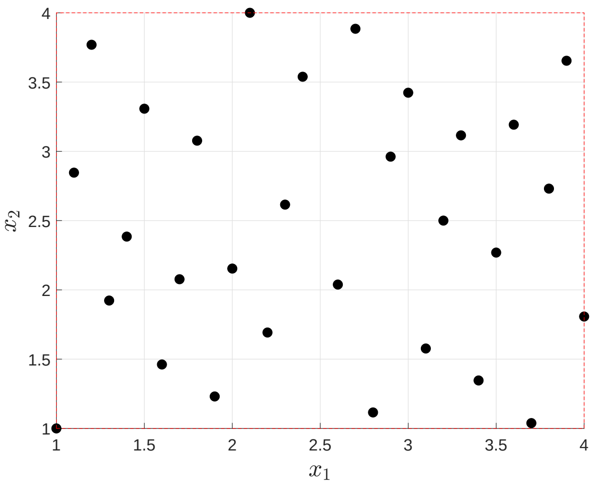


Figure 2: Halton sampling with 30 points.

5 EXAMPLES

We begin motivating the choice of convex combinations instead of affine combintions by studying the Hosaki test function, i.e.

$$
f = (1 - 8x_1 + 7x_1^2 - 7/3x_1^3 + 1/4x_1^4)x_2^2\exp(-x_2), \quad 0 \le x_i \le 5. \qquad (37)
$$

By considering (37) as a black-box, we set up Halton sampling with 30 points similar to what is used in our next example, see Figure 2, and then establish our convex combination of metamodels as well as a standard affine combination of metamodels. The two ensemble of metamodels are plotted in Figure 1. Both ensembles represent the global optimum properly. However, the boundary with values of zero is represented poorly with the affine combination. On the contrary, the convex combination also performs well for this case. This latter case could of course be a limit surface and it is then clear that our convex combination will represent the reliability much better than the affine combination for a limit surface like this.

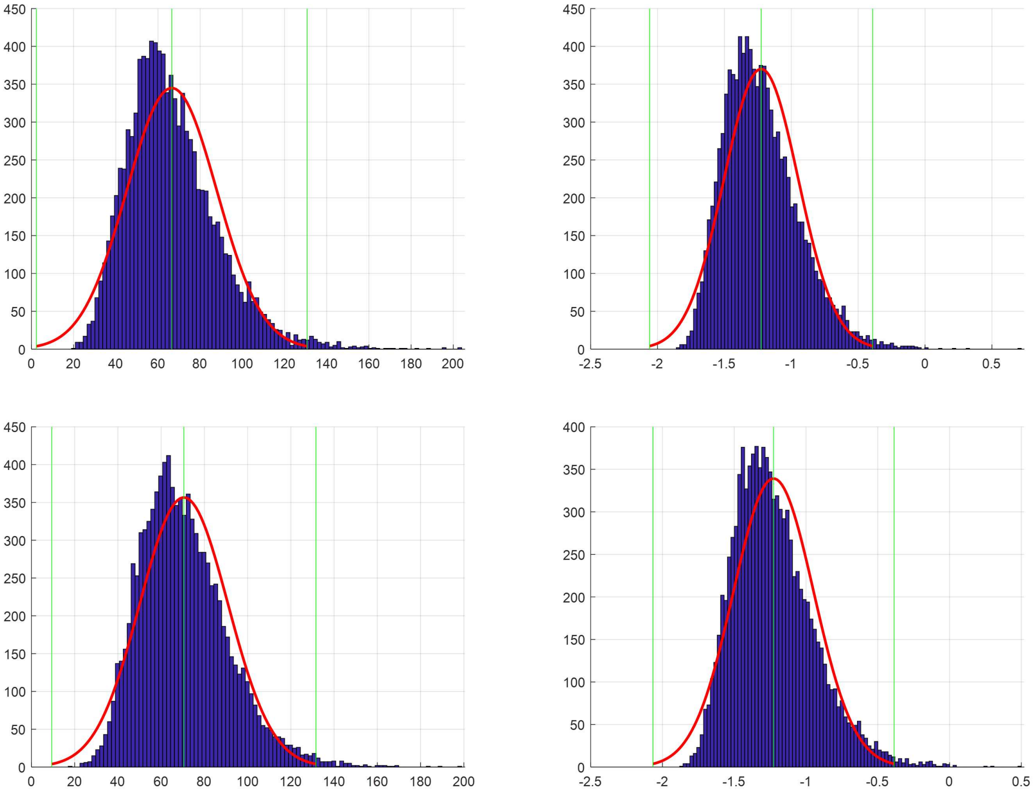


Figure 3: Histograms of the objectives and the constraints. The first row contains histograms for the analytical expressions in (38). The second row shows histograms of the OEMs.

Next, we will demonstrate by solving

$$\begin{cases} \min\limits_{\mu_i} & \sqrt{1000\left(\dfrac{4}{\mu_1}-2\right)^2+1000\left(\dfrac{4}{\mu_1}-2\right)^2} \\ \text{s.t.} & \begin{cases} \Pr[(X_1-0.5)^4+(X_2-0.5)^4 \leq 2] \geq P_s, \\ 1 \leq \mu_i \leq 4, \end{cases} \end{cases} \tag{38}$$

that the convex combinations of metamodels represent the objective as well as the limit surface most properly such that our SQP-based RBDO methodology can be applied with most satisfying results. We let $P_s = 0.999$ and $\mathrm{VAR}[X_i] = 0.1^2$. The deterministic solution is (1.5,1.5) and the minimum of the unconstrained objective function is found at (2,2). The solution to (38) obtained by our SQP-based RBDO approach is (1.2705,1.2705). The corresponding reliability is 99.9$\%$. The problem was considered in [2], where SLP-based RBDO was performed by adopting radial basis function networks as metamodels, and, in [4], it was studied by using support vector machines.

The example in (38) is now considered to be a black-box which we treat by applying design of experiments and our approach of convex OEM. Thus, first, we set up a DoE by using Halton sampling with 30 points according to Figure 2, secondly, we establish our metamodels presented in section 2, thirdly, we establish convex combinations of these metamodels for the objective function and the constraint by solving (26), and, finally, we solve (31) by applying our SQP-based RBDO approach. The solution is (1.2715,1.2694) and the corresponding reliability for the black-box in (38) is 99.9%. In conclusion, the metamodel-based solution is most accurate. This is also demonstrated in Figure 3, where the histograms of the objective function and the constraint are compared for the analytical expressions and the ensembles of metamodels.

6 CONCLUDING REMARKS

In this work we propose to use convex combinations of metamodels as the ensemble of metamodels for reliability-based design optimization. The OEM is a convex combination of quadratic regression models, Kriging, radial basis function networks, polynomial chaos expasion and support vector regression. The performance of the OEM-based RBDO approach is excellent.

REFERENCES

[1] M.A. Valdebenito & G.I. Schuëller, A Survey on Approaches for Reliabillity-Based Optimization, *Structural and Multidisciplinary Optimization*, **42**, 645–663, 2010.

[2] N. Strömberg, Reliability Based Design Optimization by using a SLP Approach and Radial Basis Function Networks, in the proceedings of *the ASME 2016 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference IDETC/CIE*, Charlotte, North Carolina, USA, August 21—24, 2016.

[3] K. Amouzgar & N. Strömberg, Radial Basis Functions as Surrogate Models with A Priori Bias in Comparison with a Posteriori Bias, *Structural and Multidisciplinary Optimization*, **55**, 1453–1469, 2017.

[4] N. Strömberg, Reliability-based Design Optimization by using Support Vector Machines, in the proceedings of *ESREL - European Safety and Reliability Conference*, June 17–21, Trondheim, Norway, 2018.

[5] N. Strömberg, Reliability Based Design Optimization by using Metamodels, in the proceedings of *the 6th International Conference on Engineering Optimization*, September 17–19, Lisboa, Portugal, 2018.

[6] T. Goel, R.T. Haftka, W. Shyy & N.V. Queipo, Ensemble of surrogates, *Structural and Multidisciplinary Optimization*, **33**, 199–216, 2007.

[7] E. Acar, Varius approaches for constructing an ensemble of metamodels using local measures, *Structural and Multidisciplinary Optimization*, **42**, 879–896, 2010.

[8] X.J. Zhou, Y. Zhong Ma & X. Fang Li, Ensembles of surrogates with recursive arithmetic avarage, *Structural and Multidisciplinary Optimization*, **44**, 651–671, 2011.

[9] Y. Lee & D.-H. Choi, Pointwise ensemble of meta-models using ν nearest points cross-validation, *Structural and Multidisciplinary Optimization*, **50**, 383–394, 2014.

[10] R. Shi, L. Liu, T. Long & J. Liu, An efficient ensemble of radial basis functions method based on quadratic programming, *Engineering Optimization*, **48**, 1202–1225, 2016.

[11] X. Song, L. Lv, J. Li, W. Sun & J. Zhang, An advanced and robust ensemble surrogate model: extended adaptive hybrid functions, *Journal of Mechanical Design*, **140**, 2018.

[12] N. Strömberg, Reliability-based Design Optimization using SORM and SQP, *Structural and Multidisciplinary Optimization*, **56**, 631-–645, 2017.

[13] A. Hasofer & N. Lind, Exact and Invariant Second Moment Code Format, *Journal of the Mecanics Division, ASCE*, **100**, 111-121, 1974.

[14] L. Tvedt, Two Second Order Approximations to the Failure Probability, *Technical report: RDIV/20-004-83*, Det Norske Veritas, 1983.

UNCECOMP 2019
3^{rd} ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24-26 June 2019

BLACK-BOX PROPAGATION OF FAILURE PROBABILITIES UNDER EPISTEMIC UNCERTAINTY

Marco De Angelis[1], Scott Ferson[1], Edoardo Patelli [1], and Vladik Kreinovich[2]

[1]Institute for Risk and Uncertainty, School of Engineering, University of Liverpool
Liverpool, L69 7ZF
marco.de-angelis@liverpool.ac.uk

[2] Computer Science, University of Texas at El Paso
El Paso, TX 79968, USA
vladik@utep.edu

Keywords: Black-box code, Cauchy-deviate method, Line sampling, Digital Twins

Abstract. *In engineering simulation a black-box code is often a complex, legacy or proprietary (secret) black-box software used to describe the physics of the system under study. Strategies to propagate epistemic uncertainty through such codes are desperately needed, for code verification, sensitivity, and validation on experimental data. Very often in practice, the uncertainty in the inputs is characterised by* imprecise *probability distributions or distributions with interval parameters, also known as probability boxes. In this paper we propose a strategy based on line sampling to propagate both aleatory and epistemic uncertainty through black-box codes to obtain interval probabilities of failure. The efficiency of the proposed strategy is demonstrated on the NASA LaRC UQ problem.*

1 INTRODUCTION

Digital twins seem to be the emerging modelling paradigm for industrial system simulation. According to ANSYS [1], a digital twin is "a complete virtual prototype of an entire system, a working system in a real world environment, a complex system integrating multiple engineering disciplines, requiring system-level simulation knowledge". ANSYS also state that digital twins " represent a new era in simulation, a new world of predictability, a new tool for engineering the future".

The rise of digital twins is justified by the progressive increase of high-fidelity methods (e.g. finite element and computational fluid dynamic) and the fast-paced growth of the computing power that has led to the solution of unprecedentedly complex models, for ever more realistic boundary conditions. The excitement about this new era of simulation must face the truth about our limitation of modeling the physics around us. Deterministic models are rarely suited to describe in detail the multifaceted reality of a system, and usually the more detailed the model the more sensitive it is to variations and uncertainty. Comparing predicted responses with measured data, however, does not generally show that the fidelity has improved as much as our ability of making more detailed models and accurate analyses.

The reason for this discrepancy is often the presence of uncertainties, for instance in the parameters of the model, which are not precisely known and must be expected to deviate from the assumed deterministic values. Another source of uncertainty is in the mathematical model, which usually involves some abstraction and simplifying assumptions to represent the actual mechanical/physical response. Given the limitations of data, quantification methods often rely on subjective judgement and assumptions and it may not always seem reasonable to characterize the uncertainties in a classical probabilistic way. To avoid the inclusion of subjective and often unjustified hypothesis, the imprecision and vagueness of the data can be treated by using generalized probabilistic methods.

The unavoidable uncertainties must be explicitly included in the computations to guarantee that the components or systems will continue to perform satisfactorily despite variability and precise models. If the effect of uncertainties in the optimized design is ignored, this design may perform unsatisfactorily in realistic conditions. Resilient/reliable systems are less sensitive to the uncertainties and hence, they reach low variability of the overall performance allowing for significant reductions in terms of e.g. the manufacturing and operating costs.

Quantifying the effect of the uncertainty is a necessary step to support decision makers. For instance, the analyst can estimate the importance of collecting additional information and identify the parameters that contribute the most to the variability of the output. One of the most important analyses is to identify the extreme response performances of the system. It is also important to determine the combination of input parameter values that causes a performance metric of interest to reach its extreme. Knowing such conditions, it might be possible to prevent those extreme performances or mitigate their consequences.

The need for efficient and robust uncertainty propagation on black-box models has been shown by the NASA Langley Uncertainty Quantification Challenge [2]. The only information that was released about the model is that it described the flight of a remotely controlled twin-jet aircraft pushed to the edge of the flight envelope, thus subject to strong uncertainties.

In this paper, a novel approach for black box failure probability propagation analysis is presented and discussed. The theoretical framework of imprecise probability is used for the representation of the uncertainty. An efficient and general computational framework based on Line Sampling simulation is proposed to perform reliability analysis [3]. The Cauchy-deviate method is used for the propagation of epistemic uncertainty [4]. We show that these two approaches can be combined to obtain interval failure probability of the aircraft performance. We show the applicability and efficiency of the methodology on the NASA Langley UQ black-box model.

2 BLACK-BOX CODES

In uncertainty quantification distinguishing between black-box models and open-source models is consequential. The research community is not united on the definition of black boxes. Researchers in machine learning often refer to black boxes as deep learning models, which are practically impenetrable due to their complexity. A more general definition of black-box code can be found in [4]. In this paper a black-box model is a quantitative model, whose source code is inaccessible.

2.1 Why such models exist

The reason why this kind of model exists can be ascribed to (i) secrecy: for example, commercial companies release to the open market only the software binaries, or the code is encrypted for verification and validation by third parties; (ii) legacy: for example, the source code is partially or totally lost and only the binaries are available, or the source code is available but does not build on the current compiler; (iii) complexity: for example, deep learning models, and large FEM and CFD models.

2.2 Need to distinguish black-box and open-source codes

In automatic uncertainty propagation the source code can be decomposed into a list of basic binary operations. This turns out to be the key for efficient rigorous propagation, as numerical strategies for interval analysis can be deployed.

2.3 Restrictions introduced by black boxes

Uncertainty quantification on black-box models is particularly challenging. This is because the model can only be queried and often the time required by a single evaluation is very long, in some cases ranging from hours to weeks.

Furthermore, when the user does not completely know the origin of the model, more evaluations are usually needed to characterise the mathematical properties of the model. For example, checking that the function is continuous or smooth may already require a significant amount of evaluations. Nevertheless, even if we know that the function is e.g. smooth, computing the partial derivatives may be computation expensive. This fact rules out the efficient use of differential algebra, local Taylor expansions, and monotonicity checks that may be alternatively possible in the case of open-source code. Another big limitation introduced by black-box codes is the difficulty of rigorous propagation of epistemic uncertainty. The rigorous propagation is only possible by means of intrusive interval analysis.

2.3.1 What can be done?

We can establish prove that the function in a black-box model is deterministic, by checking whether the outputs are different on a number of repeated computer experiments. If the black-box model is non-deterministic and single evaluations are computation expensive, then rigorous uncertainty quantification may not be practically possible. Techniques that combine surrogate modelling, and massive high performance parallel and distributed computing, may be the answer. If the model is deterministic, the aforementioned techniques may still be necessary, but it is also possible to better characterise the behaviour of the black box and deploy sampling schemes and accelerating strategies to achieve efficiency.

3 PROBLEM STATEMENT

A deterministic model can be conceptually represented in the functional form

$$\boldsymbol{y} = f(\boldsymbol{x}) \tag{1}$$

where f is a function that maps $\boldsymbol{x}$ to $\boldsymbol{y}$ [5]. The variable $\boldsymbol{x} = [x_1, ..., x_{N_X}]$ is a vector of *real* valued inputs, and $\boldsymbol{y} = [y_1, ..., y_{N_Y}]$ is a vector of real-valued model outputs. In this paper we will restrict the discussion to $N_Y = 1$, so for simplicity we assign $N = N_X$.

The uncertainty about the elements of $\boldsymbol{x}$ will be represented by a sequence of *imprecise* distributions $\mathcal{D}_1, ..., \mathcal{D}_D$, where $\mathcal{D}_j$ characterises the uncertainty associated with the element x_j of $\boldsymbol{x}$. Various correlations and other restrictions involving the elements of $\boldsymbol{x}$ may be specified. Typically, these distributions are obtained through some form of expert elicitation or expert review process.

3.1 Uncertainties

The *imprecise distribution* $\mathcal{D}_j$ fully characterises both the aleatory and epistemic uncertainty of the element x_j. The probability box or simply *p-box* [$\overline{\mathcal{D}}$, $\underline{\mathcal{D}}$] denote the set of all non-decreasing functions $\mathcal{D}$ from the real line into $[0, 1]$, such that $\underline{\mathcal{D}}(x) \leq \mathcal{D}(x) \leq \overline{\mathcal{D}}(x)$ [6]. Eq. 2 summarises in one formula the latter sentence.

$$\mathcal{D} : \mathbb{R} \rightarrow [0, 1], \quad \underline{\mathcal{D}}(x) \leq \mathcal{D}(x) \leq \overline{\mathcal{D}}(x) \tag{2}$$

So if [$\overline{\mathcal{D}}$, $\underline{\mathcal{D}}$] is a *p-box* for a random variable X whose distribution $\mathcal{D}$ is unknown except that is within the *p-box*, then $\overline{\mathcal{D}}$ and $\underline{\mathcal{D}}$ are respectively lower and upper bounds on $\mathcal{D}(x)$, which is the – imprecisely known – probability that the random variable X is smaller than x. In practical applications it is very common to construct p-boxes using known probability distributions with interval parameters. In these cases the assumption on the probability distribution type may be relaxed to include all the distributions that fall within the bounds.

3.2 Failure probability

The probability that the model output y is greater than a given, yet uncertain, threshold can be expressed as

$$P(Y > \tilde{y}) = P(f(\boldsymbol{X}) > \tilde{y}) \tag{3}$$

Often the threshold $\tilde{y}$ represents a given performance limit that the model under study should be in, with a large margin of safety. The more catastrophic are the consequences associated with exceeding the given threshold, the larger the safety margin, therefore the smaller is the failure probability.

The aim of this paper is to present a sampling strategy to compute small failure probability bounds on Y when continuous imprecise probability distributions are defined for the input variables of the black-box model.

3.3 Sampling

The probability of Eq. 3 can be approximated via sampling. The solution of the multi-dimensional integral of Eq. 4 can be obtained by averaging over the generated samples so to avoid the numerical integration.

Let $\mathcal{C}_{\boldsymbol{x}} : [0,1]^N \to [0,1]$ be the copula function of the vector of uncertain variables, and let $g(\boldsymbol{x}) = \tilde{y} - f(\boldsymbol{x})$, with $\Omega_F = g(\boldsymbol{x}) < 0$ denote the failure domain.

The probability of failure can be given as:

$$P\left(g(\boldsymbol{x}) < 0\right) = \int_{\Omega_F} \mathrm{d}\mathcal{C}_{\boldsymbol{x}} \tag{4}$$

Where, the copula expresses the aleatory dependence between the p-boxes, and can be used to represent without loss of generality, the uncertainty about the model in its entirety. Clearly, this is restricted to the case of precise aleatory dependence among the variables. The dependence among focal elements, also known as epistemic dependence, will not be treated in this paper.

In this particular formulation of the uncertainty model, every draw corresponds to a *focal element*, which is the interval counterpart of a pointwise sample. We use the *inverse transformation method* to generate focal elements from the copula function [7]. Focal elements $[\boldsymbol{x}]^{\{s\}} = [\underline{\boldsymbol{x}},\ \overline{\boldsymbol{x}}]^{\{s\}}$ are propagated through the model solving the following two problems:

$$\min_{\boldsymbol{x}\in[\boldsymbol{x}]^{\{s\}}} g([\boldsymbol{x}]^{\{s\}}), \qquad \max_{\boldsymbol{x}\in[\boldsymbol{x}]^{\{s\}}} g([\boldsymbol{x}]^{\{s\}}). \tag{5}$$

Let us define the set A of all the boxes contained in Ω_F, and the set B^c of all the boxes strictly not contained in Ω_F. Let B be the complement of B^c, then the two sets are

$$A = \left\{[\boldsymbol{x}] : [\boldsymbol{x}] \subseteq \Omega_F\right\}; \quad B = \left\{[\boldsymbol{x}] : [\boldsymbol{x}] \cap \Omega_F \neq \varnothing\right\} \tag{6}$$

Let us define the characteristic function χ_A for a set A as

$$\chi_A(\boldsymbol{e}_i) = \begin{cases} 1 & \text{if } \boldsymbol{e}_i \in A \\ 0 & \text{if } \boldsymbol{e}_i \notin A \end{cases} \tag{7}$$

where $\boldsymbol{e}_i$ is a box in Ω. Lower and upper bounds on the failure probability are obtained averaging the number of focal elements classified as in Eq. 6.

$$\overline{\hat{p}_F} = \sum_{s=1}^{N} \chi_A\left([\boldsymbol{x}]^{\{s\}} \subseteq \Omega_F\right), \qquad \underline{\hat{p}_F} = \sum_{s=1}^{N} \chi_B\left([\boldsymbol{x}]^{\{s\}} \cap \Omega_F \neq \varnothing\right). \tag{8}$$

4 LINE SAMPLING

Line sampling is a simulation method primarily developed to efficiently compute small failure probabilities for high dimensional problems [3]. Line sampling has been recently extended to deal with epistemic uncertainty in the form of intervals [8]. The method is generally applicable and it only requires the knowledge of the so-called "important direction", $a \in \mathrm{R}^N$, which can be any vector pointing towards the failure region. In this paper we use the line sampling scheme to produce *focal elements*. On each focal element we run the problem of Eq. 5 to solve the classification of Eq. 6.

5 PROPAGATION OF EPISTEMIC UNCERTAINTY

5.1 Why Monte Carlo is bad for epistemic propagation

Monte Carlo simulation works particularly well with aleatory uncertainty. This is because Monte Carlo is insensitive to the number of random variables. For example, the estimation of $E\left(f(\boldsymbol{x})\right)$, where E is the mean operator, can be done efficiently by Monte Carlo. However, this is quite the opposite in epistemic uncertainty, where the accuracy of the simulation method drastically decreases with the number of epistemic variables. Consider this simple example: we want to sum a number N of epistemic variables $x_{1:N}$ in the box $[a, b]^N$, and compute the bounds of the resulting *sum*. The exact bounds on the *sum* can be computed exactly and are

$$f(x) = \sum_{i=1:N} x_i = [N * a, N * b] = N * [a, b].$$

Now, let us pretend that the function $f(x)$ is a black-box model, so we do not know that behind the model is doing a simple *sum*. A Monte Carlo way to approach the problem consists in (i) generating random samples in $[a, b]^N$, (ii) summing the generated samples, (iii) getting the minimum and maximum of the *sum*. This process gets increasingly less accurate as the number of variables increases. In fact, for the central limit theorem, the obtained sum will approximate a Gaussian distribution, with mean μ and variance σ^2/N, where σ^2 is the sample variance. Given that the variance of the *sum* linearly decreases with the number of variables, it gets ever more improbable to sample close to the endpoints of the box where the exact bounds hold. Note that this applies to any probability distribution within the box $[a, b]^N$, because of the generality of the central limit theorem.

5.2 Cauchy-deviate method

The Cauchy-deviate method propagates epistemic uncertainty using sampling. [4]. The method exploits the properties of the Cauchy distribution, by which a linear combination of Cauchy random variates is also Cauchy distributed. The properties of the Cauchy distribution ensure that the samples do not get trapped around the arithmetic mean of the generated sample set. Thus the model can be treated as a black box. The method works particularly well when the intervals are small or the black-box is approximately monotonic over the interval.

6 THE NASA BLACK-BOX MODEL

The NASA Langley multidisciplinary uncertainty quantification (UQ) challenge was released in 2013 to seek responses from practitioners in the filed of UQ. Among the different challenge problems, NASA was seeking responses pertaining the *propagation of mixed aleatory and epistemic uncertainties through system models*. The challenge page quotes:

> "*NASA missions often involve the development of new vehicles and systems that must be designed to operate in harsh domains with a wide array of operating conditions. These missions involve high-consequence and safety-critical systems for which quantitative data is either very sparse or prohibitively expensive to collect. Limited heritage data may exist, but is also usually sparse and may not be directly applicable to the system of interest, making uncertainty quantification extremely challenging. NASA modeling and simulation standards require estimates of uncertainty and descriptions of any processes used to obtain these estimates.*" NASA LaRC UQ Challenge 2014.

The challenge problem was based upon a model of the NASA Langley Generic Transport Model (GTM). The GTM is a 5.5% dynamically scaled, remotely piloted, twin-turbine, research aircraft used to conduct experiments for the NASA Aviation Safety Program. The multidisciplinary character of the proposed problems led the challenge scientists to release the model in the form of a *black-box*. Again quoting the challenge page:

> "*Although a discipline-specific application is the focus of this challenge problem,* ***the problem was specifically structured so that specialized aircraft knowledge is not required****. We seek responses from all interested parties not only those with aircraft experience.*"

Five out-of six challenge problems involved solving a forward propagation problem with mixed aleatory and epistemic uncertainty. The epistemic uncertainty was expressed in the form of intervals, while the aleatory uncertainty in the form of beta and normal distributions.

6.1 The challenge problem

In this section we analyze a portion of the black-box model provided by the NASA Langley Research Center. The full model contains twenty-one input parameters and eight outputs, nonetheless we will limit our discussion to only the first five input parameters $\boldsymbol{p} = [p_1, p_2, p_3, p_4, p_5]$, and the first output $y = y_1$. In the original text of the challenge manifesto this output was referred to as x.

NASA provided the software binaries to evaluate h:

$$y = h(\boldsymbol{p}) = h(p_1.p_2, p_3, p_4, p_5), \tag{9}$$

Where $\boldsymbol{p}_{1:3} \in [0, 1]^3$, $\boldsymbol{p}_{4,5} \in \mathbb{R}$. Specific information about these parameters are provided in Table 1. We provide a solution to the following problem:

$$P\left(h(\boldsymbol{p}) > 0.39\right) \tag{10}$$

Or equivalently P $(g(\boldsymbol{p}) < 0)$, with $g(\boldsymbol{p}) = 0.39 - h(\boldsymbol{p})$. The problem of Eq.10 in words is: "What is the probability of $y = h(\boldsymbol{p})$ being greater than 0.39, when the uncertainty about $\boldsymbol{p}$ is given in Table 1?"

6.2 Input variables

The input variables of the problem are shown in Table 1, and are classified into Category I, II, and III.

- Category I represents random variables with a precise distributions; these variables only have aleatory component.
- Category II represents intervals, any value within the endpoints of the interval is allowed. Although interval is a pure epistemic way to characterize uncertainty, intervals do have an aleatory component. In fact, not only any value but also any distribution bounded by the endpoints of the interval is allowed.
- Category III is the more general representation of uncertainty. Dempster-Shafer structures and p-boxes fall into this category.

Table 1: Uncertain input parameters

Variable	Category	Uncertainty	Epistemic component	Marginal distribution
p_1	III	P-box	$\mu_1 = [3/5,\ 4/5]$	$B(a_1, b_1)$
		Unimodal Beta	$\sigma_1^2 = [1/50,\ 1/25]$	independent
p_2	II	Interval	$\Delta_2 = [0,\ 1]$	–
p_3	I	Random variable	–	$U(0, 1)$
p_4, p_5	III	P-box	$\mu_4 = [-5,\ 5]$	$N(\mu_4, \sigma_4^2)$
		Normal bivariate	$\sigma_4^2 = [1/400,\ 4]$	dependent on p_5
			$\mu_5 = [-5,\ 5]$	$N(\mu_5, \sigma_5^2)$
			$\sigma_5^2 = [1/400,\ 4]$	dependent on p_4
		MN($\mu_{4:5}$, $\sigma_{4:5}$, $\rho_{4:5}$)	$\rho_{4,5} = [-1,\ 1]$	

7 Results

A preliminary analysis is run with 1000 samples. The model takes approximately $30s$ to produce a thousand samples on a common desktop computer.

The 1000 focal elements are propagated through the model and the classification of Figure 1 is obtained. This preliminary classification shows clear borders between the three states, namely *safe*, *plausibility*, and *belief*. In Figure 1, the set of dots that are both red and black belong to the first class of Eq.8, i.e. the focal elements that contribute to the upper bound of the failure probability $\overline{p_F}$; while the black dots only contribute to the lower bound.

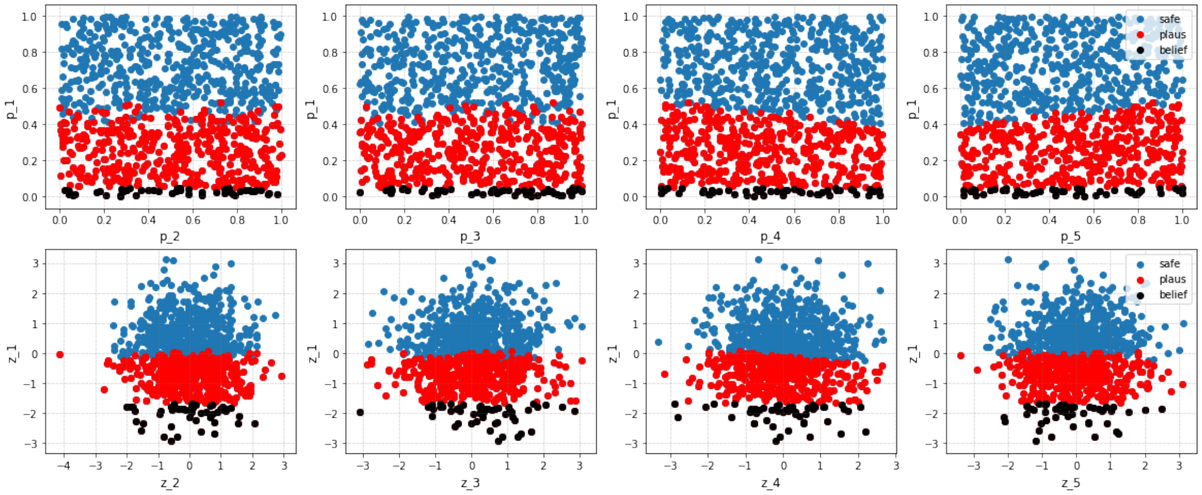


Figure 1: Focal elements p_1 v. $\boldsymbol{p}_{2:5}$ in the copula space (top row) and in the standard normal space (bottom row)

If we look at the scatter plot of classified focal elements for the remaining variables as shown in Figure 5, Figure 6, and Figure 2 we note that there is not a clear border as in the previous case. This suggests that variable p_1 is predominant with respect to the other variables. It is also to note the characteristic cross-shaped dependence between variable p_4 and p_5 depicted in Figure 2 that is typical of the case of unknown linear correlation as specified in Table 1.

With the preliminary analysis it was possible to identify the following important direction α.

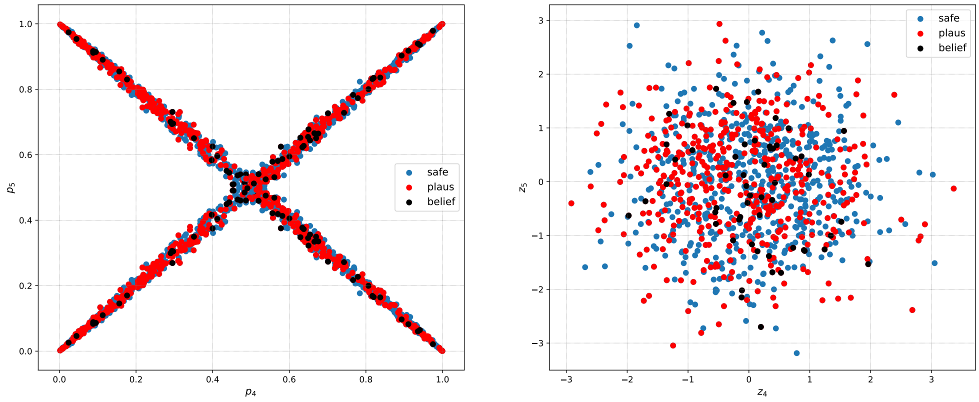


Figure 2: Classification of focal elements for variables p_4 v. p_5 in the copula (top) and the standard normal space (bottom)

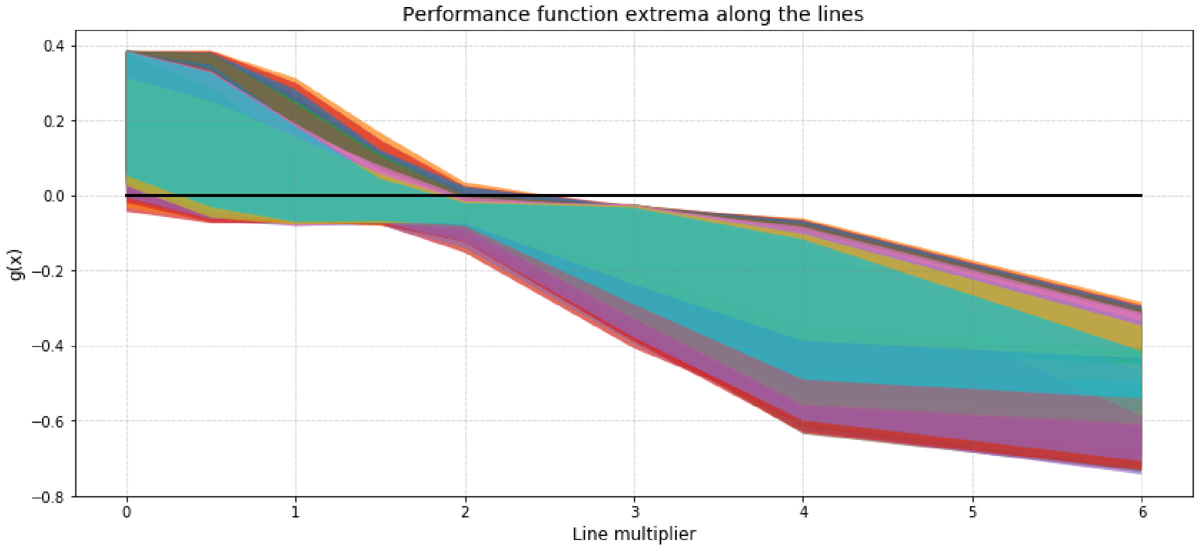


Figure 3: Performance function envelope on the line, each line displays a different color.

$$\alpha = [-0.987, -0.0087, 0.162, 0.0145, 0.00028]$$

The analysis conducted with 100 lines and a total of 500 focal elements, with the threshold of Eq. 10, led to the failure probability interval $[\hat{p}_F] = [0.0378, 0.44]$, with no update in direction. The same analysis run on the set of thresholds $t = [0.4, 0.5, 0.6, 0.7, 0.8]$ led to the upper and lower fragility curves shown in Figure 4.

t	0.4	0.5	0.6	0.7	0.8
$\overline{p_F}$	4.38E-1	1.71E-2	4.70E-3	1.06E-3	3.26E-4
$\underline{p_F}$	3.06E-2	1.17E-4	6.57E-6	3.94E-7	3.79E-8

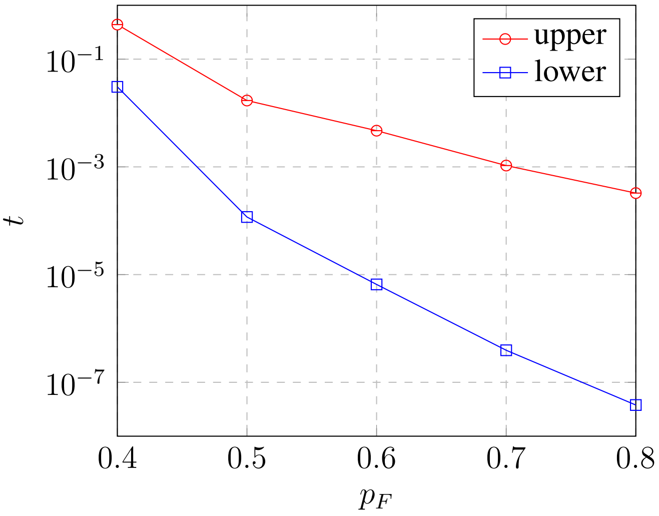


Figure 4: Upper and lower fragility curves

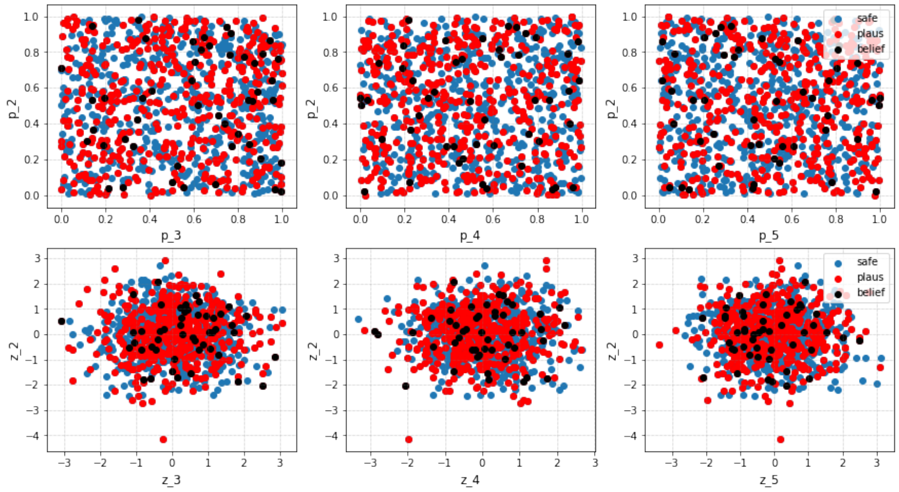


Figure 5: Classification of focal elements for variables p_2 v. $\boldsymbol{p}_{3:5}$

Acknowledgement

This research is funded by the Engineering & Physical Sciences Research Council (EPSRC) through the programme grant “Digital twins for improved dynamic design” with grant no. EP/R006768/1. EPSRC and the DigiTwin consortium are greatly acknowledged for their funding and support.

REFERENCES

[1] Ansys Inc. Delivering a digital twin. Technical report, Ansys Advantage, Issue 2, 2017.

[2] L.G. Crespo and S.P. Kenny. Special edition on uncertainty quantification of the aiaa journal of aerospace computing, information, and communication. *Journal of Aerospace Information Systems*, 12(1):9, 2015.

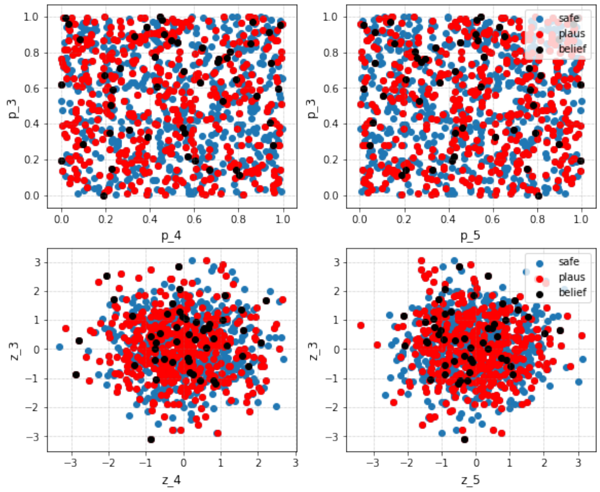


Figure 6: Classification of focal elements for variables p_3 v. $\boldsymbol{p}_{4:5}$

[3] M. de Angelis, E. Patelli, and M. Beer. Advanced line sampling for efficient robust reliability analysis. *Structural Safety*, 52(PB):170–182, 2015.

[4] V. Kreinovich and S.A. Ferson. A new cauchy-based black-box technique for uncertainty in risk analysis. *Reliability Engineering and System Safety*, 85(1-3):267–279, 2004.

[5] J.C. Helton, J.D. Johnson, W.L. Oberkampf, and C.B. Storlie. A sampling-based computational strategy for the representation of epistemic uncertainty in model predictions with evidence theory. *Computer Methods in Applied Mechanics and Engineering*, 196(37):3980 – 3998, 2007. Special Issue Honoring the 80th Birthday of Professor Ivo Babuka.

[6] S. Ferson, V. Kreinovich, L. Grinzburg, D. Myers, and K. Sentz. Constructing probability boxes and dempster-shafer structures. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2015.

[7] E. Patelli, D.A. Alvarez, M. Broggi, and M. De Angelis. Uncertainty management in multidisciplinary design of critical safety systems. *Journal of Aerospace Information Systems*, 12(1):140–169, 2015. cited By 19.

[8] E. Patelli and M. De Angelis. Line sampling approach for extreme case analysis in presence of aleatory and epistemic uncertainties. pages 2585–2593, 2015.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24—26 June 2019

META-MODELS FOR RANDOM SIGNAL ANALYSIS

Christian Bucher[1]

[1]Technische Universität Wien
Karlsplatz 13, A-1040 Wien, Austria
e-mail: christian.bucher@tuwien.ac.at

Keywords: Meta-modeling, signal analysis, time series, response surface, random process, optimization, parameter identification.

Abstract. *In time series analysis (e.g. in structural dynamics) one frequently encounters a situation that parts of a signal are either missing or they are distorted by an unacceptably high level of random noise. In such a situation it is desirable to reconstruct the missing or un-usable parts by a procedure which does not necessarily introduce a large amount of further uncertainties. Also, the signals may substantially depend on a set of parameters (e.g. system properties) and these signals have typically been measured only for a limited set of parameter values. For further application, however, one would like to predict the signals for a different set of possibly random parameter values. Based on such a signal reconstruction it is possible to solve optimization problems, e.g. carry out parameter identification with high numerical efficiency. This is especially important if the underlying problem is ill-conditioned and possesses multiple local minima. In such cases, relatively expensive global optimizers (such as e.g. Genetic Algorithms) can be used advantageously. The feasibility of such an approach heavily depends on the accuracy and the computational speed of the meta-models being used. The paper discusses some possible formalisms underlying meta-models for time series and demonstrates their usefulness by several numerical and structural applications.*

1 INTRODUCTION

The application of meta-modeling techniques for the prediction of the behavior of complex systems under varying external and internal conditions has had a long history of development. In the area of structural reliability, this type of approach has frequently been termed *Response Surface Method* [1, 2]. Essentially, the underlying concept follows the strategy that first a predetermined set on analyses with different governing parameters is carried out (the so-called *Design of Experiments, DOE*) and then the behavior in the required setting is computed by rather simple approximation (usually interpolation) formulas. This strategy shifts the main computational burden to the DOE phase and thus in typical application reduces overall computational effort substantially.

The main issue of such a meta-modeling approach lies in the question of accuracy and error estimation. Some of the most fundamental points are outlined in [3]. Some further thoughts on assessing the quality of a meta-model have been discussed in [4]. A first and widely used quality measure is the coefficient of determination (R^2). This quantity measures the correlation between the actual data Y and the model predictions Z:

$$R^2 = \left(\frac{\mathbf{E}[(Y - \bar{Y}) \cdot (Z - \bar{Z})]}{\sigma_Y \sigma_Z} \right)^2 = \rho_{YZ}^2; \; Z = \sum_{i=1}^{n} p_i g_i(X) \tag{1}$$

One well-known problem with this measure is that the CoD may be high due to overfitting (which eventually leads to bad prediction behavior). If an additional test data set T is available, then a true measure for the prediction quality can be computed, which is herein called *Coefficient of Quality* (CoQ, Nash-Sutcliffe Efficiency [5])

$$CoQ = 1 - \frac{\sum_{k=1}^{m} \left(T^{(k)} - Z_T^{(k)} \right)^2}{\sum_{k=1}^{m} \left(T^{(k)} - \bar{T} \right)^2}; \; Z_T = \sum_{i=1}^{n} p_i g_i(X_T); \quad CoQ \leq 1 \tag{2}$$

In practical application it is useful to randomly split data into training set/test set and repeat the computation of the Coefficient of Quality several times to obtain a stable statistical estimator of CoQ. This strategy allows to assess the variability of the CoQ-value at the same time such that confidence intervals can be provided.

It should be noted that the CoQ is a global measure based on second-order statistics and should not be misconstrued as a valid measure for local errors of the response surface model.

2 THEORECTICAL CONSIDERATION

Two specific, and essentially quite different, tasks will be considered in the present analysis. These tasks focus on

- Filling gaps in a stationary time series
- Identify (system) parameter values from realizations of a space-time field

Apparently, as the tasks are different, it is reasonable to follow different approaches.

In the first case, the analysis establishes a recurrence relation between the signal values over time. This recurrence is then extended to cover those time intervals from which actual data are missing. Thus, a meta-model is used to fill the gaps.

In the second case, it is assumed that a data base of signals depending on a set of system parameters is available (a so-called Design of Experiments, DOE). Based on this DOE, a meta-model relating the system parameters to the signals is established. For a new signal (whose system parameters are not known), the meta-model can be used to inversely compute the system parameters specific to this new signal.

2.1 Time series with missing data points

We consider filling gaps in a discrete time time series y_k in which several data points are missing (gaps or un-usable data). A simple case in sketched in Fig. 1.

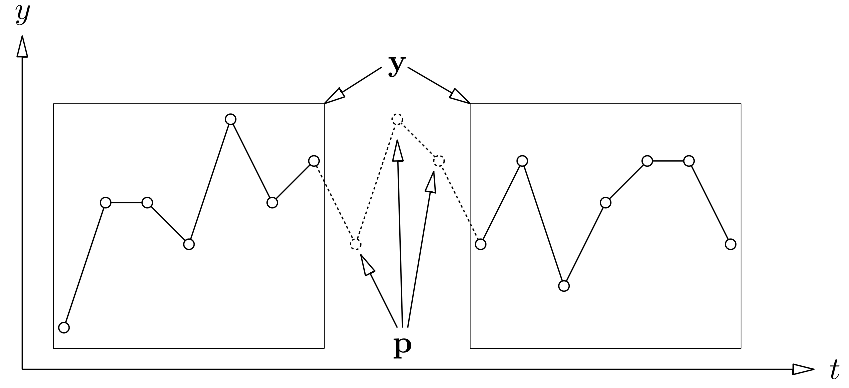


Figure 1: Time series with missing data points

If the number of missing points is small in comparison to the number of actually available data then it is reasonable to assume that the missing data should somehow be "similar" to the available ones. Such a similarity arises naturally when solving differential equations in a time-discrete manner. For homogeneous differential equations, this solution can easily be arranged into a recursive (auto-regressive, AR) scheme in which future values are computed by appropriate combinations of present and past values. In the case of linear equations, these combinations are linear as well. In this sense, we follow this idea and thus we can design the following procedure:

- collect all N known data points into vector $\mathbf{y}$
- collect all M unknown data points (in the beginning freely assumed values) into the vector $\mathbf{p}$
- Sort both vectors together into a vector $\mathbf{z}$ of size $N + M$ corresponding to strictly increasing time values
- Create Hankel matrix $\mathbf{Z}$ of size $(N - L) \cdot L$ containing time shifted values (L represents the assumed order of the AR-model)

$$\mathbf{Z} = \begin{bmatrix} z_1 & z_2 & \dots & z_L \\ z_2 & z_3 & \dots & z_{L+1} \\ \vdots & & & \\ z_{N-L} & z_{N-L+1} & \dots & z_N \end{bmatrix} \tag{3}$$

- Establish a formal linear relation between the present values and the past values

  $$\mathbf{z} = \mathbf{Z}\mathbf{a} \tag{4}$$

  containing an unknown parameter vector $\mathbf{a}$ of size M to be determined by a least-squares fit

  $$\mathbf{a} = (\mathbf{Z}^T\mathbf{Z})^{-1}(\mathbf{Z}^T\mathbf{z}) \tag{5}$$

- Compute the regression values

  $$\hat{\mathbf{z}} = \mathbf{Z}\mathbf{a} \tag{6}$$

  This actually means that the entire time series can be written as an autoregressive (AR) process without any external input.

- Compute the error

  $$E^2 = \sum_{k=1}^{N} (\hat{z}_k - z_k)^2 \tag{7}$$

- Minimize this error term with respect to the numerical values of the components of the vector $\mathbf{p}$.

2.2 Space-time fields

Here we consider a scalar- or vector-valued random field $H(\mathbf{x}, t)$ depending on a spatial coordinate vector $\mathbf{x} \in \mathcal{D}$ and time t. In addition, the random field depends on a vector $\mathbf{p}$ of time-invariant variables. We assume that there are P sample functions of the random field available. These realizations H_i of the random fields taken at the sample locations $\mathbf{x}_k; k = 1 \dots N$ therefore depend not only on the time points $t_j; j = 1 \dots M$ but also on the corresponding realizations (samples) $\mathbf{p}_i; i = 1 \dots P$ of the parameter vector. For the sake of numerical analysis, the values of each realization H_i for all time steps are arranged into a matrix $\mathbf{s}_i$ with $N \times M$ entries. For convenience, all matrices $\mathbf{s}_i$ are further arranged into an $N \times (M \cdot P)$ supermatrix $\mathbf{S}$.

From this sample matrix, the essential features are extracted by means of a Proper Orthogonal Decomposition (POD) leading m POD vectors contained columnwise in the matrix $\mathbf{U}$. Typically, m is substantially smaller than $M \cdot P$. A corresponding matrix $\mathbf{A}$ of amplitudes can be computed by projection onto $\mathbf{U}$ as

$$\mathbf{A} = \mathbf{U}^T\mathbf{S}$$

Note that this POD deals with the spatial and temporal variability simultaneously, and at the same time accounts for the effect of parameter variability contained in $\mathbf{p}_i$. Each column $\mathbf{a}_j$ of the amplitude matrix $\mathbf{A}$ is a time-signal. Of course, all these signals retain their dependency on the parameter vector $\mathbf{p}_i$.

3 NUMERICAL EXAMPLES

3.1 CATS benchmark

The data series as analyzed here consists of 5000 data points as shown in Fig. 2. For the purpose of a benchmark study [6], five small intervals, each containing 20 data points, have been removed.

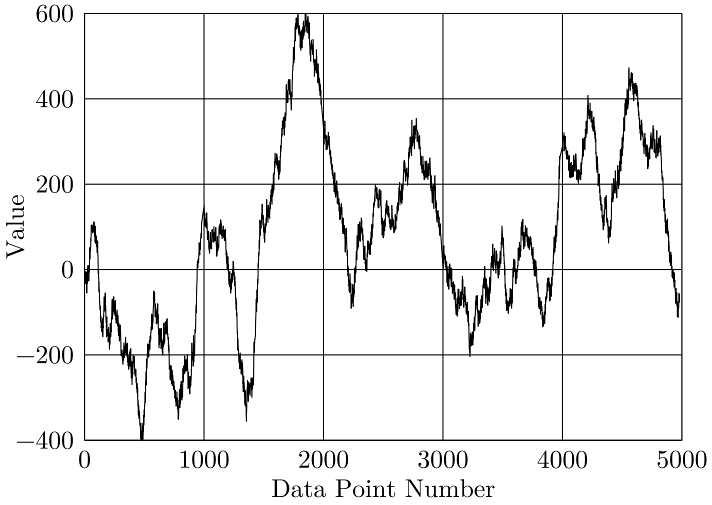


Figure 2: Time series with missing data points

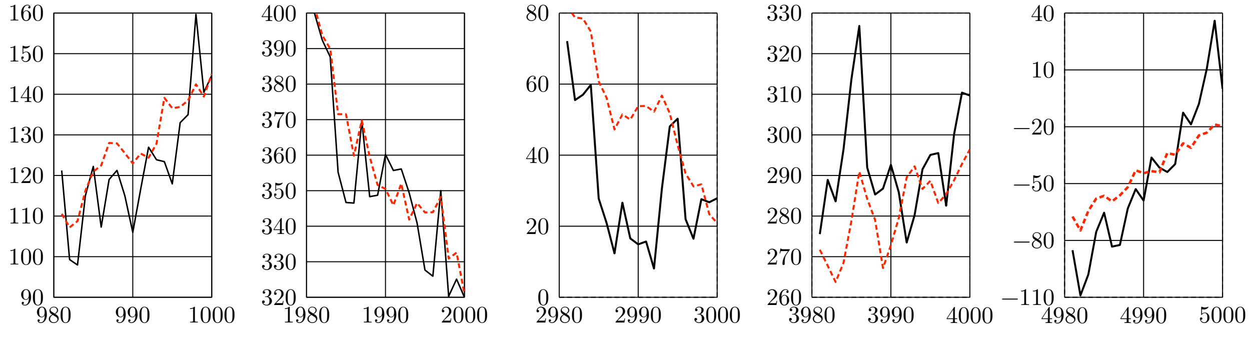


Figure 3: Missing data points (solid lines, black) and model predictions (dashed lines, red)

Table 1: Random Variables

Variable	Lower Bound	Upper Bound
a_1	1.000	1.500
a_2	0.500	0.750
a_3	0.333	0.500
a_4	0.250	0.375
a_5	0.200	0.300
ν	2.000	2.500
ζ	0.050	0.075

An AR model with a dimension L = 15 has been established. The minimization process as described above was done within the in-house software package `slangTNG` [7]. The optimization algorithm employed is CONMIN [8].

The error measure E_1 (as defined in [6]) achieved by this approach is $E_1 = 310$. This is substantially smaller than all results reported in the primary reference. In this reference, the best value of $E_1 = 408$ was achieved by a Kalman Smoother procedure [9]. A later study using a Deep Gaussian Covariance Network [10] yielded a value of $E_1 = 368$.

3.2 String Example

Consider the vibration of a string with nominal frequency ν. We assume for simplicity that the string is vibrating freely (with damping) according to

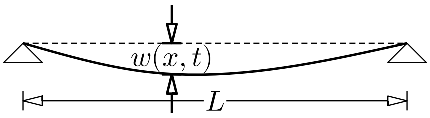


Figure 4: Vibrating string

$$w(x,t) = \sum_{k=1}^{N} a_k \exp(-\zeta\omega_k t) \cos\omega_k t \sin\frac{k\pi x}{L}$$

In this equation, the harmonics are $\omega_k = 2\pi k\nu$, L is the length of the string, and the coefficients a_k are chosen as $a_k = \frac{1}{k}$. The series is truncated at $N = 5$.

For the purpose of establishing a meta-model, the coefficients a_k, the nominal frequency ν and the damping ζ are assumed to be uniformly distributed random variables. The respective lower and upper bounds are given in Table 1.

A Latin Hypercube Design of Experiments (DOE) with a sample size of 100 is set up.

The time signals at different locations x_k along the string are shown in Fig. 5.

The variance of the field depends on both location x and time t as shown in Fig. 6.

The Karhunen-Loéve expansion (or, equivalently, POD) yields 5 relevant POD shapes as shown in Fig. 7. Of course, these shapes are basically a representation of the 5 vibration modes as considered in this example.

Projecting the time signals onto the POD shapes yields in the amplitude functions as shown in Fig. 8.

The dependency of the amplitude functions on the random input variables is now described by a meta-model of optimum quality [4]. The prognostic quality of the two most important

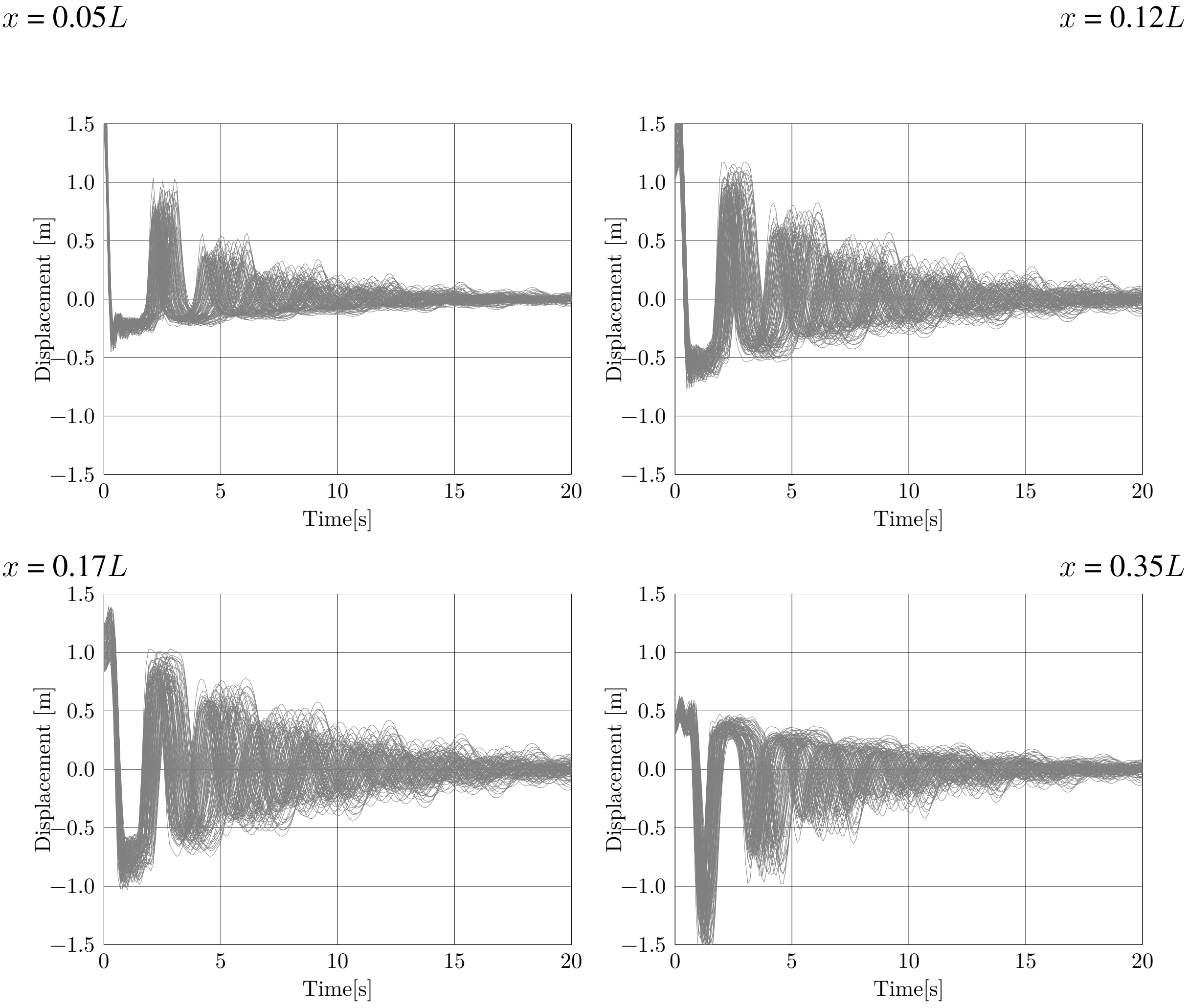


Figure 5: Time signals at different locations x

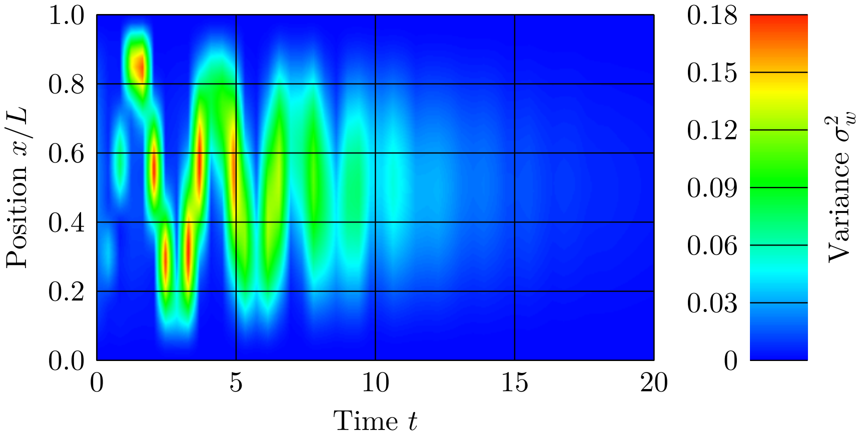


Figure 6: Variance of space-time signal

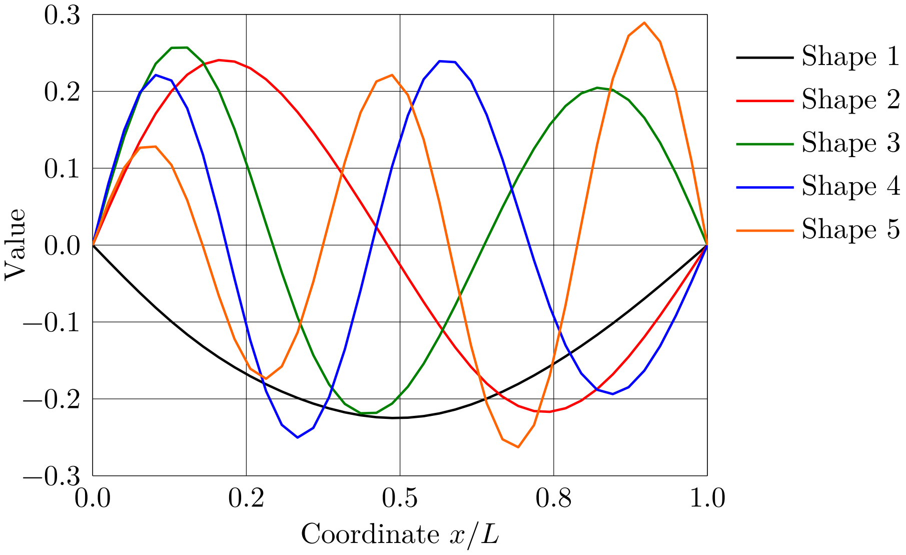


Figure 7: POD shapes

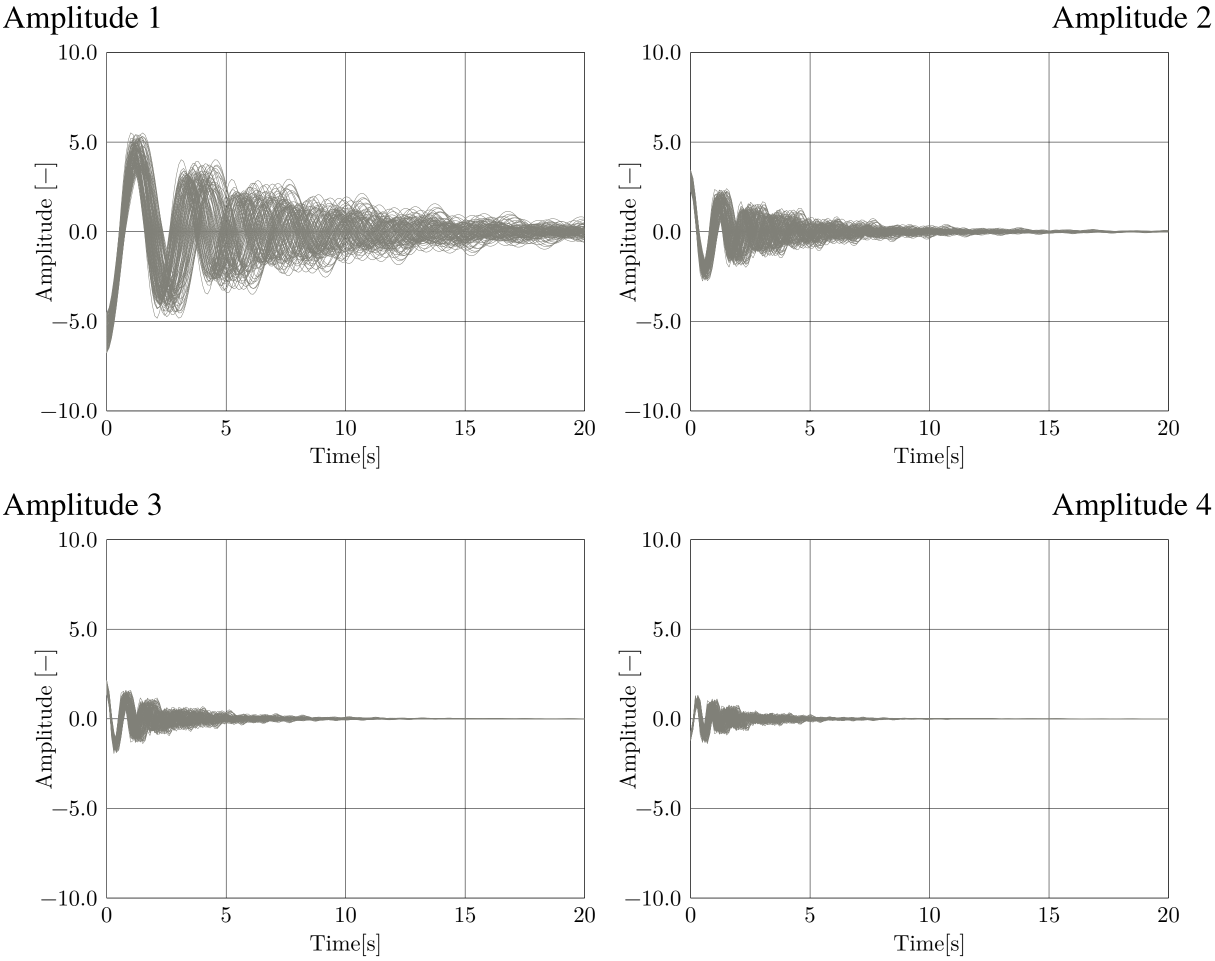


Figure 8: Time signatures

Table 2: Variance-weighted average CoQ values for Amplitudes 1–5

Quantity	$CoQ_{m,1}$	$CoQ_{m,2}$	$CoQ_{m,3}$	$CoQ_{m,4}$	$CoQ_{m,5}$
Value	0.993	0.987	0.985	0.984	0.993

amplitude functions as expressed by the Coefficient of Quality (CoQ) is shown in Figs. 9. It can be seen that in the initial phases of the signals the quality is excellent, but it does markedly decrease as the signal intensities diminish. This diminishing of the signals after $t = 10$ can also be clearly seen from the variance as shown in Fig. 6. A variance-weighted average CoQ_m can be computed according to

$$CoQ_m = \frac{\int\limits_0^T CoQ(t)\sigma^2(t)\mathrm{d}t}{\int\limits_0^T \sigma^2(t)\mathrm{d}t} \tag{8}$$

In this equation, T denotes the total time duration of the signals. For Amplitude 1, the average is $CoQ_{m,1}$ = 0.993. Data for amplitudes 1–5 are given in Table 2. These numbers certainly indicate quite satisfactory accuracy of the meta-models.

Fig. 9 also shows the sensitivities (computed as total Sobol indices, [11]) of the amplitude with respect to the system parameters. This figure shows that the two most important parameters for amplitude 1 are the coefficient a_1 and the frequency ν. In the initial phase, the importance oscillates between those parameters which in the later phase only the influence of the frequency remains.

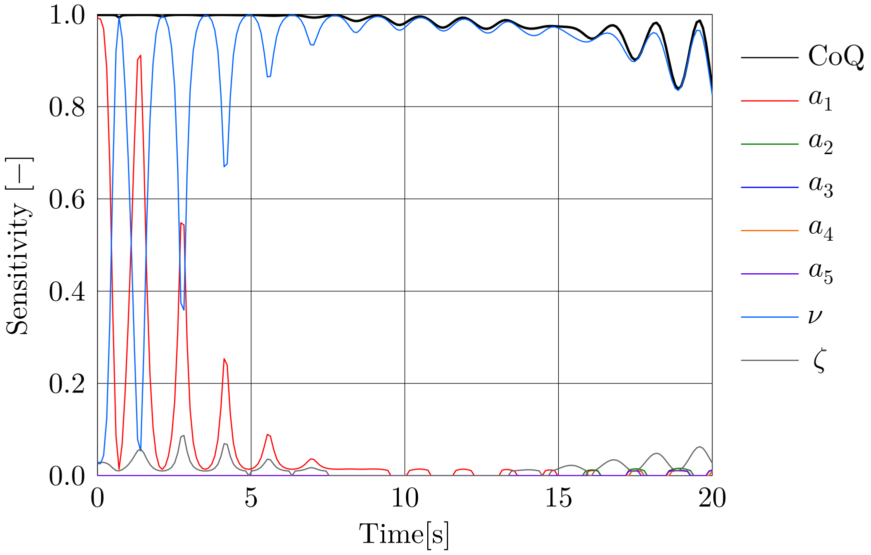


Figure 9: Coefficients of Quality and sensitivities for Amplitude 1

Finally, a system identification using the meta-model is carried out. Target values for the identification are given in Table 3.

The identification is based on the signals of points 2 and 34 as shown in Figs. 10 and 11. The objective function is chosen as to minimize the mean square difference between the true values and the meta-model values of these two signals. The resulting parameter values are

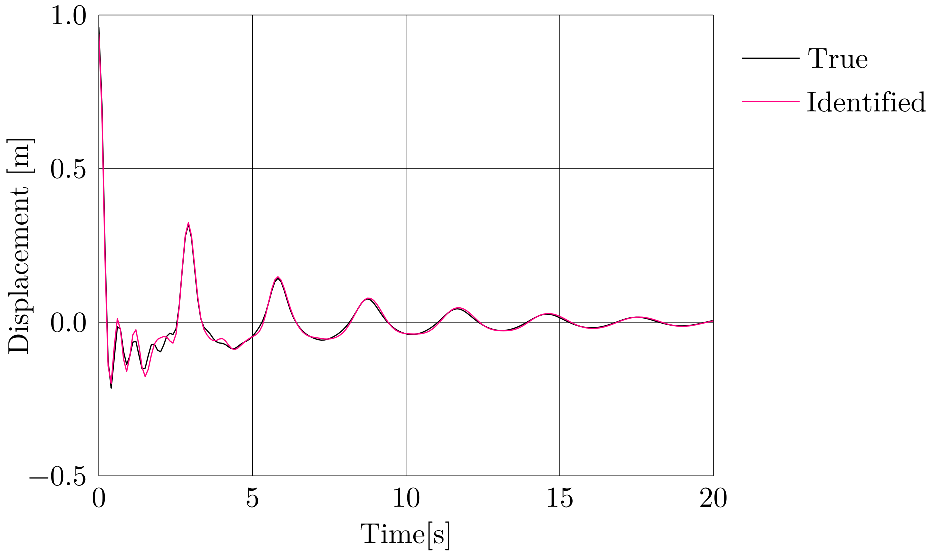


Figure 10: Comparison of true signal and meta model for point 2

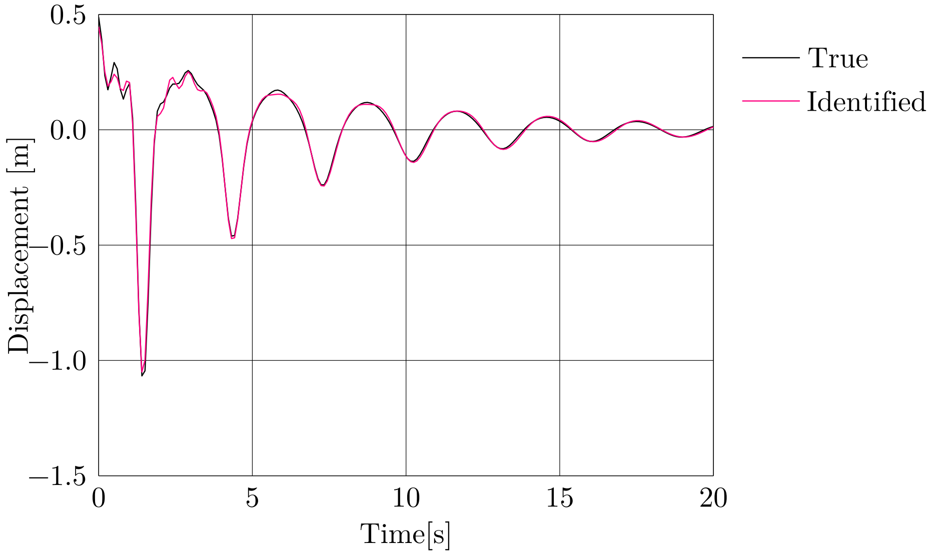


Figure 11: Comparison of true signal and meta model for point 34

Table 3: Target values for System identification

Variable	Target Value	Identified Value
a_1	0.700	0.671
a_2	0.150	0.113
a_3	0.200	0.192
a_4	0.050	0.063
a_5	0.140	0.115
ν	2.150	2.143
ζ	0.070	0.0691

given in Table 3 and the corresponding signals reconstructed using the meta-models are shown in Figs. 10 and 11.

4 CONCLUSIONS

From the preceding analysis and the numerical examples it can be seen that interpolation of missing data points of random time series may be reasonably well done by using auto-regressive schemes. Such a scheme essentially assumes that the time signal under consideration is obtainable by the numerical solution of some homogeneous ordinary differential equation. Therefore this rather simple approach is considered to be appropriate only if the time series does not have significant external dependencies.

The second example demonstrates the usefulness of meta-modeling techniques to quantitatively describe the relation between space-time signals and external or system parameters. It has been shown that inverse problems (system parameter identification) can be successfully performed on the basis of such meta-models.

One may conclude that meta-modeling techniques can be successfully applied to the reconstruction of time series with gaps and for the modeling of the parameter-dependence of space-time signals representing system responses.

References

[1] L. Faravelli. “Response-surface approach for reliability analysis”. In: *Journal of Engineering Mechanics* 115 (1989), pp. 2763–2781.

[2] C. Bucher and U. Bourgund. “A Fast and Efficient Response Surface Approach for Structural Reliability Problems”. In: *Structural Safety* 7 (1990), pp. 57–66.

[3] G. E. P. Box and N. R. Draper. “A basis for the selection of a response surface design”. In: *Journal of the American Statistical Association* 54 (1959), pp. 622–654.

[4] C. Bucher. “Metamodels of optimal quality for stochastic structural optimization”. In: *Probabilistic Engineering Mechanics* 54 (2018), pp. 131–137.

[5] J. E. Nash and J. V. Sutcliffe. “River flow forecasting through conceptual models. Part I - A discussion of principles”. In: *Journal of Hydrology* 10 (1970), pp. 282–290.

[6] A. Lendasse, E. Oja, O. Simula, and M. Verleysen. “Time Series Prediction Competition: The CATS Benchmark”. In: *IJCNN’2004 proceedings – International Joint Conference on Neural Networks Budapest (Hungary), 25-29 July 2004*. IEEE. 2004, pp. 1515–1620.

[7] C. Bucher and S. Wolff. “slangTNG - software for stochastic structural analysis made easy”. In: *Meccanica dei Materiali e delle Strutture* 3.4 (2012), pp. 10–17.

[8] G. N. Vanderplaats and F. Moses. “Structural optimization by methods of feasible directions”. In: *Computers and Structures* 3 (1973), pp. 739–755.

[9] S. Särkkä, A. Vehtari, and J. Lampinen. “Time Series Prediction by Kalman Smoother with Cross-Validated Noise Density”. In: *IJCNN’2004 proceedings – International Joint Conference on Neural Networks Budapest (Hungary), 25-29 July 2004*. IEEE. 2004, pp. 1653–1657.

[10] K. Cremanns and D. Roos. “Deep Gaussian Covariance Network”. In: *CoRR* abs/1710.06202 (2017). arXiv: `1710.06202`. URL: `http://arxiv.org/abs/1710.06202`.

[11] I. Sobol. “Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates”. In: *Mathematics and Computers in Simulation* 55 (2001), pp. 271–280.

UNCECOMP 2019
3rd ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Crete, Greece, 24–26 June 2019

A DIRECT HAMILTONIAN MCMC APPROACH FOR RELIABILITY ESTIMATION

Hamed Nikbakht[1], and Konstantinos G. Papakonstantinou[1]

[1]Department of Civil and Environmental Engineering
The Pennsylvania State University, University Park, PA 16802, USA
e-mail: {hun35, kpapakon}@psu.edu

Keywords: Hamiltonian MCMC, Quasi-Newton, Rare Event Probability, High-dimensional Parameter Space, Reliability Estimation.

Abstract. *Accurate and efficient estimation of rare events probabilities is of significant importance, since often the occurrences of such events have widespread impacts. The focus in this work is on precisely quantifying these probabilities, often encountered in reliability analysis of complex engineering systems, by introducing a gradient-based Hamiltonian Markov Chain Monte Carlo (HMCMC) framework, termed Approximate Sampling Target with Post-processing Adjustment (ASTPA). The basic idea is to construct a relevant target distribution by weighting the high-dimensional random variable space through a one-dimensional likelihood model, using the limit-state function. To sample from this target distribution we utilize HMCMC algorithms that produce Markov chain samples based on Hamiltonian dynamics rather than random walks. We compare the performance of typical HMCMC scheme with our newly developed Quasi-Newton based mass preconditioned HMCMC algorithm that can sample very adeptly, particularly in difficult cases with high-dimensionality and very small failure probabilities. To eventually compute the probability of interest, an original post-sampling step is devised at this stage, using an inverse importance sampling procedure based on the samples. The involved user-defined parameters of ASTPA are then discussed and general default values are suggested. Finally, the performance of the proposed methodology is examined in detail and compared against Subset Simulation in a series of static and dynamic low- and high-dimensional benchmark problems.*

1 INTRODUCTION

In this work, we investigate Hamiltonian Markov Chain Monte Carlo (HMCMC) schemes for estimation of rare events probabilities, a commonly encountered important problem in several engineering and scientific applications, most often observed in the form of failure probability, or alternatively, reliability estimation. Calculating such small probabilities with accuracy presents many numerical and mathematical challenges, particularly in cases with high dimensional random spaces and/or expensive computational models, that practically limit the afforded number of model calls. The well known gradient based First Order Reliability Method (FORM), and variants, have a very long history in reliability estimation problems, with numerous successes [1, 2, 3, 4]. Such asymptotic approximation methods naturally have of course limitations, however, in general settings. Hence, numerous sampling based methods have been also suggested in the literature to tackle the problem in its utmost generality, e.g. [5]. The current state-of-the-art sampling method for problems of this type is termed Subset Simulation (SuS) [6] and belongs to the family of MCMC techniques. Within the context of Subset Simulation, various random-walk and non-random-walk-based MCMC proposal steps [7, 8] have been explored and suggested, to improve the sampling efficiency of SuS, including Hamiltonian steps [9].

In this work we completely deviate from SuS and we introduce a gradient-based Hamiltonian Markov Chain Monte Carlo (HMCMC) sampling framework, termed *Approximate Sampling Target with Post-processing Adjustment* (ASTPA) [10], that is directly used for rare events probabilities estimation. The basic idea of ASTPA is to construct a relevant target distribution to sample from, by weighting the high-dimensional random variable space through a one-dimensional likelihood model, using the limit-state function, and to then utilize an original post-sampling step, using an inverse importance sampling procedure based on the acquired samples. Hamiltonian MCMC schemes are employed to perform the sampling. The Hamiltonian Monte Carlo (HMC) method, originally developed by [11], and more recently popularized mainly through the works of [12, 13, 14], is characterized by scalability [13, 15], fast mixing rates, weak sample auto-correlation, even in complex high-dimensional parameter spaces [16], and has achieved broad-spectrum successes in most general settings e.g. [17, 18, 19, 20]. Herein, we compare the performance of the typical HMCMC scheme with our newly developed Quasi-Newton based mass preconditioned HMCMC algorithm that also exploits the information about the localized geometry of the failure region, through an inexpensive BFGS approximation. The involved user-defined parameters of ASTPA are also discussed in the paper and general default values are suggested. The performance of the proposed methodology is finally examined and compared successfully against Subset Simulation, in a series of static and dynamic, low- and high-dimensional benchmark problems.

2 CONCEPTS BEHIND HAMILTONIAN MARKOV CHAIN MONTE CARLO

In HMCMC methods, Hamiltonian dynamics are used to produce distant state steps for the Metropolis proposals, thereby avoiding the slow exploration of the state space that results from the diffusive behavior of simple random-walk proposals. Given a parameter of interest $\boldsymbol{\theta}$ with (unnormalized) density $\pi_\Theta(.)$, the Hamiltonian Markov Chain Monte Carlo method introduces an auxiliary momentum variable $\mathbf{z}$ and samples from the joint distribution characterized by:

$$\pi(\boldsymbol{\theta}, \mathbf{z}) \propto \pi_\Theta(\boldsymbol{\theta})\ \pi_{Z|\Theta}(\mathbf{z}|\boldsymbol{\theta}) \tag{1}$$

where $\pi_{Z|\Theta}(.|\boldsymbol{\theta})$ is proposed to be a symmetric distribution. With $\pi_\Theta(\boldsymbol{\theta})$ and $\pi_{Z|\Theta}(\mathbf{z}|\boldsymbol{\theta})$ being uniquely described up to normalizing constants, the functions $U(\boldsymbol{\theta}) = -\log\pi_\Theta(\boldsymbol{\theta})$ and $K(\boldsymbol{\theta},\mathbf{z}) = -\log\pi_{Z|\Theta}(\mathbf{z}|\boldsymbol{\theta})$ are introduced as the potential energy and kinetic energy, owing to the physical laws which motivate the Hamiltonian Markov Chain Monte Carlo algorithm. The total energy $H(\boldsymbol{\theta},\mathbf{z})$ can be thus expressed as:

$$H(\boldsymbol{\theta},\mathbf{z}) = U(\boldsymbol{\theta}) + K(\boldsymbol{\theta},\mathbf{z}) \tag{2}$$

and is often termed the Hamiltonian H. The kinetic energy function is unconstrained and can be formed in various ways based on the implementation. In most typical cases, the momentum is given by a zero-mean normal distribution [13, 16], and accordingly the kinetic energy can be written as: $K(\boldsymbol{\theta},\mathbf{z}) = -\log\pi_{Z|\Theta}(\mathbf{z}|\boldsymbol{\theta}) = -\log\pi_Z(\mathbf{z}) = \frac{1}{2}\mathbf{z}^T\mathbf{M}^{-1}\mathbf{z}$, where the $\mathbf{M}$ is a symmetric, positive-definite covariance (mass) matrix.

HMCMC generates a Metropolis proposal on the joint state-space $(\boldsymbol{\theta},\mathbf{z})$ by sampling the momentum and simulating trajectories of Hamiltonian dynamics in which the time evolution of the state $(\boldsymbol{\theta},\mathbf{z})$ is governed by Hamilton's equations, expressed typically by:

$$\frac{d\boldsymbol{\theta}}{dt} = \frac{\partial H}{\partial \mathbf{z}} = \frac{\partial K}{\partial \mathbf{z}} = \mathbf{M}^{-1}\mathbf{z}, \quad \frac{d\mathbf{z}}{dt} = -\frac{\partial H}{\partial \boldsymbol{\theta}} = -\frac{\partial U}{\partial \boldsymbol{\theta}} = \nabla_\theta \mathcal{L}(\boldsymbol{\theta}) \tag{3}$$

where $\mathcal{L}(\boldsymbol{\theta})$ denotes the log-density of the target distribution. Hamiltonian dynamics prove to be an effective proposal generation mechanism because the distribution $\pi(\boldsymbol{\theta},\mathbf{z})$ is invariant under the dynamics of Eq. (3). These dynamics enable a proposal state, obtained by an approximate solution of Eq. (3), to be distant from the current state, yet having high probability of acceptance. The solution to Eq. (3) is in general analytically intractable and thus the Hamiltonian equations need to be numerically solved by discretizing time, using some small step size, ε. A symplectic integrator that can be used for the numerical solution is the leapfrog one, as follows:

$$\mathbf{z}_{\text{t}+\varepsilon/2} = \mathbf{z}_\text{t} - (\frac{\varepsilon}{2})\frac{\partial U}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}_\text{t}), \quad \boldsymbol{\theta}_{\text{t}+\varepsilon} = \boldsymbol{\theta}_\text{t} + \varepsilon\frac{\partial K}{\partial \mathbf{z}}(\mathbf{z}_{\text{t}+\varepsilon/2}), \quad \mathbf{z}_{\text{t}+\varepsilon} = \mathbf{z}_{\text{t}+\varepsilon/2} - (\frac{\varepsilon}{2})\frac{\partial U}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}_{\text{t}+\varepsilon}) \tag{4}$$

The main advantages of using the leapfrog integrator are its simplicity, its volume-preserving feature, and its reversibility, due to its symmetry, by simply negating **z**, facilitating a valid Metropolis proposal. See [13], [16] and[21] for details on energy-conservation, reversibility and volume-preserving integrators and their connections to HMCMC. It is noted that in the above leapfrog integration algorithm, the computationally expensive part is to acquire the $\dfrac{\partial U}{\partial \boldsymbol{\theta}}$ term at the updated location $\boldsymbol{\theta}$. Taking $L = \tau/\varepsilon$ steps of the leapfrog integrator approximates the evolution $(\boldsymbol{\theta}(0),\mathbf{z}(0)) \longrightarrow (\boldsymbol{\theta}(\tau),\mathbf{z}(\tau))$, where τ is the trajectory length or path length, and provides the exact solution in the limit $\varepsilon \longrightarrow 0$.

As discussed, the typical HMCMC version is based on a Gaussian momentum $\pi_{Z|\Theta}(\mathbf{z}|\boldsymbol{\theta}) = \pi_Z(\mathbf{z}) \sim N(\mathbf{0},\mathbf{M})$ (or $\mathbf{z} \sim N(\mathbf{0},\mathbf{M})$). The mass matrix **M** is often set to the identity matrix, **I**, but can also be adapted to precondition the sampler when relevant information about the target distribution is available (see Section 4). A standard procedure for drawing *NIter* samples via HMCMC is described in Algorithm 1, where $\mathcal{L}(\boldsymbol{\theta})$ is the log-density of the target distribution of interest. $\boldsymbol{\theta}^0$ are the initial values for the $\boldsymbol{\theta}$, and L is the number of leapfrog steps, as explained before. For each HMCMC step, we first resample the momentum and then implement the L leapfrog updates (Leapfrog($\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}}$, ε)) before we accept or reject the Metropolis proposal at the pertinent step.

Algorithm 1 Hamiltonian Markov Chain Monte Carlo

```
procedure HMCMC($\boldsymbol{\theta}^0$, $\varepsilon$, $L$, $\mathcal{L}(\boldsymbol{\theta})$, NIter)
    for $m = 1\ \ to\ \ NIter$ do
        $\mathbf{z}^0 \sim N(0, \mathbf{I})$        ▷ momentum sampling from standard normal distribution
        $\boldsymbol{\theta}^m \leftarrow \boldsymbol{\theta}^{m-1}$, $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}^{m-1}$, $\tilde{\mathbf{z}} \leftarrow \mathbf{z}^0$
        for $i = 1\ \ to\ \ L$ do
            $\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}} \leftarrow$ Leapfrog($\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}}$, $\varepsilon$)        ▷ leapfrog integration
        end for
        $with\ probability$:
        $\alpha = \min\left\{1, \dfrac{\exp(\mathcal{L}(\tilde{\boldsymbol{\theta}}) - \frac{1}{2}\tilde{\mathbf{z}}.\tilde{\mathbf{z}})}{\exp(\mathcal{L}(\boldsymbol{\theta}^{m-1}) - \frac{1}{2}\mathbf{z}^0.\mathbf{z}^0)}\right\}$        ▷ Metropolis step
        $\boldsymbol{\theta}^m \leftarrow \tilde{\boldsymbol{\theta}}$, $\mathbf{z}^m \leftarrow$ -$\tilde{\mathbf{z}}$
    end for
end procedure
```

The efficiency of HMCMC relies significantly on selecting suitable values for ε and L. In this work we select the stepsizes ε in such a way that the corresponding average acceptance rates are approximately 65%, as values between 60% and 80% are typically assumed optimal [13, 14, 15]. The dual averaging algorithm of [14] was adopted here to find these stepsizes. However, in contrast to [14] we adapt these stepsizes throughout the analysis, in both the burn-in and stationary phases of the MCMC algorithm. To determine the value of L, we estimate the trajectory length τ so as to have a sufficient so called normalized Expected Square Jumping Distance (*ESJD*) $\tau^{-1/2}\,\mathbb{E}\|\theta^{(t+1)}(\tau) - \theta^{(t)}(\tau)\|^2$, as introduced in [22], and then we randomly perturb each trajectory length $\tau^{(t)}$ in the range $[0.9\tau, 1.1\tau]$ to avoid periodicity (t denotes the t-th iteration of HMCMC). In all our experiments we determine L and control the trajectory length in this manner, as we have found it to work well in practice. The role of these parameters (ε and τ (or L)) and techniques for determining them have been quite extensively studied and for more details we refer the readers to [13, 14, 15].

3 METHODOLOGY TO CALCULATE THE FAILURE PROBABILITY

The failure probability P_F for a system, that is the probability of a defined unacceptable system performance, can be expressed as a d-fold integral, as:

$$P_F = \mathbb{E}[I_F(\boldsymbol{\theta})] = \int_{g(\boldsymbol{\theta})\leq 0} I_F(\boldsymbol{\theta})\pi_{\boldsymbol{\theta}}(\boldsymbol{\theta})d\boldsymbol{\theta} \tag{5}$$

where $\boldsymbol{\theta}$ is the random vector $[\theta_1, ..., \theta_d]^T$; $F \subset \mathbb{R}^d$ is the failure event in the parameter space; g($\boldsymbol{\theta}$) is the limit-state function that can include one or several distinct failure modes and defines the failure of the system by g($\boldsymbol{\theta}$)$\leq$ 0; *I(.)* is the indicator function with: I_F ($\boldsymbol{\theta}$) = 1 if $\boldsymbol{\theta} \in$ g($\boldsymbol{\theta}$)$\leq$ 0 and $I_F(\boldsymbol{\theta})$ = 0 otherwise; $\mathbb{E}$ is the expectation operator, and $\pi_{\boldsymbol{\theta}}$ is the joint probability density function (PDF) for Θ. It is common practice in reliability analysis to have the joint PDF of Θ be the standard normal one, due to its rotational symmetry and exponential probability decay. In most cases, this is not restrictive, since it is uncomplicated to transform the original random variables **X** to Θ, e.g. [23]. When this is not the case however, but the probabilistic characterization of **X** can be defined in terms of marginal distributions and correlations, the Nataf

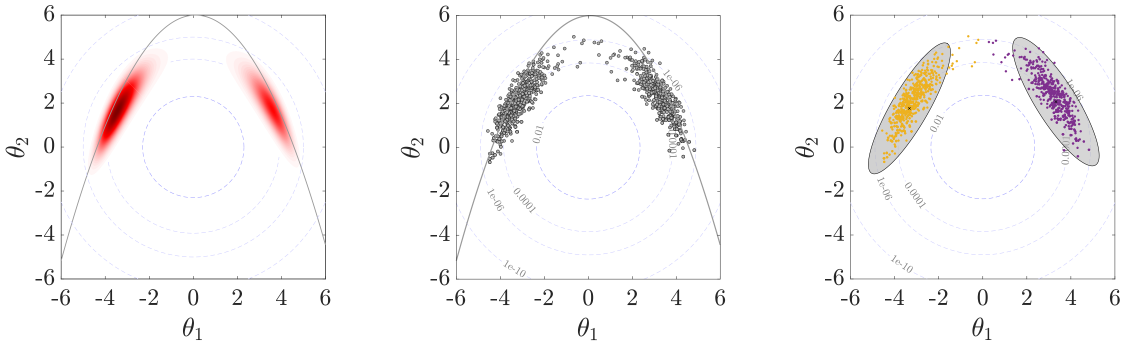


Figure 1: The above figures represent the analytical target distribution, the simulated target distribution samples based on our HMCMC-based method, and the fitted Gaussian Mixture Model describing the simulated samples, from left to right, respectively.

distribution (equivalent to Gaussian copula) can be used to model the joint PDF, and the mapping to the standard normal space can be then accomplished [24].

The main idea of our approach to calculate the failure probability is to construct an appropriate approximate target distribution to sample from, based on Hamiltonian MCMC methods that can quickly reach regions of interest and can keep the number of model calls to a minimum, and to then utilize a post-sampling step to acquire the exact probability estimation, without any additional model calls. We construct this approximate target distribution by combining the multidimensional parameter space Θ with a one-dimensional likelihood function, using the limit-state expression. This one-dimensional likelihood function is expressed as a Gaussian PDF with mean $= \mu_{g(\boldsymbol{\theta})} = 0$, where $g(\boldsymbol{\theta})$ is the limit-state function, and a dispersion factor σ:

$$\mathbf{N}\left(\frac{g(\boldsymbol{\theta})}{g_c}\;\middle|\;\mu_{g(\boldsymbol{\theta})} = 0,\ \sigma\right), \quad g_c = \begin{cases} g(\mathbf{0}), & \text{if}\, g(\mathbf{0}) > 8 \,\text{or}\, g(\mathbf{0}) < 1 \\ 1, & \text{otherwise} \end{cases} \tag{6}$$

where g_c is a normalizing constant. The reason for this normalization, $g(\boldsymbol{\theta})/g_c$, is to control the suggested upper and lower bounds of σ. The target PDF is then defined as:

$$\text{Target probability distribution} \propto \mathbf{N}\left(\frac{g(\boldsymbol{\theta})}{g_c}\;\middle|\;\mu_{g(\boldsymbol{\theta})} = 0,\ \sigma\right) \times \Big(\boldsymbol{\theta} \sim \mathbf{N}(\mathbf{0}, \mathbf{I})\Big) \tag{7}$$

Having the total number of model calls in mind, as well as the coefficient of variation of the estimator (C.O.V), the suggested value for σ is in the range $[0.1\ 0.7]$. Fine tuning σ in that range is not generally necessary. It is recommended, in general, to use higher σ values $(0.6 - 0.7)$ in nonlinear high-dimensional problems and multi-modal cases when $1 \leq g(\mathbf{0}) \leq 8$, since a larger σ usually allows longer state jumps and fewer required model calls. On the other hand, a lower σ generally increases the accuracy of the estimator, at the expense of a slightly increased number of model calls.

Fig. 1 concisely portrays the overall approach by using a bimodal target distribution. The gray curves represent the parabolic limit-state function $g(\boldsymbol{\theta})$ of this problem, with the failure domain being outside $g(\boldsymbol{\theta})$. The left figure displays the constructed target distribution, by adopting the previously described approach, which in this simple 2D case can be visualized. The middle figure shows drawn samples from the target distribution by our suggested Hamiltonian MCMC variant, described in Section 4. For their initial stage, our HMCMC samplers have an adaptive annealed phase, mainly in order to automatically tune parameters and reduce the computational cost, overall, and then follow the typical Hamiltonian approach, except in our Quasi-Newton case (Section 4) the mass matrix is appropriately preconditioned. As such, during the burn-in period, we initialize the spread of likelihood, σ_0, equal to 1 that then follows an *exponential decay* throughout the burn-in period, while at the end of this initial period, σ takes its constant value, as described above, for the stationary phase of the algorithms.

To finally compute the failure probability we have to adjust Eq. (5) accordingly, since the samples have been sampled based on our constructed approximate target distribution. An original post-sampling step is devised at this stage using our inverse importance sampling procedure, i.e. having the samples, choose a pertinent Importance Sampling Density (ISD) automatically, based on the samples. Given that, the probability of failure after some algebra (see [10] for details) can be computed as follows:

$$P_F = \int I_F(\boldsymbol{\theta}) \pi_{\boldsymbol{\theta}}(\boldsymbol{\theta}) d\boldsymbol{\theta} = \int I_F(\boldsymbol{\theta}) \frac{C \ . \ \tilde{h}(\boldsymbol{\theta})}{\ell(\boldsymbol{\theta})} d\boldsymbol{\theta} \tag{8}$$

where $C = \frac{1}{N}\sum_{i=1}^{N} \dfrac{\tilde{h}(\theta_i)}{Q(\theta_i)}$; $\tilde{h}(.)$ denotes the non-normalized target PDF, $\ell(\boldsymbol{\theta})$ is our likelihood function, and $Q(.)$ is a computed Gaussian Mixture Model (GMM), based on the already available samples and the generic Expectation Maximization (EM) algorithm, as indicatively seen in the right plot of Fig. 1.

Our described newly proposed method is termed ASTPA (Approximate Sampling Target with Post-processing Adjustment) and, as a summary, comprises of constructing a target distribution model, performing HMCMC sampling, and finally applying a post-sampling step. For more details on supplementary justifications about this method, we refer readers to [10].

4 QUASI-NEWTON EXTENSIONS AND CONNECTIONS TO HMCMC

In high-dimensional problems, the computational cost of the typical HMCMC sampler may increase considerably and a prohibitive number of model calls per leapfrog step may be required. In this work, we address this issue in a developed Newton-type context, where the Hessian information is approximated without any required additional model calls per leapfrog step. To this end, the well-known BFGS approximation [25] is used in our Quasi-Newton type Hamiltonian MCMC approach. Let $\boldsymbol{\theta} \in \mathbb{R}^d$, consistent with the previous section. Given the $k\text{-}th$ estimate $\mathbf{W}_k$, where $\mathbf{W}_k$ is an approximation to the inverse Hessian at $\boldsymbol{\theta}_k$, the BFGS update $\mathbf{W}_{k+1}$ can be expressed as:

$$\mathbf{W}_{k+1} = (\mathbf{I} - \frac{\boldsymbol{s}_k \boldsymbol{y}_k^T}{\boldsymbol{y}_k^T \boldsymbol{s}_k}) \mathbf{W}_k (\mathbf{I} - \frac{\boldsymbol{y}_k \boldsymbol{s}_k^T}{\boldsymbol{s}_k^T \boldsymbol{y}_k}) + \frac{\boldsymbol{s}_k \boldsymbol{s}_k^T}{\boldsymbol{s}_k^T \boldsymbol{y}_k} \tag{9}$$

where $\mathbf{I}$ is the identity matrix, $\boldsymbol{s}_k = \boldsymbol{\theta}_{k+1} - \boldsymbol{\theta}_k$, and $\boldsymbol{y}_k = \nabla f(\boldsymbol{\theta}_{k+1}) - \nabla f(\boldsymbol{\theta}_k)$ where $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ denotes any relevant target distribution function in this case. Our developed Quasi-Newton preconditioned Hamiltonian Markov Chain Monte Carlo (QNp-HMCMC) method is presented in detail in Algorithm 2. In the burn-in phase we are still sampling the momentum from an identity mass matrix but the ODEs of Eq. (3) now become:

$$\dot{\boldsymbol{\theta}} = \mathbf{W}\mathbf{M}^{-1}\mathbf{z}, \quad \dot{\mathbf{z}} = \mathbf{W}\nabla_{\theta}\mathcal{L}(\boldsymbol{\theta}). \tag{10}$$

where $\mathbf{W} \in \mathbb{R}^{d \times d}$ is the symmetric positive definite matrix of Eq. (9) and being the inverse Hessian matrix provides an informed approximation of the local geometry of the parameter space, accelerating exploration of the domain. The final estimation of the approximated inverse of the Hessian matrix, **W**, from the burn-in phase is then used to define the preconditioned covariance matrix to sample the momentum variable for the stationary, non-adaptive stage of the chain. It can be shown that all utilized dynamics in both phases of the algorithm enable us to maintain the desired target distribution as the invariant one. In Section 5 we empirically evaluate and compare the QNp-HMCMC performance in various settings. For further details on the QNp-HMCMC method, its performance in different settings, and its validity, see [10].

5 NUMERICAL RESULTS

In this section, four numerical examples are implemented to illustrate the efficiency of the proposed methods. In all examples, the tuning parameters (ε,τ,σ) are systematically used as mentioned in Sections 2 and 3. In the context of reliability problems, we use the default value $\tau = 0.7$ as a starting point and then employ the ESJD metric [22] as described in Section 2. The burn-in period is chosen to be on average 15% of the total number of model calls, while the upper bound of the burn-in size is limited to 20%. The described methods are compared to the Component-wise Metropolis-Hastings based Subset Simulation (CWMH-SuS). For the sake of comparison, we use two proposal distributions in CWMH-SuS, a uniform distribution of width 2 and a standard normal one. The parameters of Subset Simulation are chosen as n_s =1,000 and 2,000 for low- and high-dimensional simulations respectively, where n_s is the number of samples for each subset level, and $p_0 = 0.1$, where p_0 is the percentile of the samples that determines the intermediate subsets [6]. Comparisons are illustrated in terms of accuracy and computational cost. In particular, the tables show the P_F estimation, including the mean number of limit-state function calls in order to calculate the value and gradient of the target distribution in the HMCMC-based algorithms, and the value of the limit-state function in SuS. The analytical gradients are provided in all examples, hence one model call can provide both the value and the gradient of the target distribution. In all examples, the number of limit-state function evaluations for all methods has been set to be roughly the same to each other for comparison purposes. Results are based for all examples on 500 independently performed simulations, so that the sample mean and C.O.V of the results can be acquired. It should be noted that the ASTPA parameters are carefully chosen for all examples but are not optimized for any one. Hence, comparative and perhaps improved alternate performance might be achieved with a different set of parameters.

5.1 Example 1: parabolic/concave limit-state function

The first example is expressed by the following limit state function for two standard normal random variables [26]:

$$g(\theta) = r - \theta_2 - \kappa\,(\theta_1 - e)^2 \tag{11}$$

where r, κ and e are deterministic parameters chosen as $r = 6$, $\kappa = 0.3$ and $e = 0.1$. The probability of failure is 3.95E-5 and the limit-state function consists of two design points (failure modes), as seen in Fig. 1. For the HMCMC-based algorithms, the likelihood dispersion factor, σ, is 0.7 and the burn-in sample size is taken as 200. Consistent to the discussion in Section 3, the trajectory length is set to $\tau = 1$.

Table 1 compares the number of model calls, the coefficient of variation and the $\mathbb{E}[\hat{P}_F]$ obtained by all tested methods. The Subset Simulation results are based on n_s =1,000. It is shown that the HMCMC approach gives significantly smaller C.O.V. than SuS and also

Algorithm 2 Quasi-Newton preconditioned Hamiltonian Markov Chain Monte Carlo

```
procedure QNP-HMCMC($\boldsymbol{\theta}^0$, $\varepsilon$, $L$, $\mathcal{L}(\boldsymbol{\theta})$, BurnIn, NIter)
    $\mathbf{W} = \mathbf{I}$
    for $m = 1 \;\; to \;\; NIter$ do
        if $m \leq BurnIn$ then
            $\mathbf{z}^0 \sim \mathbf{N}(\mathbf{0}, \mathbf{M})$                                   ▷ where $\mathbf{M} = \mathbf{I}$
            $\boldsymbol{\theta}^m \leftarrow \boldsymbol{\theta}^{m-1}$, $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}^{m-1}$, $\tilde{\mathbf{z}} \leftarrow \mathbf{z}^0$, $\mathbf{B} \leftarrow \mathbf{W}$
            for $i = 1 \;\; to \;\; L$ do
                $\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}} \leftarrow$ Leapfrog-BurnIn($\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}}$, $\varepsilon$, $\mathbf{B}$)
              Update $\mathbf{W}$ using Eq. (9)
            end for
            $with\ probability$:
          $\alpha = \min\left\{1, \dfrac{\exp(\mathcal{L}(\tilde{\boldsymbol{\theta}}) - \frac{1}{2}\tilde{\mathbf{z}}.\tilde{\mathbf{z}})}{\exp(\mathcal{L}(\boldsymbol{\theta}^{m-1}) - \frac{1}{2}\mathbf{z}^0.\mathbf{z}^0)}\right\}$
          $\boldsymbol{\theta}^m \leftarrow \tilde{\boldsymbol{\theta}}$, $\mathbf{z}^m \leftarrow$ -$\tilde{\mathbf{z}}$                 ▷ If proposal rejected: $\mathbf{W} \leftarrow \mathbf{B}$
        else                                                         ▷ If $m > BurnIn$
            $\mathbf{z}^0 \sim \mathbf{N}(\mathbf{0}, \mathbf{M})$                                   ▷ where $\mathbf{M} = \mathbf{W}^{-1}$
            $\boldsymbol{\theta}^m \leftarrow \boldsymbol{\theta}^{m-1}$, $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}^{m-1}$, $\tilde{\mathbf{z}} \leftarrow \mathbf{z}^0$
            for $i = 1 \;\; to \;\; L$ do
                $\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}} \leftarrow$ Leapfrog($\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}}$, $\varepsilon$, $\mathbf{M}$)
            end for
            $with\ probability$:
          $\alpha = \min\left\{1, \dfrac{\exp(\mathcal{L}(\tilde{\boldsymbol{\theta}}) - \frac{1}{2}\tilde{\mathbf{z}}.\ \mathbf{M}^{-1}.\tilde{\mathbf{z}})}{\exp(\mathcal{L}(\boldsymbol{\theta}^{m-1}) - \frac{1}{2}\mathbf{z}^0.\ \mathbf{M}^{-1}.\mathbf{z}^0)}\right\}$
          $\boldsymbol{\theta}^m \leftarrow \tilde{\boldsymbol{\theta}}$, $\mathbf{z}^m \leftarrow$ -$\tilde{\mathbf{z}}$
        end if
    end for
end procedure

function LEAPFROG-BURNIN($\tilde{\boldsymbol{\theta}}, \tilde{\mathbf{z}}, \varepsilon, \mathbf{B}$)
    $\tilde{\mathbf{z}} \leftarrow \mathbf{z} + (\varepsilon/2)\mathbf{B}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta})$
    $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta} + \varepsilon\mathbf{B}\tilde{\mathbf{z}}$
    $\tilde{\mathbf{z}} \leftarrow \mathbf{z} + (\varepsilon/2)\mathbf{B}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\tilde{\boldsymbol{\theta}})$
return $\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}}$.
end function

function LEAPFROG($\tilde{\boldsymbol{\theta}}, \tilde{\mathbf{z}}, \varepsilon, \mathbf{M}$)
    $\tilde{\mathbf{z}} \leftarrow \mathbf{z} + (\varepsilon/2)\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta})$
    $\tilde{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta} + \varepsilon\mathbf{M}^{-1}\tilde{\mathbf{z}}$
    $\tilde{\mathbf{z}} \leftarrow \mathbf{z} + (\varepsilon/2)\nabla_{\boldsymbol{\theta}}\mathcal{L}(\tilde{\boldsymbol{\theta}})$
return $\tilde{\boldsymbol{\theta}}$, $\tilde{\mathbf{z}}$.
end function
```

Table 1: Performance of various methods for the parabolic/concave limit-state function

	500 Independent Simulations	**CWMH-SuS**		**HMCMC**	**QNp-HMCMC**
		$U(-1,1)$	$N(0,1)$		
$\sigma = 0.7$ $\tau = 1$	Number of model calls	4,559	4,565	4,391	4,926
	C.O.V	0.62	0.65	0.35	0.39
	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 3.95E-5)	4.19E-5	4.14E-5	3.86E-5	3.47E-5

Table 2: Performance of various methods for the four-branch series system

	500 Independent Simulations	**CWMH-SuS**		**HMCMC**	**QNp-HMCMC**
		$U(-1,1)$	$N(0,1)$		
$\sigma = 0.7$ $\tau = 1$ $n_s =$1,000	Number of model calls	2,841	2,852	2,867	2,887
	C.O.V	0.26	0.30	0.29	0.26
	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 2.20E-3)	2.23E-3	2.26E-3	1.98E-3	1.91E-3
$\sigma = 0.7$ $\tau = 1$ $n_s =$2,000	Number of model calls	5,634	5,657	5,688	5,740
	C.O.V	0.19	0.22	0.13	0.17
	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 2.20E-3)	2.24E-3	2.23E-3	2.16E-3	2.11E-3

outperforms it in terms of the $\mathbb{E}[\hat{P}_F]$. Fig. 1 also demonstrates that the QNp-HMCMC samples accurately describe the two important failure regions.

5.2 Example 2: four-branch series system

This example is a well-known benchmark system reliability problem, defined by the following limit-state function in the standard normal space:

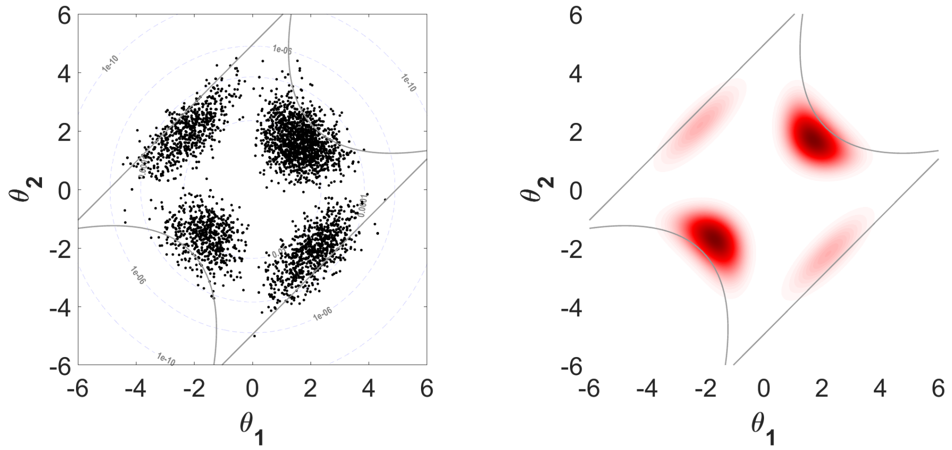
$$g(\boldsymbol{\theta}) = \min \begin{cases} 3 + 0.1(\theta_1 - \theta_2)^2 - (\theta_1 - \theta_2)/\sqrt{2} \\ 3 + 0.1(\theta_1 - \theta_2)^2 + (\theta_1 - \theta_2)/\sqrt{2} \\ (7/\sqrt{2}) + (\theta_1 - \theta_2) \\ (7/\sqrt{2}) + (\theta_2 - \theta_1) \end{cases} \tag{12}$$

The trajectory length is chosen as $\tau = 1$ and the likelihood dispersion factor, σ, is fixed to 0.7. The burn-in is set to 200 samples. Table 2 shows that the SuS with uniform proposal gives more accurate P_F estimation with smaller C.O.V than the HMCMC-based methods for the case of $n_s =$1,000. However, by increasing the sample size to $n_s =$2,000, it is seen that the HMCMC algorithm exhibits lower C.O.V compared to both SuS implementations. For the case of QNp-HMCMC, both the bias and C.O.V of the probability estimate considerably decrease with the sample size increase. Fig. 2 shows the analytical target density of the four-branch limit-state function problem and samples from the target distribution using the HMCMC approach. As seen, the method achieves to efficiently sample all four important failure regions.

5.3 Example 3: SDOF oscillator under impulse load

In this example, a nonlinear undamped single-degree-of-freedom (SDOF) oscillator subjected to a rectangular impulse load is analysed, as described in [27, 28]. The limit-state function is given as:

$$g(k_1, k_2, M, r, T_1, F_1) = 3r - \left| \frac{2F_1}{M\omega_0^2} \sin(\frac{\omega_0 T_1}{2}) \right| \tag{13}$$



(a)

(b)

Figure 2: (a) Simulated samples from the target distribution, (b) Analytical target distribution.

Table 3: Random variables of the undamped oscillator

Variable	Distribution	Mean	C.O.V
M	Gaussian	1	0.05
k_1	Gaussian	1	0.1
k_2	Gaussian	0.1	0.1
r	Gaussian	0.5	0.1
T_1	Gaussian	1	0.2
F_1	Gaussian	0.6-0.45	$\frac{1}{6}$

Table 4: Performance of various methods for the undamped oscillator example

	500 Independent Simulations	**CWMH-SuS**		**HMCMC**	**QNp-HMCMC**
		$U(-1,1)$	$N(0,1)$		
$\sigma = 0.1$	Number of model calls	5,170	5,160	5,132	5,119
$\tau = 0.7$	C.O.V	0.67	0.51	0.14	0.11
$\mu_{F_1} = 0.6$	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 9.09E-6)	9.68E-6	9.55E-6	9.10E-6	9.08E-6
$\sigma = 0.1$	Number of model calls	7,583	7,617	7,523	7,515
$\tau = 0.7$	C.O.V	0.77	0.70	0.21	0.15
$\mu_{F_1} = 0.45$	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 1.55E-8)	1.67E-8	1.50E-8	1.52E-8	1.51E-8

where $\omega_0 = \sqrt{(k_1 + k_2)/M}$ is the natural frequency of the oscillator, T_1 is the duration of the impulse load, M is the mass, k_1 and k_2 are the stiffnesses of the primary and secondary springs, r is the displacement at which one of the springs yields, and F_1 is the amplitude of the force. The description of all random variables is listed in Table 3. SuS results for both proposals are based on $n_s = 1{,}000$. All variables are first transformed to the standard normal space. Results are shown in Table 4 for two cases, by changing the mean value, μ_{F_1}, of F_1. For the HMCMC-based methods, the trajectory length and the likelihood dispersion factor are chosen as $\tau = 0.7$ and $\sigma = 0.1$ respectively. The burn-in sample size is set to 500. It is shown in this example that the QNp-HMCMC approach provides significantly more accurate and stable results in terms of the C.O.V. and $\mathbb{E}[\hat{P}_F]$. Particularly for the lowest failure probability level, QNp-HMCMC

Table 5: Performance of various methods for SDOF oscillator under white noise

	500 Independent Simulations	**CWMH-SuS**		**HMCMC**	**QNp-HMCMC**
		$U(-1,1)$	$N(0,1)$		
$\sigma = 0.2$ $\tau = 0.9$ $R = 1.8$	Number of model calls	11,000	11,011	11,063	11,059
	C.O.V	0.32	0.35	0.30	0.24
	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 2.53E-6)	2.58E-6	2.63E-6	2.57E-6	2.55E-6
$\sigma = 0.2$ $\tau = 0.9$ $R = 2$	Number of model calls	13,578	13,646	13,644	13,618
	C.O.V	0.41	0.48	0.34	0.29
	$\mathbb{E}[\hat{P}_F]$ (Exact $P_F \sim$ 1.11E-7)	1.16E-7	1.14E-7	1.13E-7	1.12E-7

approach noticeably outperforms all other methods. As results indicate, the QNp-HMCMC method is roughly insensitive to the failure probability level and there is no negative influence on the method when changing μ_{F_1}. For the two SuS variants, it is noteworthy to say here that the SuS with the standard normal proposal distribution indicates reasonably better performance in this example than the one with the uniform proposal.

5.4 Example 4: SDOF oscillator under white noise excitation

In this last example, we consider a SDOF oscillator, initially at rest, with natural frequency $\omega = 7.85\,rad/s$ and damping ratio $\xi = 0.02$, subjected to a Gaussian white noise ($W(t)$) excitation with spectral density of magnitude $S_0 = 1$. The response of the system is computed at discrete time instants $\{t_j = (j-1)\Delta t : j = 1,...n\}$ with $\Delta t = 0.05$, and the duration of study is $T = 5\,sec$. Thus, the number of time instants is equal to $n = T/\Delta t + 1 = 101$. The state vector $\boldsymbol{\theta}$ is the sequence of i.i.d. standard normal random variables that generate the $W(t_j) = \sqrt{\frac{2\pi S_0}{\Delta t}}\theta_j$ at the discrete time instants, resulting in 101 involved random variables in this example. Failure is characterized by the positive displacement response exceeding a threshold level R: $g(\boldsymbol{\theta}) = R - max\{Y(t)\}$.

The burn-in sample size is taken as 1,000 for the HMCMC-based methods. SuS results are based on $n_s =$2,000. It is seen in Table 5 that the QNp-HMCMC approach shows more accurate and efficient results in terms of C.O.V. and $\mathbb{E}[\hat{P}_F]$. Compared to the HMCMC approach, this example agrees with the additional results in [10] and confirms that the application of QNp-HMCMC in high-dimensional reliability problems is in general more attractive. By decreasing the target failure probability, results also reveal that QNp-HMCMC gives us a substantially improved estimation in comparison to all other methods.

6 CONCLUSIONS

A novel approach for estimation of rare event probabilities termed Approximate Sampling Target with Post-processing Adjustment (ASTPA), is presented in this paper, suitable for low- and high-dimensional problems, very small probabilities and multiple failure modes. ASTPA can provide an accurate unbiased estimation of the failure probabilities with an efficient number of limit-state function evaluations. The basic idea of ASTPA is to construct a relevant target distribution by weighting the high-dimensional random variable space through a one-dimensional likelihood model, using the limit-state function. To sample from this target distribution we utilize gradient-based HMCMC schemes, including our newly developed Quasi-Newton based mass preconditioned HMCMC algorithm (QNp-HMCMC) that can sample very adeptly, particularly in difficult cases with high-dimensionality and very small failure probabilities. Finally, an original post-sampling step is also devised, using an inverse importance sampling procedure based on the samples. The performance of the proposed methodology is examined and compared very successfully herein against Subset Simulation in a series of static and dynamic low- and high-dimensional benchmark problems. As a general guideline, QNp-HMCMC is recommended to be used for problems with more than 20 dimensions, where traditional HMCMC schemes may not perform that well. However, even in lower dimensions QNp-HMCMC performs reasonably well and is still a competitive algorithm. Since we are utilizing gradient-based sampling methods, all of our analyses and results are based on the fact that analytical gradients can be computed. In cases where numerical schemes are needed for the gradient evaluations, then HMCMC methods will not be competitive in relation to SuS. It should also be pointed out that different combinations of the HMCMC and QNp-HMCMC algorithms can be possible, based on problem-specific characteristics. Some of the ongoing and future work is directed towards exploring various ASTPA variants, and on estimating first-passage problems under numerous settings and high-dimensional parameter spaces.

REFERENCES

[1] Rüdiger Rackwitz. Reliability analysis—A review and some perspectives. *Structural Safety*, **23**(4):365–395, 2001.

[2] Armen Der Kiureghian. First-and second-order reliability methods. *Engineering Design Reliability Handbook*, **14**, 2005.

[3] Pei-Ling Liu and Armen Der Kiureghian. Optimization algorithms for structural reliability. *Structural Safety*, **9**(3):161–177, 1991.

[4] Karl Breitung. 40 years FORM: Some new aspects? *Probabilistic Engineering Mechanics*, **42**:71–77, 2015.

[5] Gerhart I Schuëller and Helmuth J Pradlwarter. Benchmark study on reliability estimation in higher dimensions of structural systems–an overview. *Structural Safety*, **29**(3):167–182, 2007.

[6] Siu-Kui Au and James L Beck. Estimation of small failure probabilities in high dimensions by Subset Simulation. *Probabilistic Engineering Mechanics*, **16**(4):263–277, 2001.

[7] Iason Papaioannou, Wolfgang Betz, Kilian Zwirglmaier, and Daniel Straub. MCMC algorithms for Subset Simulation. *Probabilistic Engineering Mechanics*, **41**:89–103, 2015.

[8] Konstantin M Zuev. Subset Simulation method for rare event estimation: An introduction. *Encyclopedia of Earthquake Engineering*, pages 1–25, 2015.

[9] Ziqi Wang, Marco Broccardo, and Junho Song. Hamiltonian Monte Carlo methods for Subset Simulation in reliability analysis. *Structural Safety*, **76**:51–67, 2019.

[10] Hamed Nikbakht and Konstantinos G. Papakonstantinou. Hamiltonian MCMC methods for estimating rare events probabilities in high-dimensional problems. *Journal of Computational Physics*, under review.

[11] Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics Letters B*, **195**(2):216–222, 1987.

[12] Radford M Neal. Bayesian learning for neural networks. *PhD Thesis, University of Toronto*, 1995.

[13] Radford M Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, **2**(11):2, 2011.

[14] Matthew D Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, **15**(1):1593–1623, 2014.

[15] Alexandros Beskos, Natesh Pillai, Gareth Roberts, Jesus-Maria Sanz-Serna, and Andrew Stuart. Optimal tuning of the Hybrid Monte Carlo algorithm. *Bernoulli*, **19**(5A):1501–1534, 2013.

[16] Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[17] Andrew Gelman, Hal S Stern, John B Carlin, David B Dunson, Aki Vehtari, and Donald B Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, 2013.

[18] John Kruschke. *Doing Bayesian Data Analysis: A tutorial with R, JAGS, and Stan*. Academic Press, 2014.

[19] Cole C Monnahan, James T Thorson, and Trevor A Branch. Faster estimation of Bayesian models in ecology using Hamiltonian Monte Carlo. *Methods in Ecology and Evolution*, **8**(3):339–348, 2017.

[20] Elena Akhmatskaya and Sebastian Reich. GSHMC: An efficient method for molecular simulation. *Journal of Computational Physics*, **227**(10):4934–4954, 2008.

[21] Nilesh Tripuraneni, Mark Rowland, Zoubin Ghahramani, and Richard Turner. Magnetic Hamiltonian Monte Carlo. *arXiv preprint arXiv:1607.02738*, 2016.

[22] Ziyu Wang, Shakir Mohamed, and Nando Freitas. Adaptive Hamiltonian and Riemann manifold Monte Carlo. In *International Conference on Machine Learning*, pages 1462–1470, 2013.

[23] Michael Hohenbichler and Rudiger Rackwitz. Non-normal dependent vectors in structural safety. *Journal of the Engineering Mechanics Division*, **107**(6):1227–1238, 1981.

[24] Armen Der Kiureghian and Pei-Ling Liu. Structural reliability under incomplete probability information. *Journal of Engineering Mechanics*, **112**(1):85–104, 1986.

[25] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer, 2006.

[26] Armen Der Kiureghian and Taleen Dakessian. Multiple design points in first and second-order reliability. *Structural Safety*, **20**(1):37–49, 1998.

[27] Christian G Bucher and Ulrich Bourgund. A fast and efficient response surface approach for structural reliability problems. *Structural Safety*, **7**(1):57–66, 1990.

[28] Roland Schöbi and Bruno Sudret. Structural reliability analysis for p-boxes using multi-level meta-models. *Probabilistic Engineering Mechanics*, **48**:27–38, 2017.

UNCECOMP 2019

Proceedings of the
3rd International Conference on
Uncertainty Quantification in Computational Sciences and Engineering

M. Papadrakakis, V. Papadopoulos, G. Stefanou (Eds)

First Edition, October 2019

ISBN: **978-618-82844-9-4**

Institute of Structural Analysis and Antiseismic Research
National Technical University of Athens, Greece