Available online at www.eccomasproceedia.org
Eccomas Proceedia UNCECOMP (2019) 600-619

ECCOMAS
Proceedia

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

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.*

Keywords: Ducted Propeller, Co-Kriging, Reliability-based Design Optimisation, Multi-fidelity Learning, Gaussian Markov Random Fields, Risk Averseness

ISSN:2623-3339
Peer-review under responsibility of the organizing committee of UNCECOMP 2019.
doi: 10.7712/120219.6363.18806

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

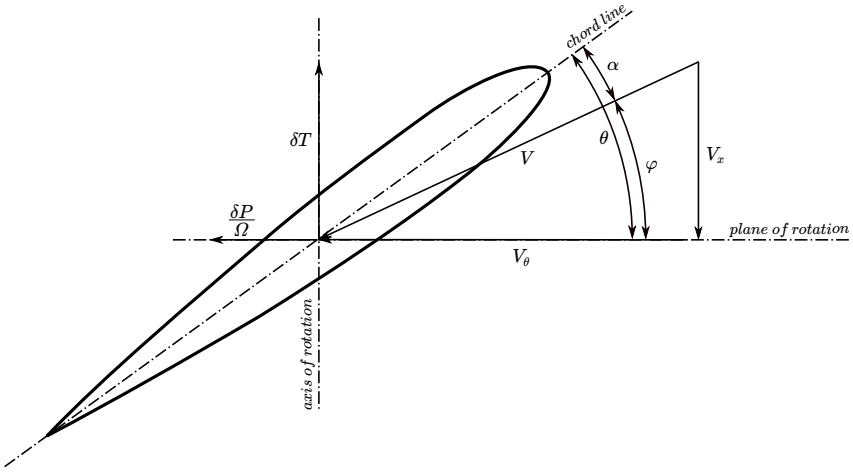


Figure 1: Blade Element velocities and forces

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:

$$\varphi = \tan \frac{V_x}{V_\theta}, \tag{6}$$

$$\alpha = \theta - \varphi, \tag{7}$$

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:

$$\delta T = \rho 4\pi r V_\infty^2 a(1+a)dr, \tag{8}$$

$$\delta P = \rho 4\pi r^3 V_\infty b(1+a)\omega^2 dr, \tag{9}$$

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):

$$\frac{a}{1+a} = \frac{\sigma_r}{4\sin^2(\varphi)}(C_l\cos(\varphi) - C_d\sin(\varphi)), \tag{10}$$

$$\frac{b}{1-b} = \frac{\sigma_r}{4\sin(\varphi)\cos(\varphi)}(C_d\cos(\varphi) + C_l\sin(\varphi)). \tag{11}$$

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

$$F\left(y_1,\ldots,y_n;\mathbf{s}_1,\ldots,\mathbf{s}_n\right) = P\left(X(\mathbf{s}_1)\le y_1,\ldots,X(\mathbf{s}_n)\le y_n\right)$$

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 $\mathbf{s}_1,\ldots,\mathbf{s}_n$,

$$\mathbf{x} \equiv \left(X(\mathbf{s}_1),\ldots,X(\mathbf{s}_n)\right)^T \sim N\left(\mu,\Sigma\right),$$

where $\Sigma_{ij} = \varsigma(\mathbf{s}_i; \mathbf{s}_j)$. For existence of a Gaussian field with a prescribed mean and covariance it is enough to ensure that ς is positive definite. A function $\varsigma(\mathbf{s}; \mathbf{t})$ is positive definite if for any finite set of locations $\mathbf{s}_1, \ldots, \mathbf{s}_n$ in $\mathcal{D}$, the covariance matrix

$$\mathbf{\Sigma} = \begin{pmatrix} \varsigma(\mathbf{s}_1, \mathbf{s}_1) & \varsigma(\mathbf{s}_1, \mathbf{s}_2) & \ldots & \varsigma(\mathbf{s}_1, \mathbf{s}_n) \\ \varsigma(\mathbf{s}_2, \mathbf{s}_1) & \varsigma(\mathbf{s}_2, \mathbf{s}_2) & \ldots & \varsigma(\mathbf{s}_2, \mathbf{s}_n) \\ \vdots & \vdots & \ddots & \vdots \\ \varsigma(\mathbf{s}_n, \mathbf{s}_1) & \varsigma(\mathbf{s}_n, \mathbf{s}_2) & \ldots & \varsigma(\mathbf{s}_n, \mathbf{s}_n) \end{pmatrix}$$

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 $z(\mathbf{s}) : \mathbf{s} \in \mathcal{D} \subset \mathbb{R}^d$, where $\mathbf{s}$ is the location of the process $z(\mathbf{s})$ and $\mathbf{s}$ varies over the index set $\mathcal{D}$.

In Kriging theory, the response $z(\mathbf{s})$ is considered as a realisation of a multivariate Gaussian process $Z(\mathbf{s})$. $Z(\mathbf{s})$ is assumed to be the sum of a deterministic regression function $m(\mathbf{s})$, constructed by observed data, and a Gaussian process $Y(\mathbf{s})$, constructed through the residuals:

$$Z(\mathbf{s}) = m(\mathbf{s}) + Y(\mathbf{s}). \tag{12}$$

The trend function $m(\mathbf{s})$ is assumed to be $m(\mathbf{s}) = \mathbf{h}(\mathbf{s})\boldsymbol{\beta}$, where $\mathbf{h}(\mathbf{s})$ is a set of p covariates associated with each site $\mathbf{s}$ and $\boldsymbol{\beta}$ is a p-dimensional vector of coefficients. $Y(\mathbf{s})$ is the Gaussian process with zero mean and covariance function $\Sigma_{ij} = \varsigma(\mathbf{s}_i, \mathbf{s}_j) = \sigma^2 c(\mathbf{s}_i, \mathbf{s}_j; \boldsymbol{\theta})$, 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 $z(\mathbf{s})$ at n known locations $\hat{\mathcal{D}} = (\mathbf{s}_1, \ldots, \mathbf{s}_n)^T \subset \mathcal{D}$. 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 $\sigma_\epsilon^2(\mathbf{s})$. 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 $Z(\mathbf{s})$ at the points $\hat{\mathcal{D}}$ and $\boldsymbol{\mathcal{E}}^{(n)} = (\sigma_\epsilon(\mathbf{s}_1)\mathcal{E}_1, \ldots, \sigma_\epsilon(\mathbf{s}_n)\mathcal{E}_n)^T$ 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 $Z(\mathbf{s})$ considering the joint distribution of $Z(\mathbf{s})$ and $\mathbf{Z}^{(n)}$:

$$\begin{pmatrix} Z(\mathbf{s}) \\ \mathbf{Z}^{(n)} \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{h}(\mathbf{s})\boldsymbol{\beta} \\ \mathbf{H}\boldsymbol{\beta} \end{pmatrix}, \begin{pmatrix} \varsigma(\mathbf{s}, \mathbf{s}) & \boldsymbol{\varsigma}^T(\mathbf{s}) \\ \boldsymbol{\varsigma}(\mathbf{s}) & \mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I} \end{pmatrix} \right), \tag{13}$$

where $\mathbf{H} = \mathbf{h}\left(\hat{\mathcal{D}}\right)$ is the $n \times p$ model matrix, $\mathbf{\Sigma}$ 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 $\mathbf{s}$ and the observation points $\hat{\mathcal{D}}$, σ_ϵ^2 is considered constant for simplicity.

Then, the conditional distribution $\left[Z(\mathbf{s}) \left| \mathbf{Z}^{(n)}, \boldsymbol{\beta}, \sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta} \right.\right]$ is Gaussian with mean and variance:

$$\hat{m}_Z(\mathbf{s}) = \mathbf{h}(\mathbf{s})\boldsymbol{\beta} + \boldsymbol{\varsigma}^T(\mathbf{s}) \left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right), \tag{14}$$

$$\hat{s}_Z^2(\mathbf{s}) = \varsigma(\mathbf{s}, \mathbf{s}) - \boldsymbol{\varsigma}^T(\mathbf{s}) \left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \boldsymbol{\varsigma}(\mathbf{s}). \tag{15}$$

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:

$$f\left(\mathbf{z}^{(n)} \left| \boldsymbol{\beta}, \sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta} \right.\right) = \frac{1}{(2\pi)^{n/2} \sqrt{|\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}|}} exp\left(-\frac{1}{2} \left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right)^T \left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \left(\mathbf{z}^{(n)} - \mathbf{H}\boldsymbol{\beta}\right)\right). \tag{16}$$

Given:

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{H}^T \left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \mathbf{H}\right)^{-1} \mathbf{H}^T \left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \mathbf{z}^{(n)}, \tag{17}$$

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:

$$\mathcal{L}\left(\sigma^2, \sigma_\epsilon^2, \boldsymbol{\theta}\right) = \left(\mathbf{z}^{(n)} - \mathbf{H}\hat{\boldsymbol{\beta}}\right)^T \left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)^{-1} \left(\mathbf{z}^{(n)} - \mathbf{H}\hat{\boldsymbol{\beta}}\right) + log\left(\left|\left(\mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}\right)\right|\right), \tag{18}$$

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 $\mathbf{s}$ 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

$$\mathcal{L}(\boldsymbol{\theta}) = n\, log\left(\hat{\sigma}^2\right) + log\left(|\mathbf{C}|\right). \tag{23}$$

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

$$p(X_i|\mathbf{X}_{-i}) = p(X_i|\{X_j|j \in \mathcal{N}_i\}), \tag{24}$$

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

$$f(\mathbf{X}) = (2\pi)^{-n/2}\,|\mathbf{Q}|^{1/2}\, exp\left(-\frac{1}{2}\mathbf{X}^T\mathbf{Q}\mathbf{X}\right). \tag{28}$$

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 $\phi_k = (\chi^2 - \Delta)^{1/2} \varphi_k$ 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 $\mathbf{Z}^{(n)} = \boldsymbol{\Phi}^T\mathbf{X} + \boldsymbol{\mathcal{E}}^{(n)}$, where $\boldsymbol{\Phi}$ 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

$$\mathbf{X} \sim \quad N(\boldsymbol{\mu}_x, \mathbf{Q}_x^{-1}), \tag{48}$$
$$\left(\mathbf{Z}^{(n)} \middle| \mathbf{X}\right) \sim \quad N\left(\boldsymbol{\Phi}\mathbf{X}, \mathbf{Q}_\epsilon^{-1}\right), \tag{49}$$

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:

$$\begin{pmatrix} \mathbf{X} \\ \mathbf{Z}^{(n)} \end{pmatrix} \sim N\left(\begin{pmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\Phi}\boldsymbol{\mu}_x \end{pmatrix}, \begin{bmatrix} \mathbf{Q}_x + \boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\boldsymbol{\Phi} & -\boldsymbol{\Phi}^T\mathbf{Q}_\epsilon \\ -\mathbf{Q}_\epsilon\boldsymbol{\Phi} & \mathbf{Q}_\epsilon \end{bmatrix}^{-1}\right). \tag{50}$$

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:

$$\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} = \boldsymbol{\mu}_x + \mathbf{Q}^{-1}_{\mathbf{X}|\mathbf{Z}^{(n)}}\boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\boldsymbol{\mu}_x\right), \tag{51}$$
$$\boldsymbol{\Sigma}_{\mathbf{X}|\mathbf{Z}^{(n)}} = \mathbf{Q}^{-1}_{\mathbf{X}|\mathbf{Z}^{(n)}}, \tag{52}$$
$$\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}} = \mathbf{Q}_x + \boldsymbol{\Phi}^T\mathbf{Q}_\epsilon\boldsymbol{\Phi}. \tag{53}$$

The variance can be computed as $\mathbf{s}^2_{\mathbf{X}|\mathbf{Z}^{(n)}} = diag\left(\mathbf{Q}^{-1}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right)$. 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

$$\hat{s}^2_Z(\mathbf{s}) = diag\left(\boldsymbol{\varphi}(\mathbf{s})\boldsymbol{\Sigma}_{\mathbf{X}|\mathbf{Z}^{(n)}}\boldsymbol{\varphi}^T(\mathbf{s})\right). \tag{55}$$

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 $\mathbf{X} = \hat{\mathbf{x}}$, the likelihood for $\mathbf{z}^{(n)}$ given the parameters $\boldsymbol{\theta}$ is:

$$\pi\left(\mathbf{z}^{(n)}\middle|\,\boldsymbol{\theta}\right) = \frac{\pi\left(\boldsymbol{\theta}\middle|\,\mathbf{z}^{(n)}\right)}{\pi\left(\boldsymbol{\theta}\right)} = \left.\frac{\pi\left(\mathbf{X}\middle|\,\boldsymbol{\theta}\right)\pi\left(\mathbf{z}^{(n)}\middle|\,\boldsymbol{\theta},\mathbf{X}\right)}{\pi\left(\mathbf{X}\middle|\,\boldsymbol{\theta},\mathbf{z}^{(n)}\right)}\right|_{\mathbf{X}=\hat{\mathbf{x}}} \tag{58}$$

so that the log-likelihood is:

$$\begin{aligned} log\,\pi\left(\mathbf{z}^{(n)}\middle|\,\boldsymbol{\theta}\right) = log\,\pi\left(\hat{\mathbf{x}}\middle|\,\boldsymbol{\theta}\right) + log\,\pi\left(\mathbf{z}^{(n)}\middle|\,\boldsymbol{\theta},\hat{\mathbf{x}}\right) - log\,\pi\left(\hat{\mathbf{x}}\middle|\,\boldsymbol{\theta},\mathbf{z}^{(n)}\right) = \\ -\frac{m+p}{2}log\left(2\pi\right) + \frac{1}{2}log\ |\mathbf{Q}_x| - \frac{1}{2}\left(\hat{\mathbf{x}}-\boldsymbol{\mu}_x\right)^T\mathbf{Q}_x\left(\hat{\mathbf{x}}-\boldsymbol{\mu}_x\right) \\ -\frac{n}{2}log\left(2\pi\right) + \frac{1}{2}log\ |\mathbf{Q}_\epsilon| - \frac{1}{2}\left(\mathbf{Z}^{(n)}-\boldsymbol{\Phi}\hat{\mathbf{x}}\right)^T\mathbf{Q}_\epsilon\left(\mathbf{Z}^{(n)}-\boldsymbol{\Phi}\hat{\mathbf{x}}\right) \\ +\frac{m+p}{2}log\left(2\pi\right) - \frac{1}{2}log\ \left|\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right| + \frac{1}{2}\left(\hat{\mathbf{x}}-\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right)^T\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}\left(\hat{\mathbf{x}}-\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right). \end{aligned} \tag{59}$$

In practice the likelihood for $\mathbf{z}^{(n)}$ given the parameters $\boldsymbol{\theta}$ is evaluated for $\hat{\mathbf{x}} = \mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}$, so that:

$$\begin{aligned} log\ \pi\left(\mathbf{z}^{(n)}\middle|\boldsymbol{\theta}\right) = -\frac{n}{2}log\left(2\pi\right) + \frac{1}{2}log\ \left|\mathbf{Q}_x\right| + \frac{1}{2}log\ \left|\mathbf{Q}_\epsilon\right| - \frac{1}{2}log\ \left|\mathbf{Q}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right| \\ -\frac{1}{2}\left(\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} - \boldsymbol{\mu}_x\right)^T \mathbf{Q}_x \left(\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}} - \boldsymbol{\mu}_x\right) \\ -\frac{1}{2}\left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right)^T \mathbf{Q}_\epsilon \left(\mathbf{Z}^{(n)} - \boldsymbol{\Phi}\mathbf{m}_{\mathbf{X}|\mathbf{Z}^{(n)}}\right). \end{aligned} \tag{60}$$

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 $(z_t(\mathbf{s}))_{t=1,\dots,l}$ sorted by increasing order of fidelity and modelled by Gaussian processes $(Z_t(\mathbf{s}))_{t=1,\dots,l}$, with $\mathbf{s} \in \mathcal{D}$. $z_l(\mathbf{s})$ is the most accurate and costly response and $(z_t(\mathbf{s}))_{t=1,\dots,l-1}$ are cheaper versions of it, with $z_1(\mathbf{s})$ the least accurate.

An auto-regressive model can be formulated for $t = 2, \dots, l$:

$$\begin{cases} Z_t(\mathbf{s}) = \rho_{t-1}(\mathbf{s})Z_{t-1}(\mathbf{s}) + \delta_t(\mathbf{s}), \\ Z_{t-1}(\mathbf{s}) \perp \delta_t(\mathbf{s}), \\ \rho_{t-1}(\mathbf{s}) = \mathbf{g}_{t-1}^T(\mathbf{s})\boldsymbol{\beta}_{\rho_{t-1}}, \end{cases} \tag{61}$$

where $\delta_t(\mathbf{s})$ is a Gaussian process, with mean $\mathbf{f}_t^T(\mathbf{s})\boldsymbol{\beta}_t$ and covariance function $\sigma_t^2 c_t(\mathbf{s}, \mathbf{s}')$, independent of $Z_{t-1}(\mathbf{s}), \dots, Z_1(\mathbf{s})$ and $\rho_{t-1}(\mathbf{s})$ represents a scale factor between $Z_t(\mathbf{s})$ and $Z_{t-1}(\mathbf{s})$. $\mathbf{g}_{t-1}(\mathbf{s})$ and $\mathbf{f}_t(\mathbf{s})$ are vectors of polynomial basis functions and $\boldsymbol{\beta}_{\rho_{t-1}}$ and $\boldsymbol{\beta}_t$ are the vectors of coefficients.

The Gaussian process $Z_t(\mathbf{s})$ modelling the response at level t is expressed as a function of the Gaussian process $Z_{t-1}(\mathbf{s})$ conditioned by the values $\mathbf{z}^{(t-1)} = (\mathbf{z}_1, \dots, \mathbf{z}_{t-1})$ at points in the experimental design sets $(\mathcal{D}_i)_{i=1,\dots,t-1}$.

Considering the joint distribution of $\delta_t(\mathbf{s}) = Z_t(\mathbf{s}) - \rho_{t-1}(\mathbf{s})Z_{t-1}(\mathbf{s})$ and $\delta_t(\mathcal{D}_t) = \mathbf{Z}^{(t)} - \rho_{t-1}(\mathcal{D}_t) \odot \mathbf{z}_{t-1}(\mathcal{D}_t)$:

$$\begin{pmatrix} Z_t(\mathbf{s}) - \rho_{t-1}(\mathbf{s})Z_{t-1}(\mathbf{s}) \\ \mathbf{Z}^{(t)} - \rho_{t-1}(\mathcal{D}_t) \odot \mathbf{z}_{t-1}(\mathcal{D}_t) \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{f}_t(\mathbf{s})\boldsymbol{\beta}_t \\ \mathbf{F}_t\boldsymbol{\beta}_t \end{pmatrix}, \begin{pmatrix} \mathbf{c}_t(\mathbf{s}, \mathbf{s}) & \mathbf{c}_t^T(\mathbf{s}) \\ \mathbf{c}_t(\mathbf{s}) & \mathbf{C}_t \end{pmatrix}\right), \tag{62}$$

we have for $t = 2, \dots, l$ and for $\mathbf{s} \in \mathcal{D}$:

$$\left[Z_t(\mathbf{s})\middle|\mathbf{Z}^{(t)} = \mathbf{z}^{(t)}, \boldsymbol{\beta}_t, \boldsymbol{\beta}_{\rho_{t-1}}, \sigma_t^2\right] \sim \mathcal{N}\left(\hat{m}_{Z_t}(\mathbf{s}), \hat{s}_{Z_t}^2(\mathbf{s})\right), \tag{63}$$

where:

$$\hat{m}_{Z_t}(\mathbf{s}) = \rho_{t-1}(\mathbf{s})\hat{m}_{Z_{t-1}}(\mathbf{s}) + \mathbf{f}_t^T(\mathbf{s})\boldsymbol{\beta}_t + \mathbf{c}_t^T(\mathbf{s})\mathbf{C}_t^{-1}(\mathbf{z}_t - \rho_{t-1}(\mathcal{D}_t) \odot \mathbf{z}_{t-1}(\mathcal{D}_t) - \mathbf{F}_t\boldsymbol{\beta}_t) \tag{64}$$

and:

$$\hat{s}_{Z_t}^2(\mathbf{s}) = \rho_{t-1}^2(\mathbf{s})\hat{s}_{Z_{t-1}}^2(\mathbf{s}) + \sigma_t^2\left(1 - \mathbf{c}_t^T(\mathbf{s})\mathbf{C}_t^{-1}\mathbf{c}_t(\mathbf{s})\right). \tag{65}$$

The notation $\odot$ represents the Hadamard product. $\mathbf{C}_t$ is the correlation matrix and $\mathbf{c}_t^T(\mathbf{s})$ is the correlation vector. We denote by $\rho_{t-1}(\mathcal{D}_t)$ the vector containing the values of $\rho_{t-1}(\mathbf{s})$ for $\mathbf{s} \in \mathcal{D}_t$, $\mathbf{z}_{t-1}(\mathcal{D}_t)$ the vector containing the known values of $Z_t(\mathbf{s})$ at points in $\mathcal{D}_t$ and $\mathbf{F}_t$ is the experience matrix containing the values of $\mathbf{f}_t(\mathbf{s})^T$ 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 $Z_t(\mathbf{s})$ are functions of mean and variance of the Gaussian process $Z_{t-1}(\mathbf{s})$.

The mean $\hat{\mu}_{Z_t}(\mathbf{s})$ 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 $\hat{s}^2_{Z_t}(\mathbf{s})$ 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:

$$log(|det(\mathbf{C}_t)|) + (n_t - p_t - q_{t-1})log(\hat{\sigma}_t^2) \tag{66}$$

for $t = 1, \dots, l$.

3.6 RECURSIVE GMRF-CO-KRIGING

Similarly to the classical recursive co-Kriging there are l levels of response $(z_t(\mathbf{s}))_{t=1,\dots,l}$ sorted by increasing order of fidelity.

An auto-regressive model using GMRF can be formulated for $t = 2, \dots, l$:

$$\begin{cases} Z_t(\mathbf{s}) = \boldsymbol{\varphi}^T(\mathbf{s})\mathbf{X}_t + \mathcal{E}_t(\mathbf{s}), \\ \mathbf{X}_t = \boldsymbol{\rho}_{t-1}^T\mathbf{X}_{t-1} + \boldsymbol{\delta}_t, \\ \mathbf{X}_{t-1} \perp \boldsymbol{\delta}_t, \end{cases} \tag{67}$$

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 $z_t(\mathbf{s})$ 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

$$\boldsymbol{\delta}_t \sim N(\boldsymbol{\mu}_{x_t}, \mathbf{Q}_{x_t}^{-1}), \tag{68}$$

$$\left(\mathbf{Z}_t^{(n_t)} \middle| \mathbf{X}_t\right) - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{X}_{t-1} \sim N\left(\boldsymbol{\Phi}_t\boldsymbol{\delta}_t, \mathbf{Q}_{\epsilon_t}^{-1}\right), \tag{69}$$

where $\mathbf{Q}_{\epsilon_t}^{-1} = \sigma_{\epsilon_t}^2\mathbf{I}$ 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:

$$\begin{gathered} \begin{pmatrix} \mathbf{X}_t - \boldsymbol{\rho}_{t-1}^T\mathbf{X}_{t-1} \\ \left(\mathbf{Z}_t^{(n_t)} \middle| \mathbf{X}_t\right) - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{X}_{t-1} \end{pmatrix} \sim \\ N\left(\begin{pmatrix} \boldsymbol{\mu}_{x_t} \\ \boldsymbol{\Phi}_t\boldsymbol{\mu}_{x_t} \end{pmatrix}, \begin{bmatrix} \mathbf{Q}_{x_t} + \boldsymbol{\Phi}_t^T\mathbf{Q}_{\epsilon_t}\boldsymbol{\Phi}_t & -\boldsymbol{\Phi}_t^T\mathbf{Q}_{\epsilon_t} \\ -\mathbf{Q}_{\epsilon_t}\boldsymbol{\Phi}_t & \mathbf{Q}_{\epsilon_t} \end{bmatrix}^{-1}\right). \end{gathered} \tag{70}$$

The conditional distribution for $\mathbf{X}_t$ given $\mathbf{Z}_t^{(n_t)}$ is $\left(\mathbf{X}_t | \mathbf{Z}_t^{(n_t)}\right) \sim N\left(\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}, \boldsymbol{\Sigma}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right)$, with:

$$\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \boldsymbol{\rho}_{t-1}^T\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_t-1})}} + \boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}, \tag{71}$$

$$\boldsymbol{\Sigma}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \boldsymbol{\rho}_{t-1}^T\boldsymbol{\Sigma}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_t-1})}}\boldsymbol{\rho}_{t-1} + \mathbf{Q}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}^{-1}, \tag{72}$$

$$\boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \boldsymbol{\mu}_{x_t} + \mathbf{Q}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}^{-1}\boldsymbol{\Phi}_t^T\mathbf{Q}_{\epsilon_t}\left(\mathbf{Z}_t^{(n_t)} - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_t-1})}} - \boldsymbol{\Phi}_t\boldsymbol{\mu}_{x_t}\right), \tag{73}$$

$$\mathbf{Q}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} = \mathbf{Q}_{x_t} + \boldsymbol{\Phi}_t^T\mathbf{Q}_{\epsilon_t}\boldsymbol{\Phi}_t. \tag{74}$$

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:

$$\begin{aligned} log\,\pi\left(\mathbf{z}_t^{(n_t)}\middle|\,\boldsymbol{\theta}_t\right) = &-\frac{n}{2}log\,(2\pi) + \frac{1}{2}log\;|\mathbf{Q}_{x_t}| + \frac{1}{2}log\;|\mathbf{Q}_{\epsilon_t}| - \frac{1}{2}log\;\left|\mathbf{Q}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right| \\ &-\frac{1}{2}\left(\boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} - \boldsymbol{\mu}_{x_t}\right)^T\mathbf{Q}_{x_t}\left(\boldsymbol{\mu}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}} - \boldsymbol{\mu}_{x_t}\right) \\ &-\frac{1}{2}\left(\mathbf{Z}_t^{(n_t)} - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_t-1})}} - \boldsymbol{\Phi}_t^T\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right)^T\mathbf{Q}_{\epsilon_t} \\ &\left(\mathbf{Z}_t^{(n_t)} - \boldsymbol{\rho}_{t-1}^T \odot \boldsymbol{\varphi}^T(\hat{\mathcal{D}}_t)\mathbf{m}_{\mathbf{X}_{t-1}|\mathbf{Z}_{\mathbf{t-1}}^{(\mathbf{n_t-1})}} - \boldsymbol{\Phi}_t^T\mathbf{m}_{\mathbf{X}_t|\mathbf{Z}_t^{(n_t)}}\right). \end{aligned} \tag{77}$$

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:

$$f_{high} = (6x - 2)^2 \sin(12x - 4), \tag{87}$$

$$f_{low} = \frac{1}{2} f_{high} + 10(x - 0.5) - 5. \tag{88}$$

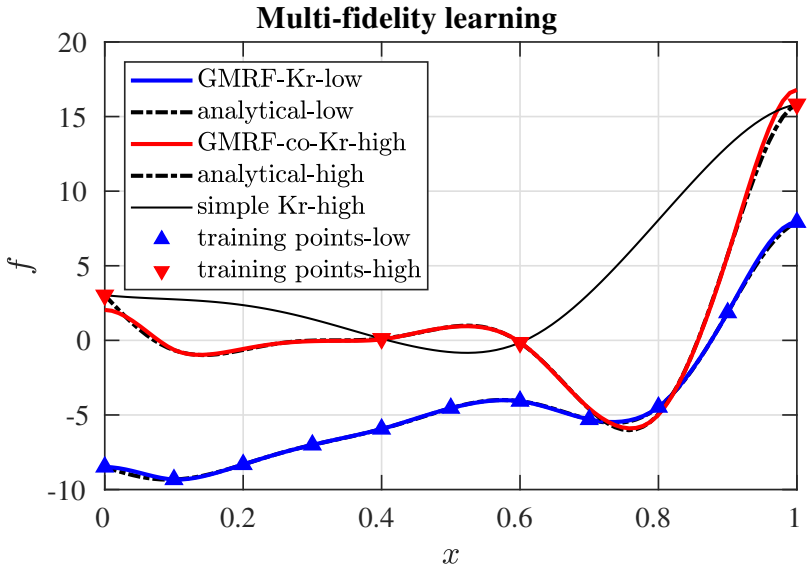


Figure 2: Multi-fidelity learning compared to single fidelity surrogate

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

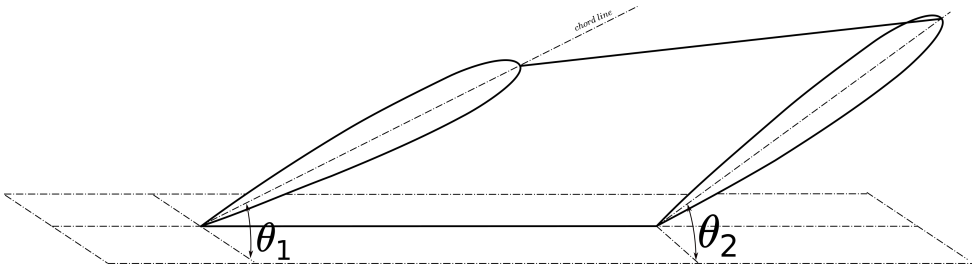


Figure 3: Design parameters of the propeller: twist at the root and at the tip.

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

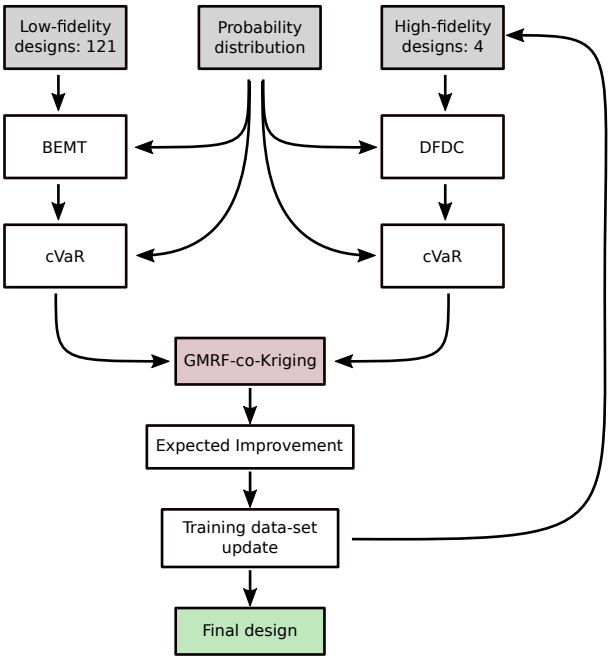


Figure 4: The optimisation workflow of the ducted propeller design optimisation

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.

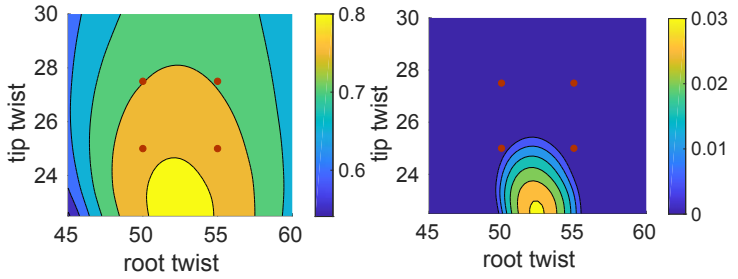
[3] A.F. El-Sayed. Aircraft propulsion and gas turbine engines. *CRC press*, 2017.

[4] H. Glauert. The elements of aerofoil and airscrew theory, *Cambridge University Press*, 1983.

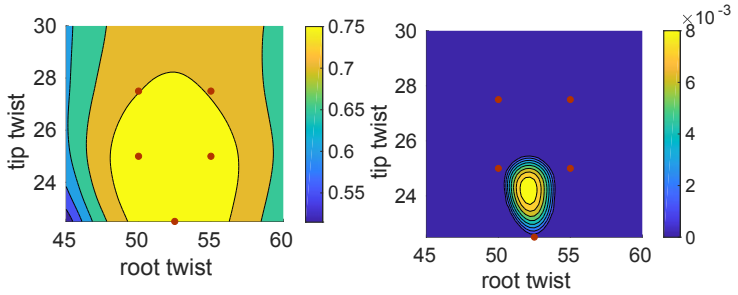
[5] L.L.M.Veldhuis. Propeller wing aerodynamic interference, *PhD thesis*, 2005.

[6] M.O.L. Hansen. Aerodynamics of wind turbines. *Routledge*, 2015.

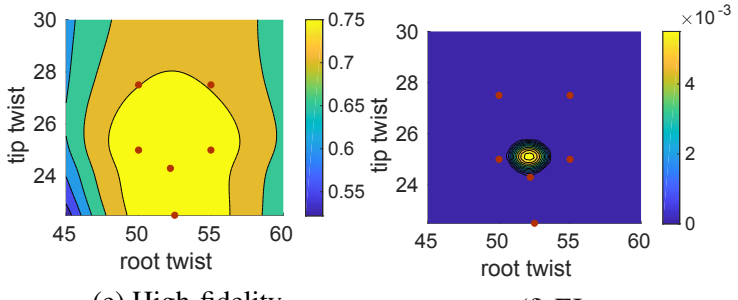
[7] M. Drela, H. Youngren, Ducted Fan Design Code (DFDC) - Axisymmetric Analysis and Design of Ducted Rotors, *Tech. Rep*, 2005



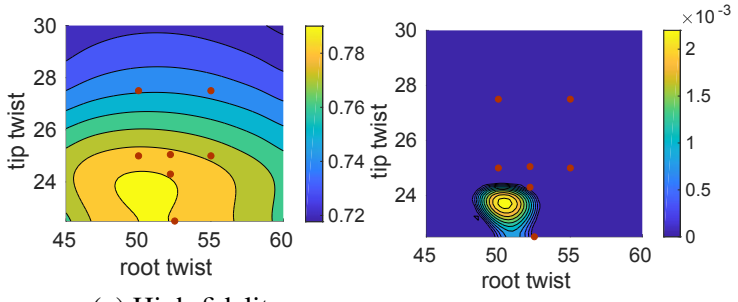
(a) High-fidelity (b) EI



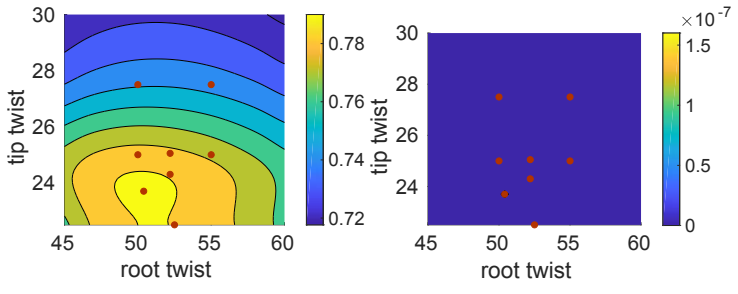
(c) High-fidelity (d) EI



(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.