Available online at www.eccomasproceedia.org
Eccomas Proceedia UNCECOMP (2021) 339-353

ECCOMAS
Proceedia

UNCECOMP 2021
4th ECCOMAS Thematic Conference on
Uncertainty Quantification in Computational Sciences and Engineering
M. Papadrakakis, V. Papadopoulos, G. Stefanou (eds.)
Streamed from Athens, Greece, 28 -30 June 2021

LINEAR ALGEBRA OF LINEAR AND NONLINEAR BAYESIAN CALIBRATION

Michaël Baudin[1], **Régis Lebrun**[2]

[1]EDF R&D
6 quai Watier, 78401, Chatou
e-mail: michael.baudin@edf.fr

[2] Airbus Group
6 rue Marceau, 92130 Issy-Les-Moulineaux
regis.lebrun@airbus.com

Abstract. *Calibration aims at combining observations with a model in order to reduce the discrepancy between the observations and the predictions of the model by updating its parameters. The Bayesian setup can improve the identifiability of the parameters of the model and amounts to regularizing the problem. The Gaussian prior is largely used by practitioners, mainly because of its ease of use and implementation. The naive implementation of the associated formula, however, can unnecessarily increase the condition number of the matrices involved in the process, in a similar way that the normal equations can increase the condition number of the matrix involved in the least squares problem. This means that the computed parameters may be more sensitive to changes in the data. In this paper, we present a way to use the Cholesky decomposition of the matrices involved in Bayesian calibration. It amounts to compute the Mahalanobis distance using the inverse of the Cholesky factors and leads to the expression of an extended residual which dimension is increased compared to the usual residual. This method can reduce the condition number of the matrices and lead to an improved accuracy in specific cases. We present applications of these ideas and an implementation of this method in the OpenTURNS library.*

Keywords: Linear Algebra, calibration, Bayesian, OpenTURNS

ISSN:2623-3339
Peer-review under responsibility of the organizing committee of UNCECOMP 2021.
doi: 10.7712/120221.8042.19074

Contents

1 Introduction

1.1 Purpose

When we want to assess the uncertainties in a computer code which simulates a physical system, calibration is an important step. Calibration reduces the discrepancies between the observations and the predictions of the model by adjusting the model parameters [10].

Least squares is the most widely used method of calibration. The implicit assumption of least squares is that the observation errors, i.e. the difference between the observations and the predictions, have a Gaussian distribution. When the computer code is linear, the least squares solution can be computed using linear algebra. In the more general case where the code is nonlinear, numerical optimization methods must be used. In both cases, problems arise when the solution is not identifiable or nearly so.

Bayesian calibration is a way to mitigate the lack of identifiability of the problem. When the prior distribution is not necessarily Gaussian, the posterior distribution of the parameter is generally not known. In this case, the most general-purpose algorithm is to use a Monte-Carlo Markov Chain (MCMC) algorithm such as the Metropolis-Hastings algorithm. The algorithm generates a sample which is designed to have the required posterior distribution. This generally requires many model evaluations, as the algorithm generates a distribution by conditional sampling. This issue can be partially solved by using a surrogate model.

In the special case where the prior is Gaussian, however, more detailed calculations can be handled and this is the main topic of this paper. In the case where the model is linear, the distribution of the posterior random vector is Gaussian, with known mean and covariance matrix. When the model is non linear, the distribution of the posterior is not known, but the

parameter value which has maximum density can be computed: this is the MAP estimator. It requires, however, to solve a non linear optimization problem.

When there is no prior distribution, the implementation of Gaussian least squares calibration based on the *normal equations* involve the Gramian matrix. These equation can be ill-conditioned, although it may happen that a special structure prevent this to happen. In the least squares context, the classical solution to this problem is to use the Cholesky decomposition or orthogonal decompositions such as the QR or SVD decomposition. In the Gaussian calibration framework, there is no known equivalent. The main purpose of this paper is to provide the formula for robust implementation of the linear and non linear Gaussian calibration. These formulas are based on the Cholesky decomposition.

This paper is structured as follows. In the first part, we present the Bayesian Gaussian calibration, and present a method to use the Cholesky decomposition for Gaussian calibration, both in linear and non linear cases.

1.2 Observations, model and parameters

We assume that we observe a quantity for different experimental conditions. These results are gathered in an observation vector $\mathbf{y}$ with dimension n, where n is the number of observations. On the other hand, we consider a model g which predicts this quantity depending on a set of experimental inputs $\mathbf{x}$ and the vector of parameters $\boldsymbol{\theta}$. These variables are formally introduced in the following definition.

Definition 1. *(*Calibration inputs*) Let $\mathbf{x} \in \mathbb{R}^m$ be the vector of experimental inputs and let $\mathbf{x}_1$, ..., $\mathbf{x}_n \in \mathbb{R}^m$ the input observables of each experiment, where n is the number of observations.*

We denote by $\boldsymbol{\theta} \in \mathbb{R}^p$ the vector of parameters to calibrate, where p is the number of parameters.

We make the hypothesis that the vector of predictions is produced by the computer model $g : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$. In other words, we assume that the relation between the i-th prediction of the computer model and the i-th experimental condition $\mathbf{x}_i$ is:

$$z_i = g(\mathbf{x}_i, \boldsymbol{\theta}) \in \mathbb{R}$$

for $i = 1, ..., n$. The vector of predictions is $\mathbf{z} \in \mathbb{R}^n$. Let $\mathbf{y} \in \mathbb{R}^n$ be the vector of observations.

In other words, the input data of calibration are the observations $\mathbf{y}$, the real scalar function g and the experimental conditions $\mathbf{x}_1$, ..., $\mathbf{x}_n$.

We make the hypothesis that $n \geq p$, i.e. the number of observations is greater than the number of parameters, which leads to an over-determined problem. Although some methods that we are going to present can be applied when $n < p$ (especially Bayesian methods), this is a situation that we do not often see in our applications, and this is why we do not consider this case in this paper.

In the remaining of the presentation, the *explicit* dependence from z_i to $\mathbf{x}_i$ is not relevant for the development and the analysis of calibration algorithms. Let $\mathbf{h} : \mathbb{R}^p \to \mathbb{R}^n$ be the function which i-th component is:

$$h_i(\boldsymbol{\theta}) = g(\mathbf{x}_i, \boldsymbol{\theta})$$

for $i = 1, ..., n$ and any $\boldsymbol{\theta} \in \mathbb{R}^p$. We then use the following compact notation:

$$\mathbf{z} = \mathbf{h}(\boldsymbol{\theta})$$

where $\mathbf{h}(\boldsymbol{\theta}) = (h_1(\boldsymbol{\theta}), ..., h_n(\boldsymbol{\theta}))^T \in \mathbb{R}^n$.

With these notations, the input data of calibration are the observations $\mathbf{y}$ and the vector model $\mathbf{h}$.

The following definition introduces the probabilistic hypothesis in which each observation y_i is the sum of the prediction $h_i(\boldsymbol{\theta})$ and a random variable.

Definition 2. *(*Standard hypothesis of probabilistic calibration*) Let us assume that:*

$$\mathbf{y} = \mathbf{h}(\boldsymbol{\theta}^\star) + \boldsymbol{\epsilon} \tag{1}$$

where $\boldsymbol{\theta}^\star \in \mathbb{R}^p$ is a fixed, but unknown, value of the parameter $\boldsymbol{\theta}$ and $\boldsymbol{\epsilon} : \Omega \to \mathbb{R}^n$ is a random vector with zero mean:

$$\mathbf{E}(\boldsymbol{\epsilon}) = \mathbf{0} \in \mathbb{R}^n$$

and finite covariance. In the special case where the error covariance matrix is diagonal, then:

$$\mathbf{Cov}(\boldsymbol{\epsilon}) = (\sigma^\star)^2 \mathbf{I}$$

where $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix and $\sigma^\star > 0$ is the fixed unknown standard error. In the most general case,

$$\mathbf{Cov}(\boldsymbol{\epsilon}) = R^\star$$

where $R^\star \in \mathbb{R}^{n \times n}$ is a symmetric covariance matrix positive semi-definite.

In some situations, we say that the value $\boldsymbol{\theta}^\star$ is the "true" value of the parameter. This allows to distinguish it from the current value $\boldsymbol{\theta}$ and the estimate $\hat{\boldsymbol{\theta}}$.

We further restrict the previous hypotheses and suppose that the $R^\star$ is positive definite, such that the associated probability density function is well defined. This condition, more restrictive, implies that eigenvalues of the matrix are positive, which simplifies the definition of the multivariate Gaussian distribution that we will use.

The calibration of the model $\mathbf{h}$ aims at reducing the discrepancy between the predictions of the computer model $\mathbf{h}(\boldsymbol{\theta}) \in \mathbb{R}^n$ and the observations $\mathbf{y}$ by tuning the value of the parameter $\boldsymbol{\theta}$. For a given value of $\boldsymbol{\theta}$, the discrepancy between the predictions and the observations is the *residual*.

Definition 3. *(*Residuals*) The residual vector is:*

$$\mathbf{r}(\boldsymbol{\theta}) = \mathbf{y} - \mathbf{h}(\boldsymbol{\theta})$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

The equation 1 implies that, when the value of $\boldsymbol{\theta}$ is the "true" value $\boldsymbol{\theta}^\star$, then the residual is:

$$(\boldsymbol{\theta}^\star) = \mathbf{y} - \mathbf{h}(\boldsymbol{\theta}^\star) = \boldsymbol{\epsilon}.$$

In order to quantify the discrepancy between observations and predictions, we chose the Euclidean norm which is, as we are going to see soon, intrinsically related to the Gaussian distribution.

One of the goals of calibration is to compute the value of the true parameter $\boldsymbol{\theta}^\star$. To do this, we define the estimator $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}^\star$.

At this stage, we have not set any restriction on the distribution of the residual $\boldsymbol{\epsilon}$. More precisely, the distribution of $\boldsymbol{\epsilon}$ is not necessarily Gaussian. In the specific setting where $\boldsymbol{\epsilon}$ is Gaussian, however, we can get more details on the solution.

Since the observation vector $\mathbf{y}$ is a random variable, the estimator $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}^\star$ depending on $\mathbf{y}$ also is a random variable and a secondary goal of calibration is to get, if possible, the distribution of $\hat{\boldsymbol{\theta}}$ which models the uncertainty of calibration produced by the observation errors.

If the problem is ill conditioned, a small change in the observations $\mathbf{y}$ can lead to a significant change in $\hat{\boldsymbol{\theta}}$. Some methods (e.g. the BLUE, 3DVAR, regularised least squares, Bayesian inversion) integrate some regularisation which mitigates the impact of observation errors on the estimation $\hat{\boldsymbol{\theta}}$ and can manage the lack of identifiability. Some optimization methods like the Levenberg-Marquardt method [6, 7] also integrate some form of regularization. We will illustrate this topic in both the theoretical and practical parts of this paper.

When we calibrate some parameters in order to reduce the discrepancy between observations and measures, different methods with different names have, in fact, the same goal. We sometimes use the term *inversion* [5], *calibration*, be it Bayesian or not, least squares [3] or *data assimilation*, Kalman filter with, often, the same goals. In this paper, we will try to clarify the link between some of these methods.

The sections 2.4 and 2.6 present how to use the Cholesky decomposition in this framework which is, up to our best knowledge, an original contribution on this topic.

2 Bayesian calibration

In this section, we present methods to perform Bayesian calibration. More precisely, we focus on methods in which the distribution of the prior is Gaussian.

2.1 Gaussian distribution

The fundamental probability distribution function in this paper is the Gaussian distribution, which is introduced in the next definition.

Definition 4. *(*Absolutely continuous multivariate Gaussian distribution.*) Let* $\mathbf{X} \in \mathbb{R}^n$ *be a random vector in* n *dimensions with mean* $\mathbf{E}(\mathbf{X}) = \boldsymbol{\mu} \in \mathbb{R}^n$ *and covariance matrix* $\mathbf{Cov}(\mathbf{X}) = \Sigma$ *where* $\boldsymbol{\mu} \in \mathbb{R}^n$ *and* $\Sigma \in \mathbb{R}^{n \times n}$ *is a symmetric positive definite matrix. We say that* $\mathbf{X}$ *has an absolutely continuous Gaussian distribution if its probability density function is:*

$$f(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det(\Sigma)}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{2}$$

for any $\mathbf{x} \in \mathbb{R}^n$.

Since the matrix Σ is, by assumption, symmetric and positive definite, its determinant $\det(\Sigma)$ is nonzero, which garantees that the denominator of the previous fraction is correctly defined.

In the special case where $\mathbf{Cov}(\mathbf{X}) = \sigma^2 \mathbf{I}$ where $\boldsymbol{\mu} \in \mathbb{R}^n$, $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix and $\sigma > 0$, therefore:

$$f(\mathbf{x}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{\|\mathbf{x}-\boldsymbol{\mu}\|_2^2}{2\sigma^2}\right) \tag{3}$$

for any $\mathbf{x} \in \mathbb{R}^n$.

The previous equation makes clear why the Euclidian norm plays a central role in the Gaussian distribution.

2.2 Bayesian calibration

In this section, we present the Bayesian calibration and, more specifically, the Gaussian Bayesian calibration.

We begin by introducing the prior and posterior distributions.

Definition 5. *(*Bayesian calibration: prior and posterior distributions.*) We assume that the parameter $\boldsymbol{\theta} \in \mathbb{R}^p$ has a true value $\boldsymbol{\theta}^\star$ unknown, but constant. We make the hypothesis that the parameter $\boldsymbol{\theta}$ has a known distribution $p(\boldsymbol{\theta})$, called the prior distribution which represents the uncertainty of the true parameter value $\boldsymbol{\theta}^\star$. For any $\mathbf{y} \in \mathbb{R}^n$ such that $p(\mathbf{y}) > 0$, Bayes theorem states that the distribution of $\boldsymbol{\theta}$ given $\mathbf{y}$ is:*

$$p(\boldsymbol{\theta}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})}$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. The expression $p(\mathbf{y}|\boldsymbol{\theta})$ is the likelihood*. The distribution of $\boldsymbol{\theta}|\mathbf{y}$ is the* posterior distribution*. Since $\mathbf{y}$ is observed, the denominator of the previous fraction is constant, so that the posterior distribution is proportional to the numerator:*

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto p(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{4}$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

The likelihood $p(\mathbf{y}|\boldsymbol{\theta})$ is the conditional probability density function of the vector of observations $\mathbf{y}$.

We can now define the Bayesian calibration with Gaussian hypotheses, or, more briefly, the Gaussian calibration.

Definition 6. *(*Gaussian calibration*) We make the hypothesis that the parameter $\boldsymbol{\theta}$ has a Gaussian distribution with known mean and covariance matrices :*

$$\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, B),$$

where $\boldsymbol{\mu} \in \mathbb{R}^p$ is the mean and $B \in \mathbb{R}^{p \times p}$ is the covariance matrix. The mean of the Gaussian distribution is called the background *in data assimilation.*

We make the hypothesis that the observations have the following conditional distribution:

$$\mathbf{y}|\boldsymbol{\theta} \sim \mathcal{N}(\mathbf{h}(\boldsymbol{\theta}), R),$$

where $R \in \mathbb{R}^{n \times n}$ is the covariance matrix of observations.

In this Bayesian setting, the values of $\boldsymbol{\mu}$, B et R are known *beforehand*.

Let us emphasise that in the Gaussian calibration framework, there are two Gaussian distributions: the first is the prior distribution of $\boldsymbol{\theta}$ and the second is the $\mathbf{y}$ distribution.

The following theorem is given in [9], p.22-23.

Theorem 1. *(*Posterior distribution of Gaussian calibration*) We consider hypotheses of definition 6. We denote by $\|\cdot\|_B$ the Mahalanobis distance associated with the matrix B:*

$$\|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 = (\boldsymbol{\theta} - \boldsymbol{\mu})^T B^{-1} (\boldsymbol{\theta} - \boldsymbol{\mu}),$$

for nay $\boldsymbol{\theta}, \boldsymbol{\mu} \in \mathbb{R}^p$*. We denote by* $\|\cdot\|_R$ *the Mahalanobis distance associated with the matrix* R*:*

$$\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 = (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))^T R^{-1} (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})).$$

Therefore, the posterior distribution of $\boldsymbol{\theta}$ *given the observations* $\mathbf{y}$ *are:*

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp\left(-\frac{1}{2}\left(\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 + \|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2\right)\right) \tag{5}$$

pour tout $\boldsymbol{\theta} \in \mathbb{R}^p$*.*

Data assimilation identifies the vector $\hat{\boldsymbol{\theta}}$ associated with the maximum value of the posterior distribution : this is the maximum *a posteriori* estimate or MAP.

Theorem 2. *(*MAP estimator of Gaussian calibration*) We consider hypotheses of definition 6. The maximum of the posterior distribution of* $\boldsymbol{\theta}$ *given the observations* $\mathbf{y}$ *is reached for:*

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2}\left(\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 + \|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2\right). \tag{6}$$

We can expand the equation 6 using the definition of the Mahalanobis norm, which leads to the minimisation of the cost function ([1], p.20, p.53):

$$c(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))^T R^{-1} (\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\mu})^T B^{-1} (\boldsymbol{\theta} - \boldsymbol{\mu}) \tag{7}$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

2.3 Gaussian non linear calibration

The solution of the data assimilation problem is the solution of the optimization problem defined by theorem 2.

Definition 7. *(*Cost function of non linear Gaussian calibration*) The cost function of the Gaussian nonlinear calibration is:*

$$c(\boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 + \frac{1}{2}\|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 \tag{8}$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$*. We consider the hypotheses of definition 6. The maximum of the posterior distribution of* $\boldsymbol{\theta}$ *given the observations* $\mathbf{y}$ *is reached at:*

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\operatorname{argmin}}\, c(\boldsymbol{\theta}). \tag{9}$$

The 3DVAR algorithm aims at solving the optimization problem in which we search for the minimum of the cost function c. In general, this requires to use a nonlinear optimization algorithm.

The gradient of the cost function C can be explicitly defined depending on the matrix R and B and the gradient of the function $\mathbf{h}$, which can improve the performance of the optimization algorithm.

Theorem 3. *(*Unicity of the solution of the optimization problem.*) The Hessian matrix of the cost function of the Gaussian nonlinear calibration problem is symmetric and positive definite. Therefore, the solution of the problem associated with the cost function 9 is unique.*

2.4 Cholesky decomposition for the nonlinear Gaussian calibration

In this section, we present a nonlinear Gaussian calibration which uses the Cholesky decomposition of the covariance matrices.

Theorem 4. *(*Mahalanobis distance and Cholesky decomposition*) Let $L_B \in \mathbb{R}^{p\times p}$ be the Cholesky factor of the matrix B:*

$$B = L_B L_B^T$$

where L_B is a lower triangular matrix. Therefore, the Mahalanobis distance between $\boldsymbol{\theta} \in \mathbb{R}^p$ and $\boldsymbol{\mu} \in \mathbb{R}^p$ is:

$$\|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 = \|L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})\|_2^2,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. Let $L_R \in \mathbb{R}^{n\times n}$ be the Cholesky factor of the matrix R:

$$R = L_R L_R^T$$

where L_R is a lower triangular matrix. Therefore, the Mahalanobis distance between $\mathbf{y} \in \mathbb{R}^n$ and $\mathbf{h}(\boldsymbol{\theta}) \in \mathbb{R}^n$ is:

$$\|\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_R^2 = \|L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})\|_2^2,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$.

Proof. We have

$$\begin{aligned}
\|\boldsymbol{\theta} - \boldsymbol{\mu}\|_B^2 &= (\boldsymbol{\theta} - \boldsymbol{\mu})^T B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\
&= (\boldsymbol{\theta} - \boldsymbol{\mu})^T \left(L_B L_B^T\right)^{-1} (\boldsymbol{\theta} - \boldsymbol{\mu}) \\
&= (\boldsymbol{\theta} - \boldsymbol{\mu})^T (L_B^{-1})^T L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\
&= \left(L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})\right)^T L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})
\end{aligned}$$

which concludes the proof for the matrix B. The proof for the matrix R is similar. □

Theorem 5. *(*3DVAR and Cholesky decomposition*) Let $\mathbf{r}_e \in \mathbb{R}^{(p+n)\times(p+n)}$ be the extended residual defined by:*

$$\mathbf{r}_e(\boldsymbol{\theta}) = \begin{pmatrix} L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta})) \end{pmatrix} \tag{10}$$

Therefore, the solution of the 3DVAR problem is equivalently the solution of the non linear least squares problem in extended dimension:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{r}_e(\boldsymbol{\theta})\|_2^2. \tag{11}$$

Proof. Indeed, the theorem 4 allows to express the cost function as:

$$c(\boldsymbol{\theta}) = \frac{1}{2}\|L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}))\|_2^2 + \frac{1}{2}\|L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu})\|_2^2,$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. It is easy to see that the previous expression is the Euclidian norm of the extended residual defined in equation 10. □

2.5 Linear Gaussian calibration

In this section, we present the calibration with Gaussian prior distribution in the special case where the model is linear. This method provides the best linear unbiased estimator in this setting, and this is why it is sometimes called the *BLUE*. In data assimilation, this is called *Kalman filter*, with the difference that the Kalman filter is often used sequentially by updating the parameter $\boldsymbol{\theta}$ within an iterative loop, instead of being managed in just one pass as we do.

Let us assume that the function $\mathbf{h}$ is linear with respect to $\boldsymbol{\theta}$. In this special case, we can compute the solution of the problem by solving a linear system of equations.

Theorem 6. *(*Solution of linear Gaussian calibration.*) We consider the hypotheses of the definition 6. We assume that* $\mathbf{h}$ *is linear with respect to* $\boldsymbol{\theta}$*, i.e., for any* $\boldsymbol{\theta} \in \mathbb{R}^p$*, we have:*

$$\mathbf{h}(\boldsymbol{\theta}) = \mathbf{h}(\boldsymbol{\mu}) + J(\boldsymbol{\theta} - \boldsymbol{\mu}). \tag{12}$$

Let A *be the matrix:*

$$A = (B^{-1} + J^T R^{-1} J)^{-1}. \tag{13}$$

Let K *be the Kalman matrix defined by:*

$$K = A J^T R^{-1}. \tag{14}$$

Therefore, the unique maximum of the posterior distribution of $\boldsymbol{\theta}$ *given the observations is ([1], p.53):*

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\mu} + K(\mathbf{y} - \mathbf{h}(\boldsymbol{\mu})). \tag{15}$$

The estimator $\hat{\boldsymbol{\theta}}$ is now defined ; the next theorem introduces the distribution of this estimator.

Theorem 7. *(*Solution of the linear Gaussian calibration*) We consider the same hypotheses as in the theorem 6. Therefore :*

$$p(\boldsymbol{\theta}|\mathbf{y}) \propto \exp\left(\frac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^T A^{-1} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})\right) \tag{16}$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$ *where* $\hat{\boldsymbol{\theta}}$ *is defined by the equation 15 and the matrix* A *is given by the equation 13. In other words,*

$$\mathbf{Cov}\left(\hat{\boldsymbol{\theta}}\right) = A = \left(B^{-1} + J^T R^{-1} J\right)^{-1}.$$

The following theorem establishes the covariance of the bayesian Gaussian calibration ([9], p.36 and 66 and [1], p.93-95).

Theorem 8. *(*Covariance matrix of the linear Gaussian calibration*) Under the hypotheses of the theorem 6, we have:*

$$\hat{\boldsymbol{\theta}} \sim \mathcal{N}(\boldsymbol{\theta}, A)$$

where $\hat{\boldsymbol{\theta}}$ *is defined by the equation 15 and matrix* A *is defined by 13.*

2.6 Cholesky decomposition for linear Gaussian calibration

In this section, we present a linear Gaussian calibration method using the Cholesky decomposition of the covariance matrices. This method extends the method presented in section 2.4.

Theorem 9. *(*Solution of linear Gaussian calibration with Cholesky decomposition*) We consider the same hypotheses as in theorem 6. Therefore, the unique maximum of the posterior distribution of $\boldsymbol{\theta}$ given the observations $\mathbf{y}$ is equivalently defined as the solution of the linear least squares problem:*

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\operatorname{argmin}} \frac{1}{2} \left\| \bar{A}(\boldsymbol{\theta} - \boldsymbol{\mu}) - \bar{\mathbf{y}} \right\|_2^2. \tag{17}$$

where $\bar{A} \in \mathbb{R}^{(p+n)\times p}$ is the extended matrix:

$$\bar{A} = \begin{pmatrix} L_B^{-1} \\ -L_R^{-1} J \end{pmatrix} \tag{18}$$

and $\bar{\mathbf{y}} \in \mathbb{R}^{p+n}$ is the extended vector:

$$\bar{\mathbf{y}} = \begin{pmatrix} \mathbf{0} \\ -L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\mu})), \end{pmatrix}$$

where the p first components of $\bar{\mathbf{y}}$ are zero.

Proof. The proof uses the equation 10 in the special case where $\mathbf{h}$ is linear. The equation 12 implies:

$$\mathbf{y} - \mathbf{h}(\boldsymbol{\theta}) = \mathbf{y} - \mathbf{h}(\boldsymbol{\mu}) - J(\boldsymbol{\theta} - \boldsymbol{\mu}),$$

for any $\boldsymbol{\theta} \in \mathbb{R}^p$. We substitute the previous equation into the extended residual defined by the equation 10:

$$\begin{aligned}
\mathbf{r}_e(\boldsymbol{\theta}) &= \begin{pmatrix} L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ L_R^{-1}\left(\mathbf{y} - \mathbf{h}(\boldsymbol{\mu}) - J(\boldsymbol{\theta} - \boldsymbol{\mu})\right) \end{pmatrix} \\
&= \begin{pmatrix} L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\mu})) - L_R^{-1} J(\boldsymbol{\theta} - \boldsymbol{\mu}) \end{pmatrix} \\
&= \begin{pmatrix} L_B^{-1}(\boldsymbol{\theta} - \boldsymbol{\mu}) \\ -L_R^{-1} J(\boldsymbol{\theta} - \boldsymbol{\mu}) \end{pmatrix} + \begin{pmatrix} \mathbf{0} \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\mu})) \end{pmatrix} \\
&= \begin{pmatrix} L_B^{-1} \\ -L_R^{-1} J \end{pmatrix} (\boldsymbol{\theta} - \boldsymbol{\mu}) + \begin{pmatrix} \mathbf{0} \\ L_R^{-1}(\mathbf{y} - \mathbf{h}(\boldsymbol{\mu})) \end{pmatrix} \\
&= \bar{A}(\boldsymbol{\theta} - \boldsymbol{\mu}) - \bar{\mathbf{y}}
\end{aligned}$$

by definition of $\bar{A}$ and $\bar{\mathbf{y}}$. □

3 Calibration of an ill-conditioned exponential model

In this section, we consider the calibration of an exponential model with the linear Gaussian calibration (BLUE). The goal of this section is to quantify the influence of the condition number of the matrices involved onto the results and compare the numerical values obtained by the two methods.

We first consider the method presented in the section 2.5 which uses the Kalman matrix. Then we use the method presented in the section 2.6 which uses the Cholesky decomposition of the covariance matrices B and R.

i	1	2	3	4	5	6	7	8	9	10
y_i	7.125	-1.414	4.099	14.58	1.381	20.19	26.82	52.52	76.96	122.4

Table 1: A sample of ten observed input and the corresponding independent realizations of the observed outputs.

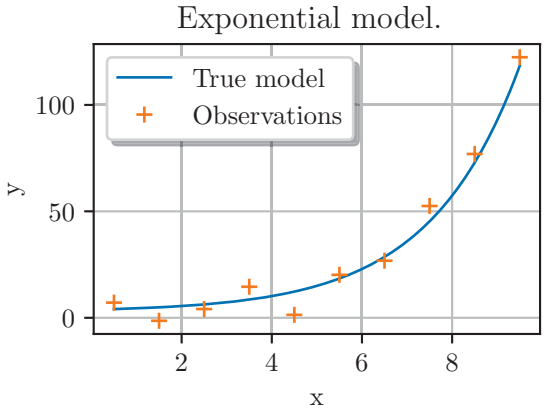

Figure 1: Observations in the exponential model.

3.1 Description of the model

We present the calibration of a model which uses the exponential function. Our aim is to provide the details of the experiment so that it can be reproduced by the interested reader.

The model is the function:

$$g(x) = \theta_1 + \exp(\theta_2 x)$$

for any $x \in [0.5, 9.5]$ where $\boldsymbol{\theta} \in \mathbb{R}^2$ is the vector of parameters. Notice that this model is linear with respect to the parameter θ_1, but not with respect to θ_2. We consider $n = 10$ observations. The observed values of the inputs are $x_i = i - 0.5$ for $i = 1, ..., n$. We assume that the observation error is the random variable $\epsilon \sim \mathcal{N}(0, 5)$, that is, an observation error which has the Gaussian distribution with zero mean and standard deviation equal to 5. We generate the noisy observations of the output by generating independent realization of the observation error:

$$y_i = g(x_i, \boldsymbol{\theta}) + \epsilon_i,$$

for $i = 1, ..., n$, where $\epsilon_1, \ldots, \epsilon_n$ are independent realizations of the random variable ϵ. The observations are the couples $\{(x_i, y_i)\}_{i=1,...,n}$.

The true values of the vector of parameters is $\boldsymbol{\theta}^\star = (2.8, 0.5)^T$.

In order to reproduce the experiment, we generated a sample of the observations once for all. This sample is presented in the table 1, rounded to 4 significant digits. Using a constant sample allows to reproduce the experiment more easily.

The figure 1 presents the observations and the model.

Let us now describe the parameters of the Gaussian calibration. The mean of the Gaussian prior is $\boldsymbol{\mu} = (1, 1)^T$. We use the prior covariance matrix:

$$B = \begin{pmatrix} 4 & \frac{1}{2} \\ \frac{1}{2} & 7 \end{pmatrix}.$$

Its condition number is $\kappa_2(B) = 1.807$, which is not perfect (because $\kappa_2(B) > 1$), but almost ideal because its order of magnitude is almost equal to the one of a perfectly conditioned matrix.

Moreover, we use a constant variance of the observation errors:

$$\sigma_Y^2 = 3.$$

The Jacobian matrix is computed based on symbolic differentiation:

$$J = \begin{pmatrix} 1 & 0.8243 \\ 1 & 6.723 \\ 1 & 30.46 \\ 1 & 115.9 \\ 1 & 405.1 \\ 1 & 1346 \\ 1 & 4323 \\ 1 & 1.356 \times 10^4 \\ 1 & 4.178 \times 10^4 \\ 1 & 1.269 \times 10^5 \end{pmatrix}$$

In the next paragraphs, we use the Gaussian linear calibration. Since the model is not linear with respect to θ_2, the BLUE estimator does not provide the exact solution of the calibration problem, but only computes the exact solution of the linearized problem. This is why the exact solution of this problem is not necessarily $\boldsymbol{\theta}^\star$, nor $\boldsymbol{\mu}$, but can only be computed from calculation.

3.2 Linear Gaussian calibration

We present the results with 4 significant digits (rounded to nearest), which is sufficient for our purpose. We use a "K" subscript for results which are obtained using the Kalman matrix and the "C" subscript for results which are obtained using the Cholesky decomposition.

Using the Kalman matrix, we get:

$$\hat{\boldsymbol{\theta}}_K = [0.9999, 0.8943].$$

When the computation uses the Kalman matrix, the covariance matrix is A, as defined by the equation 13:

$$A = \mathbf{Cov}\left(\hat{\boldsymbol{\theta}}_K\right) = \begin{pmatrix} 3.999 & -4.175 \times 10^{-5} \\ -4.175 \times 10^{-5} & 6.019 \times 10^{-10} \end{pmatrix}.$$

This leads to the 95% confidence interval:

$$\left(\hat{\theta}_1\right)_K \in [-3.267, 5.267] \qquad \left(\hat{\theta}_2\right)_K \in [0.8942, 0.8943].$$

With the method based on the extended linear least squares problem using the Cholesky decomposition, we get:

$$\hat{\boldsymbol{\theta}}_C = [-100.4, 0.8953].$$

For this method, the covariance matrix is the Gram matrix:

$$\mathbf{Cov}\left(\hat{\boldsymbol{\theta}}_C\right) = \begin{pmatrix} 0.3413 & -3.563 \times 10^{-6} \\ -3.563 \times 10^{-6} & 2.033 \times 10^{-10} \end{pmatrix}.$$

This leads to the following 95% confidence interval:

$$\left(\hat{\theta}_1\right)_C \in [-101.7, -99.08] \qquad \left(\hat{\theta}_2\right)_C \in [0.8953, 0.8954].$$

We observe that the results produced by the Kalman matrix seem to be correct, with a reduced confidence interval for the parameter θ_2, while the parameter θ_1 seem to have a relatively large confidence interval. Based on these results, it seem that the true of θ_1 is approximately in the interval from -3 to 5. This does not match the result obtained from the extended linear least squares method, which computes a relatively close value of the parameter θ_2, but has a parameter θ_1 much more negative, approximately in the interval from -102 to -99.

The figure 1 may seem to indicate that a value of θ_1 close to 0, as is the case for the method based on the Kalman matrix, may be more likely than a value of θ_1 close to -100, as is the case for the method using the Cholesky decomposition. This is, however, a false conclusion regarding the Gaussian linear calibration problem, which is different from the calibration that might be done visually. Firstly, the method uses a linearized model and, more importantly, it uses a Gaussian prior.

In order to see which result is more accurate, we use two complementary criteria.

- We evaluate the cost function defined by the equation 6: the best value is the lowest. This criterion evaluates the accuracy with respect to the consequences: in terms of stability analysis of algorithms, this the forward error analysis.
- We evaluate the condition number of the matrices involved in the methods: the best method has lowest condition numbers. This criterion evaluates the accuracy with respect to the sources: in terms of stability analysis of algorithms, this the backward error analysis.

3.3 Value of the cost function

The value of the cost function at both optimum points is equal to:

$$c\left(\hat{\boldsymbol{\theta}}_K\right) = 5.954 \times 10^4, \qquad c\left(\hat{\boldsymbol{\theta}}_C\right) = 4.449 \times 10^4.$$

We see that the method using the extended linear least squares problem has a smaller cost function value. Both methods use the same cost function, and differ only by the way they minimise it. Hence, the method which produces a lower value of the cost function achieves a better accuracy.

One possible cause for the difference in the function value may be that the function evaluation is associated with a loss of accuracy. This is in fact impossible because the matrices B and R involved in the Mahalanobis distance are respectively very well and perfectly conditioned. Therefore, the evaluation of the cost function cannot be affected by a massive loss of accuracy.

3.4 Condition number of the matrices

To see how the condition number may magnify the rounding errors in algebraic computations, we shortly describe the accuracy that can be expected when we use 64 bits floating point numbers. The IEEE754 standard for these numbers uses a precision of 53 bits, which leads to a unit roundoff approximately equal to 10^{-16}, that is approximately 16 significant digits (for normalized floating point numbers). We these numbers, when the condition number of function or algorithm is equal to 10^d, the maximum number of lost digits is equal to d (but this upper bound is not always reached). For example, if the condition number of an algorithm is equal to 10^4, therefore there are at least approximately $16 - 4 = 12$ significant digits in the result. This is why we, quite arbitrarily, write that the condition number is "low" when it is lower than 10^8

(because approximately half of the digits are corrects), "extreme" if it is greater than 10^{16} and "high" otherwise.

Let us now focus on the linear Gaussian calibration. The base 10 logarithm of the condition number of the Kalman matrix is extreme:

$$\log_{10}(\kappa_2(K)) = 20.87.$$

This shows that the use of the Kalman matrix can produce a massive loss of accuracy when we compute $\hat{\boldsymbol{\theta}}_K$ from it, with a potentially total loss of accuracy when we use 64 bits floating point numbers. The condition number of the covariance matrix of the parameter based on the Kalman matrix is also ill-conditioned:

$$\log_{10}\left(\kappa_2\left(\mathbf{Cov}\left(\hat{\boldsymbol{\theta}}_K\right)\right)\right) = 10.38,$$

although this condition number is not as bad as the previous one.

The condition number of the matrix $\bar{A}$ involved in the extended linear least squares problem 18 is:

$$\log_{10}\left(\kappa_2\left(\bar{A}\right)\right) = 4.656.$$

This shows that the matrix associated to the computation based on Cholesky decomposition is acceptable. The covariance matrix of $\hat{\boldsymbol{\theta}}_C$ is the Gram matrix of the extended linear least squares problem:

$$\mathbf{Cov}(\hat{\boldsymbol{\theta}}_C)) = \hat{\sigma}_C^2 \overline{G}^{-1}. \tag{19}$$

where $\hat{\sigma}_C^2$ is the unbiased estimator of the variance and $\overline{G}$ is the Gram matrix (also known as the information matrix) of the extended linear least squares problem:

$$\overline{G} = \bar{A}^T \bar{A}.$$

The base 10 logarithm of the condition number is:

$$\log_{10}\left(\kappa_2\left(\mathbf{Cov}(\hat{\boldsymbol{\theta}}_C)\right)\right) = 9.312,$$

which is relatively high. Let us notice, however, that the condition number of $\mathbf{Cov}(\hat{\boldsymbol{\theta}}_C)$ is necessarily equal to the square of the condition number of $\bar{A}$. Indeed, the equation 19 implies:

$$\kappa_2\left(\mathbf{Cov}(\hat{\boldsymbol{\theta}}_C)\right) = \kappa_2\left(\bar{A}\right).$$

Therefore,

$$\log_{10}\left(\kappa_2\left(\mathbf{Cov}(\hat{\boldsymbol{\theta}}_C)\right)\right) = 2\log_{10}\left(\kappa_2\left(\bar{A}\right)\right),$$

which confirms the numerical values we obtained with the exponential model, since $9.312 = 2 \times 4.656$.

In order to analyse in more depth the root causes of the condition numbers of the matrices, let us compute the condition number of the Jacobian matrix, which is involved in both methods:

$$\log_{10}\left(\kappa_2(J)\right) = 4.675.$$

This matrix, which must be managed by both two methods, has therefore a relatively low condition number. We see that its condition number is close to the one of the matrix $\bar{A}$, which is the expected result given the definition of $\bar{A}$. Hence, the method which uses the extended linear least squares problem does not artificially increase the condition number of the matrices, as opposed to the method which uses the Kalman matrix.

4 Conclusion

We have analysed the linear and non linear Gaussian calibration of a computer model and presented several methods to compute its solution. While the classical method, which uses the Kalman matrix, is mathematically satisfactory, its implementation in floating point arithmetic artificially reduces the accuracy of the solution and magnifies the errors in the data.

We presented a new method which involves an extended least squares problem. We have shown an example in which the new method actually performs with more accuracy. These methods are implemented in the OpenTURNS software [2].

REFERENCES

[1] Mark Asch, Marc Bocquet, and Maëlle Nodet. *Data assimilation. Methods, algorithms and applications.* SIAM, 2016.

[2] Michaël Baudin, Anne Dutfoy, Bertrand Iooss, and Anne-Laure Popelin. *OpenTURNS: An Industrial Software for Uncertainty Quantification in Simulation*, pages 2001–2038. Springer International Publishing, Cham, 2017.

[3] Ake Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial Applied Mathematics, 1996.

[4] Guillaume Damblin. Calage statistique des paramètres d'un modèle physique de condensation en thermohydraulique à l'échelle système. Technical report, DEN / DANS / DM2S / STMF / LGLS / NT / 2018-63096 / AA, 2018.

[5] P. C. Hansen. The l-curve and its use in the numerical treatment of inverse problems. In *in Computational Inverse Problems in Electrocardiology, ed. P. Johnston, Advances in Computational Bioengineering*, pages 119–142. WIT Press, 2000.

[6] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944.

[7] Donald Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM Journal on Applied Mathematics*, 11(2):431—-441, 1963.

[8] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley - Cambridge Press, 2009.

[9] Albert Tarantola. *Inverse problem theory*. SIAM, 2005.

[10] Timothy G Trucano, Laura Painton Swiler, Takera Igusa, William L Oberkampf, and Martin Pilch. Calibration, validation, and sensitivity analysis: What's what. *Reliability Engineering & System Safety*, 91(10-11):1331–1357, 2006.