International Journal for Uncertainty Quantification, 4 (5): 365–386 (2014)

RECURSIVE CO-KRIGING MODEL FOR DESIGN OF COMPUTER EXPERIMENTS WITH MULTIPLE LEVELS OF FIDELITY

Loic Le Gratiet[1,*] *& Josselin Garnier*[2]

[1] *Université Paris Diderot 75205 Paris Cedex 13, CEA, DAM, DIF, F-91297 Arpajon, France*

[2] *Laboratoire de Probabilites et Modeles Aleatoires & Laboratoire Jacques-Louis Lions, Universite Paris Diderot, 75205 Paris Cedex 13, France*

Original Manuscript Submitted: 01/04/2013; Final Draft Received: 12/18/2013

We consider in this paper the problem of building a fast-running approximation—also called surrogate model—of a complex computer code. The co-kriging based surrogate model is a promising tool to build such an approximation when the complex computer code can be run at different levels of accuracy. We present here an original approach to perform a multi-fidelity co-kriging model which is based on a recursive formulation. We prove that the predictive mean and the variance of the presented approach are identical to the ones of the original co-kriging model. However, our new approach allows to obtain original results. First, closed-form formulas for the universal co-kriging predictive mean and variance are given. Second, a fast cross-validation procedure for the multi-fidelity co-kriging model is introduced. Finally, the proposed approach has a reduced computational complexity compared to the previous one. The multi-fidelity model is successfully applied to emulate a hydrodynamic simulator.

KEY WORDS: *uncertainty quantification, surrogate models, universal co-kriging, recursive model, fast cross-validation, multi-fidelity computer code*

1. INTRODUCTION

Computer codes are widely used in science and engineering to describe physical phenomena. Advances in physics and computer science lead to increased complexity for the simulators. As a consequence, to perform a sensitivity analysis, an uncertainty quantification, or an optimization based on a complex computer code, a fast approximation of it—also called surrogate model—is built in order to avoid prohibitive computational cost. A very popular method of build ing asurrogate model is the Gaussian process regression, also named kriging. It corresponds to a particular class of surrogate models which makes the assumption that the response of the complex code is a realization of a Gaussian process. This method was originally introduced in geostatistics in [1] and it was then proposed in the field of computer experiments in [2]. During the last decades, this method has become widely used and investigated. The reader is referred to the books [3–5] for more detail about it.

Sometimes low-fidelity versions of the computer code are available. They may be less accurate but they are computationally cheap. A question of interest is how to build a surrogate model using data from simulations of multiple levels of fidelity. Our objective is hence to build a multi-fidelity surrogate model which is able to use the information obtained from the fast versions of the code. Such models have been presented in the literature [6], [7–11]. Besides, the highest-fidelity output could also correspond to field data and the low-fidelity ones could be obtained from physical models. In such case, the suggested methodology can be used in the context of validation (see [8, 12, 13]). Furthermore, in our framework the cheap code versions are not considered as computationally negligible. Therefore, they cannot be run intensively and considered as known as in [10].

*Correspond to Loic Le Gratiet, E-mail: loic.legratiet@cea.fr, URL: http://www.proba.jussieu.fr/l̃egratiet/

The first multi-fidelity model proposed in [6] is based on a linear regression formulation. Then this model is improved in [11] by using a Bayes linear formulation. The reader is referred to [14] for further detail about the Bayes linear approach. The methods suggested in [6, 11] have the strength to be relatively computationally cheap but as they are based on a linear regression formulation, they could suffer from a lack of accuracy. Another approach is to use an extension of kriging for multiple response models which is called co-kriging. The idea is implemented in [7], which presents a co-kriging model based on an autoregressive relation between the different code levels. This method turns out to be very efficient and it has been applied and extended significantly. In particular, the use of co-kriging for multi-fidelity optimization is presented in [9] and a Bayesian formulation is proposed in [10].

The strength of the co-kriging model is that it gives very good predictive models but it is often computationally expensive, especially when the number of simulations is large. Furthermore, large data sets can generate problems such as ill-conditioned covariance matrices. These problems are known for kriging but they become even more difficult for co-kriging since the total number of observations is the sum of the observations at all code levels.

In this paper, we adopt a new approach for multi-fidelity surrogate modeling which uses a co-kriging model but with an original recursive formulation. In fact, our model is able to build a s-level co-kriging model by building s independent krigings. An important property of this model is that it provides predictive mean and variance identical to the ones presented in [7]. However, our approach significantly reduces the complexity of the model since it divides the whole set of simulations into groups of simulations corresponding to the ones of each level. Therefore, we will have s submatrices to invert which is less expensive and ill-conditioned than a large one and the estimation of the parameters can be performed separately (Section 2.3).

Furthermore, a strength of our approach is that it allows to extend classical results of kriging to the considered co-kriging model. The two original results presented in our paper are the following ones: First, closed-form expressions for the universal co-kriging predictive mean and variance are given (Section 4). Second, the fast cross-validation method proposed in [15] is extended to the multi-fidelity co-kriging model (Section 5). Finally, we illustrate these results in a complex hydrodynamic simulator (Section 6).

2. MULTI-FIDELITY GAUSSIAN PROCESS REGRESSION

In Subsection 2.1, we briefly present the approach to build a multi-fidelity model suggested in [7] that uses a co-kriging model. In Subsection 2.2, we detail our recursive approach to build such a model. The recursive formulation of the multi-fidelity model is the first novelty of this paper. We will see in the next sections that the new formulation allows us to find original results about the co-kriging model and to reduce its computational complexity.

2.1 The Classical Autoregressive Model

Let us suppose that we have s levels of code $(z_t(x))_{t=1,\dots,s}$ sorted by increasing order of fidelity and modeled by Gaussian processes $(Z_t(x))_{t=1,\dots,s}$, $x \in U \subset \mathbb{R}^d$. x is a d-dimensional vector representing the input variables of the computer codes and U is the input parameter space. We hence consider that $z_s(x)$ is the most accurate and costly code that we want to surrogate and $(z_t(x))_{t=1,\dots,s-1}$ are cheaper versions of it with $z_1(x)$ the less accurate one. We consider the following autoregressive model with $t = 2, \dots, s$:

$$\begin{cases} Z_t(x) = \rho_{t-1}(x) Z_{t-1}(x) + \delta_t(x), \\ Z_{t-1}(x) \perp \delta_t(x), \\ \rho_{t-1}(x) = g_{t-1}^T(x)\beta_{\rho_{t-1}}, \end{cases} \tag{1}$$

where

$$\delta_t(x) \sim \mathcal{GP}(f_t^T(x)\beta_t, \sigma_t^2 r_t(x, x')), \tag{2}$$

and

$$Z_1(x) \sim \mathcal{GP}(f_1^T(x)\beta_1, \sigma_1^2 r_1(x, x')). \tag{3}$$

Here, ^{T} stands for the transpose, $\perp$ denotes the independence relationship, $\mathcal{GP}$ stands for Gaussian Process, $g_{t-1}(x)$ is a vector of q_{t-1} regression functions, $f_t(x)$ is a vector of p_t regression functions, $r_t(x, x')$ is a correlation function,

β_t is a p_t-dimensional vector, $\beta_{\rho_{t-1}}$ is a q_{t-1}-dimensional vector, and σ_t^2 is a positive real number. Since we suppose that the responses are realizations of Gaussian processes, the multi-fidelity model can be built by conditioning by the known responses of the codes at the different levels.

The previous model comes from the article [7]. It is induced by the following assumption: $\forall x \in U$; if we know $Z_{t-1}(x)$, nothing more can be learned about $Z_t(x)$ from $Z_{t-1}(x')$ for $x \neq x'$. It should be noticed that this Markov property does not imply constant adjustment coefficients $(\rho_{t-1}(x))_{t=2,\ldots,s}$. Indeed, we have for all $t = 2, \ldots, s$:

$$\rho_{t-1}(x) = \frac{\operatorname{cov}\left(Z_t(x), Z_{t-1}(x)\right)}{\operatorname{var}\left(Z_{t-1}(x)\right)}.$$

However, in the model presented in [7], the adjustment parameters $(\rho_t(x))_{t=2,\ldots,s}$ are constant. We show in a practical application (Section 6) that the extension to $(\rho_t(x))_{t=2,\ldots,s}$ depending on x is worthwhile.

Let us consider $\mathcal{Z}^{(s)} = (\mathcal{Z}_1^T, \ldots, \mathcal{Z}_s^T)^T$ the Gaussian vector containing the values of the random processes $(Z_t(x))_{t=1,\ldots,s}$ at the points of the designs of experiments (finite subsets of U) $(D_t)_{t=1,\ldots,s}$ with $D_s \subseteq D_{s-1} \subseteq \cdots \subseteq D_1$. We denote by $z^{(s)} = (z_1^T, \ldots, z_s^T)^T$ the vector containing the values of $(z_t(x))_{t=1,\ldots,s}$ at the points in $(D_t)_{t=1,\ldots,s}$. The nested property for the designs of experiments is not necessary to build the model but it allows for a simple estimation of the model parameters. Since the codes are sorted in increasing order of fidelity it is not an unreasonable constraint for practical applications. By denoting $\beta = (\beta_1^T, \ldots, \beta_s^T)^T$ the trend parameters, $\beta_\rho = (\beta_{\rho_1}^T, \ldots, \beta_{\rho_{s-1}}^T)^T$ the adjustment parameters, and $\sigma^2 = (\sigma_1^2, \ldots, \sigma_s^2)$ the variance parameters, we have for any $x \in U$:

$$[Z_s(x)|\mathcal{Z}^{(s)} = z^{(s)}, \beta, \beta_\rho, \sigma^2] \sim \mathcal{N}\left(m_{Z_s}(x), s_{Z_s}^2(x)\right),$$

where

$$m_{Z_s}(x) = h^{(s)}(x)^T\beta + t_s(x)^T(V^{(s)})^{-1}(z^{(s)} - H^{(s)}\beta), \tag{4}$$

and

$$s_{Z_s}^2(x) = v_{Z_s}^2(x) - t_s(x)^T(V^{(s)})^{-1}t_s(x). \tag{5}$$

The Gaussian process regression mean $m_{Z_s}(x)$ is the predictive model of the highest fidelity response $z_s(x)$ which is built with the known responses of all code levels $z^{(s)}$. The variance $s_{Z_s}^2(x)$ represents the predictive mean squared error of the model.

The matrix $V^{(s)}$ is the covariance matrix of the Gaussian vector $\mathcal{Z}^{(s)}$, the vector $t_s(x)$ is the vector of covariances between $Z_s(x)$ and $\mathcal{Z}^{(s)}$, $H^{(s)}\beta$ is the mean of $\mathcal{Z}^{(s)}$, $h^{(s)}(x)^T\beta$ is the mean of $Z_s(x)$, and $v_{Z_s}^2(x)$ is the variance of $Z_s(x)$. All these terms can be expressed in terms of the experience vector at level t (6) and of the covariance between $Z_t(x)$ and $Z_{t'}(x')$ (7)

$$h^{(t)}(x)^T = \left(\left(\prod_{i=1}^{t-1}\rho_i(x)\right)f_1^T(x), \left(\prod_{i=2}^{t-1}\rho_i(x)\right)f_2^T(x), \ldots, \rho_{t-1}(x)f_{t-1}^T(x), f_t^T(x)\right), \tag{6}$$

$$\operatorname{cov}(Z_t(x), Z_{t'}(x')|\sigma^2, \beta, \beta_\rho) = \left(\prod_{i=t'}^{t-1}\rho_i(x)\right)\operatorname{cov}(Z_{t'}(x), Z_{t'}(x')|\sigma^2, \beta, \beta_\rho), \quad \forall t > t', \tag{7}$$

and

$$\operatorname{cov}(Z_t(x), Z_t(x')|\sigma^2, \beta, \beta_\rho) = \sum_{j=1}^{t}\sigma_j^2\left(\prod_{i=j}^{t-1}\rho_i(x)\rho_i(x')\right)r_j(x, x'). \tag{8}$$

Furthermore, we have

$$H^{(s)} = \begin{pmatrix} [H^{(s)}]_{1,1} & \cdots & [H^{(s)}]_{1,s} \\ \vdots & \ddots & \vdots \\ [H^{(s)}]_{s,1} & \cdots & [H^{(s)}]_{s,s} \end{pmatrix},$$

where $[H^{(s)}]_{i,j}$ is a $n_i \times p_j$ matrix given by

$$[H^{(s)}]_{i,j} = \left(\bigodot_{k=j}^{i-1} \rho_k(D_i) \mathbf{1}_{p_j}^T \right) \odot f_j^T(D_i),$$

where $\mathbf{1}_{p_j}$ is a p_j-vector of ones, $\odot$ stands for the element by element matrix product, $\rho_k(D_i)$ is the vector containing the values of $\rho_k(x)$ for $x \in D_i$ and we use the convention $\left(\bigodot_{k=i}^{i-1} \rho_k(D_i) \mathbf{1}_{p_j}^T \right) = \mathbf{1}_{n_i} \mathbf{1}_{p_j}^T$.

Remark. If the cheap codes at levels $1, \ldots, t$, $t < s$, are computationally negligible, they can be considered as perfectly known and integrated into the regression functions $(f_i(x))_{i=1,\ldots,t}$.

2.2 Recursive Multi-Fidelity Model

In this section, we present the new recursive formulation of the multi-fidelity model. Let us consider the following model for $t = 2, \ldots, s$:

$$\begin{cases} Z_t(x) = \rho_{t-1}(x)\tilde{Z}_{t-1}(x) + \delta_t(x), \\ \tilde{Z}_{t-1}(x) \perp \delta_t(x), \\ \rho_{t-1}(x) = g_{t-1}^T(x)\beta_{\rho_{t-1}}, \end{cases} \tag{9}$$

where $\tilde{Z}_{t-1}(x)$ is a Gaussian process with distribution $[Z_{t-1}(x)|\mathcal{Z}^{(t-1)} = z^{(t-1)}, \beta_{t-1}, \beta_{\rho_{t-2}}, \sigma_{t-1}^2]$, $\delta_t(x)$ is a Gaussian process with distribution (2) and $D_s \subseteq D_{s-1} \subseteq \cdots \subseteq D_1$. The unique difference with the previous model is that we express $Z_t(x)$ (the Gaussian process modeling the response at level t) as a function of the Gaussian process $Z_{t-1}(x)$ conditioned by the values $z^{(t-1)} = (z_1, \ldots, z_{t-1})$ at points in the experimental design sets $(D_i)_{i=1,\ldots,t-1}$. As in the previous model, the nested property for the experimental design sets is assumed because it allows for efficient estimations of the model parameters but it is not required to derive the predictive distribution. We have for $t = 2, \ldots, s$ and for $x \in U$:

$$\left[Z_t(x)|\mathcal{Z}^{(t)} = z^{(t)}, \beta_t, \beta_{\rho_{t-1}}, \sigma_t^2 \right] \sim \mathcal{N}\left(\mu_{Z_t}(x), s_{Z_t}^2(x) \right), \tag{10}$$

where

$$\mu_{Z_t}(x) = \rho_{t-1}(x)\mu_{Z_{t-1}}(x) + f_t^T(x)\beta_t + r_t^T(x)R_t^{-1}\left(z_t - \rho_{t-1}(D_t) \odot z_{t-1}(D_t) - F_t\beta_t \right), \tag{11}$$

and

$$\sigma_{Z_t}^2(x) = \rho_{t-1}^2(x)\sigma_{Z_{t-1}}^2(x) + \sigma_t^2\left(1 - r_t^T(x)R_t^{-1}r_t(x) \right). \tag{12}$$

R_t is the correlation matrix $R_t = (r_t(x, x'))_{x,x' \in D_t}$, $r_t^T(x)$ is the correlation vector $r_t^T(x) = (r_t(x, x'))_{x' \in D_t}$, $z_t(D_{t-1})$ the vector containing the known values of $Z_t(x)$ at points in D_{t-1}, and F_t is the experience matrix containing the values of $f_t(x)^T$ on D_t.

The mean $\mu_{Z_t}(x)$ is the surrogate model of the response at level t, $1 \leq t \leq s$, taking into account the known values of the t first levels of responses $(z_i)_{i=1,\ldots,t}$ and the variance $\sigma_{Z_t}^2(x)$ represents the mean squared error of this model. The mean and the variance of the Gaussian process regression at level t being expressed in function of the ones of level $t-1$, we have a recursive multi-fidelity metamodel. Furthermore, in this new formulation, it is clearly emphasized that the mean of the predictive distribution does not depend on the variance parameters $(\sigma_t^2)_{t=1,\ldots,s}$. This is a classical result of kriging which states that for covariance kernels of the form $k(x, x') = \sigma^2 r(x, x')$, the mean of the kriging model is independent of σ^2. Another important strength of the recursive formulation is that contrary to the formulation suggested in [7], once the multi-fidelity model is built, it provides the surrogate models of all the responses $(z_t(x))_{t=1\ldots,s}$.

We have the following proposition.

Proposition 1. Let us consider nested designs of experiments $(D_t)_{t=1,\ldots,s}$, i.e., $D_s \subseteq D_{s-1} \subseteq \cdots \subseteq D_1$. We have the following equalities:

$$\begin{aligned} \mu_{Z_s}(x) &= m_{Z_s}(x), \\ \sigma_{Z_s}^2(x) &= s_{Z_s}^2(x), \end{aligned}$$

where $\mu_{Z_s}(x)$ and $m_{Z_s}(x)$ are defined in (11) and (4) and $\sigma^2_{Z_s}(x)$ and $s^2_{Z_s}(x)$ are defined in (12) and (5).

The proof of the proposition is given in Appendix A.1. It shows that the model presented in [7] and the recursive model (9) have the same predictive Gaussian distribution. Our objective in the next sections is to show that the new formulation (9) has several advantages compared to the one of [7]. First, its computational complexity is lower (Section 2.3); second, it provides closed-form expressions for the universal co-kriging mean and variance contrarily to [7] (Section 4); third, it makes it possible to implement a fast cross-validation procedure (Section 5).

2.3 Complexity Analysis

The computational cost is dominated by the inversion of the covariance matrices. In the original approach proposed in [7] one has to invert the matrix V_s of size $\sum_{i=1}^{s} n_i \times \sum_{i=1}^{s} n_i$ where $n_i = |D_i|$ denotes the number of observations at level $i = 1, \ldots, s$.

Our recursive formulation shows that building a s-level co-kriging is equivalent to build s consecutive krigings. This implies a reduction of the model complexity. Indeed, the inversion of s matrices $(R_t)_{t=1,\ldots,s}$ of size $(n_t \times n_t)_{t=1,\ldots,s}$ where n_t corresponds to the size of the vector z_t at level $t = 1, \ldots, s$ is less expensive than the inversion of the matrix V_s of size $\sum_{i=1}^{s} n_i \times \sum_{i=1}^{s} n_i$. We also reduce the memory cost since storing the s matrices $(R_t)_{t=1,\ldots,s}$ requires less memory than storing the matrix V_s. Finally, we note that the model with the recursive formulation is more interpretable since we can deduce the impact of each level of response into the model error through $(\sigma^2_{Z_t}(x))_{t=1,\ldots,s}$.

3. PARAMETER ESTIMATION

We deal in this section with the estimation of the model parameters. First, we describe the posterior distribution of $\psi = (\beta, \beta_\rho, \sigma^2)$ given the correlation kernels $(r_t(x, x'))_{t=1,\ldots,s}$ in Section 3.1. Then, we describe the considered method to estimate $(r_t(x, x'))_{t=1,\ldots,s}$ in Section 3.2.

3.1 Bayesian Estimation of Parameters

We present in this section a Bayesian estimation of the parameter $\psi = (\beta, \beta_\rho, \sigma^2)$ focusing on conjugate and non-informative distributions for the priors. This allows us to obtain closed-form expressions for the estimates of the parameters. Furthermore, from the non-informative case, we can obtain the estimates given by a maximum likelihood method. The presented formulas can hence be used in a frequentist approach. We note that the recursive formulation and the nested property of the experimental designs allow to separate the estimations of the parameters $(\beta_t, \beta_{\rho_{t-1}}, \sigma^2_t)_{t=2,\ldots,s}$ and (β_1, σ^2_1).

We address two cases in this section:

- Case (i): All the priors are informative
- Case (ii): All the priors are non-informative

It is of course possible to address the case of a mixture of informative and non-informative priors. For the non-informative case (ii), we use the "Jeffreys priors" [16]:

$$p(\beta_1|\sigma^2_1) \propto 1, \quad p(\sigma^2_1) \propto \frac{1}{\sigma^2_1}, \quad p(\beta_{\rho_{t-1}}, \beta_t|z^{(t-1)}, \sigma^2_t) \propto 1, \quad p(\sigma^2_t|z^{(t-1)}) \propto \frac{1}{\sigma^2_t}, \tag{13}$$

where $t = 2, \ldots, s$. For the informative case (i), we consider the following conjugate prior distributions:

$$[\beta_1|\sigma^2_1] \sim \mathcal{N}_{p_1}(b_1, \sigma^2_1 W_1),$$

$$[\beta_{\rho_{t-1}}, \beta_t|z^{(t-1)}, \sigma^2_t] \sim \mathcal{N}_{q_{t-1}+p_t}\left(b_t = \begin{pmatrix} b^\rho_{t-1} \\ b^\beta_t \end{pmatrix}, \sigma^2_t V_t = \sigma^2_t \begin{pmatrix} W^\rho_{t-1} & 0 \\ 0 & W^\beta_t \end{pmatrix}\right),$$

$$[\sigma^2_1] \sim \mathcal{IG}(\alpha_1, \gamma_1), \qquad [\sigma^2_t|z^{(t-1)}] \sim \mathcal{IG}(\alpha_t, \gamma_t),$$

with b_1 a vector of size p_1, b_{t-1}^{ρ} a vector of size q_{t-1}, b_t^{β} a vector of size p_t, W_1 a $p_1 \times p_1$ a correlation matrix, W_{t-1}^{ρ} a $q_{t-1} \times q_{t-1}$ correlation matrix, W_t^{β} a $p_t \times p_t$ correlation matrix, $\alpha_1, \gamma_1, \alpha_t, \gamma_t > 0$, and $\mathcal{IG}$ stands for the inverse Gamma distribution. The choice of conjugate Gaussian-inverse-Gamma priors is classic in the literature to perform Bayesian inference of multivariate Gaussian distribution (see [4]). These informative priors allow the user to prescribe the prior means and variances of all parameters. Furthermore, the choice of conjugate priors allows us to have closed-form expressions for the posterior distributions of the parameters (the reader is referred to [4] for more detail about the calculations). Indeed, we have:

$$[\beta_1 | z_1, \sigma_1^2] \sim \mathcal{N}_{p_1}(\Sigma_1 \nu_1, \Sigma_1), \qquad [\beta_{\rho_{t-1}}, \beta_t | z^{(t)}, \sigma_t^2] \sim \mathcal{N}_{q_{t-1}+p_t}(\Sigma_t \nu_t, \Sigma_t), \tag{14}$$

where, for $t \geq 2$:

$$\Sigma_t = \begin{cases} \left[H_t^T \dfrac{R_t^{-1}}{\sigma_t^2} H_t + \dfrac{W_t^{-1}}{\sigma_t^2} \right]^{-1} & (\mathbf{i}) \\ \left[H_t^T \dfrac{R_t^{-1}}{\sigma_t^2} H_t \right]^{-1} & (\mathbf{ii}) \end{cases}, \nu_t = \begin{cases} H_t^T \dfrac{R_t^{-1}}{\sigma_t^2} z_t + \dfrac{W_t^{-1}}{\sigma_t^2} b_t & (\mathbf{i}) \\ H_t^T \dfrac{R_t^{-1}}{\sigma_t^2} z_t & (\mathbf{ii}) \end{cases}, \tag{15}$$

with $H_1 = F_1$ and for $t > 1$, $H_t = [G_{t-1} \odot (z_{t-1}(D_t) \mathbf{1}_{q_{t-1}}^T) \, F_t]$ where G_{t-1} is the experience matrix containing the values of $g_{t-1}(x)^T$ in D_t and $\mathbf{1}_{q_{t-1}}$ is a q_{t-1}-vector of ones. Furthermore, we have for $t \geq 2$:

$$[\sigma_t^2 | z^{(t)}] \sim \mathcal{IG}\left(a_t, \frac{Q_t}{2}\right), \tag{16}$$

where

$$Q_t = \begin{cases} 2\gamma_t + (b_t - \hat{\lambda}_t)^T (W_t + [H_t^T R_t^{-1} H_t]^{-1})^{-1} (b_t - \hat{\lambda}_t) + \hat{Q}_t & (\mathbf{i}) \\ \hat{Q}_t & (\mathbf{ii}) \end{cases},$$

with $\hat{Q}_t = (z_t - H_t \hat{\lambda}_t)^T R_t^{-1} (z_t - H_t \hat{\lambda}_t)$, $\hat{\lambda}_t = (H_t^T R_t^{-1} H_t)^{-1} H_t^T R_t^{-1} z_t$, and

$$a_t = \begin{cases} \dfrac{n_t}{2} + \alpha_t & (\mathbf{i}) \\ \dfrac{n_t - p_t - q_{t-1}}{2} & (\mathbf{ii}) \end{cases},$$

with the convention $q_0 = 0$. One can note that the expression of Q_t for the case $(\mathbf{i})$ can be obtained thanks to the Woodbury matrix formula [17].

We highlight that the maximum likelihood estimators for the parameters β_1 and $(\beta_{\rho_{t-1}}, \beta_t)$ are given by the means of the posterior distributions in the non-informative case. Furthermore, the restricted maximum likelihood estimate of the variance parameter σ_t^2 can also be deduced from the posterior distribution of the Bayesian estimation in the non-informative case and is given by $\hat{\sigma}_{t,\mathrm{REML}}^2 = Q_t / 2a_t$. The restricted maximum likelihood estimation is a method which allows to reduce the bias of the maximum likelihood estimation [18].

3.2 Estimation of the Hyper-Parameters

In the previous sections, we have considered the correlation kernels $(r_t(x, x'))_{t=1,\ldots,s}$ as known. In practical applications, we choose these kernels in a parameterized family. Therefore, we consider kernels such that $r_t(x, x') = r_t(x, x'; \theta_t)$. For $t = 1, \ldots, s$ the hyperparameter θ_t can be estimated by maximizing the concentrated restricted log-likelihood [4] with respect to θ_t:

$$\log\left(\det\left(R_t\right)\right) + (n_t - p_t - q_{t-1}) \log\left(\hat{\sigma}_{t,\mathrm{REML}}^2\right), \tag{17}$$

with the convention $q_0 = 0$ and $\hat{\sigma}_{t,\mathrm{REML}}^2$ is the restricted likelihood estimate of the variance σ_t^2 (see Section 3.1). This minimization problem has to be solved numerically.

It is a common choice to estimate the hyperparameters by maximum likelihood [4]. It is also possible to estimate the hyperparameters $(\theta_t)_{t=1,\ldots,s}$ by minimizing a loss function of a Leave-One-Out Cross-Validation procedure. Usually, the complexity of this procedure is $\mathcal{O}\left(\left(\sum_{i=1}^{s} n_i\right)^4\right)$. Nonetheless, thanks to Proposition 3, it is reduced to $\mathcal{O}\left(\sum_{i=1}^{s} n_i^3\right)$ since it is essentially determined by the inversions of the s matrices $(R_t)_{t=1,\ldots,s}$. Therefore, the complexity for the estimation of $(\theta_t)_{t=1,\ldots,s}$ is substantially reduced. Furthermore, the recursive formulation of the problem allows us to estimate the parameters $(\theta_t)_{t=1,\ldots,s}$ one at a time.

4. UNIVERSAL CO-KRIGING PREDICTIVE MEAN AND VARIANCE

We can see in Eq. (10) that the predictive distribution of $Z_s(x)$ is conditioned by the observations $z^{(s)}$ and the parameters β, β_ρ, and σ^2. The objective of a Bayesian prediction is to integrate the uncertainty due to the parameter estimations into the predictive distribution. Indeed, in the previous subsection, we have expressed the posterior distributions of the variance parameters $(\sigma_t^2)_{t=1,\ldots,s}$ conditionally to the observations and the posterior distributions of the trend parameters β_1 and $(\beta_{\rho_{t-1}}, \beta_t)_{t=2,\ldots,s}$ conditionally to the observations and the variance parameters. Thus, using the Bayes formula, we can easily obtain a predictive distribution only conditioned by the observations by integrating into it the posterior distributions of the parameters.

As a result of this integration, the predictive distribution is not Gaussian. In particular, we cannot have a closed-form expression for the predictive distribution. However, it is possible to obtain closed-form expressions for the posterior mean $\mathbb{E}[Z_s(x)|\mathcal{Z}^{(s)} = z^{(s)}]$ and variance $\mathrm{Var}(Z_s(x)|\mathcal{Z}^{(s)} = z^{(s)})$.

The following proposition giving the closed-form expressions of the posterior mean and variance of the predictive distribution only conditioned by the observations is a novelty. The proof of this proposition is based on the recursive formulation which emphasizes the strength of this new approach. Indeed, the derivation of the posterior mean and variance from the model suggested in [7] involves complex algebra based on the evaluation of higher crossed moments of regression parameters and adjustment parameters and it has not been carried out anywhere as far as we know.

Proposition 2. Let us consider s Gaussian processes $(Z_t(x))_{t=1,\ldots,s}$ and $\mathcal{Z}^{(s)} = (\mathcal{Z}_t)_{t=1,\ldots,s}$ the Gaussian vector containing the values of $(Z_t(x))_{t=1,\ldots,s}$ at points in $(D_t)_{t=1,\ldots,s}$ with $D_s \subseteq D_{s-1} \subseteq \cdots \subseteq D_1$. If we consider the conditional predictive distribution in Eq. (10) and the posterior distributions of the parameters given in Eqs. (14) and (16), then we have for $t = 1, \ldots, s$:

$$\mathbb{E}[Z_t(x)|\mathcal{Z}^{(t)} = z^{(t)}] = h_t^T(x)\Sigma_t\nu_t + r_t^T(x)R_t^{-1}\left(z_t - H_t\Sigma_t\nu_t\right), \tag{18}$$

with $h_1^T = f_1^T$ and for $t > 1$, $h_t^T(x) = \left(g_{t-1}(x)^T\mathbb{E}[Z_{t-1}(x)|\mathcal{Z}_{t-1} = z_{t-1}]\, f_t^T(x)\right)$. Furthermore, we have

$$\begin{aligned}\mathrm{Var}(Z_t(x)|\mathcal{Z}^{(t)} = z^{(t)}) = \hat{\sigma}^2_{\rho_{t-1}}(x)\mathrm{Var}(Z_{t-1}(x)|\mathcal{Z}^{(t-1)} = z^{(t-1)}) + \frac{Q_t}{2\,(a_t-1)}\left(1 - r_t^T(x)R_t^{-1}r_t^T(x)\right)\\ + \left(h_t^T - r_t^T(x)R_t^{-1}H_t\right)\Sigma_t\left(h_t^T - r_t^T(x)R_t^{-1}H_t\right)^T,\end{aligned} \tag{19}$$

with $\hat{\sigma}^2_{\rho_{t-1}}(x) = \hat{\rho}^2_{t-1}(x) + g_{t-1}(x)^T\Sigma_{t,\rho}g_{t-1}(x)$, $\hat{\rho}_{t-1}(x) = g_{t-1}(x)^T[\Sigma_t\nu_t]_{1,\ldots,q_{t-1}}$, and $\Sigma_{\rho,t}$ is the submatrix of elements $(1,\ldots,q_{t-1}) \times (1,\ldots,q_{t-1})$ of Σ_t.

The proof of Proposition 2 is given in Appendix A.2. We note that, in the mean of the predictive distribution, the parameters have been replaced by their posterior means. Furthermore, in the variance of the predictive distribution, the variance parameter has been replaced by its posterior mean and the term $\left(h_t^T - r_t^T(x)R_t^{-1}H_t\right)\Sigma_t\left(h_t^T - r_t^T(x)R_t^{-1}H_t\right)^T$ has been added. It represents the uncertainty due to the estimation of the regression parameters (including the adjustment coefficient). We call these formulas the universal co-kriging equations due to their similarities with the well-known universal kriging equations (they are identical for $s = 1$).

5. FAST CROSS-VALIDATION FOR CO-KRIGING SURROGATE MODELS

The basic principle of validation is to split the experimental design set into two disjoint sets; one is used for training and the other for monitoring the performance of the surrogate model. Cross-validation extends it by considering

several different couples of test and training subsets. The idea is that the performances on the test sets can be used as a proxy for the generalization error. A particular case of this method is the Leave-One-Out Cross-Validation (noted LOO-CV) where as many test and training sets as observations are obtained by removing one observation at a time. Another, widely used procedure is the k-cross validation where the experimental design set is partitioned into k equal size subsets. Then, one of the subsets is used as a test set and the others are used for training. The procedure is repeated k times such that each subset is used once as a test set. These procedures can be time-consuming for a kriging model but it is shown in [5, 15, 19] that there are computational shortcuts. Our recursive formulation allows to extend these ideas to co-kriging models. Furthermore, the cross-validation equations proposed in this section extend the previous ones even for $s = 1$ (i.e., the classical kriging model) since they do not suppose that the regression and the variance coefficients are known and they concern every kind of cross-validation procedures (i.e., in particular the LOO-CV and the k-fold CV). Therefore, those parameters are re-estimated for each training set. We note that the re-estimation of the variance coefficient is a novelty which is important since fixing this parameter can lead to huge errors for the estimation of the cross-validation predictive variance when the number of observations is small or when the number of points in the test set is important.

If we denote by ξ_s the set of indices of n_{test} points in D_s constituting the test set D_{test} and ξ_t, $1 \leq t < s$, the corresponding set of n_{test} indices in D_t—indeed, we have $D_s \subset D_{s-1} \subset \cdots \subset D_1$, therefore $D_{\text{test}} \subset D_t$. The nested experimental design assumption implies that, in the cross-validation procedure, if we remove a set of points from D_s we can also remove it from D_t, $1 \leq t \leq s$.

The following proposition gives the vectors of the cross-validation predictive errors and variances at points in the test set D_{test} when we remove them from the levels u to s where $u \leq s$. In the proposition, we consider that we are in the non-informative case for the parameter estimation (see Section 3.1) but it can be easily extended to the informative case presented in Section 3.1.

Notations: If ξ is a set of indices, then $A_{[\xi,\xi]}$ is the submatrix of elements $\xi \times \xi$ of A, $a_{[\xi]}$ is the subvector of elements ξ of a, $B_{[-\xi]}$ represents the matrix B in which we remove the rows of index ξ, $C_{[-\xi,-\xi]}$ is the submatrix of C in which we remove the rows and columns of index ξ, and $C_{[-\xi,\xi]}$ is the submatrix of C in which we remove the rows of index ξ and keep only the columns of index ξ.

Proposition 3. Let us consider s Gaussian processes $(Z_t(x))_{t=1,\ldots,s}$ and $\mathcal{Z}^{(s)} = (\mathcal{Z}_t)_{t=1,\ldots,s}$ the Gaussian vector containing the values of $(Z_t(x))_{t=1,\ldots,s}$ at points in $(D_t)_{t=1,\ldots,s}$ with $D_s \subseteq D_{s-1} \subseteq \cdots \subseteq D_1$. We note D_{test} a set made up with the points of indices ξ_s of D_s and ξ_t the corresponding indices of the points in D_t with $1 \leq t \leq s$. Furthermore, we denote by $\lambda_{t,-\xi_t}$, $t = 1, \ldots, s$, the posterior mean of the regression and the adjustment parameters $(\beta^T_{\rho_{t-1}} \ \beta^T_t)^T$ processed without the observations indexed by ξ_t. Then, if we note ε_{Z_t,ξ_t} the errors (i.e., real values minus predicted values) of the cross-validation procedure at level t, $u \leq t \leq s$, when we remove the points of D_{test} from levels u to t, we have

$$\left(\varepsilon_{Z_t,\xi_t} - \hat{\rho}_{t-1}(D_{\text{test}}) \odot \varepsilon_{Z_{t-1},\xi_{t-1}}\right) \left[R_t^{-1}\right]_{[\xi_t,\xi_t]} \quad = \quad \left[R_t^{-1}\left(z_t - H_t\lambda_{t,-\xi_t}\right)\right]_{[\xi_t]}, \tag{20}$$

with $\varepsilon_{Z_i,\xi_i} = 0$ when $i < u$,

$$\lambda_{t,-\xi_t}\left([H_t^T]_{[-\xi_t]}K_t[H_t]_{[-\xi_t]}\right) = [H_t^T]_{[-\xi_t]}K_t z_t(D_t \setminus D_{\text{test}}),$$

$\hat{\rho}_{t-1}(D_{\text{test}}) = g_{t-1}^T(D_{\text{test}})[\lambda_{t,-\xi_t}]_{1,\ldots,q_{t-1}}$ and

$$K_t = \left[R_t^{-1}\right]_{[-\xi_t,-\xi_t]} - \left[R_t^{-1}\right]_{[-\xi_t,\xi_t]} \left(\left[R_t^{-1}\right]_{[\xi_t,\xi_t]}\right)^{-1} \left[R_t^{-1}\right]_{[\xi_t,-\xi_t]}. \tag{21}$$

Furthermore, if we note $\sigma^2_{Z_t,\xi_t}$ the variances of the corresponding cross-validation procedure, we have

$$\sigma^2_{Z_t,\xi_t} = \hat{\sigma}^2_{\rho_{t-1},-\xi_t}(D_{\text{test}}) \odot \sigma^2_{Z_{t-1},\xi_{t-1}} + \sigma^2_{t,-\xi_t}\text{diag}\left(\left(\left[R_t^{-1}\right]_{[\xi_t,\xi_t]}\right)^{-1}\right) + \mathcal{V}_s, \tag{22}$$

with $\Sigma_{\rho_{t-1},-\xi_t} = \left[\left([H_t^T]_{[-\xi_t]} K_t [H_t]_{[-\xi_t]}\right)^{-1}\right]_{[1,\ldots,q_{t-1},1,\ldots,q_{t-1}]}$,

$$\hat{\sigma}^2_{\rho_{t-1},-\xi_t}(D_{\text{test}}) = g_{t-1}^T(D_{\text{test}}) \left(\Sigma_{\rho_{t-1},-\xi_t} + [\lambda_{t,-\xi_t}]_{1,\ldots,q_{t-1}} [\lambda_{t,-\xi_t}]^T_{1,\ldots,q_{t-1}}\right) g_{t-1}(D_{\text{test}}),$$

and

$$\sigma^2_{t,-\xi_t} = \frac{\left(z_t(D_t \setminus D_{\text{test}}) - [H_t]_{[-\xi_t]}\lambda_{t,-\xi_t}\right)^T K_t \left(z_t(D_t \setminus D_{\text{test}}) - [H_t]_{[-\xi_t]}\lambda_{t,-\xi_t}\right)}{n_t - p_t - q_{t-1} - n_{\text{test}}}, \tag{23}$$

where $\sigma^2_{i,-\xi_i} = 0$ when $i < u$, n_{test} is the length of the index vector ξ_s, $H_t = [G_{t-1} \odot (z_{t-1}(D_t)\mathbf{1}^T_{q_{t-1}})\ F_t]$ and

$$\mathcal{V}_t = \mathcal{U}_t^T \left([H_t^T]_{[-\xi_t]} K_t [H_t]_{[-\xi_t]}\right)^{-1} \mathcal{U}_t, \tag{24}$$

with $\mathcal{U}_t = \left(\left([R_t^{-1}]_{[\xi_t,\xi_t]}\right)^{-1} \left[R_t^{-1} H_t\right]_{[\xi_t]}\right)$.

We note that these equations are also valid when $s = 1$, i.e., for the kriging model. We hence have closed-form expressions for the equations of a k-fold cross-validation with a re-estimation of the regression and variance parameters. These expressions can be deduced from the universal co-kriging equations. The complexity of this procedure is essentially determined by the inversion of the matrices $\left(\left[R_u^{-1}\right]_{[\xi_u,\xi_u]}\right)_{u=t,\ldots,s}$ of size $n_{\text{test}} \times n_{\text{test}}$. Furthermore, if we suppose the parameters of variance and/or trend as known, we do not have to compute $\sigma^2_{t,-\xi_t}$ and/or $\lambda_{t,-\xi_t}$ (they are fixed to their estimated value, i.e., $\sigma^2_{t,-\xi_t} = Q_t/[2(a_t - 1)]$ and $\lambda_{t,-\xi_t} = \Sigma_t \nu_t$; see Section 3.1) which reduces substantially the complexity of the method. Finally, the term $\mathcal{V}_s$ is the additive term due to the parameter estimations in the universal co-kriging. Therefore, if the trend parameters are supposed to be known, this term is equal to 0. The proof of Proposition 3 is given in Appendix A.3.

Remarks: We must recognize that our closed-form cross-validation formulas do not allow for the re-estimation of the hyperparameters of the correlation functions. However, as discussed in Subsection 3.2, Proposition 3 is useful even in that case to reduce the computational complexity of the cross-validation procedure. Furthermore, from the cross-validation predictive errors and variances, one can compute some overall measures of error in order to assess the relevance of the model [20].

6. ILLUSTRATION: HYDRODYNAMIC SIMULATOR

In this section we apply our co-kriging method to the hydrodynamic code "MELTEM" (see [21]). The aim of the study is to build a prediction as accurately as possible using only a few runs of the complex code and to assess the uncertainty of this prediction. In particular, we show the efficiency of the co-kriging model compared to the kriging one. We also illustrate the difference between simple and universal co-kriging and the results of the LOO-CV procedure. These illustrations are made possible and fast by the closed-form formulas for the predictive mean and variance for universal co-kriging and by the fast cross-validation procedure described in Section 5 and 4, respectively. Finally, we show that considering an adjustment coefficient $\rho_1(x)$ depending on x can be worthwhile.

The code MELTEM simulates a second-order turbulence model for gaseous mixtures induced by Richtmyer-Meshkov instability [21]. Two input parameters x_1 and x_2 are considered. They are phenomenological coefficients used in the equations of the energy of dissipation of the turbulent flow. These two coefficients vary in the region $[0.5, 1.5] \times [1.5, 2.3]$. The considered code outputs, called eps and L_c, are, respectively, the dissipation factor and the mixture characteristic length. The simulator is a finite-elements code which can be run at $s = 2$ levels of accuracy by altering the finite-elements mesh. The cheap code $z_1(.)$, using a coarse mesh, takes 15 s to produce an output whereas the expensive code $z_2(.)$, using a fine mesh, takes 8 min. We note that no prior information is available: We are hence in the noninformative case.

6.1 Nested Space-Filling Design

As presented in Section 2 we consider nested experimental design sets: $\forall t = 2, \ldots, s$, $D_t \subseteq D_{t-1}$. Therefore, we have to adopt particular design strategies to uniformly spread the inputs for all D_t.

We consider here another strategy for space-filling design, described in the following algorithm, which is very simple and not time-consuming. The number of points n_t for each design D_t is prescribed by the user, as well as the experimental design method applied to determine the coarsest grid D_s used for the most expensive code z_s (see [22] for a review of different methods).

Algorithm 1.

build $D_s = \{x_j^{(s)}\}_{j=1,\ldots,n_s}$ with the experimental design method prescribed by the user.

for t = s to 2 **do**:

build design $\tilde{D}_{t-1}$ with the experimental design method prescribed by the user.

for i = 1 to n_t **do**:

find $\tilde{x}_j^{(t-1)} \in \tilde{D}_{t-1}$ the closest point from $x_i^{(t)} \in D_t$ where $j \in [1, n_{t-1}]$.

remove $\tilde{x}_j^{(t-1)}$ from $\tilde{D}_{t-1}$.

end for

$D_{t-1} = \tilde{D}_{t-1} \cup D_t$.

end for

This strategy allows us to use any space-filling design method and it does not change the experimental design D_s of the most accurate code. This is not the case for a strategy based on selection of subsets of an experimental design for the less accurate code as presented in [7] and [9]. We hence can ensure that D_s has excellent space-filling properties. Moreover, the experimental design D_{t-1} being equal to $\tilde{D}_{t-1} \cup D_t$, this method ensures the nested property. Nevertheless, it alters the properties of the cheap code designs. Although this alteration is slight in our application, it could be much more severe in higher dimension. Indeed, in high dimension, the closest point to be removed may be far from the point in the upper level design. In that case, it could be relevant to not remove the point of the lower level design. Furthermore, to have good design properties for all levels, one can use the nested orthogonal array-based Latin hypercubes presented in [23]. However, this method has constraints on the number of observations per level.

In the presented application, we consider $n_2 = 5$ points for the expensive code $z_2(x)$ and $n_1 = 25$ points for the cheap one $z_1(x)$. We apply the previous algorithm to build D_2 and D_1 such that $D_2 \subset D_1$. For the experimental design set D_2, we use a Latin-Hypercube-Sampling [24] optimized with respect to the S-optimality criterion which maximizes the mean distance from each design point to all the other points [25]. Furthermore, the set $\tilde{D}_1$ is built using a maximum entropy design [26] optimized with the Fedorov-Mitchell exchange algorithm [27]. These algorithms are implemented in the R library lhs. The obtained nested designs are shown in Fig. 1.

6.2 Multi-Fidelity Surrogate Model for the Dissipation Factor eps

We build here co-kriging models for the dissipation factor eps. First, this example is used to illustrate the efficiency of the co-kriging method compared to the kriging in Section 6.2.1. Second, it will allow us to highlight the difference between the simple and the universal co-kriging in Section 6.2.2.

To build the different correlation matrices, we consider a tensorized matern $5/2$ kernel (see [5]):

$$r_t(x, x') = r(x, x'; \theta_t) = r_{1d}(x_1, x_1'; \theta_{t,1}) r_{1d}(x_2, x_2'; \theta_{t,2}), \tag{25}$$

with $x = (x_1, x_2) \in [0.5, 1.5] \times [1.5, 2.3]$, $\theta_{t,1}, \theta_{t,2} \in (0, +\infty)$ and

$$r_{1d}(x_i, x_i'; \theta_{t,i}) = \left(1 + \sqrt{5}\frac{|x_i - x_i'|}{\theta_{t,i}} + \frac{5}{3}\frac{(x_i - x_i')^2}{\theta_{t,i}^2}\right) \exp\left(-\sqrt{5}\frac{|x_i - x_i'|}{\theta_{t,i}}\right). \tag{26}$$

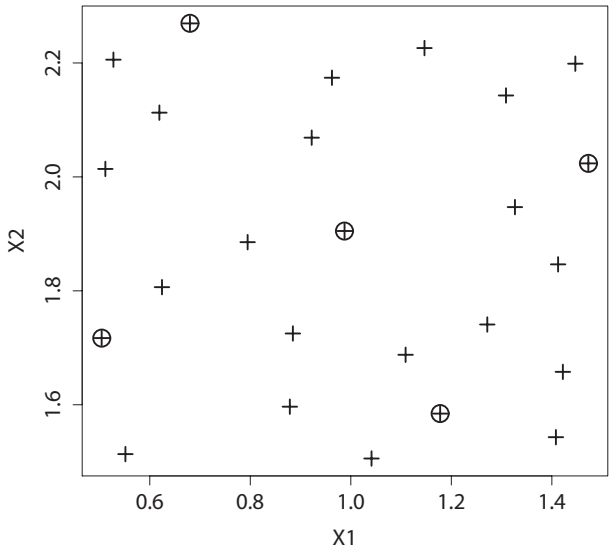

FIG. 1: Nested experimental design sets for the hydrodynamic application. The crosses represent the $n_1 = 25$ points of the experimental design set D_1 of the cheap code and the circles represent the $n_2 = 5$ points of the experimental design set D_2 of the expensive code.

Then, we consider $g_1(x) = 1$, $f_2(x) = 1$, $f_1(x) = 1$ (see Sections 2.1 and 2.2) and the hyperparameters (θ_1, θ_2) are estimated with a concentrated maximum likelihood method.

Furthermore, we build a test set of 175 points uniformly spread on $[0.5, 1.5] \times [1.5, 2.3]$ to test the accuracy of the models.

6.2.1 Comparison between Kriging and Multi-Fidelity Co-Kriging

First of all, we propose in this section a comparison between the kriging and co-kriging models when the number of runs n_2 for the expensive code varies such that $n_2 = 5, 10, 15, 20, 25$. For the co-kriging model, we consider $n_1 = 25$ runs for the cheap code.

To perform the comparison, we generate randomly 500 experimental design sets $(D_{2,i}, D_{1,i})_{i=1,\ldots,500}$ such that $D_{2,i} \subset D_{1,i}$, $i = 1, \ldots, 500$, $D_{1,i}$ has n_1 points, and $D_{2,i}$ has n_2 points.

The accuracies of the two models are evaluated on the test set composed of 175 observations. From them, the Root Mean Squared Error (RMSE) is computed: $\mathrm{RMSE} = \left[(1/175)\sum_{i=1}^{175}(\mu_{Z_2}(x_i^{\mathrm{test}}) - z_2(x_i^{\mathrm{test}}))^2\right]^{1/2}$.

Figure 2 gives the mean and the quantiles of probability 5% and 95% of the RMSE computed from the 500 sets $(D_{2,i}, D_{1,i})_{i=1,\ldots,500}$ when the number of runs for the expensive code n_2 varies. In Fig. 2, we can see that the errors converge to the same value when n_2 tends to n_1. Indeed, due to the Markov property given in Section 2.1, when $D_2 = D_1$, only the observations z_2 are taken into account. Furthermore, we can see that for small values of n_2, it is worth considering the co-kriging model since its accuracy is significantly better than the one of the kriging model.

6.2.2 Comparison between Simple and Universal Multi-Fidelity Co-Kriging

In this section, two comparisons are performed. The first one is between kriging and co-kriging models and the second is between simple and universal co-kriging. We remind the reader that for the simple co-kriging the trend and adjustment parameters are considered as known, whereas for the universal one their posterior distributions are integrated into the predictive distribution (see Section 4).

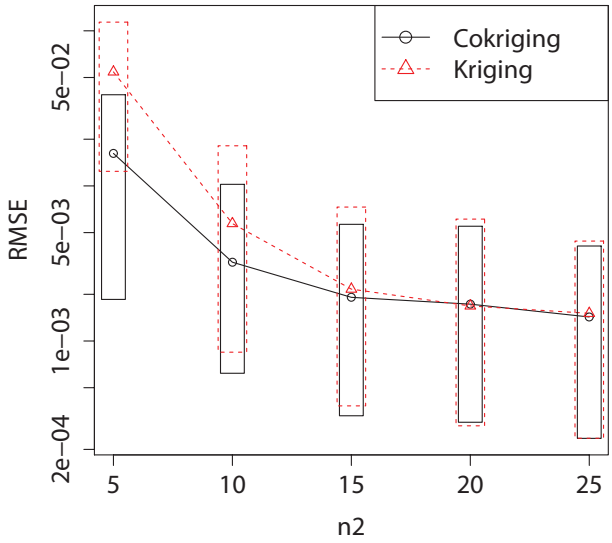

FIG. 2: Comparison between kriging and co-kriging with $n_1 = 25$ runs for the cheap code (500 nested design sets have been randomly generated for each n_2). The solid line represents the averaged RMSE of the co-kriging, the dashed line represents the averaged RMSE of the kriging, the dashed barplots represent the quantiles of probability 5% and 95% for the kriging RMSE and the solid barplots represent the quantiles of probability 5% and 95% of the co-kriging RMSE. Co-kriging predictions are better than the ordinary kriging ones for small n_2 and they converge to the same accuracy when n_2 tends to $n_1 = 25$.

In this subsection, we use 5 runs for the expensive code $z_2(x)$ and 25 runs for the cheap code $z_1(x)$. This represents 8 min on a hexa-core processor, which is our constraint for an operational use. This is actually a toy example but in industrial applications it is common to have a limited CPU budget on a multicore processor. Furthermore, the nested design sets are those built in Section 6.1 and illustrated in Fig. 1 and to validate and compare our models, the 175 simulations of the expensive code are used.

Using the concentrated maximum likelihood (see Section 3.2), we have the following estimations for the correlation hyperparameters: $\hat{\theta}_1 = (0.69, 1.20)$ and $\hat{\theta}_2 = (0.27, 1.37)$. According to the values of the hyperparameter estimates, the co-kriging model is smooth since the correlation lengths are of the same order as the size of the input parameter space. Furthermore, the estimated Pearson correlation between the two codes is 82.64%, which shows that the amount of information contained in the cheap code is substantial.

Table 1 presents the results of the parameter estimations (see Section 3.1). We see in Table 1 that the correlation between β_{ρ_1} and β_2 is important which highlights the importance of taking into account the correlation between these two coefficients for the parameter estimation. We also see that the adjustment parameter β_{ρ_1} is close to 1, which means that the two codes are highly correlated [we note that $g_1(x) = 1$, i.e., $\rho_1 = \beta_{\rho_1}$].

Figure 3 illustrates the contour plot of the kriging and co-kriging means; we can see significant differences between the two surrogate models.

Table 2 compares the prediction accuracy of the co-kriging and the kriging models. The different coefficients are estimated with the 175 responses of the expensive code on the test set:

- MaxAE: Maximal absolute value of the observed error.
- RMSE: Root mean squared value of the observed error.
- $Q_2 = 1 - ||\mu_{Z_2}(D_{\text{test}}) - z_2(D_{\text{test}})||^2 / ||\mu_{Z_2}(D_{\text{test}}) - \bar{z}_2||^2$, with $\bar{z}_2 = (\sum_{i=1}^{n_2} z_2(x_i^{\text{test}}))/n_2$.
- RIMSE: Root of the average value of the kriging or co-kriging variance.

TABLE 1: Application: hydrodynamic simulator. Parameter estimation results for the response eps [see Eqs. (14) and (16)]

Trend coefficient	$\Sigma_t \nu_t$	Σ_t / σ_t^2
β_1	8.84	0.48
$\begin{pmatrix} \beta_{\rho_1} \\ \beta_2 \end{pmatrix}$	$\begin{pmatrix} 0.92 \\ 0.74 \end{pmatrix}$	$\begin{pmatrix} 1.98 & -18.13 \\ -18.13 & 165.82 \end{pmatrix}$
Variance coefficient	Q_t	$2\alpha_t$
σ_1^2	6.98	24
σ_2^2	0.06	3

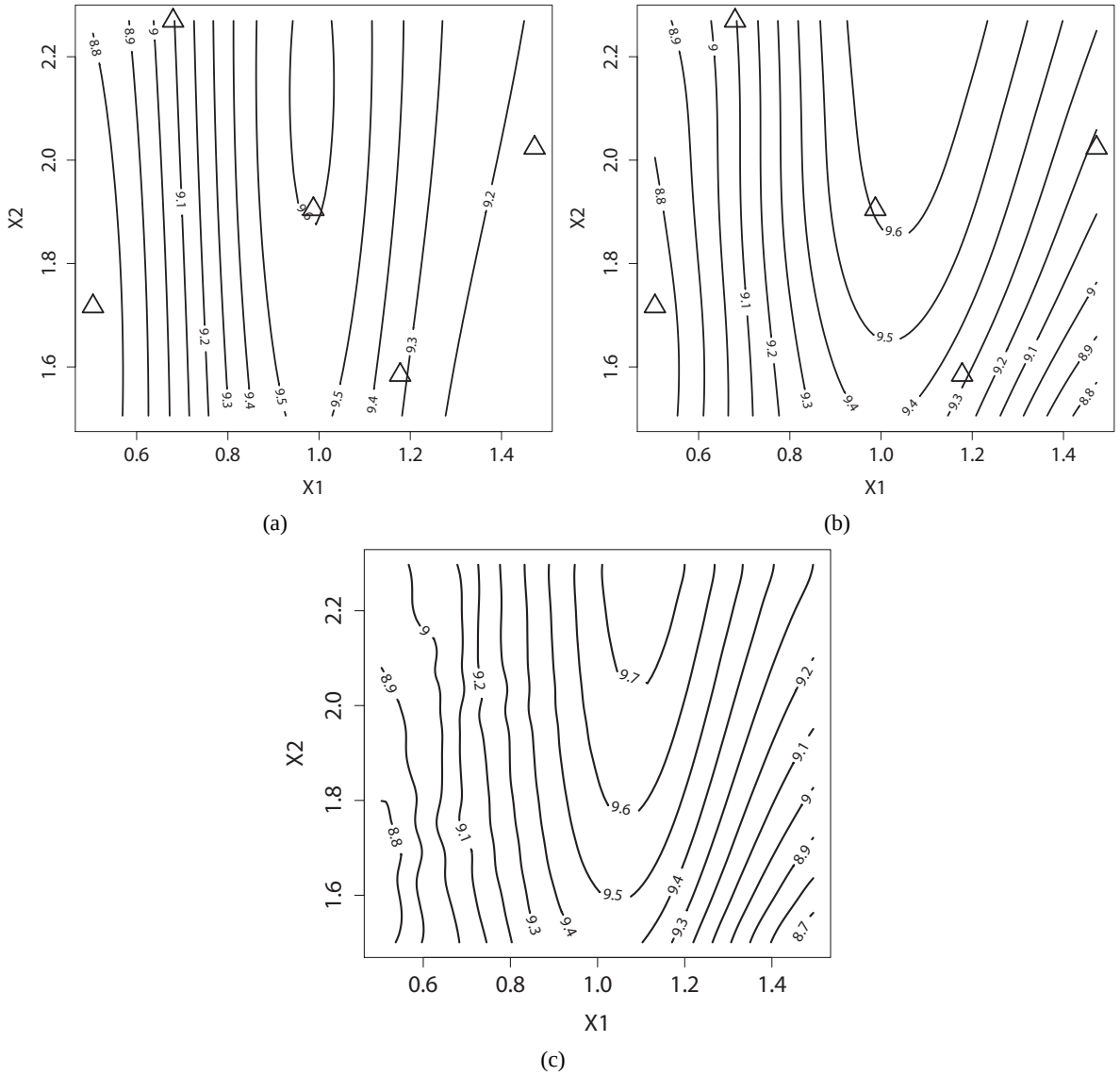

FIG. 3: Contour plot of the kriging mean (a), the co-kriging mean (b), and the true function (c). The triangles represent the $n_2 = 5$ points of the experimental design set of the expensive code.

TABLE 2: Application: hydrodynamic simulator. Comparison between kriging and co-kriging. The co-kriging model provides predictions significantly better than the ones of the kriging model

	Q_2	RMSE	MaxAE	RIMSE
kriging	75.83%	0.133	0.49	0.110
co-kriging	98.01%	0.038	0.14	0.046

We can see that the difference of accuracy between the two models is large. Indeed, that one of the co-kriging model is significantly better. Furthermore, RIMSE appears as an accurate estimation of RMSE (see Table 2). We note that the predictive variance for the co-kriging is obtained with a simple co-kriging model. Therefore, it will be slightly larger in the universal co-kriging case. Indeed, by computing the universal co-kriging equations, we find $\mathrm{RIMSE} = 0.058$.

We can compare the RMSE obtained with the test set with the RMSE obtained with a Leave-One-Out cross-validation procedure (see Section 5). For this procedure, we test our model on $n_2 = 5$ validation sets obtained by removing one observation at a time. As presented in Section 5, we can either choose to remove the observations from z_2 or from z_2 and z_1. The root mean squared error of the Leave-One-Out cross-validation procedure obtained by removing observations from z_2 is $\mathrm{RMSE}_{z_2,\mathrm{LOO}} = 4.80.10^{-3}$, whereas the one obtained by removing observations from z_2 and z_1 is $\mathrm{RMSE}_{z_1,z_2,\mathrm{LOO}} = 0.10$. Comparing $\mathrm{RMSE}_{z_2,\mathrm{LOO}}$ and $\mathrm{RMSE}_{z_1,z_2,\mathrm{LOO}}$ to the RMSE obtained with the external test set, we see that the procedure which consists in removing points from z_2 and z_1 provides a better proxy for the generalization error. Indeed, $\mathrm{RMSE}_{z_2,\mathrm{LOO}}$ is a relevant proxy for the generalization error only at points where z_1 is available. Therefore, it underestimates the error at locations where z_1 is unknown.

Figure 4 represents the mean and confidence intervals at plus or minus 1.96 times the standard deviation of the simple and universal co-krigings for points along the vertical line $x_1 = 0.99$ and the horizontal line $x_2 = 1.91$ [$x = (0.99, 1.91)$ corresponds to the coordinates of the point of D_2 in the center of the domain $[0.5, 1.5] \times [1.5, 2.3]$ in Fig. 1]. Note that the term "ordinary" co-kriging could also be appropriate in this example since $g_1(x) = 1$,

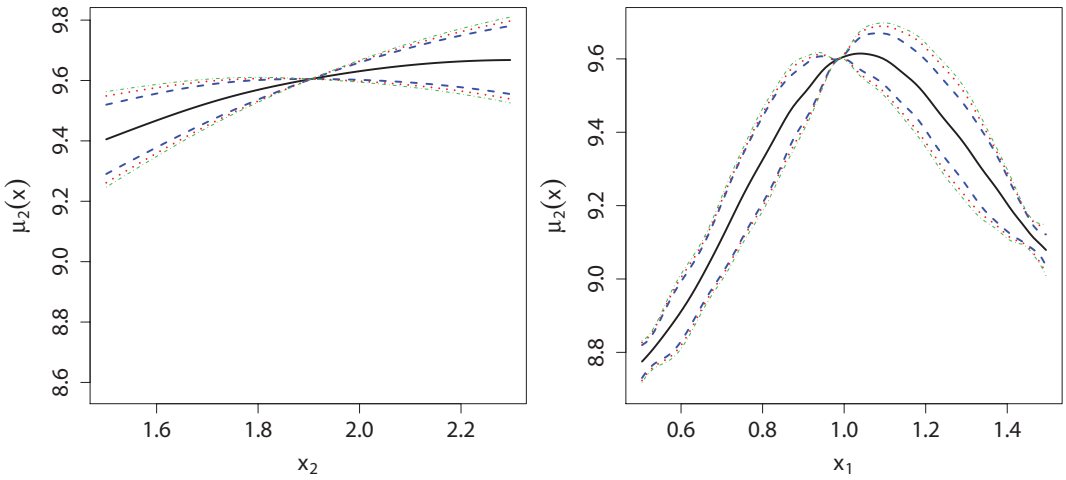

FIG. 4: Mean and confidence intervals for the simple and the universal co-kriging. The figure on the left hand side represents the predictions along the vertical line $x_1 = 0.99$ and the figure on the right hand side represents the predictions along the horizontal line $x_2 = 1.91$. The solid black lines represent the mean of the two co-kriging models, the dashed lines represent the confidence interval at plus or minus 1.96 times the standard deviation of the simple co-kriging, the dotted lines represent the same confidence intervals for the universal co-kriging, the thin dashed-dotted lines represent the empirical quantiles of order 2.5% and 97.5% estimated by a Monte-Carlo procedure with 10,000 samples.

TABLE 3: Application: hydrodynamic simulator. Estimations of β_1 and σ_1^2 for the response L_c [see Eqs. (14) and (16)]

Trend coefficient	$\Sigma_1 \nu_1$	Σ_1/σ_1^2
β_1	1.26	0.97

Variance coefficient	Q_1	$2\alpha_1$
σ_1^2	15.62	24

$f_2(x) = 1$, and $f_1(x) = 1$ do not depend on x and it is the usual term for kriging when the trend function is a constant. In Fig. 4 on the right-hand side, we see a narrowing of the confidence band for x_1 around 1.5, which corresponds to the x_1-coordinate of the upper right point of D_2 (Fig. 4), since, in the direction of x_2, the correlation hyper-parameters length for $Z_1(x)$ and $\delta_2(x)$ are large ($\theta_{1,2} = 1.20$ and $\theta_{2,2} = 1.37$). Moreover, as the predictive distribution for the universal co-kriging is not Gaussian, no exact quantiles can be associated to the confidence intervals at plus or minus 1.96 times the standard deviation. In Fig. 4, we compare them with the empirical quantiles of orders 2.5% and 97.5% estimated by a Monte Carlo procedure with 10,000 samples. These quantiles correspond to the confidence intervals at plus or minus 1.96 times the standard deviation for a normal distribution. We see in Fig. 4 that the two confidence intervals are very close (though it is a bit larger for the empirical ones). Therefore, the Gaussian assumption slightly underestimates the confidence intervals.

6.3 Multi-Fidelity Surrogate Model for the Mixture Characteristic Length L_c

In this section, we build a co-kriging model for the mixture characteristic length L_c. The aim of this example is to highlight that it can be worth having an adjustment coefficient ρ_1 depending on x. We use the same training and test sets as in the previous section and we consider a tensorized matern-$5/2$ kernel (25). Let us consider the two following cases:

- Case 1: $g_1(x) = 1$, $f_2(x) = 1$ and $f_1(x) = 1$,
- Case 2: $g_1^T(x) = (1\ x_1)$, $f_2(x) = 1$, and $f_1(x) = 1$.

We have the following hyperparameter maximum likelihood estimates for the two cases:

- Case 1: $\hat{\theta}_1 = (0.52, 1.09)$ and $\hat{\theta}_2 = (0.03, 0.02)$,
- Case 2: $\hat{\theta}_1 = (0.52, 1.09)$ and $\hat{\theta}_2 = (0.14, 1.37)$.

The estimation of $\hat{\theta}_1$ is identical in the two cases since it does not depend on ρ_1 and it is estimated with the same observations. Furthermore, we can see an important difference between the estimates of $\hat{\theta}_2$. Indeed, they are larger in Case 2 than in Case 1 which indicates that the model is smoother in Case 2. Table 3 presents the estimations of β_1 and σ_1^2 for the two cases (see Section 3.1).

Then, Table 4 presents the estimations of β_2, β_{ρ_1}, and σ_2^2 for Case 1, i.e., when ρ_1 is constant (see Section 3.1).

Finally, Table 5 presents the estimations of β_2, β_{ρ_1}, and σ_2^2 for Case 2, i.e., when ρ_1 depends on x (see Section 3.1).

We see in Table 4 that the adjustment coefficient is around 1.5 which indicates that the magnitude of the expensive code is slightly larger than that of the cheap code. Furthermore, we see in Table 5 that if we consider an adjustment coefficient which linearly depends on x_1 [i.e., with $g_1^T(x) = (1\ x_1)$], the constant part of ρ_1 is more important (it is around 1.66) and there is a negative slope in the direction x_1 (it is around -0.48). Since $x \in [0.5, 1.5]$, the averaged value of ρ_1 is 1.18 and goes from 1.42 at $x_1 = 0.5$ to 0.94 at $x_1 = 1.5$. We see also that the variance estimate in Case 1 (see Table 4) is much more important than the one in Case 2 (see Table 5). This is coherent with results of Table 5 (see below).

Figure 5 illustrates the contour plot of the two co-kriging models, i.e., when ρ_1 is constant and when ρ_1 depends on x.

TABLE 4: Application: hydrodynamic simulator. Estimations of β_2, β_{ρ_1} and σ_2^2 for Case 1, i.e., when ρ_1 is constant, for the response L_c [see Eqs. (14) and (16)]

Trend coefficient	$\Sigma_2 \nu_2$	Σ_2/σ_2^2
$\begin{pmatrix} \beta_{\rho_1} \\ \beta_2 \end{pmatrix}$	$\begin{pmatrix} 1.49 \\ -0.26 \end{pmatrix}$	$\begin{pmatrix} 0.83 & -0.79 \\ -0.79 & 0.95 \end{pmatrix}$
Variance coefficient	Q_2	$2\alpha_2$
σ_2^2	0.01	3

TABLE 5: Application: hydrodynamic simulator. Estimations of β_2, β_{ρ_1} and σ_2^2 for Case 2, i.e., when ρ_1 depends on x, for the response L_c [see Eqs. (14) and (16)]

Trend coefficient	$\Sigma_2 \nu_2$	Σ_2/σ_2^2
$\begin{pmatrix} \beta_{\rho_1} \\ \beta_2 \end{pmatrix}$	$\begin{pmatrix} 1.66 \\ -0.48 \\ -0.04 \end{pmatrix}$	$\begin{pmatrix} 2.34 & -3.50 & 0.44 \\ -3.50 & 9.18 & -3.67 \\ 0.44 & -3.67 & 2.60 \end{pmatrix}$
Variance coefficient	Q_2	$2\alpha_2$
σ_2^2	$3.24.10^{-4}$	2

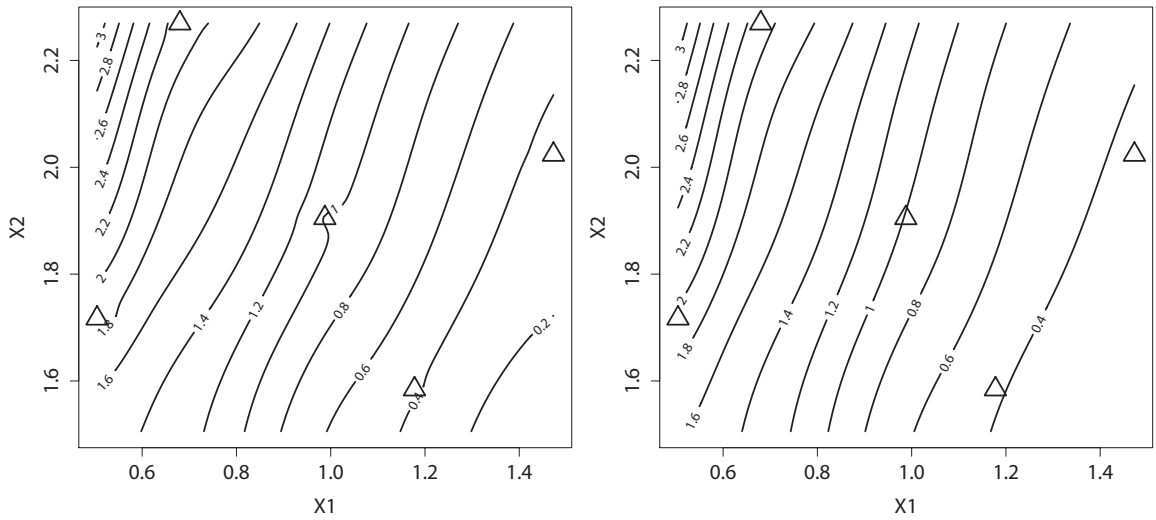

FIG. 5: Contour plot of the co-kriging mean when ρ_1 is constant (on the left hand side) and when ρ_1 depends on x (of the right hand side). The triangles represent the $n_2 = 5$ points of the experimental design set of the expensive code.

Furthermore, Table 6 compares the prediction accuracy of the co-kriging in the two cases. The precision is computed on the test set of 175 observations.

We see that the co-kriging model in Case 2 is clearly better than the one in Case 1. Therefore, we illustrate in this application that it can be worth considering an adjustment coefficient which is not constant contrarily to the model presented in [7] and [9].

7. CONCLUSION

We have presented in this paper a recursive formulation for a multi-fidelity co-kriging model. This model allows us to build surrogate models using data from simulations of different levels of fidelity.

TABLE 6: Application: hydrodynamic simulator. Comparison between co-kriging when ρ_1 is constant (Case 1) and co-kriging when ρ_1 depends on x (Case 2). The Case 2 provides predictions better than the Case 1, it is hence worthwhile to consider an adjustment coefficient that is not constant

	RMSE	MaxAE
Case 1	$7.26.10^{-3}$	0.23
Case 2	$1.53.10^{-3}$	0.16

The strength of the suggested approach is that it considerably reduces the complexity of the co-kriging model while preserving its predictive efficiency. Furthermore, one of the most important consequences of the recursive formulation is that the construction of the surrogate model is equivalent to build s consecutive krigings. Consequently, we can naturally adapt results of kriging to the co-kriging model.

First, we present a Bayesian estimation of the model parameters which provides closed-form expressions for the parameters of the posterior distributions. We note that, from these posterior distributions, we can deduce the maximum likelihood estimates of the parameters. Second, thanks to the joint distributions of the parameters and the recursive formulation, we can deduce closed-form formulas for the mean and covariance of the posterior predictive distribution. Due to their similarities with the universal kriging equations, we call these formulas the universal co-kriging equations. Third, we present closed-form expressions for the cross-validation equations of the co-kriging surrogate model. These expressions reduce considerably the complexity of the cross-validation procedure and are derived from the one of the kriging model that we have extended.

The suggested model has been successfully applied to a hydrodynamic code. We also present in this application a practical way to design the experiments of the multi-fidelity model.

From this work, three points can be investigated. The first one is the case when the experimental design sets are not nested. In such a situation, the predictive mean and variance of the recursive multi-fidelity co-kriging model can easily be derived. Furthermore, the parameters can be estimated recursively from the level 1 to the level s with maximum likelihood procedures. However, there are no more closed-form expressions for the estimates and they must be estimated jointly for each level. Moreover, the complexity of the optimization problem is controlled by the inversion of a matrix of size $n_t \times n_t$ where n_t is the number of observations at level t. As far as we know, there are no works dealing with the issue of the parameter estimation in this framework.

The second point is about the use of sequential design strategies to improve the model accuracy. Co-kriging models are well-suited to perform sequential designs since it provides an estimate of the model mean squared error through the predictive variance. However, in a multi-fidelity context, finding the locations to perform new simulations is not the only point of interest. Indeed, we have also to determine at which code levels these new simulations have to be run. Generalization of the classical kriging-based sequential design strategies can of course be envisioned. This would require to define a strategy which allocates the new simulations on the most appropriate code level as possible. It certainly should take into account the contribution of each code level on the model error and the time-rations between the code levels.

The third point is the issue of computer code validation. Indeed, it is worth noticing that the highest level of response could be field data and the lower levels could be outputs from physical models with different level of fidelity. The presented model can be used to predict a real phenomenon from both field data and computer codes. Furthermore, in such a case, a nugget effect can be required to model measurement errors for the field data and the presented multi-fidelity co-kriging model can naturally integrate it. This nugget effect can also be used to deal with ill-conditioned covariance matrices or to take into account the variability of the output of a code relying on a Monte Carlo numerical integration.

ACKNOWLEDGMENT

The authors thank Dr. Claire Cannamela for providing the data for the application and for interesting discussions.

REFERENCES

1. Krige, D. G., A statistical approach to some basic mine valuation problems on the witwatersrand, *Technometrics*, 52:119–139, 1951.
2. Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., Design and analysis of computer experiments, *Stat. Sci.*, 4:409–423, 1989.
3. Stein, M. L., *Interpolation of Spatial Data*, Springer Series in Statistics, New York, Springer, 1999.
4. Santner, T. J., Williams, B. J., and Notz, W. I., *The Design and Analysis of Computer Experiments*, New York, Springer, 2003.
5. Rasmussen, C. E. and Williams, C. K. I., *Gaussian Processes for Machine Learning*, Cambridge, MIT Press, 2006.
6. Craig, P. S., Goldstein, M., Seheult, A. H., and Smith, J. A., Constructing partial prior specifications for models of complex physical systems, *Appl. Stat.*, 47:37–53, 1998.
7. Kennedy, M. C. and O'Hagan, A., Predicting the output from a complex computer code when fast approximations are available, *Biometrika*, 87:1–13, 2000.
8. Higdon, D., Kennedy, M., Cavendish, J. C., Cafeo, J. A., and Ryne, R. D., Combining field data and computer simulation for calibration and prediction, *SIAM J. Sci. Comput.*, 26:448–466, 2004.
9. Forrester, A. I. J., Sobester, A., and Keane, A. J., Multi-fidelity optimization via surrogate modelling, *Proc. R. Soc. A*, 463:3251–3269, 2007.
10. Qian, P. Z. G. and Wu, C. F. J., Bayesian hierarchical modeling for integrating low-accuracy and high-accuracy experiments, *Technometrics*, 50:192–204, 2008.
11. Cumming, J. A. and Goldstein, M., Small sample bayesian designs for complex high-dimensional models based on information gained using fast approximations, *Technometrics*, 51:377–388, 2009.
12. Kennedy, M. C. and O'Hagan, A., Bayesian calibration of computer models, *J. R. Stat.l Soc., Ser. B*, 63(3):425–464, 2001.
13. Bayarri, M. J., Berger, J. O., Paulo, R., Sacks, J., Cafeo, J. A., Cavendish, J., Lin, C.-H., and Tu, J., A framework for validation of computer models, *Technometrics*, 49(2):138–154, 2007.
14. Goldstein, M. and Wooff, D. A., *Bayes Linear Statistics: Theory and Methods*, Chichester, UK, Wiley, 2007.
15. Dubrule, O., Cross validation of kriging in a unique neighborhood, *Math. Geol.*, 15:687–699, 1983.
16. Jeffreys, H., *Theory of Probability*, Oxford University Press, London, 1961.
17. Harville, D. A., *Matrix Algebra from a Statistician's Perspective*, New York, Springer-Verlag, 1997.
18. Patterson, H. and Thompson, R., Recovery of interblock information when block sizes are unequal, *Biometrika*, 58:545–554, 1971.
19. Zhang, H. and Wang, Y., Kriging and cross-validation for massive spatial data, *Environmetrics*, 21:290–304, 2009.
20. Bastos, L. S. and O'Hagan, A., Diagnostics for Gaussian process emulators, *Technometrics*, 51(4):425–438, 2009.
21. Grégoire, O., Souffland, D., and Serge, G., A second order turbulence model for gaseous mixtures induced by Richtmyer-Meshkov instability, *J. Turbul.*, 6:1–20, 2005.
22. Fang, K.-T., Li, R., and Sudjianto, A., *Design and Modeling for Computer Experiments*, Computer Science and Data Analysis Series, London, Chapman & Hall, 2006.
23. Qian, P. Z. G., Ai, M., and Wu, C. F. J., Construction of nested space-filling designs, *Ann. Stat.*, 37:3616–3643, 2009.
24. Stein, M. L., Large sample properties of simulations using latin hypercube sampling, *Technometrics*, 29:143–151, 1987.
25. Stocki, R., A method to improve design reliability using optimal latin hypercube sampling, *Comput. Assist. Mech. Eng. Sci.*, 12:87–105, 2005.
26. Shewry, M. C. and Wynn, H. P., Maximum entropy sampling, *J. Appl. Stat.*, 14:165–170, 1987.
27. Currin, C., Mitchell, T., Morris, M., and Ylvisaker, D., Bayesian prediction of deterministic functions with applications to the design and analysis of computer experiments, *J. Ame. Stat. Assoc.*, 86:953–963, 1991.

APPENDIX A. PROOFS

APPENDIX A.1 Proof of Proposition 1

Let us consider the co-kriging mean of the model (1) presented in [7] for a t-level co-kriging with $t = 2, \ldots, s$:

$$m_{Z_t}(x) = h^{(t)}(x)^T \beta^{(t)} + t_t(x)^T (V^{(t)})^{-1} (z^{(t)} - H^{(t)} \beta^{(t)}),$$

where $\beta^{(t)} = (\beta_1^T, \ldots, \beta_t^T)^T$, $z^{(t)} = (z_1^T, \ldots, z_t^T)^T$, and $h^{(t)}(x)^T$ is defined in Eq. (6). We have

$$\begin{aligned} h^{(t)}(x)^T \beta^{(t)} &= \rho_{t-1}(x) \left(\left(\prod_{i=1}^{t-2} \rho_i(x) \right) f_1^T(x), \left(\prod_{i=2}^{t-2} \rho_i(x) \right) f_2^T(x), \ldots, f_{t-1}^T(x) \right) \beta^{(t-1)} + f_t^T(x) \beta_t, \\ &= \rho_{t-1}(x) h^{(t-1)}(x)^T \beta^{(t-1)} + f_t^T(x) \beta_t. \end{aligned}$$

Then, from Eqs. (7) and (8), we have the following equality:

$$V^{(t)} = \begin{pmatrix} V^{(t-1)} + \begin{pmatrix} 0 & 0 \\ 0 & \dfrac{\left(\rho_{t-1}(D_t) \rho_{t-1}(D_t)^T\right) \odot R_t^{-1}}{\sigma_t^2} \end{pmatrix} & -W \\ -W^T & \dfrac{R_t^{-1}}{\sigma_t^2} \end{pmatrix},$$

where $\odot$ stands for the element by element matrix product and

$$W = \begin{pmatrix} 0 \\ \dfrac{\left(\rho_{t-1}(D_t) \mathbf{1}_{n_t}^T\right) \odot R_t^{-1}}{\sigma_t^2} \end{pmatrix}.$$

Therefore, we can deduce that

$$\begin{aligned} t_t(x)^T (V^{(t)})^{-1} z^{(t)} &= \rho_{t-1}(x) t_{t-1}(x)^T (V^{(t-1)})^{-1} z^{(t-1)} - \left(\rho_{t-1}^T(D_t)\right) \odot \left(r_t^T(x) R_t^{-1} z_{t-1}(D_t)\right) \\ &+ r_t^T(x) R_t^{-1} z_t, \end{aligned}$$

and with Eq. (6):

$$t_t(x)^T (V^{(t)})^{-1} H^{(t)} \beta^{(t)} = \rho_{t-1}(x) t_{t-1}(x)^T (V^{(t-1)})^{-1} H^{(t-1)} \beta^{(t-1)} + r_t^T(x) R_t^{-1} F_t \beta_t.$$

We hence obtain the recursive relation:

$$m_{Z_t}(x) = \rho_{t-1}(x) m_{Z_{t-1}}(x) + f_t^T(x) \beta_t + r_t^T(x) R_t^{-1} \left[z_t - \rho_{t-1}(D_t) \odot z_{t-1}(D_t) - F_t \beta_t \right].$$

The co-kriging predictive mean of the model (9) satisfies the same recursive relation, and we have $m_{Z_1}(x) = \mu_{Z_1}(x)$. This proves the first equality of Proposition 1:

$$\mu_{Z_s}(x) = m_{Z_s}(x).$$

We follow the same guideline for the co-kriging covariance:

$$s_{Z_t}^2(x, x') = v_{Z_t}^2(x, x') - t_t^T(x) (V^{(t)})^{-1} t_t(x'),$$

where $v_{Z_t}^2(x, x')$ is the covariance between $Z_t(x)$ and $Z_t(x')$ and $s_{Z_t}^2(x, x')$ is the covariance function of the conditioned Gaussian process $[Z_t(x) | \mathcal{Z}^{(t)} = z^{(t)}, \beta, \beta_\rho, \sigma^2]$ for the model (1). From Eq. (8), we can deduce the following equality:

$$v_{Z_t}^2(x, x') = \rho_{t-1}(x) \rho_{t-1}(x') v_{Z_{t-1}}^2(x, x') + v_t^2(x, x'),$$

where $v^2_{Z_t}(x,x')$ is the covariance function of the conditioned Gaussian process $[Z_t(x)|\mathcal{Z}^{(t)}=z^{(t)},\beta_t,\beta_{\rho_{t-1}},\sigma_t^2]$ of the recursive model (9). Then, from Eqs. (7) and (8), we have

$$t_t^T(x)(V^{(t)})^{-1}t_t(x')=\rho_{t-1}(x)\rho_{t-1}(x')t_{t-1}^T(x)(V^{(t-1)})^{-1}t_{t-1}(x')+\sigma_t^2 r_t^T(x)R_t^{-1}r_t(x').$$

Finally we can deduce the following equality:

$$s^2_{Z_t}(x,x')=\rho_{t-1}(x)\rho_{t-1}(x')\left(v^2_{Z_{t-1}}(x,x')-t_{t-1}^T(x)(V^{(t-1)})^{-1}t_{t-1}(x')\right)+\sigma_t^2\left(1-r_t^T(x)R_t^{-1}r_t(x')\right),$$

which is equivalent to

$$s^2_{Z_t}(x,x')=\rho_{t-1}(x)\rho_{t-1}(x')s^2_{Z_{t-1}}(x,x')+\sigma_t^2\left(1-r_t^T(x)R_t^{-1}r_t(x')\right).$$

This is the same recursive relation as the one satisfied by the co-kriging covariance $\sigma^2_{Z_t}(x,x')$ of the model (9) [see Eq. (12)]. Since $s^2_{Z_1}(x,x')=\sigma^2_{Z_1}(x,x')$, we have

$$\sigma^2_{Z_s}(x,x')=s^2_{Z_s}(x,x').$$

This equality with $x=x'$ proves the second equality of Proposition 1. □

APPENDIX A.2 Proof of Proposition 2

Noting that the mean of the predictive distribution in equation (11) does not depend on σ_t^2 and thanks to the law of total expectation, we have the following equality:

$$\mathbb{E}\left[Z_t(x)|\mathcal{Z}^{(t)}=z^{(t)}\right]=\mathbb{E}\left[\mathbb{E}\left[Z_t(x)|\mathcal{Z}^{(t)}=z^{(t)},\sigma_t^2,\beta_t,\beta_{\rho_{t-1}}\right]\left|\mathcal{Z}^{(t)}=z^{(t)}\right.\right].$$

From Eqs. (11) and (14), we directly deduce Eq. (18). Then, we have the following equality:

$$\mathrm{var}\left(\mu_{Z_t}(x)\left|z^{(t)},\sigma_t^2\right.\right)=(h_t^T(x)-r_t(x)^TR_t^{-1}H_t)\Sigma_t(h_t^T(x)-r_t(x)^TR_t^{-1}H_t)^T. \tag{A.1}$$

Furthermore, from (12) and (14), we can deduce

$$\begin{aligned}\mathbb{E}\left[\mathrm{var}(Z_t(x)|z^{(t)},\beta_t,\beta_{\rho_{t-1}},\sigma_t^2)\left|z^{(t)},\sigma_t^2\right.\right]&=\hat{\sigma}^2_{\rho_{t-1}}(x)\mathrm{var}(Z_{t-1}(x)|\mathcal{Z}^{(t-1)}\\&=z^{(t-1)},\sigma_t^2)+\sigma_t^2\left(1-r_t^T(x)R_t^{-1}r_t^T(x)\right),\end{aligned} \tag{A.2}$$

where $\hat{\sigma}^2_{\rho_{t-1}}(x)=g_{t-1}^T(x)\left(\Sigma_{t,\rho}+[\sigma_t\nu_t]_{1,\ldots,q_{t-1}}[\sigma_t\nu_t]^T_{1,\ldots,q_{t-1}}\right)g_{t-1}(x)$. The law of total variance states that

$$\begin{aligned}\mathrm{var}(Z_t(x)|z^{(t)},\sigma_t^2)=&\,\mathbb{E}\left[\mathrm{var}(Z_t(x)|z^{(t)},\beta_t,\beta_{\rho_{t-1}},\sigma_t^2)\left|z^{(t)},\sigma_t^2\right.\right]\\&+\mathrm{var}\left(\mathbb{E}\left[Z_t(x)|z^{(t)},\beta_t,\beta_{\rho_{t-1}},\sigma_t^2\right]\left|z^{(t)},\sigma_t^2\right.\right).\end{aligned}$$

Thus, from Eqs. (A.1) and (A.2), we obtain

$$\begin{aligned}\mathrm{var}(Z_t(x)|\mathcal{Z}^{(t)}=z^{(t)},\sigma_t^2)=&\,\hat{\sigma}^2_{\rho_{t-1}}(x)\mathrm{var}(Z_{t-1}(x)|\mathcal{Z}^{(t-1)}=z^{(t-1)},\sigma_t^2)+\sigma_t^2\left(1-r_t^T(x)R_t^{-1}r_t^T(x)\right)\\&+\left(h_t^T-r_t^T(x)R_t^{-1}H_t\right)\Sigma_t\left(h_t^T-r_t^T(x)R_t^{-1}H_t\right)^T.\end{aligned} \tag{A.3}$$

Again using the law of total variance and the independence between $\mathbb{E}\left[Z_t(x)|\mathcal{Z}^{(t)}=z^{(t)},\beta_t,\beta_{\rho_{t-1}}\right]$ and σ_t^2, we have

$$\mathrm{var}(Z_t(x)|z^{(t)})=\mathbb{E}\left[\mathrm{var}(Z_t(x))|z^{(t)},\sigma_t^2\right]. \tag{A.4}$$

We obtain Eq. (19) from Eq. (16) by noting that the mean of an inverse Gamma distribution $\mathcal{IG}(a,b)$ is $b/(a-1)$. □

APPENDIX A.3 Proof of Proposition 3

For notational convenience, let us consider that ξ_s is the index of the n_{test} last points of D_s. We denote by D_{test} these points. First we consider the variance and the trend parameters as fixed, i.e., $\sigma^2_{t,-\xi_t} = Q_t/[2(a_t-1)]$ and $\lambda_{t,-\xi_t} = \Sigma_t \nu_t$, and $\mathcal{V}_s = 0$; i.e., we are in the simple co-kriging case. Thanks to the blockwise inversion formula, we have the following equality:

$$R_s^{-1} = \begin{pmatrix} A & B \\ B^T & \mathcal{Q}^{-1} \end{pmatrix}, \tag{A.5}$$

with $A = \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1} + \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1} [R_s]_{[-\xi_s,\xi_s]} \mathcal{Q}^{-1} [R_s]_{[\xi_s,-\xi_s]} \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1}$,
$B^T = -\mathcal{Q}^{-1} [R_s]_{[\xi_s,-\xi_s]} \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1}$, and

$$\mathcal{Q} = [R_s]_{[\xi_s,\xi_s]} - [R_s]_{[\xi_s,-\xi_s]} \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1} [R_s]_{[-\xi_s,\xi_s]}. \tag{A.6}$$

We note that $[Q_s/2(a_s-1)]\mathcal{Q} = [Q_s/2(a_s-1)] \left(\left[R_s^{-1}\right]_{[\xi_s,\xi_s]}\right)^{-1}$ represents the covariance matrix of the points in D_{test} with respect to the covariance kernel of a Gaussian process of kernel $[Q_s/2(a_s-1)]r_s(x,x')$ [which is the one of $\delta_s(x)$] conditioned by the points $D_s \setminus D_{\text{test}}$. Therefore, from the previous remark and Eq. (12), we can deduce Eq. (22).

Furthermore, from (A.5) we have the following equality:

$$\begin{aligned}\left[R_s^{-1}\left(z_s - H_s\lambda_{s,-\xi_s}\right)\right]_{[\xi_s]} &= B^T z_s(D_s \setminus D_{\text{test}}) + \mathcal{Q}^{-1} z_s(D_{\text{test}}) \\ &\quad + B^T [H_s^T]_{[-\xi_s]}\Sigma_s\nu_s + \mathcal{Q}^{-1} h_s^T(D_{\text{test}})\Sigma_s\nu_s.\end{aligned}$$

From which we can deduce the following one:

$$\begin{aligned}\left(\left[R_s^{-1}\right]_{[\xi_s,\xi_s]}\right)^{-1} \left[R_s^{-1}\left(z_s - H_s\lambda_{s,-\xi_s}\right)\right]_{[\xi_s]} &= z_s(D_{\text{test}}) - h_s^T(D_{\text{test}})\Sigma_s\nu_s \\ &\quad - [R_s]_{[\xi_s,-\xi_s]} \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1} \\ &\quad \times \left(z_s(D_s \setminus D_{\text{test}}) - [H_s^T]_{[-\xi_s]}\Sigma_s\nu_s\right).\end{aligned} \tag{A.7}$$

From this equation and Eq. (11), we can directly deduce Eq. (20) with $\varepsilon_{Z_s,\xi_s} = z_s(D_{\text{test}}) - \mu_{Z_s}(D_{\text{test}})$.

Then, we suppose the trend and the variance parameters as unknown and we have to re-estimate them when we remove the observations. Thanks to the parameter estimations presented in Section 3.1, we can deduce that the estimates of $\sigma^2_{t,-\xi_t}$ and $\lambda_{t,-\xi_t}$ when we remove observations of index ξ_t are given by the following equations:

$$\lambda_{s,-\xi_s}\left([H_s^T]_{-\xi_s}K_s[H_s]_{-\xi_s}\right) = [H_s^T]_{-\xi_s}K_s z_s(D_s \setminus D_{\text{test}}), \tag{A.8}$$

and

$$\sigma^2_{s,-\xi_s} = \frac{\left(z_s(D_s \setminus D_{\text{test}}) - [H_s]_{-\xi_s}\lambda_{s,-\xi_s}\right)^T K_s \left(z_s(D_s \setminus D_{\text{test}}) - [H_s]_{-\xi_s}\lambda_{s,-\xi_s}\right)}{n_s - p_s - q_{s-1} - n_{\text{test}}}, \tag{A.9}$$

with $K_s = \left([R_s]_{[-\xi_s,-\xi_s]}\right)^{-1}$.

From the equality (A.5), we can deduce that $K_s = A - B\mathcal{Q}B^T$ from which we obtain Eq. (21). Finally, to obtain the cross-validation equations for the universal co-kriging, we just have to estimate the following quantity [see Eq. (19)]

$$\left(h_s^T(D_{\text{test}})^T - [R_s]_{[\xi_s,-\xi_s]} K_s[H_s]_{-\xi_s}\right)\Sigma_s \left(h_s^T(D_{\text{test}})^T - [R_s]_{[\xi_s,-\xi_s]} K_s[H_s]_{-\xi_s}\right)^T, \tag{A.10}$$

with $\Sigma_s = \left([H_s^T]_{-\xi_s} K_s [H_s]_{-\xi_s}\right)^{-1}$. From Eq. (A.5), we can deduce the following equality:

$$\left[R_s^{-1} H_s\right]_{[\xi_s]} = B^T [H_s]_{-\xi_s} + \mathcal{Q}^{-1} h_s^T (D_{\text{test}})^T,$$

from which we can deduce the following equality:

$$\left(h_s^T (D_{\text{test}})^T - [R_s]_{[\xi_s,-\xi_s]} K_s [H_s]_{-\xi_s}\right) = \left(\left([R_s^{-1}]_{[\xi_s,\xi_s]}\right)^{-1} \left[R_s^{-1} H_s\right]_{[\xi_s]}\right), \tag{A.11}$$

which allows us to obtain Eq. (24) and completes the proof. □