

Variable-fidelity modeling of structural analysis of assemblies

Nicolas Courrier¹ · Pierre-Alain Boucard¹ ·
Bruno Soulier¹

Received: 11 July 2014 / Accepted: 26 July 2015 / Published online: 13 August 2015
© Springer Science+Business Media New York 2015

Abstract This paper deals with the advantages of using variable-fidelity metamodeling strategies in order to develop a valid metamodel more rapidly than by using traditional methods. In our mechanical assembly design, we use the term “variable-fidelity” in reference to the convergence (or accuracy) level of the iterative solver being used. Variable-fidelity metamodeling is a way to improve the prediction of the output of a complex system by incorporating rapidly available auxiliary lower-fidelity data. This work uses two fidelity levels, but more levels can be added. The LATIN iterative algorithm is used along with a “multiparametric” strategy to calculate the various data and their different fidelity levels by means of an error indicator. Three main categories of variable-fidelity strategies are currently available. We tested at least one method from each of these categories, which comes to a total of five methods for calculating a valid metamodel using low- and high-fidelity data. Here, our objective is to compare the performances of these five methods in solving three mechanical examples.

Keywords Metamodel · Variable-fidelity · Partially converged data · Kriging · Cokriging

List of symbols

\mathcal{D}	The design space
n	The dimension of \mathcal{D}
\mathbf{x}^i	A point $\in \mathcal{D}$
$y(\mathbf{x})$	The objective function at point \mathbf{x}
$\hat{y}(\mathbf{x})$	An approximation of the objective function at point \mathbf{x} obtained through a metamodel

✉ Nicolas Courrier
courrier@lmt.ens-cachan.fr

¹ LMT Cachan (ENS Cachan/CNRS/Université Paris-Saclay), 61 Avenue du Président Wilson, 94230 Cachan, France

\mathbf{X}	The matrix of all the calculated points $\mathbf{X} = [\mathbf{x}^1 \cdots \mathbf{x}^p]^T$
y^i	The objective function at point \mathbf{x}^i
\mathbf{Y}	The vector of all the objective function $\mathbf{Y} = [y^1 \cdots y^p]^T$
• f_{cv}	Any of the above in the case of fully converged data
• p_{cv}	Any of the above in the case of partially converged data
$\mathbf{f}(\mathbf{x})$	The regression function. (If one chooses a mean regression, $\mathbf{f}(\mathbf{x}) = 1$; for a linear regression, $\mathbf{f}(\mathbf{x}) = [1 \ x_1 \cdots x_n]^T$)
$\boldsymbol{\beta}$	The parameters of the regression function. (For a linear regression, $\boldsymbol{\beta} = [\beta_1 \cdots \beta_{n+1}]^T$)
$\mathbf{F} = [\mathbf{f}(\mathbf{x}^1) \cdots \mathbf{f}(\mathbf{x}^p)]^T$	The matrix of the regression functions at the calculation points (for universal kriging case)
$z(\mathbf{x})$	A stochastic function
\mathbf{Z}	The stochastic vector of the calculated approximations $\mathbf{Z} = [z(\mathbf{x}^1) \cdots z(\mathbf{x}^p)]^T$
\mathbf{C}	The covariance matrix
\mathbf{R}	The correlation matrix

1 Introduction

Over the past few years, the development of new mechanical models and numerical tools has led to improvements in the design of complex systems. In many disciplines, optimizations must be carried out using these increasingly complex models which can represent the actual physics of the problem very closely. In general, however, the high cost of these models makes direct optimization impossible. In the last two decades, metamodeling (also known as metamodeling or response surface modeling) has gained popularity. A metamodel is an approximation of the real model being studied. The optimization can be carried out on this metamodel, which becomes a suitable tool for an optimization strategy. This approximation is defined through an interpolation or a regression of some specific output data of a computer code [1]. The work which is presented here focuses on parameterized assembly design. The main numerical difficulties and high computation costs are related to contact between parts with or without friction and gaps. In this context, the use of metamodels is mandatory.

More generally, as mentioned in [2,3], metamodels can be classified into three categories:

- Response surfaces: a response surface is a functional mapping of several input parameters to a single output feature. The mapping can be of polynomial form, whose regression coefficients are determined using the least-squares method, or can resort to more complex techniques, such as kriging or RBFs modeling. Typically, the properties of interest of a response surface are those which characterize the model's representativeness, the contribution of an individual variable to the variance of the whole model, the model's resolution, etc. A taxonomy of the various processes used to create a surface is given in [1].
- Reduced models derived from Proper Orthogonal Decomposition (POD) or Proper Generalized Decomposition (PGD). Details can be found in [4,5]. Recently, PGD was applied to parametric problems such as parametric optimization, inverse identification and real

time simulation. Some examples of parametric modeling can be found in [6, 7]. PGD has also been used in the multiparametric strategy context [8].

- Hierarchical models, also known as multifidelity, variable-fidelity or variable-complexity models. In the particular case of two fidelity levels, corrections can be carried out from the low-fidelity model to the high-fidelity model, such as in [9], which uses a kriging model to correct the low-fidelity model based on the available high-fidelity model. This type of correction corresponds to what is called a scaling model or a correction response surface [10, 11]. The low-fidelity model can be the same as the high-fidelity model, but – as in the case of this work – converged to a different accuracy level [3, 12], or calculated using a coarser grid [13]. Alternatively, as mentioned in [2], the low-fidelity model can be a simple engineering model in which some of the physics taken into account by the high-fidelity method is disregarded.

Computational metamodeling is widely used in fluid dynamics simulation and optimization. The techniques presented in this paper are derived from that domain of mechanics, but can be used in numerous other disciplines. The purpose of this metamodel is to help locate the zone of interest, which is updated using specific methods to refine the search of the minimum [1, 14]. Then, the mathematical model associated with the metamodel plays an important role. Several construction approaches can be used, such as kriging [15, 16], gradient-enhanced kriging [17–19] or RBFs [20, 21]. Kriging is a popular metamodel construction method which was introduced by Krige [15] and further developed by, among others, Matheron [16], who laid out its mathematical foundations.

The computation cost of the data which are necessary to build the metamodel remains a major stumbling block. The use of a metamodel based only on high-fidelity data can be too expensive. Metamodels can be improved by introducing additional information, e.g. concerning the gradients as in the GEK method [17].

In order to reduce computation times, one can use auxiliary variables (cokriging), leading to new computing frameworks [22–24]. It is because of these developments that the focal point of this work is the variable-fidelity model. The main goal of this paper is to develop a metamodel which can be used to perform an optimization and find the global optimum. The savings in computation time obtained thanks to these methods are discussed in the paper.

In the context of Variable-Fidelity Modeling (VFM), as mentioned in [25], three main approaches can be followed:

- the use of an additive, multiplicative or hybrid bridge function (also called a correction response surface) which corrects the discrepancy between the low-fidelity model and the high-fidelity model [3, 10, 12, 26]
- cokriging, which establishes a relation between the primary and auxiliary variables (the high-fidelity and low-fidelity data) [22, 27, 28]
- hierarchical kriging, which is an extension of kriging with the same computational complexity [25, 29, 30].

The originality of this work is that, as in [3], the high-fidelity and low-fidelity levels are calculated using an iterative solver (the LATIN solver [31]) in a solid mechanical context. The main goal of this work is to compare the performances of several of VFM approaches in the context of non-linear structural computations. Let us recall that the low-fidelity and high-fidelity models are based on the same mechanical model, the only difference being that in the low-fidelity model the iterations of the solver are stopped prematurely, leading to only partially converged data. The high-fidelity model is defined after the iterative algorithm has converged completely, hence the reference to fully converged data. Such an approach would

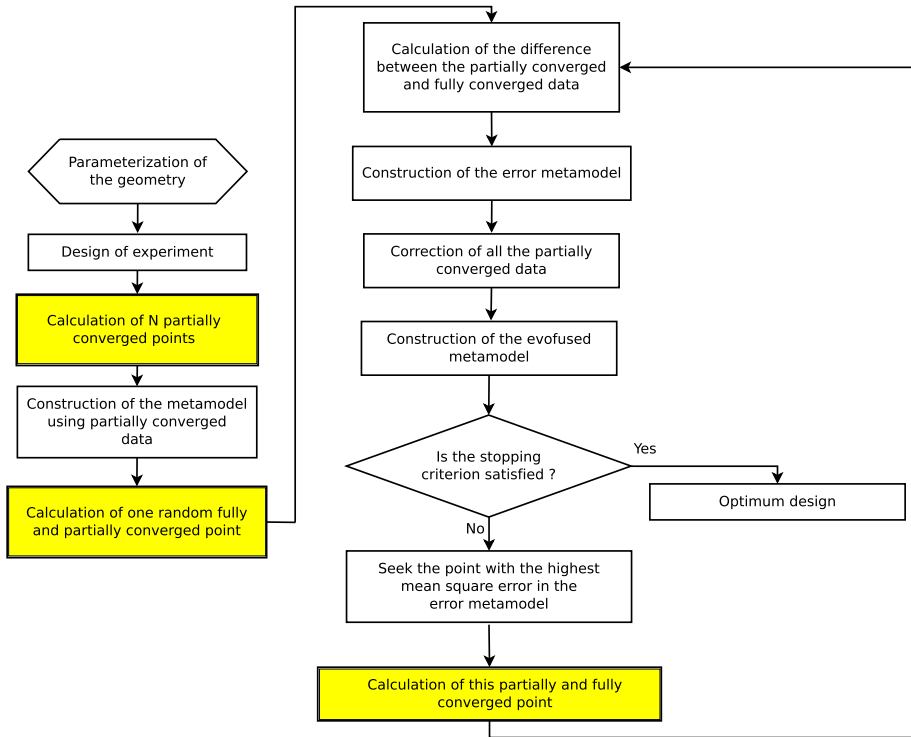


Fig. 1 The evofusion algorithm used in this work

be impossible if one used the classical incremental algorithm. In this paper, we test a variety of techniques from the three previous categories of strategies:

- an additive bridge function along with the evofusion algorithm [12] to construct the final metamodel
- three cokriging methods with different ways of calculating the cross correlation [22, 23, 27]
- the hierarchical kriging technique [30].

According to [26], an additive bridge function works better than a multiplicative function; the use of a hybrid function could be interesting, but the coding would be more difficult and would lead to an iterative process. Therefore, we used an additive bridge function along with the evofusion algorithm to enrich our final metamodel. So, a first metamodel based on partially converged data is computed. Then, the difference between partially and fully converged data is calculated in some specific points. The additive bridge function is estimated thanks to a metamodel called “error metamodel” and to get our final metamodel or “evofused” metamodel we add the first metamodel computed based on partially converged data and the error metamodel. The principle of evofusion, as presented in [3], is described in Fig. 1.

Cokriging methods were first developed in geostatistics [32] in order to enable the incorporation of auxiliary variables. In computer experiments, the information from the low-fidelity model is added to the high-fidelity data to build a final metamodel. In our case, the partially converged points are used to construct an interpolation metamodel among the fully converged points. One of the main difficulties is in finding the relation between the partially and

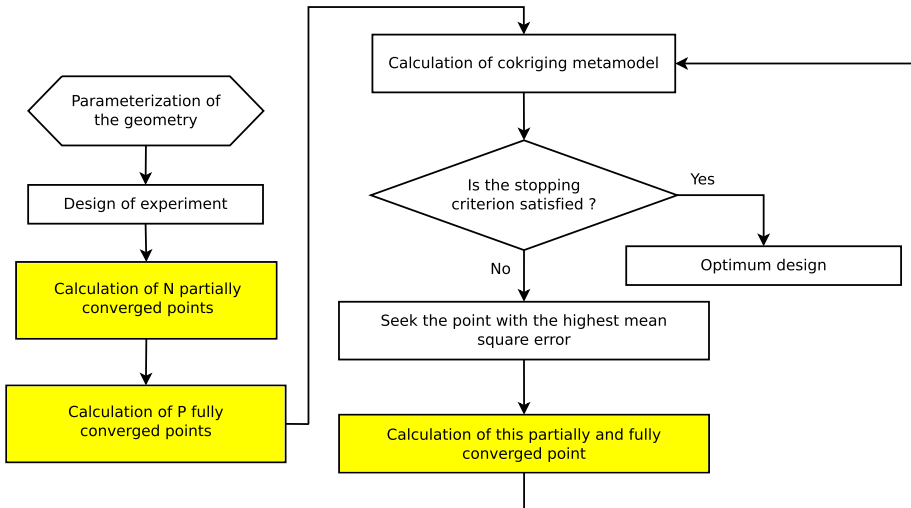


Fig. 2 The algorithm of the construction of a metamodel based on the cokriging method

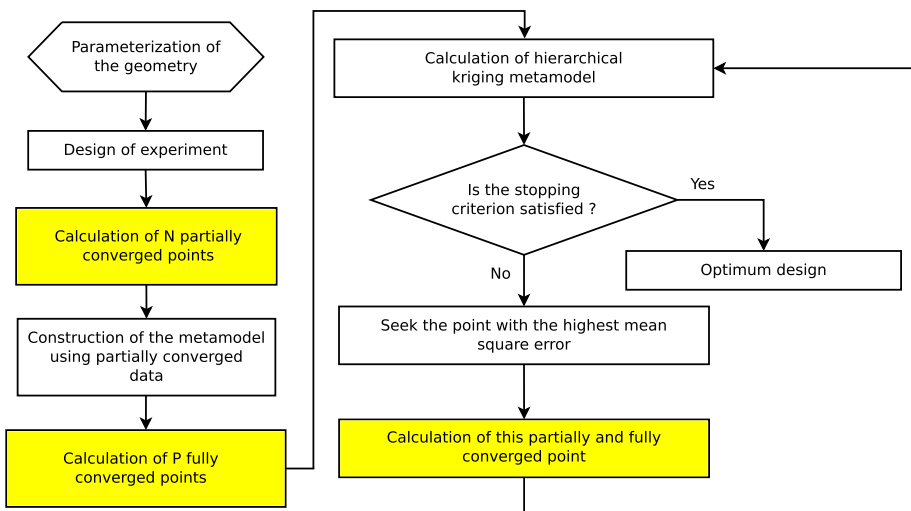


Fig. 3 The algorithm of the construction of a metamodel based on the hierarchical kriging method

fully converged data (*i.e.* the low-fidelity and high-fidelity models). We will compare three different methods. We used the algorithm in Fig. 2, which is similar to the one described previously.

Hierarchical kriging is an easier way to use partially converged data to build a metamodel because it is very similar to kriging. Indeed, the low-fidelity metamodel based on partially converged data is used as a regression function of the high-fidelity metamodel. The algorithm used in this paper is described in Fig. 3.

In order to improve the computation time, we use a specific feature of the LATIN iterative solver, called the “multiparametric strategy”, which enables one to reuse previously calculated data when performing new calculations [33].

The combination of VFM and multiparametric computing can be very efficient in terms of computation time. The main purposes of this paper is to use partially converged data as an auxiliary data layer and more especially to compare the performances and robustness of several VFM methods on different mechanical examples. A Matlab toolbox was developed to compute all of these methods with several correlation functions. This toolbox have also the possibility to take into account the gradient informations [17]. Section 2 is a review of the multiparametric LATIN method. In Sect. 3, kriging and cokriging are presented in matrix form. In Sect. 4, several tools used in this work are introduced. Section 5 introduces the examples used for comparing the various methods and presents the results of the comparison.

2 The LATIN method [31]

Frictional contact laws involve a nonlinear and nonsmooth behavior at the boundary of the body. To solve this mechanical problem, the nonincremental LARge Time INcrement (LATIN) method is used. This method is well-known for its ability to solve difficult nonlinear large problems (nonlinear material, contact problems...) [31] with a global time–space approach. This method is close to augmented Lagrangian methods. Its great advantages are non refactorization of matrices (stiffness matrix remains constant through LATIN iterations). Moreover, as it proceeds in a global space–time approach, an approximation of the solution controlled by a stopping criterion is obtained at each iteration. Depending of the value of this stopping criterion, the approximation of the solution can be considered as a low fidelity model as it is based on a partially converged run, or a high fidelity model evaluated using a fully converged run. Such variable-fidelity models can be readily constructed using the LATIN method as iterative solver. This remarkable property makes it a suitable computation method for our purpose of building a metamodel based on partially and fully converged data.

2.1 Principle

The first concept underlying the LATIN method is domain decomposition, in which the interfaces are mechanical entities in their own right, with their unknowns and constitutive relations. An evolution law which depends on the problem being studied (friction, contact, etc.) is associated with each interface. The second main point is that the difficulties are separated by considering two sets of equations: the local (possibly nonlinear) equations, and the linear (possibly global) equations.

In order to simplify the presentation, let us consider only two substructures Ω_E and $\Omega_{E'}$ connected by an interface $\Gamma^{EE'}$. The interface variables are two force fields $(\mathbf{f}^E, \mathbf{f}^{E'})$ and two dual velocity fields $(\dot{\mathbf{w}}^E, \dot{\mathbf{w}}^{E'})$, as shown in Fig. 4. By convention, $(\mathbf{f}^E, \mathbf{f}^{E'})$ are the actions of the interface onto the substructures and $(\dot{\mathbf{w}}^E, \dot{\mathbf{w}}^{E'})$ are the velocities of the substructures observed from the interface. In the case of an assembly, each of the parts is considered to be a substructure. Therefore, the interfaces describe the behavior of the assembly (including friction, gaps, etc.)

Let $\mathbf{u}^E(M, t)$ be the displacement field at point M of Ω_E and at time t in $[0, T]$, and let $\mathcal{U}_{ad}^{[0, T]}$ be the associated space. Then, the problem to be solved in each substructure is: find the evolutions of the displacement field $\mathbf{u}^E(M, t)$ and of the stress field $\sigma^E(M, t)$. One introduces the kinematic variable $\mathbf{w}^E(M, t)$ defined by $\mathbf{u}^E(M, t)|_{\partial\Omega_E} = \mathbf{w}^E(M, t)$. In our case, since the problem is quasi-static, we use the quantity $\dot{\mathbf{w}}^E(M, t)$.

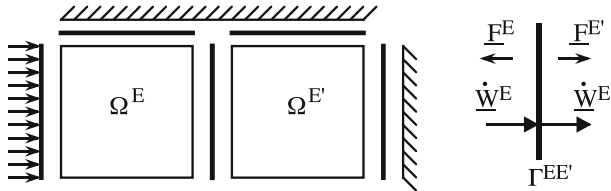


Fig. 4 Decomposition of an assembly; interface variables

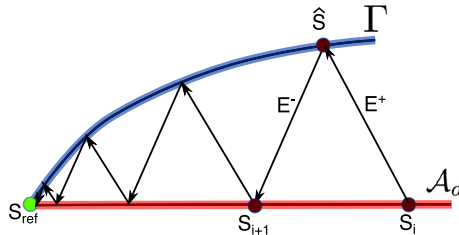


Fig. 5 Schematic representation of the iterations of the LATIN method

2.2 The algorithm

The solution s is described *a priori* as a set of time-dependent fields relative to the interface and the substructures. Here, the substructures have linear elastic behavior and the interior solution, (*i.e.* the displacement $\mathbf{u}^E(M, t)$ and the stress $\sigma^E(M, t)$), can be easily calculated from the boundary quantities $\dot{\mathbf{w}}^E(M, t)$ and $\mathbf{f}^E(M, t)$. The solution s can be represented using only the force and velocity fields on both sides of the interface:

$$s = \sum_E s^E, \quad s^E = \left\{ \dot{\mathbf{w}}^E(M, t), \mathbf{f}^E(M, t) \right\}, \quad \forall t \in [0, T] \tag{1}$$

Assuming that the substructures are elastic and that all the nonlinearities are concentrated at the interface, the equations can be divided into two groups:

- the set \mathcal{A}_d of the solutions s^E which satisfy the **linear** equations relative to the substructures, and
- the set Γ of the solutions s^E which satisfy the **local** (possibly nonlinear) equations relative to the interface.

Then, the solution of the problem is determined iteratively by seeking successive approximations s which satisfy the two groups of equations alternatively, using search directions E^+ and E^- (Fig. 5). Thus, the two steps of the iterative algorithm are:

- (local step) given $s_n \in \mathcal{A}_d$, find \hat{s} such that:

$$\hat{s} \in \Gamma \quad (\text{interfaces}) \tag{2}$$

$$\hat{s} - s_n \in E^+ \quad (\text{search direction}) \tag{3}$$

- (global step) given $\hat{s} \in \Gamma$, find s_{n+1} such that:

$$s_{n+1} \in \mathcal{A}_d \quad (\text{substructures}) \tag{4}$$

$$s_{n+1} - \hat{s} \in E^- \quad (\text{search direction}) \tag{5}$$

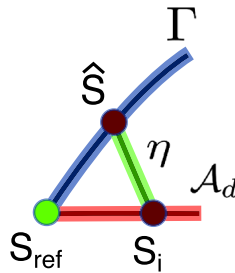


Fig. 6 Illustration of the error indicator of the LATIN method

Here, we use conjugate search directions which depend on a single scalar parameter k_0 :

$$\widehat{s} - s_n \in E^+ \iff (\widehat{\mathbf{f}}^E - \mathbf{f}_n^E) = k_0 (\widehat{\mathbf{w}}^E - \dot{\mathbf{w}}_n^E) \tag{6}$$

$$s_{n+1} - \widehat{s} \in E^- \iff (\mathbf{f}_{n+1}^E - \widehat{\mathbf{f}}^E) = -k_0 (\dot{\mathbf{w}}_{n+1}^E - \widehat{\mathbf{w}}^E) \tag{7}$$

The solution of the problem is independent of the value of parameter k_0 , which affects only the convergence rate of the algorithm. In our case of quasi-static calculations, k_0 is given by:

$$k_0 = \frac{ET}{L_c} \tag{8}$$

where E is the Young’s modulus, $[0, T]$ the time interval being considered and L_c the largest dimension of the structure.

2.3 Global error indicator

Since the LATIN algorithm is iterative, it is important to have an error indicator in order to be able to identify full or partial convergence. This error indicator is defined by:

$$\eta = \frac{\sum \|s_n^E - \widehat{s}^E\|^2}{\sum \|s_n^E\|^2 + \|\widehat{s}^E\|^2} \text{ with:} \tag{9}$$

$$\|s\|^2 = \int_{\partial\Omega^E} \mathbf{f}^T k_0 \mathbf{f} dS + \int_{\partial\Omega^E} \dot{\mathbf{w}}^T k_0 \dot{\mathbf{w}} dS \tag{10}$$

This is a global error indicator, which is what one uses to determine the accuracy of a solution. The error η is a relative distance between spaces \mathcal{A}_d and Γ (Fig. 6).

Definition A fully converged quantity is an approximate quantity whose estimated error is smaller than a reference value; a partially converged quantity is a quantity whose error is greater than the reference. Thus, the choice of this reference level is important. As we use an iterative algorithm that gives us an approximation of the solution of the problem at each iteration, we could stop the algorithm after a certain number of iterations. Moreover, at each iteration, we have an energy error indicator that qualifies the quality of the solution and is used as a stopping criterion. The reference level is set when the quantities of interest do not change. In our study, the reference levels were set, based on common engineering practice, at 1×10^{-4} for plane 2D problems. In this work, the partially converged data are computed for different values of η from 6×10^{-2} or 5×10^{-2} to 1×10^{-2} . The idea is to have in some iterations a trend of the result (around 10 iterations for a partially computation against 300

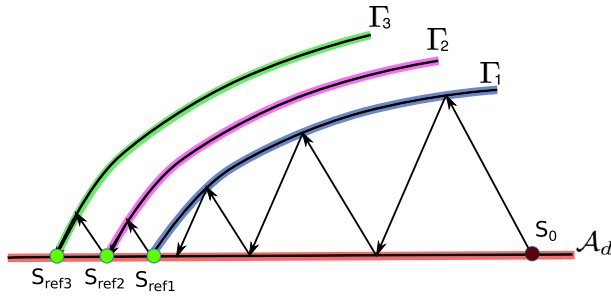


Fig. 7 Illustration of the multiparametric strategy

for a fully computation, 4 s of computation for partially converged data and 120 s for fully converged data for the third case test). The auxiliary data level (called partially converged data as the iterative algorithm is stopped before convergence) is chosen based on the study described in [3]. These levels have been chosen to have the best compromise possible between computational time, and robustness.

2.4 Multiparametric strategy

A parametric optimization involves many calculations, each carried out with a different set of the parameters of the problem. In our case, the changing parameters are the gap and the friction coefficient. Therefore, the problems to be solved are mathematically very similar. The method we use, called the MultiParametric Strategy (MPS), was introduced in [34] and applied to the construction of metamodels in [17].

This method uses the fact that, at each iteration, a solution is available over the whole loading path and at all points of the structure. In our case, space Γ alone is affected by a change in a friction coefficient or a gap. Thus, as shown in Fig. 7, it is possible to restart a resolution using a previous solution.

With this schematic representation, the number of iterations required in order to reach the converged solution S_{ref1} is greater than that required for S_{ref2} and S_{ref3} . The main problem with this method is to decide which restarting point to use: for example, in order to obtain the solution S_{ref3} , one can choose to restart from solution S_{ref1} or from solution S_{ref2} . In this work, the restarting point is chosen to be the nearest point in a Euclidean space. For further details, see [35].

3 The equations of the variable-fidelity model

As mentioned previously, this paper uses several variable-fidelity modeling techniques. In the first case (the use of a correction surface), we used the kriging principle in order to create the various metamodels. Therefore, we begin with a presentation of the kriging equation. Then, we introduce the cokriging equations (used for our second variable-fidelity case study) along with several cross correlations of the correlation matrix. Finally, we conclude this section with a presentation of hierarchical kriging.

3.1 The kriging equation

The following equations were developed for universal kriging and, therefore, can be applied to ordinary kriging (see [15, 16, 36]).

Let us assume that the objective function has the form:

$$y(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} + z(\mathbf{x}) \tag{11}$$

$z(\mathbf{x})$ is a stochastic process with the following properties:

$$\begin{aligned} E[z(\mathbf{x})] &= 0 \\ E[z(\mathbf{x})^2] &= \sigma^2 \end{aligned} \tag{12}$$

We define the predictor by means of a linear combination:

$$\hat{y}(\mathbf{x}) = \mathbf{c}^T(\mathbf{x})\mathbf{Y} \tag{13}$$

For this error indicator to be unbiased, we set:

$$E[\hat{y}(\mathbf{x}) - y(\mathbf{x})] = 0 \tag{14}$$

Thus, the unbiased condition can be written as:

$$E[\hat{y}(\mathbf{x}) - y(\mathbf{x})] = 0 \iff \mathbf{F}^T \mathbf{c}(\mathbf{x}) - \mathbf{f}(\mathbf{x}) = \mathbf{0} \tag{15}$$

In order to create the metamodel, we seek to minimize the mean square error:

$$MSE = \sigma^2 + \mathbf{c}^T \mathbf{C} \mathbf{c} - 2\mathbf{c}^T E[\mathbf{Z}z] \tag{16}$$

The covariance matrix is defined as:

$$\mathbf{C} = \begin{bmatrix} Cov(z(\mathbf{x}^1), z(\mathbf{x}^1)) & \dots & Cov(z(\mathbf{x}^1), z(\mathbf{x}^p)) \\ \vdots & \ddots & \vdots \\ Cov(z(\mathbf{x}^p), z(\mathbf{x}^1)) & \dots & Cov(z(\mathbf{x}^p), z(\mathbf{x}^p)) \end{bmatrix} \tag{17}$$

The calculation of this covariance matrix is the main difficulty of this strategy. In order to do that, one can use a correlation function (see [36]). Therefore, the only important parameter is the distance between the points. One can define the covariance between two points as: $Cov(z(\mathbf{x}^i), z(\mathbf{x}^j)) = \sigma^2 R(\mathbf{x}^i, \mathbf{x}^j)$, where R is a correlation function. We chose to use a generalized exponential correlation function $EXPG = \exp(\sum_{i=1}^n -\theta_i |w_i - x_i|^{\theta_{n+1}})$ with $1 < \theta_{n+1} < 2$.

Consequently, one has:

$$\mathbf{C} = \sigma^2 \begin{bmatrix} R(\mathbf{x}^1, \mathbf{x}^1) & \dots & R(\mathbf{x}^1, \mathbf{x}^p) \\ \vdots & \ddots & \vdots \\ R(\mathbf{x}^p, \mathbf{x}^1) & \dots & R(\mathbf{x}^p, \mathbf{x}^p) \end{bmatrix} \tag{18}$$

\mathbf{C} is a symmetric matrix due to the symmetry properties of function R . Therefore, one can rewrite Eq. 16 as:

$$MSE = \sigma^2 + \sigma^2 \mathbf{c}^T \mathbf{R} \mathbf{c} - 2\sigma^2 \mathbf{c}^T \mathbf{r} \tag{19}$$

$$\mathbf{r}(\mathbf{x}) = [R(\mathbf{x}^1, \mathbf{x}) \dots R(\mathbf{x}^p, \mathbf{x})]^T$$

Now, the objective is to minimize the mean square error under the unbiased constraint. The Lagrangian has the expression:

$$\mathcal{L}(\mathbf{c}, \boldsymbol{\lambda}) = \sigma^2(1 + \mathbf{c}^T \mathbf{R} \mathbf{c} - 2\mathbf{c}^T \mathbf{r}) + \boldsymbol{\lambda}^T (\mathbf{F}^T \mathbf{c} - \mathbf{f}) \tag{20}$$

Therefore to find the optimum parameter the following system has to solved:

$$\begin{bmatrix} \mathbf{R} & \mathbf{F} \\ \mathbf{F}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{r} \\ \mathbf{f} \end{bmatrix} \tag{21}$$

Then:

$$\begin{aligned} \frac{\lambda}{2\sigma^2} &= (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{r} - \mathbf{f}) \\ \mathbf{c} &= \mathbf{R}^{-1} \left(\mathbf{r} - \mathbf{F} \frac{\lambda}{2\sigma^2} \right) \end{aligned} \tag{22}$$

Finally, the solution takes the form:

$$\begin{aligned} \hat{y}(\mathbf{x}) &= \mathbf{c}(\mathbf{x})^T \mathbf{Y} \\ \hat{y}(\mathbf{x}) &= \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} + \mathbf{r}(\mathbf{x})^T \mathbf{R}^{-1} (\mathbf{Y} - \mathbf{F}\boldsymbol{\beta}) \end{aligned} \tag{23}$$

with $\boldsymbol{\beta} = (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{Y})$.

The difficulty consists in determining the parameter of function R . In order to do that, we seek the maximum of the likelihood function:

$$\frac{1}{\sqrt{(2\pi)^n |\mathbf{C}|}} e^{-\frac{(\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})^T \mathbf{C}^{-1} (\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})}{2}} \tag{24}$$

$$\iff \frac{1}{\sqrt{(2\pi\sigma^2)^n |\mathbf{R}|}} e^{-\frac{(\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})^T \mathbf{R}^{-1} (\mathbf{Y}-\mathbf{F}\boldsymbol{\beta})}{2\sigma^2}} \tag{25}$$

Since \mathbf{R} is positive definite, so is its inverse \mathbf{R}^{-1} . Thus, one has:

$$\boldsymbol{\beta} = (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{Y}) \tag{26}$$

$$\sigma^2 = \frac{(\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})^T \mathbf{R}^{-1} (\mathbf{Y} - \mathbf{F}\boldsymbol{\beta})}{n} \tag{27}$$

Finally, we obtain the error indicator of the mean square error as:

$$\begin{aligned} MSE &= \sigma^2 \left(1 + \mathbf{u}^T (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{u} - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} \right) \\ &\text{with } \mathbf{u} = \mathbf{F}^T \mathbf{R}^{-1} \mathbf{r} - \mathbf{f} \end{aligned} \tag{28}$$

3.2 The cokriging equation

We tested three different variable-fidelity cokriging methods. Their general construction is similar, but they differ in the covariance matrix \mathbf{C} , especially in the cross correlation term [22,23,27].

3.2.1 Cokriging in general

In our work, we used only two fidelity levels, but one could use more. We have two types of data: low-fidelity data and high-fidelity data. Since we are using an iterative solver, the high-fidelity data are obtained after full convergence of the calculation, whereas the low-fidelity data correspond to a partially converged calculation. Let $\mathbf{X}_{pcv} = [\mathbf{x}_{pcv}^1 \cdots \mathbf{x}_{pcv}^n]^T$ and \mathbf{Y}_{pcv} denote respectively the points of the partially converged design space \mathcal{D} and the corresponding

evaluations. Similarly, let $\mathbf{X}_{f_{cv}} = [\mathbf{x}_{f_{cv}}^1 \cdots \mathbf{x}_{f_{cv}}^{n_{f_{cv}}}]^T$ and $\mathbf{Y}_{f_{cv}}$ denote respectively the points of the fully converged design space \mathcal{D} and the corresponding evaluations. Of course, in our case, $n_{pcv} \gg n_{f_{cv}}$ where n_{pcv} and $n_{f_{cv}}$ are respectively the number of evaluated points for the partially and fully converged data.

With these two different evaluations, one can write two different formulations:

$$\begin{aligned} y_{f_{cv}}(\mathbf{x}) &= \mathbf{f}_{f_{cv}}^T(\mathbf{x})\boldsymbol{\beta}_{f_{cv}} + z_{f_{cv}}(\mathbf{x}) \\ y_{pcv}(\mathbf{x}) &= \mathbf{f}_{pcv}^T(\mathbf{x})\boldsymbol{\beta}_{pcv} + z_{pcv}(\mathbf{x}) \end{aligned} \tag{29}$$

Thus, if the regression is linear, we get the following matrices: $\mathbf{F}_{pcv} = [\mathbf{f}_{pcv}(\mathbf{x}^1) \cdots \mathbf{f}_{pcv}(\mathbf{x}^{n_{pcv}})]^T \in \mathbb{R}^{n_{pcv} \times n+1}$ and $\mathbf{F}_{f_{cv}} = [\mathbf{f}_{f_{cv}}(\mathbf{x}^1) \cdots \mathbf{f}_{f_{cv}}(\mathbf{x}^{n_{f_{cv}}})]^T \in \mathbb{R}^{n_{f_{cv}} \times n+1}$

The objective now is to build a metamodel based on these two sources of information:

$$\hat{y}_{f_{cv}}(\mathbf{x}) = \mathbf{c}_{f_{cv}}^T(\mathbf{x})\mathbf{Y}_{f_{cv}} + \mathbf{c}_{pcv}^T(\mathbf{x})\mathbf{Y}_{pcv} \tag{30}$$

Let us seek the unbiased condition:

$$\hat{y}_{f_{cv}}(\mathbf{x}) - y_{f_{cv}}(\mathbf{x}) = \mathbf{c}_{f_{cv}}^T(\mathbf{x})\mathbf{Y}_{f_{cv}} + \mathbf{c}_{pcv}^T(\mathbf{x})\mathbf{Y}_{pcv} - y_{f_{cv}}(\mathbf{x}) \tag{31}$$

This unbiased condition can be written as:

$$\mathbf{F}_{f_{cv}}^T \mathbf{c}_{f_{cv}} - \mathbf{f}_{f_{cv}} = 0 \tag{32}$$

$$\mathbf{c}_{pcv}^T \mathbf{F}_{pcv} = 0 \tag{33}$$

As we did previously, we can calculate the mean square error:

$$MSE(\hat{y}_{f_{cv}}(\mathbf{x}) - y_{f_{cv}}(\mathbf{x})) = E[(\hat{y}_{f_{cv}}(\mathbf{x}) - y_{f_{cv}}(\mathbf{x}))^2] \tag{34}$$

Then we define the Lagrangian:

$$\mathcal{L}(\mathbf{c}_{f_{cv}}, \mathbf{c}_{pcv}, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) = MSE + \boldsymbol{\lambda}_1^T (\mathbf{F}_{f_{cv}}^T \mathbf{c}_{f_{cv}} - \mathbf{f}_{f_{cv}}) + \boldsymbol{\lambda}_2^T (\mathbf{F}_{pcv}^T \mathbf{c}_{pcv}) \tag{35}$$

Finally we obtain the system of equations:

$$\begin{bmatrix} \sigma_{f_{cv}}^2 \mathbf{R}_{f_{cv}} & \sigma_{f_{cv}} \sigma_{pcv} \mathbf{R}_{f-pcv} & \mathbf{F}_{f_{cv}} & \mathbf{0} \\ \sigma_{f_{cv}} \sigma_{pcv} \mathbf{R}_{f-pcv}^T & \sigma_{pcv}^2 \mathbf{R}_{pcv} & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{f_{cv}}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{f_{cv}} \\ \mathbf{c}_{pcv} \\ \boldsymbol{\lambda}_1/2 \\ \boldsymbol{\lambda}_2/2 \end{bmatrix} = \mathbf{r} \tag{36}$$

with

$$\begin{aligned} \mathbf{r} &= [\sigma_{f_{cv}}^2 \mathbf{r}_{f_{cv}} \quad \sigma_{f_{cv}} \sigma_{pcv} \mathbf{r}_{f-pcv} \quad \mathbf{f}_{f_{cv}} \quad \mathbf{0}]^T \\ \mathbf{r}_{f_{cv}}(\mathbf{x}) &= [R(\mathbf{x}_{f_{cv}}^1, \mathbf{x}) \cdots R(\mathbf{x}_{f_{cv}}^{n_{f_{cv}}}, \mathbf{x})]^T \\ \mathbf{r}_{f-pcv}(\mathbf{x}) &= [R_{f-pcv}(\mathbf{x}_{pcv}^1, \mathbf{x}) \cdots R_{f-pcv}(\mathbf{x}_{pcv}^{n_{pcv}}, \mathbf{x})]^T \end{aligned} \tag{37}$$

3.2.2 The first cross correlation strategy

This method is described more precisely in [22]. In this strategy, we assume that the primary and secondary data have the same spatial intercorrelation, which means that:

$$E[z_{f_{cv}}(\mathbf{x}), z_{f_{cv}}(\hat{\mathbf{x}})] = E[z_{pcv}(\mathbf{x}), z_{pcv}(\hat{\mathbf{x}})] \quad \forall \mathbf{x}, \hat{\mathbf{x}} \tag{38}$$

This is equivalent to $\sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}, \hat{\mathbf{x}}) = \sigma_{fcv}^2 \mathbf{R}_{fcv}(\mathbf{x}, \hat{\mathbf{x}})$. Thus, if $\mathbf{x} = \hat{\mathbf{x}}$, then $\sigma_{pcv} = \sigma_{fcv} = \sigma$. This final relation requires that $\mathbf{R}_{pcv}(\mathbf{x}, \hat{\mathbf{x}}) = \mathbf{R}_{fcv}(\mathbf{x}, \hat{\mathbf{x}}) = \mathbf{R}(\mathbf{x}, \hat{\mathbf{x}})$. [22] introduced an additional parameter $\gamma \in [0, 1]$ in order to model the cross correlation $\mathbf{R}_{f-pcv}(\mathbf{x}, \hat{\mathbf{x}}) = \gamma \mathbf{R}(\mathbf{x}, \hat{\mathbf{x}})$.

$$\begin{bmatrix} \sigma^2 \mathbf{R}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) & \gamma \sigma^2 \mathbf{R}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) & \mathbf{F}_{fcv} & \mathbf{0} \\ \gamma \sigma^2 \mathbf{R}(\mathbf{x}_{pcv}, \mathbf{x}_{fcv}) & \sigma^2 \mathbf{R}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{pcv} \\ \lambda_1/2 \\ \lambda_2/2 \end{bmatrix} = \begin{bmatrix} \sigma^2 \mathbf{r}_{fcv}(\mathbf{x}) \\ \sigma^2 \mathbf{r}_{f-pcv}(\mathbf{x}) \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \tag{39}$$

We end up with the following linear system:

$$\begin{bmatrix} \mathbf{R}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) & \gamma \mathbf{R}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) & \mathbf{F}_{fcv} & \mathbf{0} \\ \gamma \mathbf{R}(\mathbf{x}_{pcv}, \mathbf{x}_{fcv}) & \mathbf{R}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{pcv} \\ \lambda_1/(2\sigma^2) \\ \lambda_2/(2\sigma^2) \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{fcv}(\mathbf{x}) \\ \mathbf{r}_{f-pcv}(\mathbf{x}) \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \tag{40}$$

One can see that this system can be written in exactly the same form as Eq. 21:

$$\begin{bmatrix} \tilde{\mathbf{R}} & \tilde{\mathbf{F}} \\ \tilde{\mathbf{F}}^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{c}} \\ \tilde{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{r}} \\ \tilde{\mathbf{f}} \end{bmatrix} \tag{41}$$

Finally, this leads to the following predictor:

$$\hat{\mathbf{y}}(\mathbf{x}) = \tilde{\mathbf{f}}(\mathbf{x})^T \boldsymbol{\beta} + \tilde{\mathbf{r}}(\mathbf{x})^T \tilde{\mathbf{R}}^{-1} (\tilde{\mathbf{Y}} - \tilde{\mathbf{F}} \boldsymbol{\beta}) \tag{42}$$

with $\boldsymbol{\beta} = (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{F}})^{-1} (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{Y}})$ and $\tilde{\mathbf{Y}} = [\mathbf{Y}_{fcv} \ \mathbf{Y}_{pcv}]^T$.

With these notations, the mean square error can be written exactly as in Eq. 28:

$$MSE = \sigma^2 \left(1 + \tilde{\mathbf{u}}^T (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{F}})^{-1} \tilde{\mathbf{u}} - \tilde{\mathbf{r}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{r}} \right) \tag{43}$$

with $\tilde{\mathbf{u}} = \tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{r}} - \tilde{\mathbf{f}}$

An important point is the way to tune the parameter γ . To do this, the likelihood function is used and γ is an extra parameter to optimize. The equation of the likelihood function has the same form as Eq. 25.

3.2.3 The second cross correlation strategy

This method is more general than the previous one, but some choices had to be made (see [23,37]).

$$\begin{bmatrix} \sigma_{fcv}^2 \mathbf{R}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) & \sigma_{fcv} \sigma_{pcv} \mathbf{R}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) & \mathbf{F}_{fcv} & \mathbf{0} \\ \sigma_{fcv} \sigma_{pcv} \mathbf{R}(\mathbf{x}_{pcv}, \mathbf{x}_{fcv}) & \sigma_{pcv}^2 \mathbf{R}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{fcv} \\ \mathbf{c}_{pcv} \\ \lambda_1/2 \\ \lambda_2/2 \end{bmatrix} = \mathbf{r} \tag{44}$$

This matrix can be modified into:

$$\begin{bmatrix} \mathbf{R}_{fcv} & \mathbf{R}_{f-pcv} & \mathbf{F}_{fcv} & \mathbf{0} \\ \mathbf{R}_{f-pcv}^T & \mathbf{R}_{pcv} & \mathbf{0} & \mathbf{F}_{pcv} \\ \mathbf{F}_{fcv}^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{pcv}^T & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{c}_{fcv} \\ \frac{\sigma_{pcv}}{\sigma_{fcv}} \mathbf{c}_{pcv} \\ \lambda_1/(2\sigma_{fcv}^2) \\ \lambda_2/(2\sigma_{fcv} \sigma_{pcv}) \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{fcv} \\ \mathbf{r}_{f-pcv} \\ \mathbf{f}_{fcv} \\ \mathbf{0} \end{bmatrix} \tag{45}$$

Let us use the notations $\tilde{\mathbf{c}}_{fcv} = \mathbf{c}_{fcv}$, $\tilde{\mathbf{c}}_{pcv} = \frac{\sigma_{pcv}}{\sigma_{fcv}} \mathbf{c}_{pcv}$, $\tilde{\lambda}_1 = \lambda_1 / (2\sigma_{fcv}^2)$ and $\tilde{\lambda}_2 = \lambda_2 / (2\sigma_{fcv}\sigma_{pcv})$. We also write $\tilde{\mathbf{c}} = [\tilde{\mathbf{c}}_{fcv} \ \tilde{\mathbf{c}}_{pcv}]^T$ and $\tilde{\boldsymbol{\lambda}} = [\tilde{\lambda}_1 \ \tilde{\lambda}_2]^T$. Equation 30 becomes:

$$\hat{y}_{fcv}(\mathbf{x}) = \tilde{\mathbf{c}}_{fcv}(\mathbf{x})\mathbf{Y}_{fcv} + \tilde{\mathbf{c}}_{pcv}(\mathbf{x})\frac{\sigma_{fcv}}{\sigma_{pcv}}\mathbf{Y}_{pcv} \tag{46}$$

We define $\tilde{\mathbf{Y}}_s = \left[\mathbf{Y}_{fcv} \ \frac{\sigma_{fcv}}{\sigma_{pcv}}\mathbf{Y}_{pcv} \right]^T$.

Finally, we get the system:

$$\begin{bmatrix} \tilde{\mathbf{R}} & \tilde{\mathbf{F}} \\ \tilde{\mathbf{F}}^T & 0 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{c}} \\ \tilde{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{r}} \\ \tilde{\mathbf{f}} \end{bmatrix} \tag{47}$$

and we get the same interpolator as previously:

$$\hat{y}(\mathbf{x}) = \tilde{\mathbf{f}}(\mathbf{x})^T \boldsymbol{\beta} + \tilde{\mathbf{r}}(\mathbf{x})^T \tilde{\mathbf{R}}^{-1} (\tilde{\mathbf{Y}}_s - \tilde{\mathbf{F}}\boldsymbol{\beta}) \tag{48}$$

with $\boldsymbol{\beta} = (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{F}})^{-1} (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{Y}}_s)$.

The mean square error is defined as before:

$$\begin{aligned} MSE &= \sigma_{fcv}^2 \left(1 + \tilde{\mathbf{u}}^T (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{F}})^{-1} \tilde{\mathbf{u}} - \tilde{\mathbf{r}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{r}} \right) \\ &\text{with } \tilde{\mathbf{u}} = \tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{r}} - \tilde{\mathbf{f}} \end{aligned} \tag{49}$$

Moreover, we define $\mathbf{R}_{fcv} = 0.9999\mathbf{R}_{f-pcv} = \mathbf{R}_{pcv}$. The parameter 0.9999 is introduced to avoid singularity in the correlation matrix. Again, the main difficulty is to obtain the ratio $\frac{\sigma_{fcv}}{\sigma_{pcv}}$. In order to do that, one also uses the likelihood function L as in[23]. We precise that further details can be found in [37].

$$L \left(\boldsymbol{\beta}, \frac{\sigma_{fcv}}{\sigma_{pcv}}, \sigma_{pcv}^2, \theta_{fcv}, \theta_{pcv}, \theta_{f-pcv} \right) = \frac{1}{\sqrt{(2\pi\sigma_{fcv}^2)^{n_{fcv}+n_{pcv}} |\tilde{\mathbf{R}}|}} e^{-\frac{(\tilde{\mathbf{Y}}_s - \tilde{\mathbf{F}}\boldsymbol{\beta})^T \tilde{\mathbf{R}}^{-1} (\tilde{\mathbf{Y}}_s - \tilde{\mathbf{F}}\boldsymbol{\beta})}{2\sigma_{fcv}^2}} \tag{50}$$

By taking logarithm and derivatives, it is possible to find the optima of the parameters $\boldsymbol{\beta}$, $\frac{\sigma_{fcv}}{\sigma_{pcv}}$ and σ_{pcv}^2 . We find respectively :

$$\boldsymbol{\beta} = (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{F}})^{-1} (\tilde{\mathbf{F}}^T \tilde{\mathbf{R}}^{-1} \tilde{\mathbf{Y}}_s) \tag{51}$$

$$\frac{\sigma_{fcv}}{\sigma_{pcv}} = \left(\begin{bmatrix} 0 \\ \mathbf{Y}_{pcv} \end{bmatrix}^T \tilde{\mathbf{R}}^{-1} \begin{bmatrix} 0 \\ \mathbf{Y}_{pcv} \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ \mathbf{Y}_{pcv} \end{bmatrix}^T \tilde{\mathbf{R}}^{-1} \begin{bmatrix} \mathbf{F}_{fcv}\boldsymbol{\beta}_{fcv} - \mathbf{Y}_{fcv} \\ \mathbf{F}_{pcv}\boldsymbol{\beta}_{pcv} \end{bmatrix} \tag{52}$$

and

$$\sigma^2 = \frac{(\tilde{\mathbf{Y}}_s - \tilde{\mathbf{F}}\boldsymbol{\beta})^T \tilde{\mathbf{R}}^{-1} (\tilde{\mathbf{Y}}_s - \tilde{\mathbf{F}}\boldsymbol{\beta})}{n_{fcv} + n_{pcv}} \tag{53}$$

Equations 51, 52 and 53 allow to compute values of the parameters. For example, in [37] an iterative strategy is used to determined these parameters. The others parameters θ_{fcv} , θ_{pcv} and θ_{f-pcv} can be found by maximizing the log-likelihood function.

3.2.4 The third cross correlation strategy: the autoregressive method

In this method, we use an autoregressive model as described in [38]. The predictor used is defined by Forrester [27], which is slightly different from the others. The difference is that we assume that $\beta_{pcv} = \beta_{fcv}$, so the regression terms are equal. This method is based on an auto-regressive model based on the assumption that $\text{cov}(Y_{fcv}(\mathbf{x}_i), Y_{pcv}(\mathbf{x}) | Y_{pcv}(\mathbf{x}^i)) = 0, \forall \mathbf{x} \neq \mathbf{x}^i$. This Markov property means that nothing more can be learned about $Y_{fcv}(\mathbf{x}^i)$ from the less expensive model if the value of the more expensive function at \mathbf{x}^i is known (see [38]).

The auto-regressive model we use approximates the fully converged model using the partially converged model with a scaling factor γ plus a Gaussian process \hat{y}_{cor} :

$$\hat{y}_{fcv}(\mathbf{x}) = \gamma \hat{y}_{pcv}(\mathbf{x}) + \hat{y}_{cor}(\mathbf{x}) \tag{54}$$

We assume that \hat{y}_{cor} and \hat{y}_{pcv} are independent, which leads to $\text{Cov}(\hat{y}_{cor}(\mathbf{x}), \hat{y}_{pcv}(\hat{\mathbf{x}})) = 0 \forall (\mathbf{x}, \hat{\mathbf{x}})$

This relation leads to the following terms:

$$\begin{aligned} \text{Cov}(\mathbf{Y}_{fcv}(\mathbf{x}_{fcv}), \mathbf{Y}_{fcv}(\mathbf{x}_{fcv})) &= \text{Cov}(\mathbf{Z}_{fcv}(\mathbf{x}_{fcv}), \mathbf{Z}_{fcv}(\mathbf{x}_{fcv})) \\ &= \gamma^2 \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) + \sigma_{cor}^2 \mathbf{R}_{cor}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) \end{aligned} \tag{55}$$

$$\begin{aligned} \text{Cov}(\mathbf{Y}_{fcv}(\mathbf{x}_{fcv}), \mathbf{Y}_{pcv}(\mathbf{x}_{pcv})) &= \text{Cov}(\mathbf{Z}_{fcv}(\mathbf{x}_{fcv}), \mathbf{Z}_{pcv}(\mathbf{x}_{pcv})) \\ &= \gamma \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) \end{aligned} \tag{56}$$

$$\begin{aligned} \text{Cov}(\mathbf{Y}_{pcv}(\mathbf{x}_{pcv}), \mathbf{Y}_{pcv}(\mathbf{x}_{pcv})) &= \text{Cov}(\mathbf{Z}_{pcv}(\mathbf{x}_{pcv}), \mathbf{Z}_{pcv}(\mathbf{x}_{pcv})) \\ &= \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) \end{aligned} \tag{57}$$

The final matrix is:

$$\mathbf{C} = \begin{pmatrix} \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{pcv}) & \gamma \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{pcv}, \mathbf{x}_{fcv}) \\ \gamma \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{pcv}) & \gamma^2 \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) + \sigma_{cor}^2 \mathbf{R}_{cor}(\mathbf{x}_{fcv}, \mathbf{x}_{fcv}) \end{pmatrix} \tag{58}$$

The final predictor and its mean square error are:

$$\begin{aligned} \hat{y}(\mathbf{x}) &= \tilde{\mathbf{f}}(\mathbf{x})^T \boldsymbol{\beta} + \tilde{\mathbf{c}}(\mathbf{x})^T \mathbf{C}^{-1} (\tilde{\mathbf{Y}} - \tilde{\mathbf{F}} \boldsymbol{\beta}) \\ \text{MSE} &= \sigma_{cor}^2 + \gamma \sigma_{pcv}^2 + \left(\mathbf{u}^T (\mathbf{F}^T \mathbf{C}^{-1} \mathbf{F})^{-1} \mathbf{u} - \tilde{\mathbf{c}}^T \mathbf{C}^{-1} \tilde{\mathbf{c}} \right) \\ \text{with } \tilde{\mathbf{c}}(\mathbf{x}) &= \begin{pmatrix} \gamma \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{pcv}, \mathbf{x}) \\ \gamma^2 \sigma_{pcv}^2 \mathbf{R}_{pcv}(\mathbf{x}_{fcv}, \mathbf{x}) + \sigma_{cor}^2 \mathbf{R}_{cor}(\mathbf{x}_{fcv}, \mathbf{x}) \end{pmatrix} \\ \text{with } \mathbf{u} &= \mathbf{F}^T \mathbf{C}^{-1} \tilde{\mathbf{c}} - \mathbf{f} \end{aligned} \tag{59}$$

The parameter γ is defined as an extra parameter of the likelihood function of the meta-model \hat{y}_{cor} .

3.3 Hierarchical kriging

Hierarchical kriging is a very attractive method because it is very simple to use. The only difference between kriging and hierarchical kriging is in the regression function. Rather than using a basis of functions, one uses the low-fidelity metamodel. As shown in [30], $y(\mathbf{x})$ can

be expressed as:

$$y(\mathbf{x}) = \hat{y}_{pcv}(\mathbf{x})\boldsymbol{\beta} + z(\mathbf{x}) \tag{60}$$

The regression term is replaced by the low-fidelity model (in this case, the model derived from partially converged points). Thus, the only change compared to kriging is that \mathbf{F} is replaced by $\hat{\mathbf{Y}}_{pcv} \in \mathbb{R}^{n_{fcv} \times 1}$, with:

$$\hat{\mathbf{Y}}_{pcv} = (\hat{y}_{pcv}(\mathbf{x}^1) \dots \hat{y}_{pcv}(\mathbf{x}^{n_{fcv}}))^T \tag{61}$$

So, the important parameter of this method is $\boldsymbol{\beta}$. This parameter is defined thanks to the likelihood function defined in 25 with the only change that \mathbf{F} is replaced by $\hat{\mathbf{Y}}_{pcv}$:

$$L(\boldsymbol{\beta}, \sigma_{fcv}, \theta_{fcv}) = \frac{1}{\sqrt{(2\pi\sigma_{fcv}^2)^{n_{fcv}} |\mathbf{R}|}} e^{-\frac{(\mathbf{Y} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta})^T \mathbf{R}^{-1} (\mathbf{Y} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta})}{2\sigma_{fcv}^2}} \tag{62}$$

By taking the logarithm of this expression and derivating we find :

$$\boldsymbol{\beta} = \left(\hat{\mathbf{Y}}_{pcv}^T \mathbf{R}^{-1} \hat{\mathbf{Y}}_{pcv} \right)^{-1} \left(\hat{\mathbf{Y}}_{pcv}^T \mathbf{R}^{-1} \mathbf{Y} \right) \tag{63}$$

$$\sigma_{fcv}^2 = \frac{\left(\mathbf{Y} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta} \right)^T \mathbf{R}^{-1} \left(\mathbf{Y} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta} \right)}{n_{fcv}} \tag{64}$$

So, it is now possible to define the predictor of the metamodel and its mean square error :

$$\begin{aligned} \hat{y}(\mathbf{x}) &= \hat{y}_{pcv}(\mathbf{x})\boldsymbol{\beta} + \mathbf{r}(\mathbf{x})^T \mathbf{R}^{-1} (\mathbf{Y} - \hat{\mathbf{Y}}_{pcv}\boldsymbol{\beta}) \\ MSE &= \sigma_{fcv}^2 \left(1 + \mathbf{u}^T \left(\hat{\mathbf{Y}}_{pcv}^T \mathbf{R}^{-1} \hat{\mathbf{Y}}_{pcv} \right)^{-1} \mathbf{u} - \mathbf{r}^T \mathbf{R}^{-1} \mathbf{r} \right) \\ \text{with } \mathbf{u} &= \hat{\mathbf{Y}}_{pcv}^T \mathbf{R}^{-1} \mathbf{r} - \hat{y}_{pcv} \end{aligned} \tag{65}$$

The implementation of this method is not quite difficult, it is similar to ordinary kriging except that the regression function $\mathbf{f}(\mathbf{x})$ is no more equal to 1 but is equal to the estimation of the metamodel constructed thanks to the partially converged points.

3.4 Characterizations of the methods

EvoFusion: the correction-based VFM with additive bridge function [39]. It is well suited for the case when the low fidelity data is sufficiently correlated with high-fidelity data (or have the similar variation trend). But when the correlation is relatively small, we get less benefit. The risk of using this kind of method is that when the low-fidelity data misses the correct trend of high-fidelity data, it can be even worse than the method of building surrogate model only using the high-fidelity data (the test case 3 of this manuscript also validate this point).

First cokriging method: it also assumes that the low- and high-fidelity data are sufficiently correlated to each other, and then have the same risk as the first method does. But the introduction of an additional parameter called γ makes it more flexible and could be more accurate than method 1.

Second cokriging method: theoretically, it can be viewed as a generalization of the first cokriging method. It provides an additional parameter called $\sigma_{fcv}/\sigma_{pcv}$ which can act as an indicator of how the low- and high-fidelity data are correlated to each other. The introduction of this parameter helps to automatically adjust the influence of low-fidelity data on the resulting VFM prediction, which in turn helps to avoid the risk of building VFM.

Third cokriging method: it differs from the 1st and 2nd cokrigings for the way of how to calculate the so-called cross variance. This is a well-accepted method and theoretically it should be as accurate as the 1st and 2nd cokrigings. A restriction of this method is that the high-fidelity sample sites have to be the subset of the low-fidelity sample sites; otherwise interpolation has to be performed. There is a parameter called γ which is an scaling factor of low-fidelity data to the prediction of high fidelity data.

Hierarchical kriging: a simple and robust method. Compared to the cokrigings, the formulation as well as the implementation is much simpler. The correlation matrix is much smaller and there is no need to calculate the so-called cross correlation or cross covariance as a cokriging does. A parameter called β is introduced to account for the influence of low-fidelity data on VFM prediction. Theoretically, it should be almost as accurate as methods of cokrigings 1–3, and should not be worse than evofusion.

The main interest of this work is to test these methods in a engineering contest, and some results differ from theoretical results and some results found in the literature. Two explanations can be given, the first one is that the parameters which control the different methods are not well tuned because of the genetic algorithm used to optimize the likelihood which miss the optimum parameters. Indeed, even for an example in 1D, this function can be difficult to optimize. The second reason for the differences is that for a large part of the test performed even the fully converged points give a poor metamodel describing not well the high-fidelity model, contrary to example in [27] where only with the high-fidelity data a kriged metamodel has a correlation r^2 equal to 0.949 whereas with a cokriged metamodel we have a correlation of 0.96. There is also the fact that the partially converged data are not very well correlated with the high-fidelity model contrary as for example in [22]. And between the partially and fully converged data there is no simple relation as for the analytical example as $\hat{y}_{fcv} = \gamma \hat{y}_{pcv} + \hat{y}_{cor}$.

3.5 Validation of the methods

In this section, we illustrate the cokriging and hierarchical kriging techniques on analytical example. We use two examples the first one comes from [27], the second one from [22]. The first example has its high-fidelity and its low-fidelity models defined by:

$$\begin{aligned} \text{HF} : x \mapsto y_{hf}(x) &= (6x - 2)^2 \times \sin(12x - 4) \\ \text{LF} : x \mapsto y_{lf}(x) &= A \times (6x - 2)^2 \times \sin(12x - 4) + 10(x - 0.5) - 5 \end{aligned} \tag{66}$$

The factor A can take several values, here we have used $A = 0.5$, $A = 0.08$ or $A = 100$. The second analytical example is defined by:

$$\begin{aligned} \text{HF} : x \mapsto y_{hf}(x) &= \sin\left(\frac{2\pi x}{5}\right) - 0.5x + 5 \\ \text{LF} : x \mapsto y_{lf}(x) &= 0.9 \sin\left(\frac{2\pi x}{5}\right) - 0.3x + 5 \end{aligned} \tag{67}$$

To check the implementation of these methods (especially the cokriging one), we chose to use the same correlation function as in the original paper dedicated to the cokriging technique. The red points correspond to the low-fidelity model evaluations and the green square correspond to the high-fidelity model evaluations (see Fig. 8).

So for the first cokriging method used in this paper, as in the original paper [22], the second analytical example is treated. The correlation function used is the gaussian function. As in [22], ordinary cokriging (OCK), universal kriging with a regression function at the order one

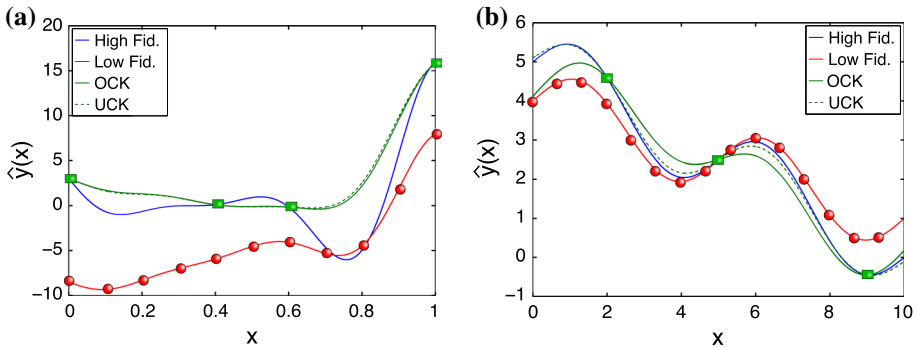


Fig. 8 Illustration of the first cokriging method. **a** First analytical example, **b** second analytical example, OCK: $\theta = 0.1609$ and $\gamma = 0.8040$, UCK: $\theta = 0.1609$ and $\gamma = 0.9990$ [22]

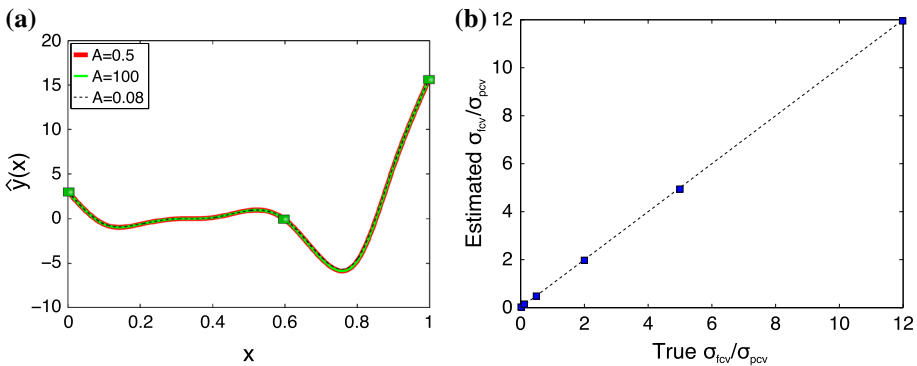


Fig. 9 Results for the second cokriging method. **a** Second cokriging method, **b** comparison of estimated and true value of $\sigma_{fcv}/\sigma_{pcv}$

(UCK) are computed. The parameters of cokriging are the ones written in [22] and the same response as in the original paper is obtained. We have also tested the first example to show that this cokriging method is not well suited on this specific example (see Fig. 8).

The second cokriging method is tested with the first example for several values of A . The correlation function used is the spline function [37]. The comparison of the estimated $\sigma_{fcv}/\sigma_{pcv}$ and the “true” $\sigma_{fcv}/\sigma_{pcv}$ is shown in Fig. 9. As in [37], the estimated $\sigma_{fcv}/\sigma_{pcv}$ is obtained by the model fitting method, while the “true” $\sigma_{fcv}/\sigma_{pcv}$ is calculated based on a $\sigma_{fcv}^2/\sigma_{pcv}^2$ obtained by fitting kriging models based on a large number of high- and low-fidelity samples.

The third cokriging method is used with the generalized exponential [27]. The hierarchical kriging is illustrated on the Fig. 10.

The results obtained are the same as in the original paper.

4 The tools we used

4.1 Sampling plans

In developing these strategies, an important parameter is whether $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ or not. Indeed, in the case of the evofusion principle, $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$.

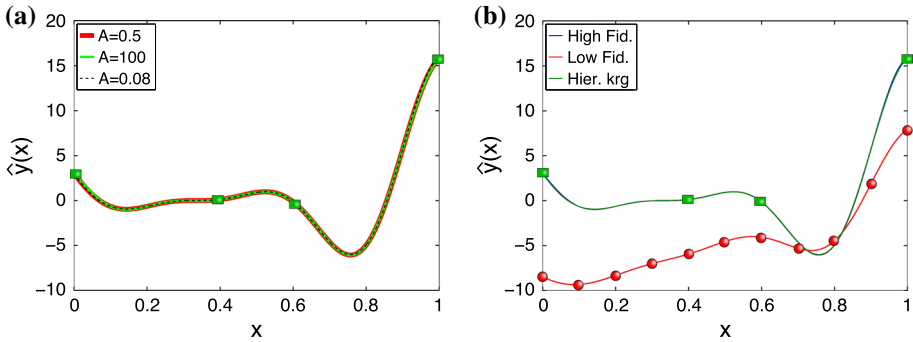


Fig. 10 Implementation of the third cokriging and hierarchical kriging. **a** Third cokriging method, **b** hierarchical kriging with $A = 0.5$

The equations of the auto-regressive model (the third cokriging method) assume that $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ (see [27]). One can also write these equations if $\mathbf{x}_{fcv} \not\subset \mathbf{x}_{pcv}$, in which case one can evaluate $y_{pcv}(\mathbf{x}_{fcv})$ as $\hat{y}_{pcv}(\mathbf{x}_{fcv})$. The other two cokriging strategies as well as hierarchical kriging work the same in both cases $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ and $\mathbf{x}_{fcv} \not\subset \mathbf{x}_{pcv}$. Therefore, in this paper, we compare three strategies (the first and second cokriging methods and hierarchical kriging) on the basis of whether property $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ is used or not.

An important aspect of this work was to build a good data set. The partially converged points were generated by a Latin Hypercube method (LHS) [40] using the function “lhsdesign” in Matlab. This approach is quite classical, but an important point was the selection of a subset \mathbf{x}_{fcv} of \mathbf{x}_{pcv} . This had to be a “smart” selection because we needed to span the whole parameter space. Thus, as in [27], we selected these points using a Morris–Mitchell criterion [41] (i.e. the minimum of $\phi_p = [\sum_j d_j^{-p}]^{1/p}$, with j being a pair of points and d_j the distance between these points). This is nothing but a combinatorial problem, but the difficulty is in the number of possible selections of point pairs. For example, if $n_{pcv} = 30$ and $n_{fcv} = 9$ (which is a rather small example), the number of possibilities is $n_{pcv} C_{n_{fcv}} = n_{pcv}! / n_{fcv}!(n_{pcv} - n_{fcv})!$, $n_{pcv} C_{n_{fcv}} \approx 14 \times 10^6$. Thus, it can be difficult to test all the possibilities. In this work, we used an exchange algorithm similar to that proposed in [27] to make a selection. We worked with a random selection subset \mathbf{x}_{fcv} . We calculated the Morris–Mitchell criterion. The first point \mathbf{x}_{fcv}^1 was exchanged with each of the remaining points in $\mathbf{x}_{pcv} / \mathbf{x}_{fcv}$ and we retained the case which gave the minimum Morris–Mitchell criterion. The process was repeated for each remaining point. With this exchange algorithm, the number of possibilities tested is $n_{fcv} * (n_{pcv} - n_{fcv}) + 1$, which, in our example, equals 190 (Fig. 11).

4.2 The correlation coefficient

In this study, for the stopping criterion of our updating strategies described in Figs. 1, 2 and 3, we needed a correlation coefficient which takes into account both the amplitude and the shape of the metamodels. We used the concordance correlation coefficient, defined by [42], in the case of two observers for one experiment:

$$r_{ccc} = \frac{2 \cdot S_{\hat{F}_{app}} \hat{F}_{ref}}{S_{\hat{F}_{app}}^2 + S_{\hat{F}_{ref}}^2 + (\bar{\hat{F}}_{app} - \bar{\hat{F}}_{ref})^2} \tag{68}$$

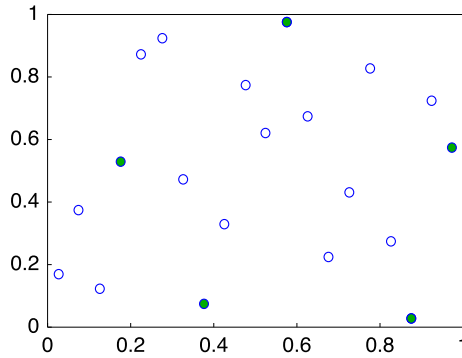


Fig. 11 Illustration of an LHS and the selection of some points

with

$$\begin{aligned} \bar{F} &= \frac{1}{N_s} \sum_{n=1}^{N_s} \hat{F}, & S_{\bar{F}}^2 &= \frac{1}{N_s} \sum_{n=1}^{N_s} (\hat{F} - \bar{F})^2 \\ S_{\hat{F}_{app}\hat{F}_{ref}} &= \frac{1}{N_s} \sum_{n=1}^{N_s} (\hat{F}_{app} - \bar{F}_{app}) (\hat{F}_{ref} - \bar{F}_{ref}) \end{aligned} \tag{69}$$

where N_s is the number of points to be correlated, which is the number of points used to compute the reference metamodel \hat{F}_{ref} . These points are taken from an independent set and are numerous. \hat{F}_{app} are approximate values of the objective function taken from the metamodels at the locations of \hat{F}_{ref} .

We are also interested in the correlation taking into account only the shape, but not the amplitude. Indeed, the shape is the dominant parameter when it comes to locating the minimum of the function of interest. We used the coefficient defined by [43] and also used in [3, 12, 44]:

$$r^2 = \left(\frac{N_s \sum \hat{F}_{ref} \hat{F}_{app} - \sum \hat{F}_{ref} \sum \hat{F}_{app}}{\sqrt{\left[N_s \sum \hat{F}_{ref}^2 - \left(\sum \hat{F}_{ref} \right)^2 \right] \left[N_s \sum \hat{F}_{app}^2 - \left(\sum \hat{F}_{app} \right)^2 \right]}} \right)^2 \tag{70}$$

One can note that, in our case, the correlation coefficient is exactly the same as that defined by Bravais-Paerson.

4.3 The stopping criterion

The updating strategy was illustrated in Figs. 1, 2 and 3. In this work, we use stopping criteria, one of which consists in calculating the correlation of the μ previously constructed metamodels. First, let us define the criterion proposed by [12, 44], which measures the evolution of the construction of the metamodel:

$$r_{ccc\mu} = \frac{2 \cdot S_{\hat{F}_{app-\mu}\hat{F}_{app}}}{S_{\hat{F}_{app-\mu}}^2 + S_{\hat{F}_{app}}^2 + (\bar{F}_{app-\mu} - \bar{F}_{app})^2} \tag{71}$$

Table 1 The bounds of the three design parameters

Parameters	Min	Max
$Cf_{Str2/Str3}$	0.0	0.95
$Cf_{Str2/Str4}$	0.0	0.95
J2 (in mm)	-0.028	0.048

where \hat{F}_{app} denotes the latest calculated metamodel and $\hat{F}_{app-\mu}$ denotes the μ^{th} previously calculated metamodel. As stated in [12, 44], “this is similar to a leave-one-out cross correlation, but in this case it is an add-one-in validation”.

Then, let us define the criterion:

$$\bar{r}_{ccc}^v = \frac{1}{v} \sum_{\mu=1}^v r_{ccc\mu} \tag{72}$$

This criterion, with $v = 4$, was used as the stopping criterion. In order to guarantee good results, the updating strategy was stopped once the condition $\bar{r}_{ccc}^v > 0.99$ had been satisfied three consecutive times. The advantage of this criterion is that it is based solely on known metamodels. Thus, it can be used even in the absence of a reference model.

5 The problems studied

In this section, we present comparisons based on three different mechanical problems. We used the five methods described above to construct a valid metamodel. To compare the methods for each treated example, a reference model is computed. The results given under the form of correlation is the correlation between the reference model and the VFM metamodel computed. The first method consists in using a correction function defined through an additive bridge function and the evofusion algorithm. The next three methods are based on the cokriging principle using three different cross correlation techniques. Let us recall that the first strategy assumes that the covariances of the fully and partially converged data are identical; the second strategy is more general and is based on the ratio of the covariances of the fully and partially converged data; the last strategy is based on an auto-regressive model between the fully and partially converged data. The last method used is based on hierarchical kriging. To search the maximum of the likelihood function for our mechanical examples described in Sect. 5, a genetic algorithm is used. Two stopping criteria were considered. The first criterion consists in stopping the procedure once the correlation with respect to some reference model exceeds 95 %. The second criterion is the one described in Sect. 4.3, which does not require a reference.

5.1 The first problem

The first problem (Fig. 12) is a three-parameter problem (Table 1) which consists in finding the reaction load $F(\mathbf{x})$ of a cube (Str2) against a wall.

This is a typical problem involving contact with friction. We constructed a first metamodel using partially converged points. We carried out the analyses for $10 \times n$ initial partially converged points (with $n = 3$ in this example) and (as in [3]) for several error levels and 20 different draws. Thus, for the case being studied, we first calculated 30 partially converged points.

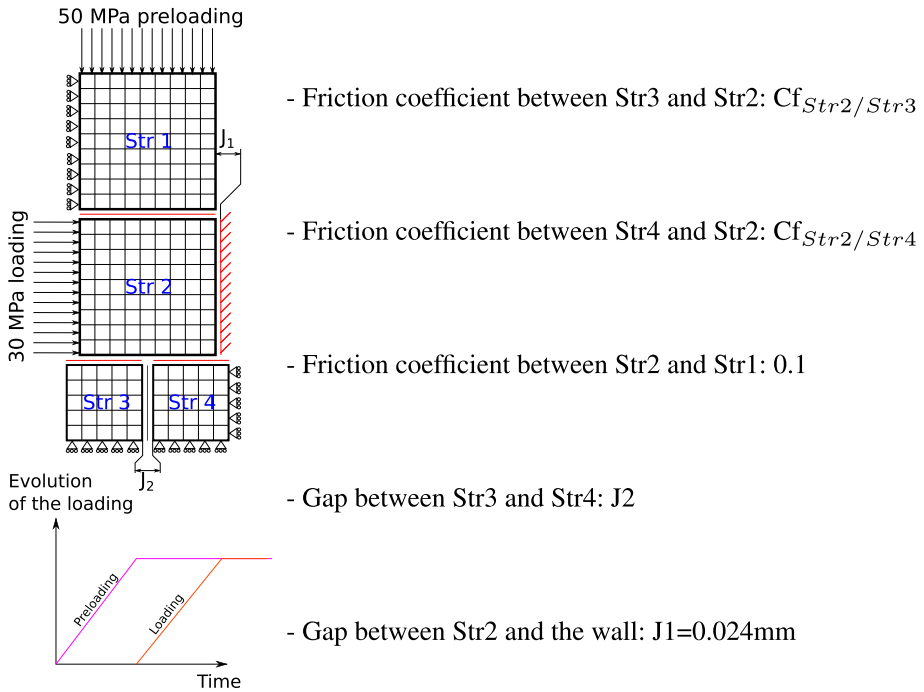


Fig. 12 The reference problem

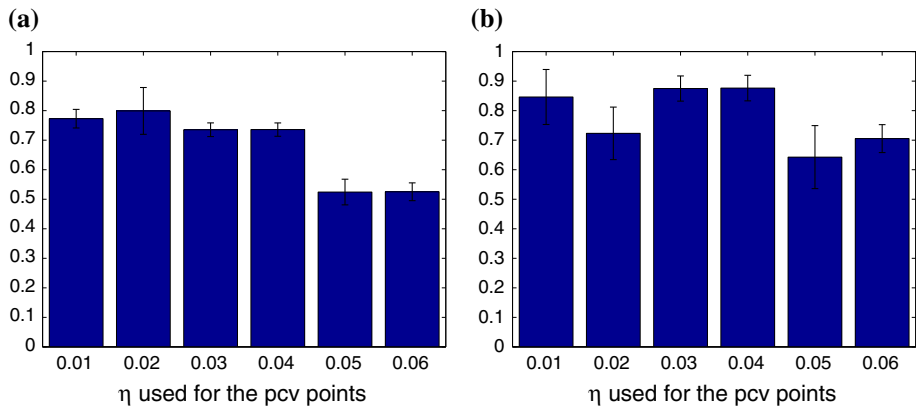


Fig. 13 Initial correlations for the first case test. **a** Initial amplitude correlation, **b** initial shape correlation

In order to prove that all these methods can be efficient, one must show that the first meta-model (built from only partially converged data) has a low correlation (or a lower correlation than what would be obtained using any of these strategies). As a reminder, η is the LATIN error indicator.

Figure 13a and b show that the initial amplitude correlation and the initial shape correlation have the same order of magnitude.

For each cokriging method and for the hierarchical method, we calculated $2 \times n$, $3 \times n$, $4 \times n$ and $5 \times n$ fully converged points. In this section, we used the exchange algorithm to select

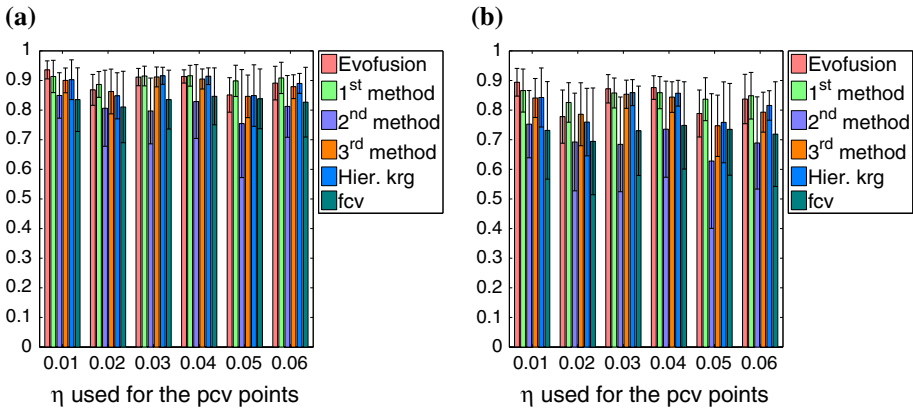


Fig. 14 Correlations with 9 additional fcv points for the first case test. **a** Correlation, **b** shape correlation

the fully converged points. For the evofusion method we calculated $2 \times n$, $3 \times n$, $4 \times n$ and $5 \times n$ additional partially and fully converged data. Thus, the computation time was slightly greater than for the other methods, but one can ignore this small difference. The first study is based on the comparison of the correlations using these fully converged points. The use of $3 \times n$ fully converged points leads to a good result in term of accuracy and computation time. In order to show the interest of each of these methods, we also compare the correlations obtained with only fully converged points. The notations used are the following:

- Evofusion: as described in Fig. 1
- 1st method: as described in Fig. 2 and in Sect. 3.2.2
- 2nd method: as described in Fig. 2 and in Sect. 3.2.3
- 3rd method: as described in Fig. 2 and in Sect. 3.2.4
- Hier. krg: as described in Fig. 3 and and in Sect. 3.3
- fcv: refers to the case where only fully converged data are used to create a metamodel.

Figure 14 shows that the most promising methods are: evofusion, the first cokriging method, the third cokriging method and the hierarchical kriging method. The use of the fully converged points alone leads to a clearly inferior correlation, especially in term of shape. Therefore, the methods which include partially converged data are definitely the best methods.

In order to show the advantage of using fully converged points in addition to the partially converged points, let us examine the cases where $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ and $\mathbf{x}_{fcv} \not\subset \mathbf{x}_{pcv}$. As before, this comparison is based on $3 \times n$ fully converged points (Fig. 15)

Even though the metamodels which include a selection of fully converged points among the partially converged points performed slightly better, one can note that, except for the second cokriging method, the correlations are almost identical. In the case of this example, the second cokriging method seems to be unsuitable for our metamodel construction.

Another study which we carried out with this mechanical example consisted in assessing whether these methods are efficient when it comes to achieving 95% correlation. We found the best method to be the first cokriging method.

To complete the study, we also included in the comparison the use of only fully converged points from the updating strategy of the first cokriging method (Fig. 16)

This example shows that if one considers only the fully converged points obtained from the updating strategy of the first cokriging method, this stopping criterion can be satisfied

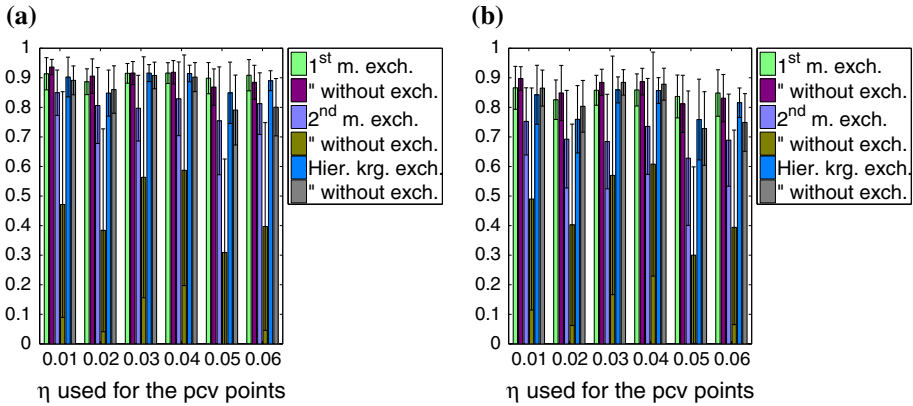


Fig. 15 Comparison of the correlations with 9 additional fcv points for the first case test. **a** Correlation, **b** shape correlation

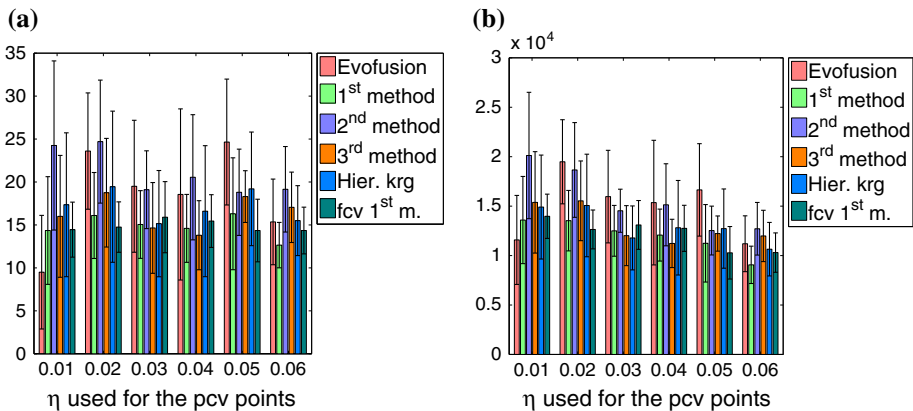


Fig. 16 The number of points and the time necessary to reach 95 % correlation for the first case test. **a** Number of fcv points to reach 95 % correlation, **b** time to reach 95 % correlation (in hundredths of a second)

slightly more rapidly than with the other strategies (especially for the error level of 0.06 which is the most attractive). This can be explained by the fact that the response surface consists of planes.

The last study we carried out with this mechanical example consisted in testing the stopping criterion presented in Sect. 4.3. One can clearly see, on the Fig. 17, that the stopping criterion tested is well suited for the construction of a valid metamodel. Moreover, using the first cokriging method, this metamodel can be obtained rapidly.

One can observe that if one takes into account only the fully converged point issued from the first cokriging method the criterion is satisfied slightly more rapidly, which pinpoints the quality of the chosen points and explains the efficiency of the first cokriging method. However, the evofusion method and the hierarchical kriging method are efficient, too.

One should note that for the first cokriging method and for hierarchical kriging (Fig. 18) the fact that $\mathbf{x}_{fcv} \not\subset \mathbf{x}_{pcv}$ or $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ has no serious impact on the number of fully converged calculated points. Concerning the second cokriging method, a smaller number of points was used, resulting in a poorer-quality solution.

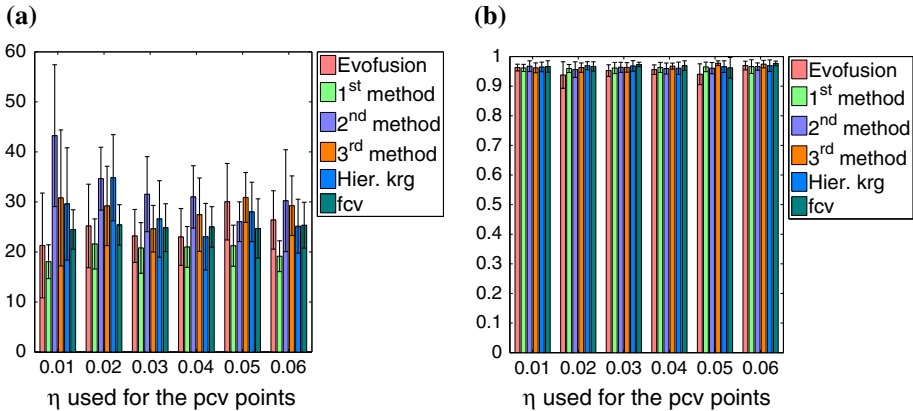


Fig. 17 The study carried out with the second stopping criterion for the first case test. **a** Number of fcv points to reach the stopping criterion, **b** final correlation

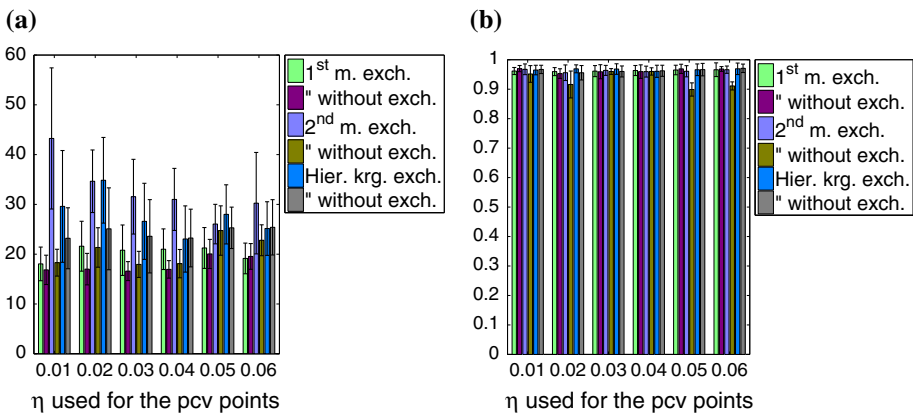


Fig. 18 Comparison of the methods with and without the exchange algorithm for the first case test. **a** Number of fcv points to reach the second stopping criterion, **b** final correlation

Conclusion Our study shows that, in order to create a valid metamodel rapidly, evofusion, the first cokriging method, the third cokriging method and the hierarchical kriging method work equally well. All these strategies, especially evofusion and the first cokriging method, are totally suitable for the purpose of achieving a high final correlation at a small number of points. Moreover, the proposed stopping criterion described in Sect. 4.3 appears to be a good criterion which enables a high correlation level to be achieved in the absence of a reference.

5.2 The second problem

The second test case concerns a more industrial problem. A shrink disk (Fig. 19) is a technological component which consists of a biconical inner ring which is fitted to the pinion and two external conical pressure flanges, one of which is threaded. A clamping load is applied between the external flanges through a series of screws distributed along the circumference. The tightening of the screws presses the conical surfaces against each other and generates radial forces which create the adhesion binding which is necessary to transmit the torque M_a

Table 2 The bounds of the design parameters

Parameters	Min	Max
$Cf_{shaft/gear}, Cf_{gear/inner}, Cf_{inner/outer}$	0.05	0.5
$J_{shaft/gear} (\mu m)$	0	50
$J_{gear/inner} (\mu m)$	36	106

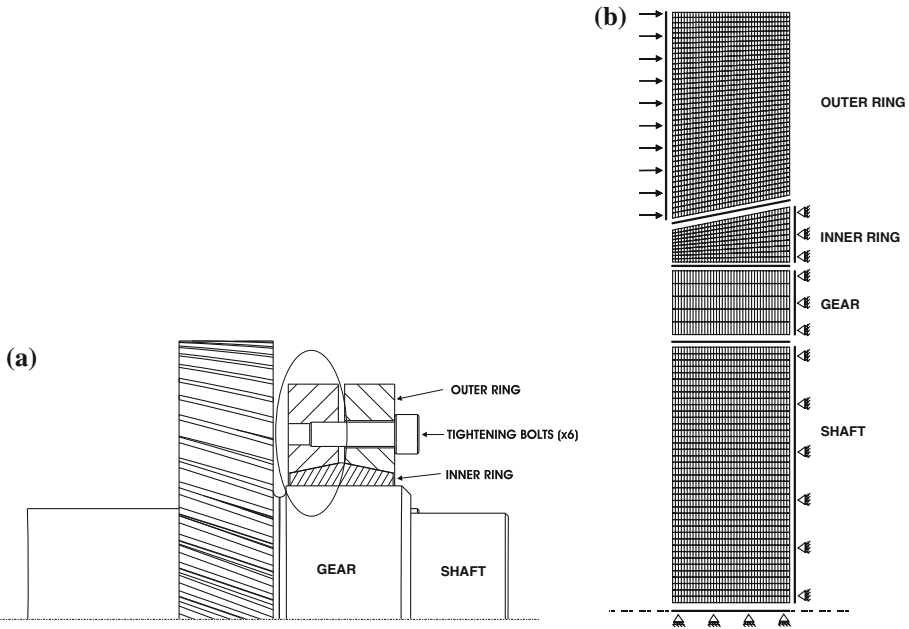


Fig. 19 The shrink disk problem. **a** A shrink disk, **b** the axisymmetric mesh

and the axial load F_a from the shaft to the pinion. The quantity of interest which is to be studied is the torque M_a .

This problem has 5 parameters as shown in Table 2. In this case, our objective is to check whether the previously described strategies are suitable for this mechanical example. Again, we used $10 \times n$ (in this case, 50) initial partially converged points and $3 \times n$ (in this case, 15) fully converged points. Only the results obtained using the exchange algorithm are reported here. First, Fig. 20 shows the initial correlations. Timewise, the most interesting correlation level is 0.05, but this level is very low.

Then, as in Sect. 5.1, we look at the correlation after the introduction of $3 \times n$ additional, fully converged points to the model.

One can observe (Fig. 21) that none of these methods is more efficient in achieving a good correlation than the simple use of only fully converged points (15 in this case). The eufusion strategy even leads to the worst results in terms of the level of accuracy of the partially converged points at 0.05. This is due to the poor initial correlation for both the amplitude and the shape. Conversely, one can note in Fig. 22 that if our updating strategy is stopped at 95% correlation all the strategies are suitable except for the third cokriging method, which is the least performing method.

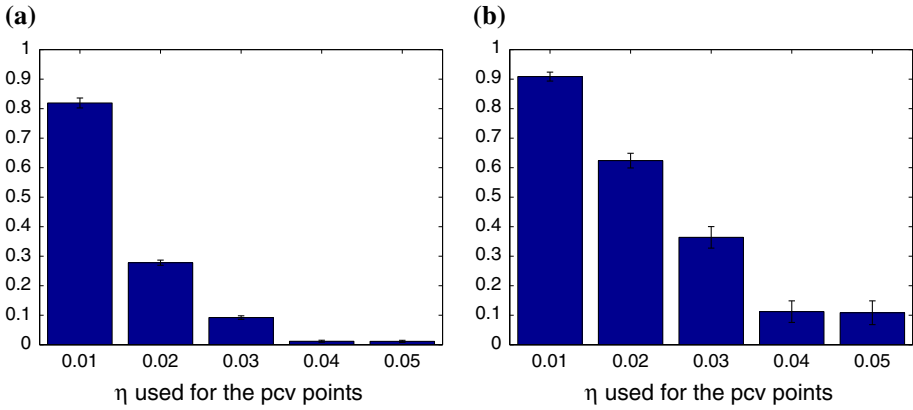


Fig. 20 Initial correlations for the second test case. **a** Initial amplitude correlation, **b** initial shape correlation

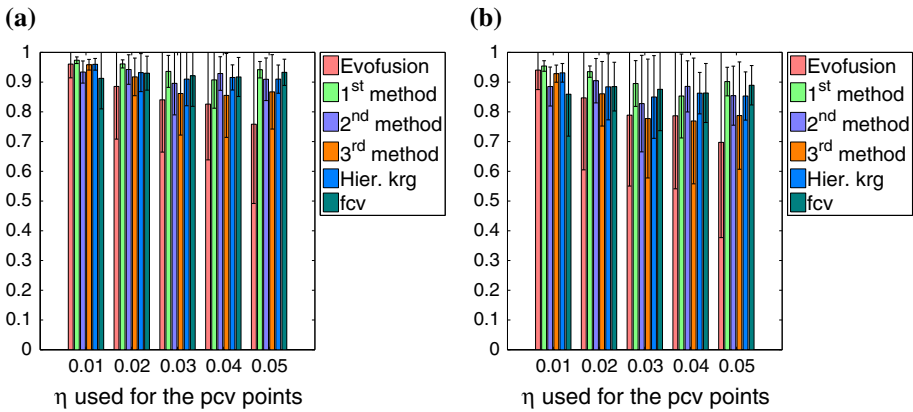


Fig. 21 The correlations with 15 additional fcv points for the second test case. **a** Correlation, **b** shape correlation

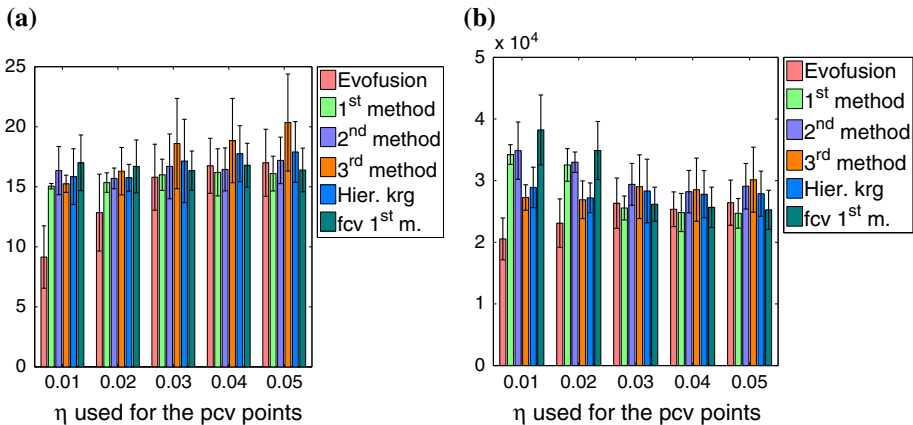


Fig. 22 The number of points and the time necessary to reach 95% correlation for the second test case. **a** Number of fcv points to reach 95% correlation, **b** the time to reach 95% correlation (in hundredths of second)

Table 3 The bounds of the design parameters

Parameters	Min	Max
X (in mm)	35	208
L (in mm)	1	30

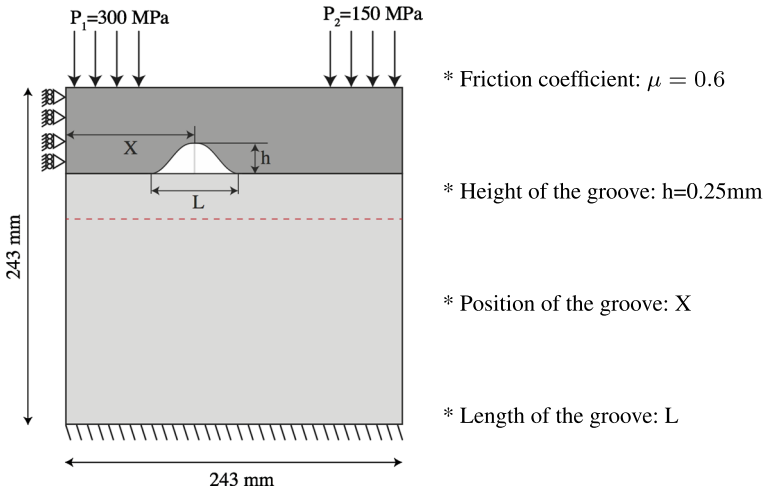


Fig. 23 The reference problem for the third test case

Conclusion As in the previous example, the first cokriging method, the hierarchical kriging method and evofusion are the most efficient strategies. In both mechanical examples, the second and third cokriging methods did not appear to work well. One can consider that the use of a larger error indicator is appropriate, even though, in the case of evofusion, an error indicator equal to 0.01 leads to an approximate 15 % saving in computation time compared to an error indicator equal to 0.05. Figure 21 shows that Evofusion is the worst method, especially for $\eta = 0.05, 0.04$, even if on Fig. 22, the results are good. Two reasons can explain this situation. The first one is that for the evofusion, points are added one by one up to have 15 fully converged points, and it sometimes happens that one reaches the correlation of 0.95 before having computed 15 points. The second reason is that when the correlation with the reference model is computed (so, when 15 fully converged points are available), the process of evofusion is not very stable, because the 15 fully converged points are often on the edge of the space domain (due the enrichment based on the maximum of mean square error) and can be not representative of the design space (dimension 5 in this example). So in this context the genetic algorithm can miss the maximum likelihood. On our 20 design of experiments, 3 of them have this problem, if the results are computed without these one, the results on the Fig. 21 are comparable with the first cokriging method.

5.3 The third problem

In order to further test the efficiency of these three methods, we studied a third example. This time, we considered a two-dimensional, multimodal problem with two parameters (Fig. 23; Table 3).

The objective function is the mean pressure along the dotted line in Fig. 23. The response surface is shown in Fig. 24.

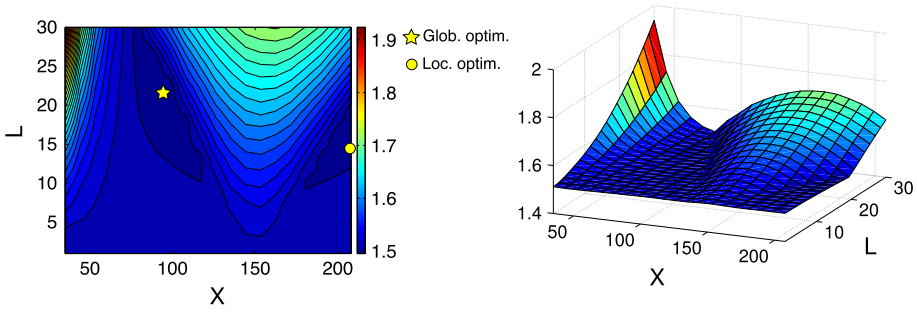


Fig. 24 The response surface of the groove problem

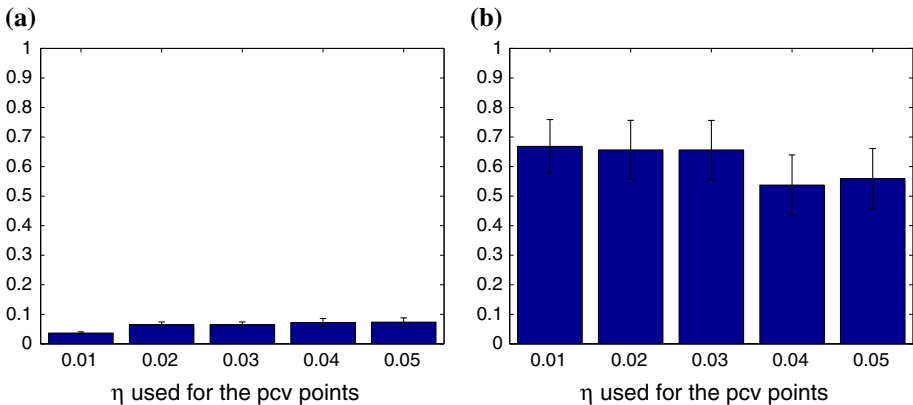


Fig. 25 Initial correlations for the third case test. **a** Initial amplitude correlation, **b** initial shape correlation

We can note that the response surface is multimodal and, therefore, more complex than the simple superposition of several planes as was the case of the other examples. With this problem, the initial correlations show that only the shape correlation is good (see Fig. 25), but, globally, there is an offset between the high-fidelity and low-fidelity levels.

As before, after having calculated $3 \times n$ fully converged points, we calculated the correlation and the shape correlation (Fig. 26).

For this mechanical example, the VFM methods were found to be very efficient in reaching the first stopping criterion (*i.e.* 95 % correlation) as we can see on Fig. 27.

It is interesting to compare the cases $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$ and $\mathbf{x}_{fcv} \not\subset \mathbf{x}_{pcv}$ in order to assess the impact of this property. Figure 28 shows that it is better to have $\mathbf{x}_{fcv} \subset \mathbf{x}_{pcv}$.

Finally, we tested our second stopping criterion defined in 4.3. The results are very interesting (Fig. 29) because evofusion and the first cokriging method satisfied this stopping criterion with about 20 fully converged points for a final 97% correlation, while with 20 fully converged Latin Hypercube points the correlation was only about 86%! Moreover, in this case, the multiparametric strategy is attractive because it reduces the computation time by a factor of about 2.5. This example clearly shows that, using the same number of points, the VFM techniques lead to a better correlation, even when $\mathbf{x}_{fcv} \not\subset \mathbf{x}_{pcv}$ (see Fig. 30).

Moreover the area of the global minimum is found for 16 different initial draws on 20 (global minimum: $X = 89.9$ mm $L = 22.9$ mm for a value of 1.49). The 4 other draws leads to the local minimum (local minimum: $X = 207$ mm $L = 13.1$ mm for a value of 1.50).

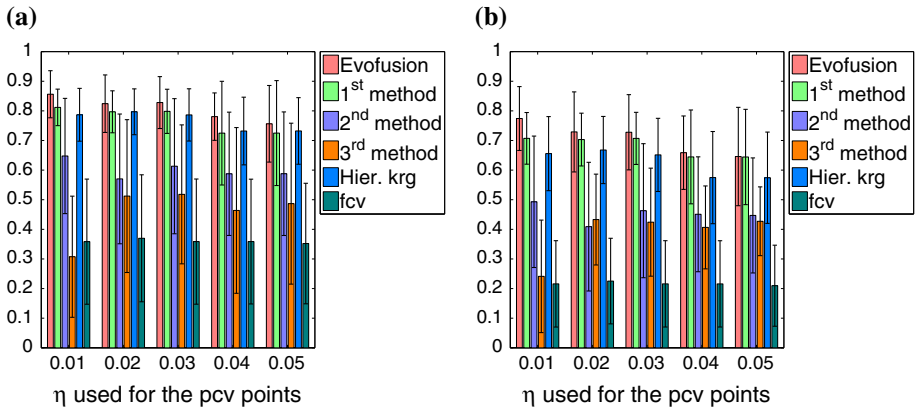


Fig. 26 The correlations with 6 additional fcv points for the third case test. **a** Correlation, **b** shape correlation

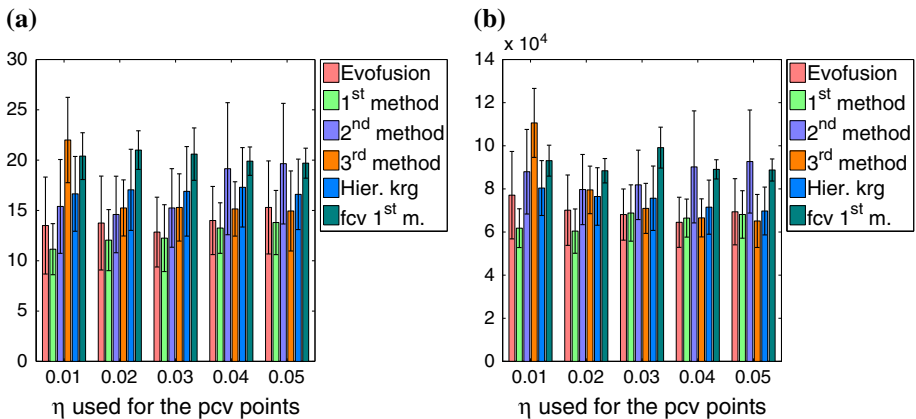


Fig. 27 The number of points and the time necessary to reach 95% correlation for the third case test. **a** Number of fcv points to reach 95% correlation, **b** time to reach 95% correlation (in hundredths of a second)

So these strategies enable to find the global minimum of a problem even if the local and global minimum are close in term of value. For the two other examples treated previously, the minimum or the maximum of the function have been found.

6 Optimization with VFM methods

We can notice that three methods are clearly promising, efficient and robust on our mechanical examples, these methods are Evofusion, 1st cokriging method and hierarchical kriging. We purpose in this paragraph to show the interest to use the variable-fidelity methods in the context of optimization. To illustrate this point, an EGO approach [14] is performed on the test case 3 defined in paragraph 5.3. This mechanical example get two minima (one global and one local) which have almost the same value (see Fig. 24). In this section, the comparison

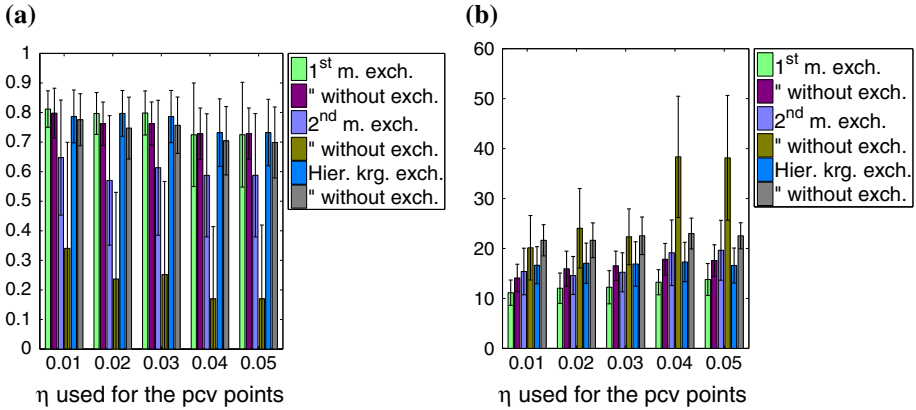


Fig. 28 Comparison of the methods with and without the exchange algorithm for the third case test. **a** Correlation after the introduction of 6 additional fcv converged points, **b** comparison of the number of additional fcv points to reach 95 % correlation

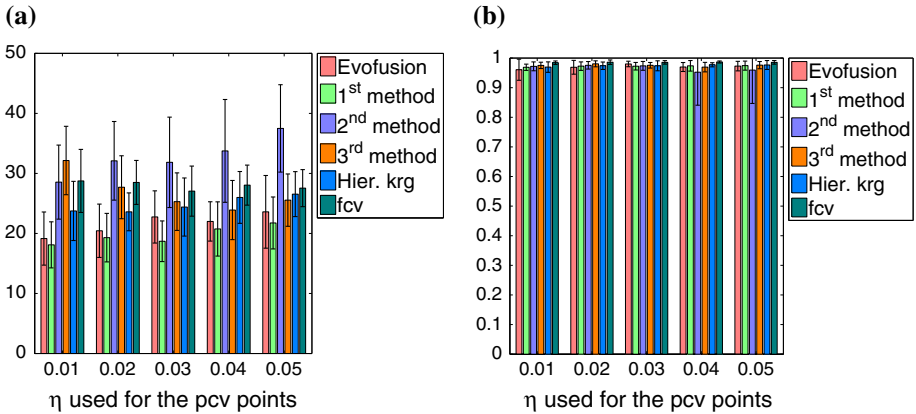


Fig. 29 The study with the stopping criterion for the third case test. **a** Number of fcv points to satisfy the stopping criterion, **b** final correlation

of the position of the minima found thanks to the EGO approach is given as well as the convergence history of the optimization process.

The previous cited methods are used to construct a metamodel to lead the optimization. As previously 20 different sampling sets are used for each multi-fidelity method. For the EGO approach, the same schemes of construction of the metamodel as previously are kept, the only made change is the enrichment of the points. To do this enrichment, the maximum of expected improvement [14] is the method used.

6.1 Expected improvement

The goal of the expected improvement criterion is to locate the area where an improvement relative to the best computed optimum can be consider. To have more details about this famous technique please see [14].

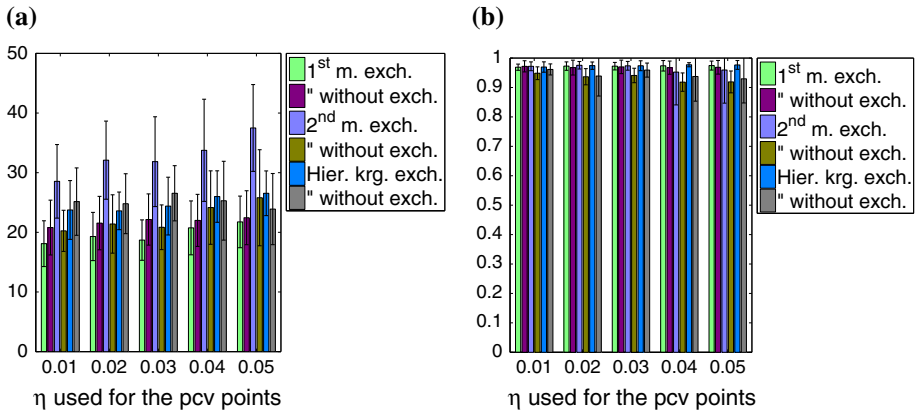


Fig. 30 The study with the second stopping criterion and the influence of the exchange algorithm for the third case test. **a** Number of fcv points to satisfy the stopping criterion, **b** final correlation

We note $s(\mathbf{x})$ the value of the mean square error at the point \mathbf{x} . The expected improvement can be expressed in a kriging context as:

$$EI(\mathbf{x}) = \begin{cases} (\min(Y) - \hat{y}(\mathbf{x}))\Phi\left(\frac{\min(Y) - \hat{y}(\mathbf{x})}{s(\mathbf{x})}\right) + s(\mathbf{x})\phi\left(\frac{\min(Y) - \hat{y}(\mathbf{x})}{s(\mathbf{x})}\right) & \text{if } s > 0 \\ \{0\} & \text{if } s = 0 \end{cases} \quad (73)$$

In the cokriging and hierarchical kriging context as in [27] the expected improvement is expressed as:

$$EI(\mathbf{x}) = \begin{cases} (\min(Y_{fcv}) - \hat{y}_{fcv}(\mathbf{x}))\Phi\left(\frac{\min(Y_{fcv}) - \hat{y}_{fcv}(\mathbf{x})}{s(\mathbf{x})}\right) + s(\mathbf{x})\phi\left(\frac{\min(Y_{fcv}) - \hat{y}_{fcv}(\mathbf{x})}{s(\mathbf{x})}\right) & \text{if } s > 0 \\ \{0\} & \text{if } s = 0 \end{cases} \quad (74)$$

Another question is how the expected improvement of the metamodel constructed by Evofusion can be expressed. Indeed, to obtain the high-fidelity metamodel, a first low-fidelity metamodel is constructed and corrected by an error metamodel (see Fig. 1), so it is not simple to determine the mean square error of the high-fidelity metamodel. In this work we have taken the same option as in [12,44]: all the points used to create the high-fidelity metamodel are considered with a null error. Moreover all the added points are enriched with the maximum expected improvement, so contrary at the other methods there is no fully converged points when starting the enrichment. The main reason of this decision is that we wanted to compare proposed methods in the literature. These three methods are compared to a more traditional one based on kriging of fully converged points. First, for this approach, 20 LHS points are computed and then enriched thanks to maximum of expected improvement.

For all the methods we decide to stop the enrichment when three times in a row the value $\frac{\max EI}{\max Y_{fcv} - \min Y_{pcv}}$ is inferior to 0.001 see [45].

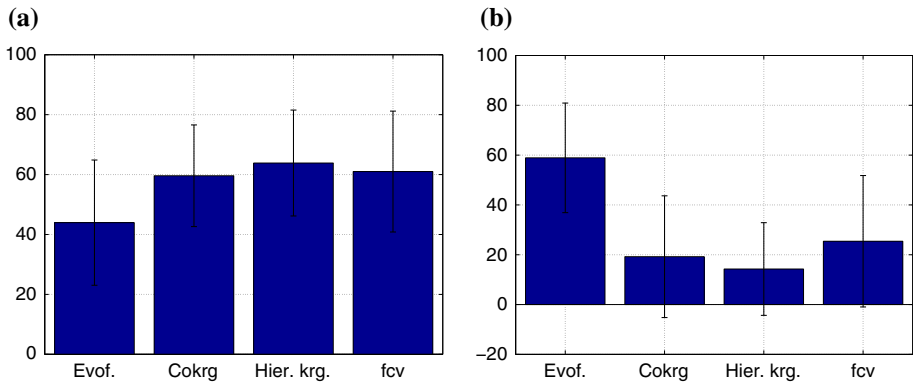


Fig. 31 Result of the optimization. **a** Number of points to reach the stopping criterion, **b** quality of the final result, distance in (%) to the global minimum

6.2 Comparison of the VFM methods

The results presented in Fig. 31 show clearly that for a same computation effort the accuracy of the position of the minimum is better with cokriging and hierarchical kriging than with kriging of fully converged data. The distance in %, is defined as:

$$\frac{\| \mathbf{x}_{\min}^{EGO} - \mathbf{x}_{\min} \|}{\max_{\mathbf{x} \in \mathcal{D}} \{ \| \mathbf{x} - \mathbf{x}_{\min} \| \}} \times 100$$

with \mathbf{x}_{\min} the global minimum and \mathbf{x}_{\min}^{EGO} the minimum provided by the metamodel. One can note that the computation time of partially converged data can be neglected, because the computation time of one partially converged data is 30 times less expensive than one fully converged data. We can notice that on the twenty tests performed, the Evofusion results are not good. Several solutions can be considered to improve the results.

- A first enrichment of a few points based on the maximum mean square error can be done to improvement the quality of the response surface. After, the enrichment based on expected improvement can be computed. (The results obtained are better for the same number of computed points.)
- The second solution is to modify the evaluation of the mean square error and used the one proposed by [26] as for example.

6.3 Convergence of the results

Since the local and the global minima have almost the same value it is interesting to see which method is the best one at fixed computation time. The results presented on Fig. 32 show the speed convergence of the found minimum thanks to the metamodels. These results show the mean of the minimum found for the 20 numerical experiments run for each method.

We can notice that all the multi-fidelity strategies are better than the classical strategy based only on the fully converged data. It is worth noting that the localisation of the global minimum is not reached with the Evofusion method. Nevertheless, as the local minima is very close from the global one, the final result obtained is still very useful. For information, the convergence history as a function of the distance to the global minimum is given in

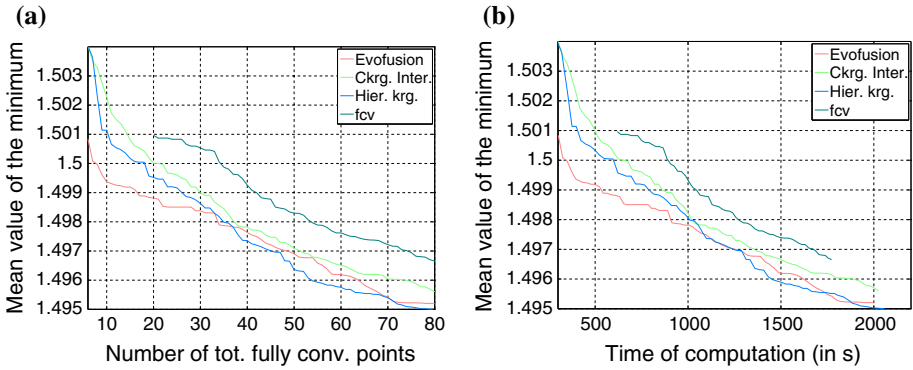


Fig. 32 Convergence history of the found minimum. **a** Mean value of minimum for the 20 numerical experiments as a function of the number of fully converged points computed, **b** mean value of minimum for the 20 numerical experiments as a function of the CPU time

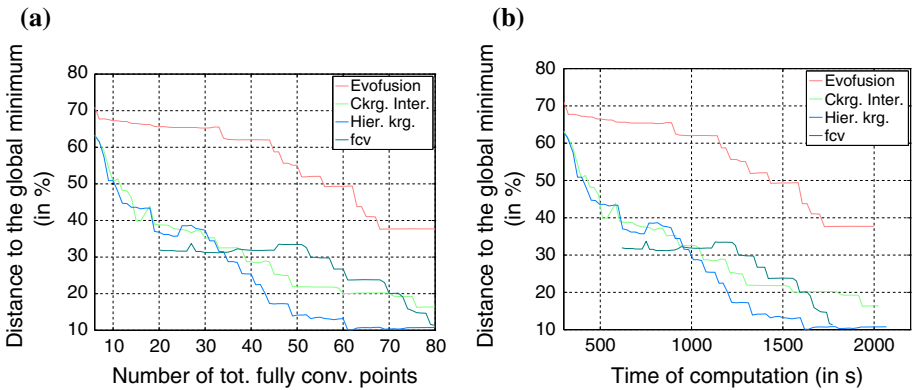


Fig. 33 Convergence history of the found minimum, **a** Mean distance to the global minimum as a function of the number of fully converged points, **b** mean distance to the global minimum as a function of CPU time

Fig. 33 and shows again that the Evofusion strategy is less efficient to locate the real global minimum.

7 Conclusion

In this paper, we tested several variable-fidelity methods in an attempt to create a valid metamodel which would be appropriate for optimization purposes. Three main types of strategies were compared: the additive bridge function, cokriging and hierarchical kriging. Out of the five methods tested, these three were found to be very efficient for creating a metamodel:

- the additive bridge function
- cokriging using the first cross correlation method
- hierarchical kriging

The results are slightly different that the ones found in the literature, but this can be easily explained. The first reason is that the correlation between low-fidelity model and high-

fidelity model are low and even very low in some cases. The second reason is that the kriging metamodel is constructed with a low number of fully converged points ($3 \times \text{dim}$), so that it has a low level of correlation with the reference solution. So in our engineering context, the previous methods cited are the most robust: other methods are less accurate (even though they may have certain advantages in other contexts). So tuning, the parameter scaling the low-fidelity data for the second and third cokriging method can be difficult. The same observation was performed in [22] which shows that the first cokriging method is very efficient in real-life applications.

It is important to note that in our examples all three main families of methods turned out to be efficient. We were able to create a very accurate metamodel using only a few fully converged points ($3 \times n$). These methods are more efficient than using only fully converged data when it comes to reaching 95% correlation. Moreover, one should note that with the stopping criterion based on the constructed metamodel alone as defined in Sect. 4.3, a very high final correlation was obtained with fewer points than the usual rule which says that a metamodel should be constructed with $10 \times n$ fully converged points. If, as in the third example, the number of points used corresponds to that rule, the correlation is even higher and, therefore, the resulting metamodel is better.

In conclusion, these three techniques were found to be efficient for the construction of a valid metamodel, which can be enriched through a variety of techniques. Our review of several types of techniques shows that they can be used efficiently along with our mechanical model. One can note that all the techniques developed in this paper can take into account some gradient information with which the multiparametric strategy can be applied rapidly. Once the metamodel has been obtained, one can seek the global optimum using an enrichment method such as the “expected improvement” method since it has been shown (especially for the third example) that these strategies enable to find the area of this one. These methods also lend themselves well to the introduction of constraints or the resolution of multiobjective optimization problems.

References

1. Jones, D.R.: A taxonomy of global optimization methods based on response surface. *J. Glob. Optim.* **21**, 345–383 (2001)
2. Robinson, T.D., Eldred, M.R., Willcox, K.E., Haimes, R.: Surrogate-based optimization using multifidelity models with variable parametrization and corrected space mapping. *AIAA J.* **46**, 2814–2822 (2008)
3. Courrier, N., Boucard, P.-A., Soulier, B.: The use of partially converged simulations in building surrogate models. *Adv. Eng. Softw.* **67**, 186–197 (2014)
4. Chinesta, F., Ammar, A., Cueto, E.: Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Arch. Comput. Methods Eng.* **17**, 327–350 (2010)
5. Chinesta, F., Ladeveze, P., Cueto, E.: A short review on model order reduction based on proper generalized decomposition. *Arch. Comput. Methods Eng.* **18**(4), 395–404 (2011)
6. Lamari, H., Chinesta, F., Ammar, A., Cueto, E.: Nonconventional numerical strategies in the advanced simulation of materials and processes. *Int. J. Mod. Manuf. Technol.* **1**, 49–56 (2009)
7. Pruliere, E., Chinesta, F., Ammar, A.: On the deterministic solution of multidimensional parametric models using the proper generalized decomposition. *Math. Comput. Simul.* **81**, 791–810 (2010)
8. Heyberger, C., Boucard, P.-A., Neron, D.: Multiparametric analysis within the proper generalized decomposition framework. *Comput. Mech.* **49**, 277–289 (2012)
9. Gano, S.E., Renaud, J.E., Martin, J.D., Simpson, T.W.: Update strategies for kriging models used in variable fidelity optimization. *Struct. Multidiscipl. Optim.* **32**(4), 287–298 (2006)
10. Vitali, R., Haftka, R.T., Sankar, B.V.: Multi-fidelity design of stiffened composite panel with a crack. *Struct. Multidiscipl. Optim.* **23**(5), 347–356 (2002)

11. Knill, D.L., Giunta, A.A., Baker, C.A., Grossman, B., Mason, W.H., Haftka, R.T., Watson, L.T.: Response surface model combining linear and euler aerodynamics for super sonic transport design. *J. Aircr.* **36**(1), 75–86 (1999)
12. Forrester, A.I.J., Bressloff, N.W., Keane, A.J.: Optimization using surrogate models and partially converged computational fluid dynamics simulations. *Proc. R. Soc. A* **462**, 2177–2204 (2006)
13. Dadone, A., Grossman, B.: Progressive optimization of inverse fluid dynamic design problems. *Comput. Fluids* **29**, 1–32 (2000)
14. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. *J. Glob. Optim.* **13**, 455–492 (1998)
15. Krige, D.G.: A statistical approach to some mine valuation and allied problems on the witwatersrand. Master's thesis, University of the Witwatersrand (1951)
16. Matheron, G.: Principles of geostatistics. *Econ. Geol.* **58**(8), 1246–1266 (1963)
17. Laurent, L., Boucard, P.-A., Soulier, B.: A dedicated multiparametric strategy for the fast construction of a cokriging metamodel. *Comput. Struct.* **124**, 61–73 (2013)
18. Morris, M.D., Mitchell, T.J., Ylvisaker, D.: Bayesian desing and analysis of computer experiments: use of derivatives in surface prediction. *Technometrics* **35**(3), 243–255 (1993)
19. Koehler, J., Owen, A.: Computer experiments. *Handb. Stat.* **13**, 261–308 (1996)
20. Praveen, C., DuVigneau, R.: Radial basis function and kriging metamodels for aerodynamic optimization. Technical report 6151 INRIA (2007)
21. Buhmann, M.D.: Radial basis functions. *Acta Numer.* **9**, 1–38 (2000)
22. Zimmermann, R., Han, Z.-H.: Simplified cross-correlation estimation for multi-fidelity surrogate cokriging models. *Adv. Appl. Math. Sci.* **7**(2), 181–201 (2010)
23. Han, Z.-H., Zimmermann, R., Görtz, S.: Alternative cokriging model for variable-fidelity surrogate modeling. *AIAA J.* **50**(5), 1205–1210 (2012)
24. Forrester, A.I.J., Keane, A.J.: Recent advances in surrogate-based optimization. *Prog. Aerosp. Sci.* **45**, 50–79 (2009)
25. Rosenbaum, B., Schulz, V.: Efficient response surface method based on generic surrogate models. *SIAM J. Sci. Comput.* **35**(2), B529–B550 (2013)
26. Han, Z.-H., Görtz, S., Zimmermann, R.: Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerosp. Sci. Technol.* **25**(1), 177–189 (2013)
27. Forrester, A.I.J., Sobester, A., Keane, A.J.: Multi-fidelity optimization via surrogate modelling. *Proc. R. Soc. A* **463**, 3251–3269 (2007)
28. Yamazaki, W., Mavriplis, D.J.: Derivative-enhanced variable fidelity surrogate modeling for aerodynamic functions. *AIAA J.* **51**(1), 126–137 (2013)
29. Rosenbaum, B.: Efficient global surrogate models for responses of expensive simulations. PhD thesis, Universität Trier (2013)
30. Han, Z.-H., Görtz, S.: A hierarchical kriging model for variable-fidelity surrogate modeling of aerodynamic functions. *AIAA J.* **50**(9), 1885–1896 (2012)
31. Ladevèze, Pierre: Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation. *Mechanical Engineering Series*. Springer, Berlin (1999)
32. Wachernagel, P.: Multivariate geostatistics: an introduction with applications. Springer, Berlin (2003)
33. Soulier, B., Boucard, P.-A.: A multiparametric strategy for the two step optimization of structural assemblies. *Struct. Multidiscipl. Optim.* **47**(4), 539–553 (2013)
34. Boucard, P.-A., Ladevèze, P.: A multiple solution method for non-linear structural mechanics. *Mech. Eng.* **50**(5), 317–328 (1999)
35. Soulier, B., Boucard, P.-A.: A multiparametric strategy for the large-scale multilevel optimization of structural assemblies. In: 8th World Congress on Structural and Multidisciplinary Optimization, Lisbon, Portugal (2009)
36. Sacks, J., Welch, W.J., Mitchell, T.J., Wynn, H.P.: Design and analysis of computer experiments. *Stat. Sci.* **4**, 409–423 (1989)
37. Han, Z.-H., Zimmermann, R., Görtz, S.: A new cokriging method for variable-fidelity surrogate modeling of aerodynamic data. In: 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, FL (2010). AIAA 2010-1225
38. Kennedy, M.C., O'Hagan, A.: Predicting the output from a complex computer code when fast approximations are available. *Biometrika* **87**(1), 1–13 (2000)
39. Han, Z.-H., Görtz, S., Hain, R.: A variable-fidelity modeling method for aero-loads prediction. *New Results Numer. Exp. Fluid Mech.* VII **112**, 17–25 (2010)
40. McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21**(2), 239–245 (1979)

41. Morris, M.D., Mitchell, T.J.: Exploratory designs for computational experiments. *J. Stat. Plan. Interf.* **43**(3), 381–402 (1995)
42. Lawrence, I., Lin, K.: A concordance correlation coefficient to evaluate reproducibility. *Biometrics* **45**, 255–268 (1989)
43. Edwards, A.L.: *An introduction to linear regression and correlation*. W.H.Freeman, New York (1976)
44. Forrester, A.I.J.: *Efficient global aerodynamic optimisation using expensive computational fluid dynamics simulations*. PhD thesis, University of Southampton, School of Engineering Sciences, Southampton, England (2004)
45. Huang, D., Allen, T.T., Notz, W.I., Miller, R.A.: Sequential kriging optimization using multiple-fidelity evaluations. *Struct. Multidiscipl. Optim.* **32**(5), 369–382 (2006)