



HAL
open science

Coupling multi-fidelity kriging and model-order reduction for the construction of virtual charts

Stéphane Nachar, Pierre-Alain Boucard, David Néron, Felipe Bordeu

► **To cite this version:**

Stéphane Nachar, Pierre-Alain Boucard, David Néron, Felipe Bordeu. Coupling multi-fidelity kriging and model-order reduction for the construction of virtual charts. *Computational Mechanics*, 2019, 64 (6), pp.1685-1697. 10.1007/s00466-019-01745-9 . hal-02169052

HAL Id: hal-02169052

<https://hal.science/hal-02169052>

Submitted on 3 Jul 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Coupling multi-fidelity kriging & model-order reduction for the construction of virtual charts

Stéphane Nachar · Pierre-Alain Boucard · David Néron · Felipe Bordeu

Received: date / Accepted: date

Abstract This article presents the coupling between multi-fidelity kriging and a database generated on-the-fly by model reduction to accelerate the generation of a surrogate model. The two-level multi-fidelity kriging method Evofusion is used for data fusion. The remarkable point is the generation of low-fidelity and high-fidelity observations from the same solver using the Proper Generalized Decomposition, a model-order reduction method. A $17\times$ speedup is obtained here on an elasto-viscoplastic test case.

Keywords Metamodels, Reduced-order models, Viscoplasticity, Data fusion

Introduction

Engineering simulation provides the best design products by allowing many conception options to be quickly explored and tested, reducing the need for physical prototyping. Many companies also want to generate deeper insights with each simulation, by integrating more physics variables, investigating more nonlinear materials, and evaluating more complex environmental conditions.

But fast-time-to-result requirement remains a critical factor to meet aggressive time-to-market demands and using high-fidelity direct resolution solvers is not suitable for (virtual) charts generation for engineering design and optimization. Metamodels are commonly considered to explore design options without computing the solution for the whole set.

Metamodels are here models of input-output data that are obtained from numerical simulations. Inputs are design parameters chosen in a design space and outputs are scalar quantity of interest (QoI) generated from the full spatio-temporal mechanical fields. So the computation of each value of the QoI requires several hours.

Nonlinear relationships require a large amount of data, especially when the number of input parameters increases. Finding the best design space sampling is difficult without any *a priori* information. A possibility is to use further data sources to generate a multi-fidelity surrogate model. With this aspect, many low-fidelity and low-cost calculations can be performed first to estimate the QoI function. In a second phase, local enrichment is made by doing high-fidelity computations. Low-fidelity information is also calculated to add correction on all low-fidelity data. Further developments have already made on this subject by using different methods of multi-fidelity kriging [9, 12, 39].

Another possibility is to take advantage of the redundancy of information in mechanical fields by using model reduction. Model reduction techniques constitute one of the possible tools to bypass the limited calculation budget by seeking a solution to a problem on a reduced order basis (ROB). Popular methods for model reduction are Proper Orthogonalized Decomposition (POD) [4, 38], reduced-basis methods [27, 33] or Proper Generalized Decomposition [22, 23, 5].

The purpose of this paper is to show how to pair these two approaches to quickly create a metamodel of the QoI. Note that a coupling between multi-fidelity kriging and POD reduction has already been performed for the quantification of uncertainty [21, 1] with snapshots of a high-fidelity model calculated offline.

In this paper, an online method for building a multi-fidelity metamodel is given by calculating the QoI from the PGD modes generated on-the-fly with the LATIN-PGD framework. Low-fidelity fields are obtained by stopping the solver before convergence, and high-fidelity information is obtained with converged fields. In addition, the solver ability to reuse information from previously calculated PGD modes is exploited. The demonstration of the coupling of these methods was done on elasto-viscoplastic elasticity test cases and an acceleration of $17\times$ was obtained on a test case, allowing a first highlight of performances of this approach.

Generation of virtual charts with multi-fidelity kriging is described on Section 1. Data observations are computed thanks to the LATIN-PGD solver given in Section 2 and some results obtained with this solver are in Section 3. The link between multi-fidelity kriging and model reduction can be found on Section 4 with first results.

1 Generation of virtual charts with multi-fidelity kriging

1.1 Gaussian Process Regression - Kriging

Let us consider here a spatio-temporal mechanical problem defined by some input parameters \underline{x} belonging to a design space \mathcal{D} . For each value $\underline{x} \in \mathcal{D}$, the solution of the problem allows to compute a given quantity of interest (QoI), denoted Y , which is assumed to be scalar herein. The corresponding virtual chart is the mapping $\underline{x} \in \mathcal{D} \mapsto Y(\underline{x}) \in \mathbb{R}$, but it is not affordable directly for computational cost reasons. Then, this mapping is replaced by a metamodel $\underline{x} \in \mathcal{D} \mapsto \hat{Y}(\underline{x}) \in \mathbb{R}$, where \hat{Y} is built from a set of resolutions for n selected points \underline{x}_i and the corresponding values of the QoI $Y_i = Y(\underline{x}_i)$.

Herein, the metamodel \hat{Y} is built using Gaussian process regression, also called kriging. One possible strategy to select n initial samples \underline{x}_i in the design space \mathcal{D} is to use Latin Hypercube Sampling [28].

A simple way to generate a metamodel is to consider a linear regression model \hat{Y} :

$$\hat{Y}(\underline{x}) = \underline{f}(\underline{x})^T \underline{\beta} + Z(\underline{x}) \quad (1)$$

$$\forall i, \quad Y_i = \underline{x}_i^T \underline{\beta} + Z_i \quad (2)$$

\underline{x} is the input vector, $\underline{\beta}$ the vector of weights for the linear model, \underline{f} user-defined regression function, and Z_i the error on observations. In case of noise-free observations, $Z_i = 0$.

Gaussian process regression [34] is an extension of the linear model regression by considering fitting error Z as a zero-mean Gaussian process:

$$Z(\underline{x}) \sim N(\underline{0}, \sigma^2 r) \quad (3)$$

$\sigma^2 r$ is a user-defined covariance function with σ a scalar parameter. With Equation (3), \hat{Y} is also a Gaussian process.

Let us introduce matrix \mathbf{X} , that contains the location of all the observed points. The i -th column contains the location of the i -th observed point \underline{x}_i . The vector with all the observation responses \underline{Y} is such that $\underline{Y}_i = Y_i$.

To fit observations, the Gaussian process is conditioned to them: assume \underline{Y} as a random vector defined by observations, then joint distribution between them is defined by:

$$\begin{pmatrix} \hat{Y}(\underline{x}) \\ \underline{Y} \end{pmatrix} \sim N \left(\begin{pmatrix} \underline{f}(\underline{x})^T \underline{\beta} \\ \mathbf{F}' \underline{\beta} \end{pmatrix}, \sigma^2 \begin{pmatrix} \mathbf{1} & \underline{r}'(\underline{x}) \\ \underline{r}(\underline{x}) & \mathbf{R} \end{pmatrix} \right) \quad (4)$$

where \mathbf{F} designates the regression matrix defined by $\mathbf{F}_{ij} = \underline{f}_i(\underline{x}_j)$, \mathbf{R} the correlation matrix between data observed defined by $\mathbf{R}_{ij} = r(\underline{x}_i, \underline{x}_j)$ and $\underline{r}(\underline{x})$ the correlation vector between observed and predicted data defined by $\underline{r}(\underline{x})_j = r(\underline{x}_j, \underline{x})$.

$\hat{Y}(\underline{x})$ conditionally to \underline{Y} is given by Schur complement of \mathbf{R} in correlation matrix of joint distribution.

Then, the generated model $\underline{x} \mapsto \hat{Y}(\underline{x})$ is a Gaussian process $N(m(\underline{x}), C(\underline{x}, \underline{x}'))$ conditionally to observations, with mean $m(\underline{x})$ and covariance $C(\underline{x}, \underline{x}')$:

$$\begin{cases} m(\underline{x}) &= \underline{f}(\underline{x})^T \underline{\beta} + \underline{r}(\underline{x})^T \mathbf{R}^{-1} (\underline{Y} - \mathbf{F} \underline{\beta}) \\ C(\underline{x}, \underline{x}') &= \sigma^2 (1 - \underline{r}(\underline{x})^T \mathbf{R}^{-1} \underline{r}(\underline{x}')) \end{cases} \quad (5)$$

The Gaussian definition (5) is for Simple Kriging when $\underline{\beta}$ is assumed to be known. Usually, the case of Universal Kriging is considered when $\underline{\beta}$ is supposed unknown. To do that, \hat{Y} is considered as the Best Linear Unbiased Predictor [36], that is:

- Based on linear combination of observations

$$\exists \underline{w} \in \mathbb{R}^n, \hat{Y}(\underline{x}) = \underline{w}^T(\underline{x}) \underline{Y} \quad (6)$$

- Unbiased (with \mathbb{E} the mean of a random variable)

$$\mathbb{E}(\hat{Y}(\underline{x}) - Y(\underline{x})) = 0 \quad (7)$$

$$\text{That gives: } \mathbf{F}^T \underline{w}(\underline{x}) - Y(\underline{x}) = \underline{0}$$

- Which minimizes mean square error

$$\min \mathbb{E} \left[\left(\hat{Y}(\underline{x}) - Y(\underline{x}) \right)^2 \right] = \sigma^2 (1 + \underline{w}^T \mathbf{R} \underline{w} - \underline{w}^T \underline{k}) \quad (8)$$

Lagrange multipliers are used to minimize the quadratic form from Equation (7). The details concerning the resolution can be found in [7].

These properties define the Gaussian process regressor with respect to the covariance function and its hyper-parameters:

$$\begin{cases} m(\underline{x}) &= \underline{f}(\underline{x})^T \hat{\underline{\beta}} + \underline{\mathbf{r}}(\underline{x})^T \mathbf{R}^{-1} (\underline{\mathbf{Y}} - \mathbf{F} \hat{\underline{\beta}}) \\ C(\underline{x}, \underline{x}') &= 1 - \underline{\mathbf{r}}(\underline{x})^T \mathbf{R}^{-1} \underline{\mathbf{r}}(\underline{x}') \\ &\quad + \underline{\mathbf{u}}(\underline{x})^T (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \underline{\mathbf{u}}(\underline{x}') \\ \hat{\underline{\beta}} &= (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \mathbf{R}^{-1} \underline{\mathbf{Y}} \\ \underline{\mathbf{u}}(\underline{x}) &= \mathbf{F}^T \mathbf{R}^{-1} \underline{\mathbf{r}}(\underline{x}) - \underline{f}(\underline{x}) \end{cases} \quad (9)$$

The kriging variance — also called Mean Squared Error (MSE) — is defined by:

$$\text{MSE} = \sigma^2 (1 - \underline{\mathbf{r}}^T \mathbf{R}^{-1} \underline{\mathbf{r}} + \underline{\mathbf{u}}^T (\mathbf{F}^T \mathbf{R}^{-1} \mathbf{F})^{-1} \underline{\mathbf{u}}) \quad (10)$$

Covariance function being chosen, its hyper-parameters could be obtained by maximizing the likelihood function associated or by Cross-Validation [19]. The impact of the covariance function choice can be seen on [34]. Here, computations are made with the function Matern 5/2.

1.2 Using all data sources with multi-fidelity kriging

Usually, during the design process, engineers start by testing several configurations with a fast but inaccurate solver (coarse mesh, linear model,...) and validate their choice after with an accurate solver. A way to reduce design step is to reuse this process for kriging by considering multiple solvers from the coarse one to the finest one and merge data observations. An example can be found in [21] and the reader can refer to [8] for a review of surrogate modeling and multi-fidelity approach.

Coming back to Equation (4), a possibility is to build joint distribution with multiple random observation vectors which include low-fidelity observations $(\mathbf{X}_{LF}, \underline{\mathbf{Y}}_{LF})$ and high-fidelity one $(\mathbf{X}_{HF}, \underline{\mathbf{Y}}_{HF})$. This approach is called cokriging. Some hypotheses are needed for defining interaction between observations and can be found in [40, 14, 18, 25]. All these methods modify covariance matrix to take into account solver quality. So they allow expert judgement, but they are intrusive.

Other methods based on recursive metamodeling exist: Hierarchical Kriging [13] which replaces regression function by a Gaussian process conditioned to low-fidelity data, and Evofusion [10]. Details of Evofusion algorithm can be seen in Algorithm 1.

The Evofusion method is the simplest one. To illustrate how it works, we consider that the dotted curve in figure 1 is obtained from a low-fidelity solver and the continuous one is obtained from a high-fidelity solver. Data observed from both solvers are dots on curves.

Algorithm 1 Evofusion Algorithm

Require: low-fidelity and high-fidelity observations

- 1: Build a low-fidelity metamodel with only low-fidelity data $(\mathbf{X}_{LF}, \underline{\mathbf{Y}}_{LF}) \rightarrow \hat{\mathbf{Y}}_{LF}$
 - 2: Compute the gap between the low-fidelity metamodel $\hat{\mathbf{Y}}_{LF}$ and high-fidelity observations $\underline{\mathbf{Y}}_{HF}$ on high-fidelity points \mathbf{X}_{HF} :
 $\mathbf{Y}_{corr} = \underline{\mathbf{Y}}_{HF} - \hat{\mathbf{Y}}_{LF}(\mathbf{X}_{HF})$
 - 3: Build a correction metamodel with correction data $(\mathbf{X}_{HF}, \underline{\mathbf{Y}}_{corr}) \rightarrow \hat{\mathbf{Y}}_{corr}$
 - 4: Modify low-fidelity observations with the correction metamodel:
 $\underline{\mathbf{Y}}_{LFcorr} = \underline{\mathbf{Y}}_{LF} + \hat{\mathbf{Y}}_{corr}(\mathbf{X}_{LF})$
 - 5: Build the fused metamodel with corrected data and high-fidelity data:
 $(\mathbf{X}_{LF}, \underline{\mathbf{Y}}_{corr}) \cup (\mathbf{X}_{HF}, \underline{\mathbf{Y}}_{HF}) \rightarrow \hat{\mathbf{Y}}$
 - 6: **return** Fused metamodel: $\hat{\mathbf{Y}}$
-

Firstly, a Gaussian process with low-fidelity data is created. The gap between LF regressor and HF data is used to create a correction model. A Gaussian process conditioned by corrected LF and HF data is built with this and is visible on figure 2. The major drawback of this method is the wrong variance associated to low-fidelity corrected data, which is treated as zero.

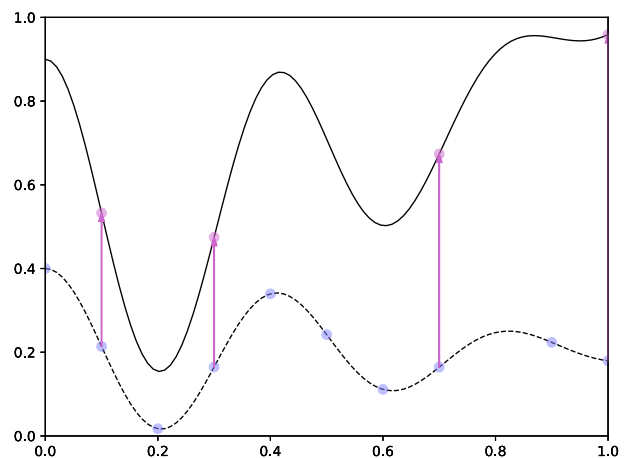


Fig. 1: Data observed (dotted curve is from a LF solver, continuous one is from a HF solver, bullets • are data observed and pink arrows describe LF-HF exact gap)

Benchmarks of these methods can be found in [6, 24]. With these methods, results from multiple solvers can be well exploited. One of the principal interests in kriging is the variance of the Gaussian process. The kriging variance is a powerful tool to enrich an experimental design as a model error estimator. In the case of Evofusion, Mean Squared Error from correction metamodel is going to be used.

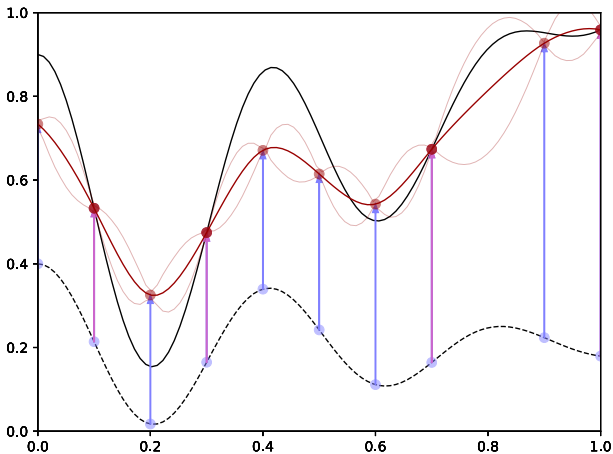


Fig. 2: Evofusion method: Gaussian process with corrected LF and HF data (red continuous curve is the metamodel and the upper and lower dotted ones delimit the estimated confidence interval computed from the metamodel. Blue arrows are LF-HF estimated gap from pink arrows)

1.3 Enriching surrogate model with adaptive resampling

Usually, kriging is used with two steps. During the first one, a surrogate model is made with simulations chosen from a design set sampling like Latin Hypercube Sampling (LHS) [28], or low-discrepancy suite. The second stage consists in adding simulations sequentially at new design points by maximizing information gain for the surrogate model until the quality indicator reaches tolerance. Here, new points improve the global accuracy of the kriging model. Sequential enrichment can also be used in optimization with notable EGO Algorithm [16], and probabilities of failure estimation.

Among information gain predictor, Mean Square Error is mainly used and is fast to compute as the variance of prediction. Some other exist like the Integrated MSE (IMSE) [32], which consists in integrating kriging variance over the design space for the predicted surrogate model with the new point:

$$IMSE(\underline{x}) = \int_{\mathcal{D}} MSE[\hat{Y}_x](\underline{s}) d\underline{s} \quad (11)$$

where \hat{Y}_x is a metamodel build with the same data and the new observation $(\underline{x}, \hat{Y}(\underline{x}))$. The optimal point will be the global maximum one.

This method can be costly because each point tested needs a surrogate model building and prediction. A way to reduce costs could be to create kriging model on IMSE or do Bayes optimisation with EGO algorithm.

If data are sufficient, it could be interesting to consider cross-validation (CV) based criteria like Leave-One Out (LOO) [20]. These criteria pilot enrichment process. About quality indicators, further methods exist. We can cite MSE, all determination coefficients like r-squared, Bravais-Pearson coefficient, or also cross validation criteria like LOO.

Defining the enrichment strategy in a multi-fidelity framework is still an open problem and is critical for time computation gain. Le Gratiet [25] developed sequential design for multi-fidelity co-kriging models with a reflexion about the time-ratios between the code levels and the contribution of each code level to the total predictor MSE. Here, a generic method is used with a first step using only the low-fidelity solver, and a second one using both solvers has described in Algorithm 2. But some more sophisticated strategies could be developed with a great estimated gain.

Algorithm 2 Multi-fidelity Surrogate Model Generation

Require: \mathcal{D} , Solver Y

- 1: Design Sampling on $\mathcal{D} \rightarrow \mathbf{X}$
 - 2: LF Solve (Error η_{LF}) $\mathbf{X} \rightarrow \underline{Y}_{LF}$
 - 3: **while** $r \geq r_{tol}$ **do**
 - 4: Fit surrogate model \hat{Y} to observations $(\mathbf{X}, \underline{Y}_{LF})$
 - 5: Predict quality indicator r on \mathcal{D}
 - 6: Find optimum \underline{x}^* of the information gain predictor (MSE, IMSE, LOO, ...) on \mathcal{D}
 - 7: LF Solve \underline{x}^*
 - 8: Add $(\underline{x}^*, Y_{LF}(\underline{x}^*))$ on surrogate model observations
 - 9: **if** LF Points $\geq n$ **then**
 - 10: HF Solve (Error η_{HF}) $\underline{x}^* \rightarrow Y_{HF}(\underline{x}^*)$
 - 11: Add $(\underline{x}^*, Y_{HF}(\underline{x}^*))$ on surrogate model observations
 - 12: **end if**
 - 13: **end while**
-

The two algorithm parameters η_{LF} and n on Algorithm 2 are possible keys here to reduce time computation. About this aspect, multiple studies have been already made by using coarse and fine models [11] or meshes [21] but considering partially and totally converged solver results is recent. Here, the solver uses a mixed formulation and generate on-the-fly space-time ROM fields. This allows to have during computation a space and time approximation of mechanical fields.

2 Access to multiple fidelity data with the LATIN-PGD framework

Multi-fidelity kriging allows using multiple data sources and multiple solvers in the case of computer experiments. But using multiple solvers could be difficult because of solver and I/O configuration. With the LATIN-

PGD framework, approximated but complete spatio-temporal mechanical fields are computed at each convergence iteration. So QoI can be estimated by stopping the solver before convergence. LATIN-PGD framework is used as a demonstrative example in the context of viscoplastic problems. Detailed equations could also be found in [2].

2.1 Elasto-viscoplasticity reference problem

The reference problem is a quasi-static isothermal evolution defined over the time-space domain $I \times \Omega$, assuming the hypothesis of small perturbations. The structure is subjected to body forces \underline{f}_d , traction forces \underline{F}_d over a part $\partial_2\Omega$ of the boundary, and to prescribed displacements \underline{U}_d over the complementary part $\partial_1\Omega$. The state of the structure is defined by the set of fields $\mathbf{s} = (\dot{\underline{\varepsilon}}_p, \underline{\sigma}, \underline{A}_k, \underline{V}_k)$ where:

- $\underline{\varepsilon}$ designates strain field corresponding to the displacement field \underline{U} , split between elastic part $\underline{\varepsilon}_e$, and inelastic part $\underline{\varepsilon}_p$;
- $\underline{\sigma}$ defines Cauchy stress field;
- \underline{A}_k and \underline{V}_k respectively primal and dual internal variable fields.

All these quantities are defined over the time-space domain $I \times \Omega$ and assumed to be sufficiently regular.

Kinematically admissibility $\underline{U} \in \mathcal{V}$, $\underline{\varepsilon} \in \mathcal{E}$ and static admissibility $\underline{\sigma} \in \mathcal{S}$ must be fulfilled. Spaces and the corresponding homogeneous vector spaces (with superscript *) are described below:

- the space \mathcal{V} of the kinematically admissible fields \underline{U} (\mathcal{V}^* is obtained with $\underline{U}_d = \underline{0}$):

$$\mathcal{V} = \left\{ \underline{U} \mid \underline{U}|_{t=0} = \underline{U}_0, \quad \underline{U} = \underline{U}_d \text{ on } \partial_1\Omega \right\} \quad (12)$$

- the space \mathcal{S} of the statically admissible fields $\underline{\sigma}$ such that

$$\begin{aligned} & - \int_{[0,T] \times \Omega} \underline{\sigma} : \underline{\varepsilon}(\underline{U}^*) d\Omega dt \\ & + \int_{[0,T] \times \Omega} \underline{f}_d \cdot \underline{U}^* d\Omega dt \\ & + \int_{[0,T] \times \partial_2\Omega} \underline{F}_d \cdot \underline{U}^* dS dt = 0 \end{aligned} \quad (13)$$

- the space \mathcal{E} of the kinematically admissible fields $\underline{\varepsilon}$ such that $\exists \underline{U} \in \mathcal{V}, \underline{\varepsilon} = \nabla_{sym}\underline{U}$.

2.2 Elasto-viscoplasticity behavior law

Unified viscoplastic framework previously presented in [26] is considered. In this constitutive law, strain $\underline{\varepsilon}$ is split between elastic reversible strain $\underline{\varepsilon}_e$ and plastic strain $\underline{\varepsilon}_p$ like $\underline{\varepsilon} = \underline{\varepsilon}_p + \underline{\varepsilon}_e$. Stress is driven by $\underline{\sigma} = \mathbf{K}_e \underline{\varepsilon}_e$ where \mathbf{K}_e is the Hooke matrix.

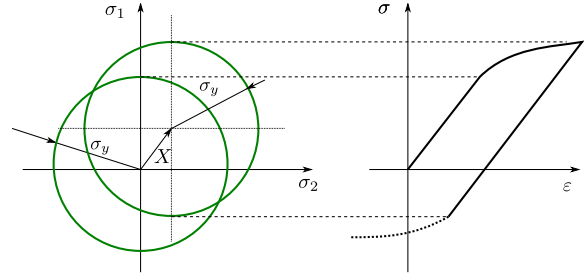


Fig. 3: Yield surface influence on strain and stress

Extreme loadings create plastic behavior zone when stress goes over an elastic limit f . This limit is usually represented by an ellipsis in deviatoric stress principal component space also called yield surface (figure 3). Size and origin of the ellipsis are driven respectively by isotropic hardening R (drag effect) and a unique linear kinematic hardening \mathbf{X} :

$$f = (\underline{\sigma} - \mathbf{X})_{eq} - \sigma_0 \quad (14)$$

where $J_2 = (\underline{\sigma} - \mathbf{X})_{eq}$ is the Von Mises equivalent stress and $\sigma_0 = \sigma_y - R$ is the yield surface size. We define the plastic strain p and the primal field linked with \mathbf{X} , α . Primal fields $\underline{A}_k = (p, \alpha)$ are associated with $\underline{V}_k = (R, \mathbf{X})$.

The Norton-Hoff law drives the plastic strain p :

$$\dot{p} = \left\langle \frac{f}{k} \right\rangle_+^N \quad (15)$$

where k , N are material dependant scalars, $\langle \cdot \rangle_+$ are Macaulay brackets.

State laws are:

$$\underline{\sigma} = \mathbf{K}_e \underline{\varepsilon}_e \quad (16)$$

$$\mathbf{X} = \frac{2}{3} C \alpha \quad (17)$$

$$R = R_\infty (1 - e^{-bp}) \quad (18)$$

where C, R_∞, b are material dependant scalars.

A pseudo-dissipation potential F is defined:

$$F = f + \frac{3\gamma}{4C} \mathbf{X} : \mathbf{X} - \frac{2\gamma C}{3} \alpha : \alpha \quad (19)$$

So, the evolution equations are described by:

$$\frac{d}{dt} \begin{bmatrix} \varepsilon_p \\ -\alpha \\ -p \end{bmatrix} = \left\langle \frac{f}{k} \right\rangle_+^N \begin{bmatrix} -\sqrt{\frac{3}{2}} \sqrt{\frac{3}{2}} \mathbb{N} \\ -\sqrt{\frac{3}{2}} \sqrt{\frac{3}{2}} \mathbb{N} + \frac{3\gamma}{2C} \mathbf{X} \\ -1 \end{bmatrix} \quad (20)$$

With \mathbb{N} the unitary normal vector

$$\mathbb{N} = \sqrt{\frac{3}{2}} \frac{\boldsymbol{\sigma}_D - \mathbf{X}}{(\boldsymbol{\sigma}_D - \mathbf{X})_{eq}}, \quad (\mathbb{N})_{eq} = 1 \quad (21)$$

The constitutive law can be described by decoupled linear state laws (18), and nonlinear, coupled evolution laws (20). A_d is defined as the admissibility space of fields which respects state linear equations and admissibility ones $\varepsilon \in \mathcal{E}$ and $\boldsymbol{\sigma} \in \mathcal{S}$. By the same way, Γ is defined as the local manifold with respect of nonlinear behavior laws. The mechanical solution is $\Gamma \cap A_d$ and can be found by alternating search in the global linear space and the local manifold.

2.3 LATIN-PGD algorithm

The LATIN framework [23] is a powerful method to obtain the linear equilibrium of the structure with the respect of nonlinear behavior laws. This iterative solver consists in searching alternately the solution on Γ and A_d , with search directions Θ^+ and Θ^- which allow finding a solution on one space which minimizes distance projection to the other one.

Unlike Newton-like techniques, this approach provides a complete solution at each iteration of the algorithm. The schematic representation in Figure 4 illustrates the differences between the two solvers.

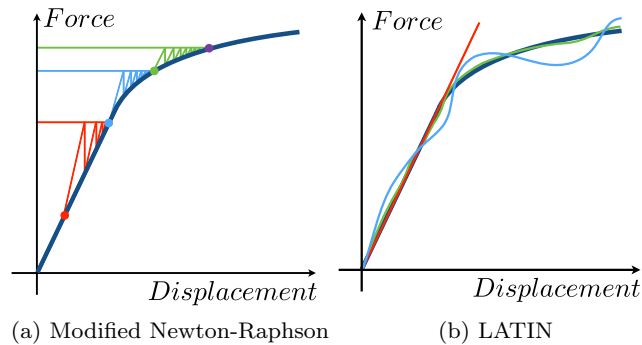


Fig. 4: Schematic representation of the solvers

Such strategy has also been applied for other types of problems: multi-scale, domain decomposition, multi-physics... Here, model-order reduction (PGD) is used

to reduce the cost of resolution of the global problem. More details is given in [23, 30] and a short analysis can be found in [37].

Steps can be briefly described below and on figure 5:

- **Elastic initialization:** The algorithm is classically initialized by computing the elastic solution of the problem, such that $\mathbf{s}_0 \in A_d$. \mathbf{s}_0 can be enriched by adding kinematically admissible with null condition information from previous computations. This possibility allows the implementation of the multi-parametric strategy presented in the section 2.5.
- **Local stage:** Knowing a solution \mathbf{s} in A_d , local stage consists in finding a local solution $\hat{\mathbf{s}}$ in Γ using search direction $(\hat{\mathbf{s}} - \mathbf{s}) \in \Theta^+$. The local problem can be solved at each time step at each Gauss point. Here, formulation is written to be explicit.
- **Global stage:** The linear stage consists in finding \mathbf{s} in A_d knowing $\hat{\mathbf{s}} \in \Gamma$ and using the search direction Θ^- .

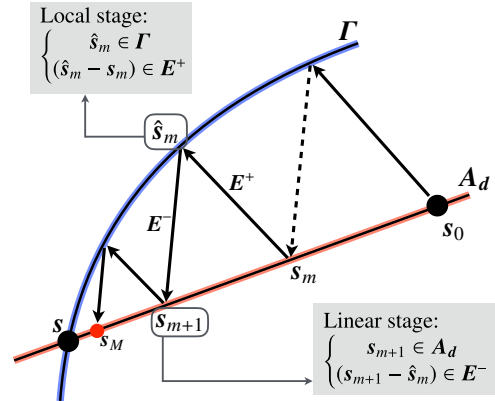


Fig. 5: Schematic representation of the LATIN strategy

The chosen error indicator is a stagnation criterion on consecutive solutions \mathbf{s} and $\hat{\mathbf{s}}$:

$$\eta_{LATIN} = \frac{1}{2} \frac{\|\dot{\hat{\varepsilon}}_p - \dot{\varepsilon}_p\|_{Ke}}{\|\frac{1}{2}(\dot{\hat{\varepsilon}}_p + \dot{\varepsilon}_p)\|_{Ke}} + \frac{1}{2} \frac{\|\hat{\boldsymbol{\sigma}} - \boldsymbol{\sigma}\|_{Ke^{-1}}}{\|\frac{1}{2}(\hat{\boldsymbol{\sigma}} + \boldsymbol{\sigma})\|_{Ke^{-1}}} \quad (22)$$

where $\|\bullet\|_{Ke}$ is the norm associated to the Hooke operator.

The local search direction Θ^+ (which can be seen as a local approximation of the A_d space) is defined by $\Theta^+ = \infty$, which means that dual quantities are conserved between the two stages. This operator evolves along LATIN iterations and depends on space and time.

At the global linear stage of the LATIN method, a linear approximation of the elastic-viscoplastic problem is known. The PGD technique can be introduced easily in this context. The unknowns are searched as sums of products of time functions $\lambda_i(t)$ and space functions $\underline{A}_i(M)$. For example, with $m - 1$ pairs,

$$\boldsymbol{\sigma}(t, M) = \sum_{i=1}^{m-1} \lambda_i(t) \mathbf{C}(\underline{A}_i)(M) \quad (23a)$$

$$\dot{\boldsymbol{\epsilon}}_p(t, M) = \sum_{i=1}^{m-1} \dot{\lambda}_i(t) \underline{A}_i(M) \quad (23b)$$

With PGD modes, a preliminary step can be made based on updating time modes (λ_i) to minimize search direction. If updating is not sufficient, a new time-space mode is created assuming $\mathbf{s} \in A_d$.

The interested reader can refer to [31] for a review of the different algorithms to solve a linear problem with the PGD and [35] for the special case of the viscoplastic LATIN algorithm. A progressive algorithm with update is considered, based on the minimization of the linear search direction. The stop criterion will use LATIN error estimator even if some other criterion can be used like a QoI-based error.

2.4 LATIN-PGD algorithm in action

To illustrate, a simple test case is used here. An elasto-viscoplastic beam (figure 6) is considered with symmetry boundary conditions on origin planes $x = 0$, $y = 0$, $z = 0$. The dimensions in (x, y, z) are $100 \times 10 \times 10$ mm. A prescribed displacement $U_d(t)$ is considered on the $x = 100$ mm plane. The temporal evolution of $U_d(t)$ is visible on the figure 7 with $U_{max} = 0.05$ mm. The eps-sigma curve is drawn at for some convergence iterations on figure 8.

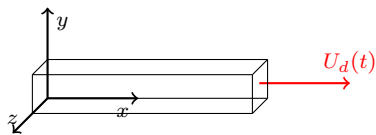


Fig. 6: Elasto-viscoplastic beam test-case

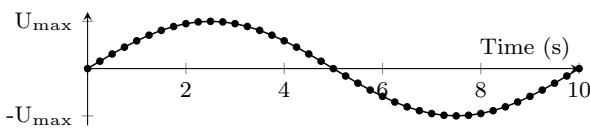


Fig. 7: Prescribed displacement evolution

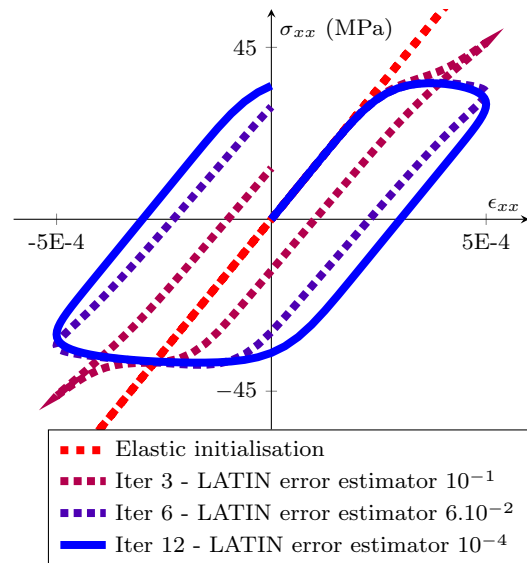


Fig. 8: Evolution of $\epsilon - \sigma$ curve during convergence loops

An iteration of the LATIN frame gives an approximate but complete space-time solution field (Fig. 4b). The first argument in favour of the LATIN-PGD framework is that the calculation could be stopped before convergence if the approximation is sufficient to be considered as a low-fidelity observation. For example, the maximum of the equivalent Von Mises stress calculated at iteration #6 and convergence are close. Thus, non-converging fields can give a good approximation of QoI values.

2.5 Multi-parametric strategy to accelerate computation

For surrogate modelling, the computation of a quantity of interest Y_i is done sequentially on certain points \underline{x}_i in the design space \mathcal{D} . A major feature of the LATIN framework is its multi-parametric strategy which allows to start a new calculation with fields created from previous calculations. Its objective is to provide very quickly the solution of a nonlinear evolution problem for several parameter values of the model and reduce the number of iterations to reach the required error estimator level. This strategy is presented on Algorithm 3 and a schematic representation is shown on figure 9.

The figure 10 shows an application of this strategy on the one-side clamped elasto-viscoplastic beam. The result obtained on figure 8 is used to start resolution of a similar problem, but with a softer material.

The exploration path of the design space which minimizes time computation is not trivial, and this question has already been addressed in [3, 15]. An approach sim-

Algorithm 3 Multi-parametric Latin-PGD algorithm

Require: Model informations

- 1: Elastic resolution - Generate CA modes $\in \mathcal{V}$
- 2: Load mode database - Load old CA₀ modes $(\lambda, \underline{\Delta})$
- 3: **while** $\eta_{LATIN} > \eta_{LATIN,tol}$ **do**
- 4: Local stage: $\mathbf{s} \rightarrow \hat{\mathbf{s}} \in \Gamma$
- 5: Preliminary stage: Update all time modes
- 6: **if** $\|\Delta\lambda\| \leq \nu \|\lambda\|$ **then**
- 7: Global stage: Compute new CA₀ mode $(\lambda, \underline{\Delta})$
- 8: **end if**
- 9: **end while**
- 10: **return** Solution at $\eta_{LATIN,tol}$: \mathbf{s}

ilar to the RB method to select the most relevant sets of parameters have been made to compute the enrichment.

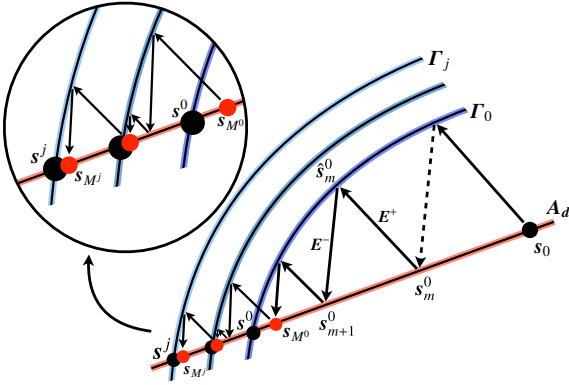


Fig. 9: Schematic representation of the multi-parametric strategy [37]

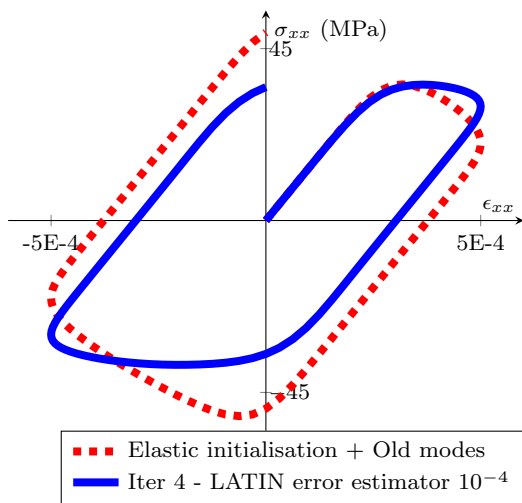


Fig. 10: $\varepsilon - \sigma$ curve obtained from a LATIN-PGD resolution started from another solution

3 Impact of the elasto-viscoplasticity LATIN-PGD framework on time computation

3.1 Implementation

A 3D solver implementation was developed for the test using MATLAB. To reach fast and multithreading computation, LATIN operators are organized as nd -arrays and a parallel implementation of Einstein summation [29] exploits variable broadcasting to deal with time- and spatial-dependent operators.

3.2 Test case 1: Plate with an elliptic inclusion

Two academic examples are tested. The first one is an eighth part of 3D plate with an elliptic inclusion presented on figure 11. The typical dimensions in (x,y,z) are $10 \times 1 \times 20$ mm. Material behaviors are described by the Chaboche constitutive law (see Section 2.2) and Table 2 for the material coefficients.

The plate is subjected to symmetry boundary conditions on $x=0$, $y=0$, $z=0$ planes and to a prescribed displacement $\underline{U} = \underline{U}_d$ at the upper side $z = 20$ mm. \underline{U}_d evolution can be seen on figure 7 with $U_{max} = 0.03$ mm. Structure came with 316 Steel parameters at 600°K. Structure and inclusion have the same parameters except for the inclusion Young modulus αE_{316} and the structure power coefficient of Norton-Hoff law.

The aim here is to obtain an approximation of the maximum of the Rankine stress during one loading cycle on the 2D design space $\mathcal{D} = [0.1, 10] \times [3, 7]$ as described on Table 1.

Objective function	$Y(\underline{x}) = \max_{\Gamma \times \Omega} \sigma_{Rankine}$
Parameters	$\underline{x} = (\boldsymbol{\alpha}, \mathbf{n}) \in \mathcal{D} = [0.1, 10] \times [3, 7]$
Space element type	Linear triangular (1379 DOFs)
Loading cycles	One cycle (10s — 41 time steps)
Boundary conditions	Symmetry conditions on bottom and two side faces $\underline{U} = U_d(t)\underline{e}_z$ at the top

Table 1: Inclusion test case parameters

part	E	σ_y	n	K
Structure	137.6 GPa	8 MPa	\mathbf{n}	$150 \text{ MPa} \cdot \mathbf{s}^{1/n}$
Inclusion	$\boldsymbol{\alpha} \times 137.6 \text{ GPa}$	8 MPa	5	$150 \text{ MPa} \cdot \mathbf{s}^{1/n}$

Table 2: Elastic-viscoplastic constitutive coefficients for inclusion test case parts (**Bold** symbols are design parameters)

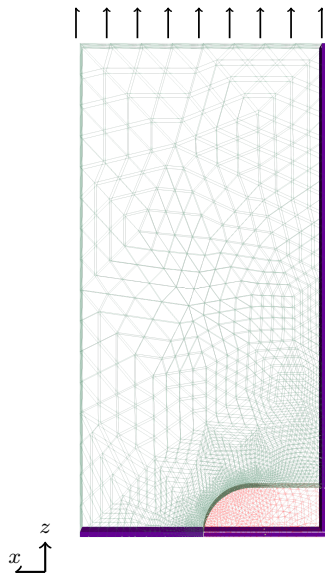


Fig. 11: Inclusion test case with symmetry conditions on purple parts and the prescribed displacement on the upper part. A symmetry condition is also prescribed on the back side (not visible on the figure)

3.3 Test case 2: Turbine blade

This second test-case is freely inspired from a blade of the Vulcain engine of the Ariane 5 launcher [30]. The typical dimensions in (x,y,z) are $45 \times 70 \times 50$ mm.

The mechanical part is clamped on the upper, left and right sides (Blue parts on figure 12). The turbine nose is visible on the bottom left of the figure 12. This one is loaded on the front and rear sides (on green and red on the figure). The two loading pressures have the same intensity $P(t)$ described by figure 14 with $P_{max} = 80$ MPa, but their direction are different. Each loading direction is driven by two angles (θ, ϕ) which could go between $[0^\circ, 90^\circ]$. The description of the angle-driven loading conditions is shown on figure 13. The material is also a 316 Steel at 600°K .

The aim here is to obtain an approximation of the maximum of the Von Mises stress for every possible load directions during one loading cycle. All loading options describe a 4D design space $\mathcal{D} = [0^\circ, 90^\circ]^4$.

Objective function	$Y(\underline{x}) = \max_{I \times \Omega} \sigma_{VonMises}$
Parameters	$\underline{x} = (\theta_1, \phi_1, \theta_2, \phi_2) \in \mathcal{D} = [0^\circ, 90^\circ]^4$
Space element type	Linear triangular (30k DOFs)
Loading cycles	One cycle (10s — 41 time steps)
Boundary conditions	Clamped on blue side faces $\underline{P}_1(t) = P(t) \underline{e}_1$ on green $\underline{P}_2(t) = P(t) \underline{e}_2$ on red

Table 3: Second test case characteristics

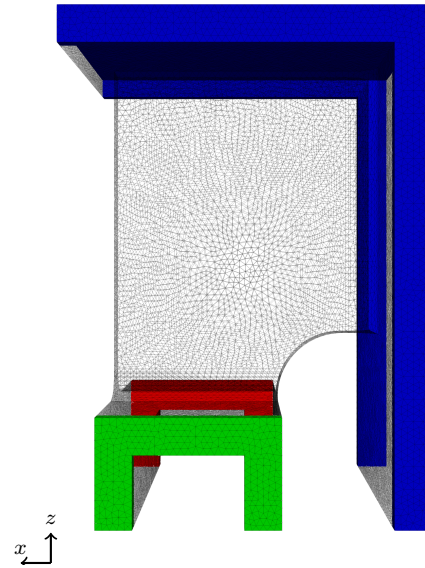


Fig. 12: Turbine blade test case

The blue part is clamped as the hidden upper and back part of the support, the green part is loaded as shown on figure 13, and the hidden part of the nose is also loaded

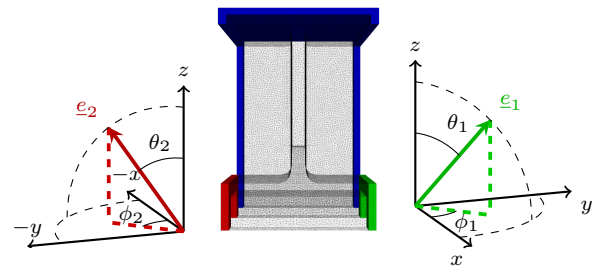


Fig. 13: Angle-driven loading directions on the green and red parts of the figure 12

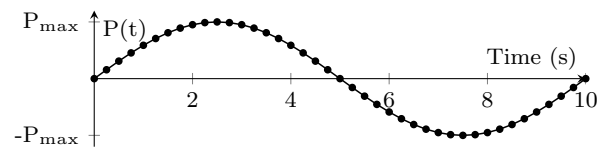


Fig. 14: Pressure evolution $P(t)$

3.4 Multi-parametric impact

Multi-parametric impact is tested with the first test case (figure 11). A 1D-design space \mathcal{D} is defined by $\alpha = [1, 10]$ and the sequential sampling is chosen as k equally spaced points between the bounds of the design space. In this test, computations from scratch and computations started with modes already computed from other computations are tested. Results are summarized in Table 4. In comparison with a Newton solver with restarts from scratch, the LATIN-PGD one reuses information from the closest design point. So finer is the discretisation, better is the speedup. The used sampling is

k sequential test cases	10	20	50	100
From scratch	119s	372s	1468s	4059s
On-the-fly database	16s	37s	117s	245s
Speedup	7.4	10.1	12.6	16.5

Table 4: Sequential sampling: computation time

an easy one because it minimizes distances between two consecutive design points [3]. The use of adaptive sampling will increase the distances between the already calculated points and the next one. The acceleration obtained will therefore be lower than these results.

3.5 Link between solver and quantity of interest error estimators

For each computation, at each solver iteration, full spatio-temporal mechanical fields are computed. So, quantities of interest $\max_{I \times \Omega} \sigma_{VonMises}$, $\max_I \int_{\Omega} p$ and the LATIN error estimator (Equation (22)) can be computed at each solver iteration. So the correlation between QoI error estimator and the LATIN one can be done easily and a low-fidelity stopping criterion can be defined by finding interactions between QoI error estimator $\frac{f-f_{exact}}{f_{exact}}$ and LATIN error estimator. A study is made on the first test case (figure 11) with computations selected on a LHS design space sampling 15×15 .

Figures 15, 16 allow us to estimate the LATIN stopping criterion needed to have numerous computations (50%, 95% or all design points) under the quantity of interest error estimator. For example, for having all design points under 10% of QoI error estimator, computations should be stopped when LATIN error estimator is under 10^{-2} .

The results suggest that $\eta_{LATIN} \approx 10^{-2}$ could be a good stopping criterion for the low-fidelity computation with about 1% error on the first QoI (Maximum Von Mises). This result is also visible in the following

Section when multi-fidelity kriging and model reduction are coupled.

Also, some computations need a lower number of LATIN iterations to achieve a defined QoI error. Having different low-fidelity stop criteria on design space could be a good idea to minimize time computation.

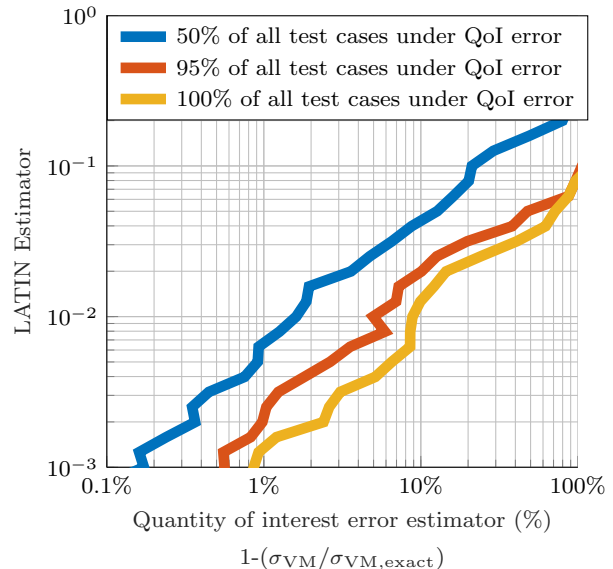


Fig. 15: Comparison between LATIN error estimator and QoI error estimator - QoI: maximum Von Mises stress

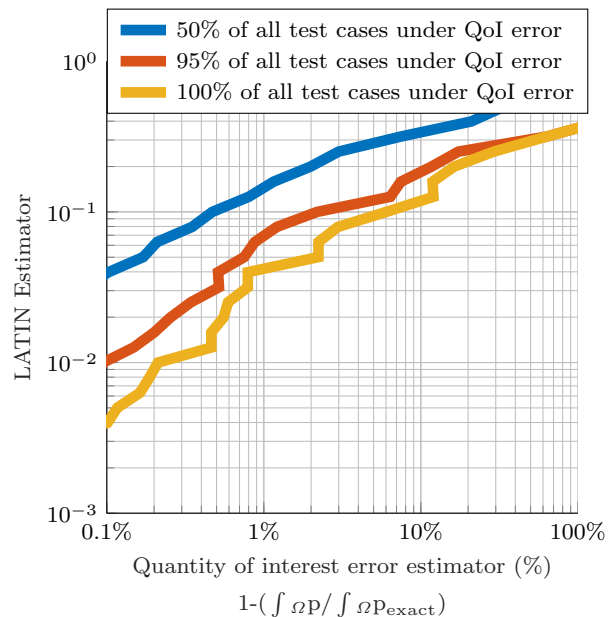


Fig. 16: Comparison between LATIN error estimator and QoI error estimator - QoI: maximum plastic strain rate

4 Accelerate virtual charts generation with Kriging-ROM algorithm

First Section gives the right regressor with multi-fidelity kriging. We can use this method with multiple solvers, or with one multi-quality solver which improve accuracy of QoI at every convergence iteration.

Algorithms 2 will be used with solver 3. Some questions remain about kriging parameters on algorithm 2. How many points are needed to start metamodel generation? And which low-fidelity and high-fidelity quality are the best choice to reduce the time computation?

4.1 Sampling and quality levels

The aim is to determine best kriging parameters to minimize the generation time of a good metamodel. Metamodel quality is determined by considering a variation of Bravais-Pearson r coefficient, the constant conditional correlation r_{ccc} (the higher the better, 1 is the exact correlation):

$$r_{ccc} = \frac{2\sigma_{ex,app}^2}{\sigma_{ex}^2 + \sigma_{app}^2 + (m_{app} - m_{ex})^2} \quad (24)$$

Considering a fine regular grid on design space, σ^2 is the standard deviation of values on this and m the associated mean.

A reference metamodel is generated with a fine grid on the design space and converged data. Metamodel generation is accepted when $r_{ccc} > 0.95$. Ten solver stop criteria for low-fidelity data η_{LF} were tested in the range $[10^{-3}, 10^{-1}]$. For each, the parameter n_{dim} governing the number of initial points calculated only in low-fidelity was tested by varying it between 4 and 13 points per dimension. To avoid the influence of initial random sampling on the choice of parameters, 20 samples are tested by modifying the random seed generator for each couple (η_{LF}, n_{dim}) .

4.2 Results from test cases

We reconsider the first test case explained in Section 3.1. To see impact of the LATIN indicator on objective function, two surrogate models of maximum of Rankine stress are created: one at convergence $\eta_{LATIN} = 10^{-4}$ (figure 18), the other before at $\eta_{LATIN} = 10^{-2}$ (Figure 17).

This example shows that approximated fields already give information about the evolution of the objective function. Kriging configurations have been tested and give results presented on Table 5. Both multi-fidelity

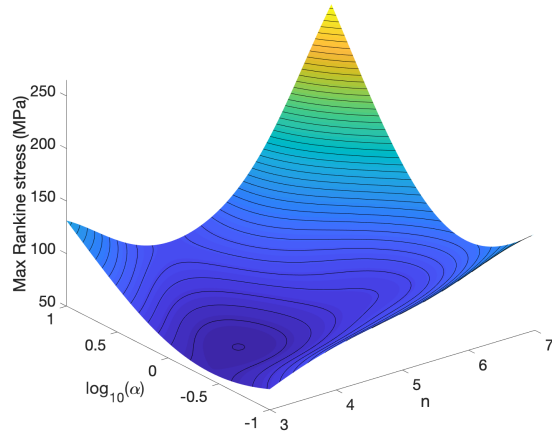


Fig. 17: QoI metamodel of the first test case obtained by considering $\eta_{LATIN} = 10^{-2}$ (Mean: 33s per computation)

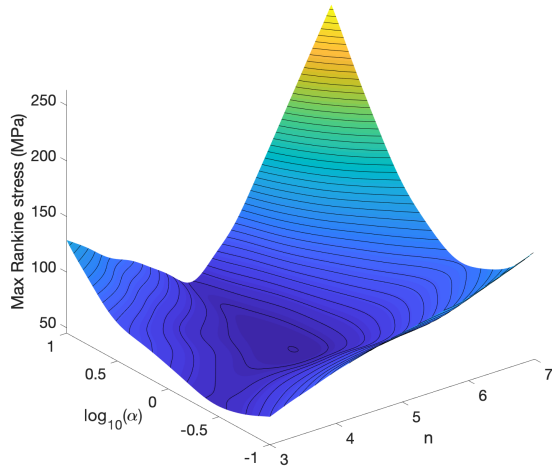


Fig. 18: QoI metamodel of the first test case obtained by considering $\eta_{LATIN} = 10^{-4}$ (Mean: 143s per computation)

and multi-parametric accelerate computations and gives good speedup: $5.2\times$ from multi-fidelity and $1.5\times$ from multi-parametric (411s in the case of LF error estimator $\eta_{LATIN} 10^{-2}$ and 12 LF points per dimension) so a $7.8\times$ projected speedup. Impact of multi-parametric will be more important as spatial degrees of freedom increased.

For the second test case, kriging configurations have also been tested and give results presented on Table 6. Multi-fidelity and multi-parametric also gives speedup: $5.5\times$ from multi-fidelity and $3.2\times$ from multi-paramet-

ric (3146s in the case of LF error estimator $\eta_{LATIN} = 10^{-2}$ and 12 LF points per dimension) so a **17.6** \times projected speedup.

The analysis of these two test cases show that the use of multi-fidelity data and the multi-parametric strategy allow significant computational gains. An optimal parameters pair emerges for the generation strategy with $\eta_{LATIN} = 10^{-2}$ as the low-fidelity stop criterion and 12 initial points per dimension calculated only in low-fidelity.

Number of LF points \ LF Latin Error	10^{-3}	10^{-2}	10^{-1}
4/dim	1623s	1457s	2094s
8/dim	921s	537s	2782s
12/dim	755s	272s	2685s

Table 5: Mean of time computation to generate meta-model for bi-material test case (Only HF data: 1440s)

Number of LF points \ LF Latin Error	10^{-3}	10^{-2}	10^{-1}
4/dim	11840s	10510s	8946s
8/dim	6350s	5709s	7583s
12/dim	4496s	2072s	2162s

Table 6: Mean of time computation to generate meta-model for blade test case (Only HF data : 11420s)

Conclusion

In this work, a new strategy has been proposed for the generation of virtual charts by sequentially sampling the design space and using a two-level of fidelity surrogate model. The solver presented reuses all previous information on physical fields to accelerate computation. With this, multi-fidelity kriging is coupled with data from the multi-parametric, multi-quality solver LATIN-PGD. The effectiveness of the proposed approach is demonstrated on two test cases.

The complexity of these test cases is related to the use of nonlinear elasto-viscoplastic models. The multi-parameter strategy speeds up the calculation according to the distance between data points. In addition, the coupling strategy outperforms classical one by given estimated speedups 7.8 \times on bi-material test case and 17.6 \times on the blade one when considering 12 low-fidelity

points per dimension, and a low-fidelity stop criterion $\eta_{LATIN} < 10^{-2}$.

Only the generation of a complete metamodel for virtual charts is presented in this article. In the case of optimisation, EGO strategy [17] allows to quickly find the global optimum by performing adaptive sampling only on a predicted optimal zone. The use of multi-fidelity coupling with model reduction could also give good results with potentially coarser low-fidelity fields.

Acknowledgements This work was supported by Ministry of Higher Education, Research and Innovation (France) and SAFRAN Tech. This work was also performed using HPC resources from the "Mesocentre" computing center of Centrale-Supélec and École normale supérieure Paris-Saclay supported by CNRS and Région Île-de-France (<http://mesocentre.centralesupelec.fr/>).

References

1. Aversano, G., Parra-Alvarez, J.C., Isaac, B.J., Smith, S.T., Coussement, A., Gicquel, O., Parente, A.: PCA and Kriging for the efficient exploration of consistency regions in Uncertainty Quantification. Proceedings of the Combustion Institute (2018). DOI 10.1016/j.proci.2018.07.040 **1**
2. Bhattacharyya, M., Fau, A., Nackenhorst, U., Néron, D., Ladevèze, P.: A LATIN-based model reduction approach for the simulation of cycling damage. Computational Mechanics pp. 1–19 (2017). DOI 10.1007/s00466-017-1523-z **5**
3. Boucard, P.A., Buytet, S., Guidault, P.A.: A multiscale strategy for structural optimization. International Journal for Numerical Methods in Engineering **78**(1), 101–126 (2009). DOI 10.1002/nme.2484 **7, 10**
4. Chatterjee, A.: An introduction to the proper orthogonal decomposition. Current Science **78**(7), 808–817 (2000) **1**
5. Chinesta, F., Keunings, R., Leygue, A.: The Proper Generalized Decomposition for Advanced Numerical Simulations. SpringerBriefs in Applied Sciences and Technology. Springer International Publishing, Cham (2014) **1**
6. Courier, N., Boucard, P.A., Soulier, B.: Variable-fidelity modeling of structural analysis of assemblies. Journal of Global Optimization **64**(3), 577–613 (2016). DOI 10.1007/s10898-015-0345-9 **3**
7. Cressie, N.: Statistics for Spatial Data. John Wiley & Sons (2015) **2**
8. De Lozzo, M.: Substitution de modèle et approche multifidélité en expérimentation numérique. Journal de la Société Française de Statistique **156**(4), 21–55 (2015) **3**
9. Forrester, A.I., Bressloff, N.W., Keane, A.J.: Optimization using surrogate models and partially converged computational fluid dynamics simulations. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **462**(2071), 2177–2204 (2006). DOI 10.1098/rspa.2006.1679 **1**
10. Forrester, A.I.J., Keane, A.J., Bressloff, N.W.: Design and Analysis of "Noisy" Computer Experiments. AIAA Journal **44**(10), 2331–2339 (2006) **3**
11. Forrester, A.I.J., Sóbester, A., Keane, A.J.: Multi-fidelity optimization via surrogate modelling. Proceedings of the Royal Society of London A: Mathematical, Physical and

- Engineering Sciences **463**(2088), 3251–3269 (2007). DOI 10.1098/rspa.2007.1900 [4](#)
12. Han, Z., Zimmerman, R., Görtz, S.: Alternative Cokriging Method for Variable-Fidelity Surrogate Modeling. *AIAA Journal* **50**(5), 1205–1210 (2012). DOI 10.2514/1.J051243 [1](#)
 13. Han, Z.H., Görtz, S.: A hierarchical kriging model for variable-fidelity surrogate modeling of aerodynamic functions. *AIAA Journal* **50**(9), 1885–1896 (2012) [3](#)
 14. 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. AIAA 2010-1225, Orlando, FL (2010) [3](#)
 15. Heyberger, C., Boucard, P.A., Néron, D.: A rational strategy for the resolution of parametrized problems in the PGD framework. *Computer Methods in Applied Mechanics and Engineering* **259**, 40–49 (2013). DOI 10.1016/j.cma.2013.03.002 [7](#)
 16. Jones, D.R.: A taxonomy of global optimization methods based on response surface. *Journal of Global Optimization* **21**, 345–383 (2001) [4](#)
 17. Jones, D.R., Schonlau, M., Welch, W.J.: Efficient global optimization of expensive black-box functions. *Journal of Global optimization* **13**(4), 455–492 (1998) [12](#)
 18. 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) [3](#)
 19. Kleijnen, J.P.: Experimental design for sensitivity analysis, optimization, and validation of simulation models. *Handbook of simulation* **1998**, 173–223 (1998) [3](#)
 20. Kleijnen, J.P.C., van Beers, W.C.M.: Application-driven sequential designs for simulation experiments: Kriging metamodelling. *Journal of the Operational Research Society* **55**(8), 876–883 (2004). DOI 10.1057/palgrave.jors.2601747 [4](#)
 21. Kramer, B., Marques, A.N., Peherstorfer, B., Villa, U., Willcox, K.: Multifidelity probability estimation via fusion of estimators (2017) [1](#), [3](#), [4](#)
 22. Ladevèze, P.: Sur une famille d’algorithmes en mécanique des structures. *Comptes-rendus des séances de l’Académie des sciences. Série 2, Mécanique-physique, chimie, sciences de l’univers, sciences de la terre* **300**(2), 41–44 (1985) [1](#)
 23. Ladevèze, P.: *Nonlinear Computational Structural Mechanics: New Approaches and Non-Incremental Methods of Calculation*. Mechanical Engineering Series. Springer Verlag (1999) [1](#), [6](#)
 24. Laurent, L., Riche, R.L., Soulier, B., Boucard, P.A.: An overview of gradient-enhanced metamodelling with applications. *Archives of Computational Methods in Engineering* pp. pp 1–46 (2017). DOI 10.1007/s11831-017-9226-3 [3](#)
 25. Le Gratiet, L.: Recursive co-kriging model for Design of Computer experiments with multiple levels of fidelity with an application to hydrodynamic (2013) [3](#), [4](#)
 26. Lemaitre, J., Chaboche, J.L.: *Mechanics of Solid Materials*. Cambridge university press (1994) [5](#)
 27. Maday, Y., Ronquist, E.: The Reduced Basis Element Method: Application to a Thermal Fin Problem. *SIAM Journal on Scientific Computing* **26**(1), 240–258 (2004). DOI 10.1137/S1064827502419932 [1](#)
 28. 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* **42**(1), 55–61 (2000) [2](#), [4](#)
 29. Nachar, S.: Einstein summation for MATLAB. Zenodo (2018). DOI 10.5281/zenodo.1297570 [8](#)
 30. Néron, D., Boucard, P.A., Relun, N.: Time-space PGD for the rapid solution of 3D nonlinear parametrized problems in the many-query context. *International Journal for Numerical Methods in Engineering* **103**(4), 275–292 (2015) [6](#), [9](#)
 31. Nouy, A.: A priori model reduction through Proper Generalized Decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering* **199**(23–24), 1603–1626 (2010). DOI 10.1016/j.cma.2010.01.009 [7](#)
 32. Picheny, V., Ginsbourger, D., Roustant, O., Haftka, R.T., Kim, N.H.: Adaptive Designs of Experiments for Accurate Approximation of Target Regions. *Journal of Mechanical Design* **132**(7), 1–9 (2010) [4](#)
 33. Quarteroni, A., Manzoni, A., Negri, F.: *Reduced Basis Methods for Partial Differential Equations, UNITEXT*, vol. 92. Springer International Publishing, Cham (2016) [1](#)
 34. Rasmussen, C.E., Williams, C.K.I.: *Gaussian Processes for Machine Learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass (2006). OCLC: ocm61285753 [2](#), [3](#)
 35. Relun, N., Néron, D., Boucard, P.A.: A model reduction technique based on the PGD for elastic-viscoplastic computational analysis. *Computational Mechanics* **51**(1), 83–92 (2013). DOI 10.1007/s00466-012-0706-x [7](#)
 36. Robinson, G.K.: That BLUP is a Good Thing: The Estimation of Random Effects. *Statistical Science* **6**(1), 15–32 (1991). DOI 10.1214/ss/1177011926 [2](#)
 37. Vitse, M., Néron, D., Boucard, P.A.: Virtual charts of solutions for parametrized nonlinear equations. *Computational Mechanics* **54**(6), 1529–1539 (2014). DOI 10.1007/s00466-014-1073-6 [6](#), [8](#)
 38. Willcox, K.: Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Computers & fluids* **35**(2), 208–226 (2006) [1](#)
 39. Zimmerman, D.L., Holland, D.M.: Complementary cokriging: Spatial prediction using data combined from several environmental monitoring networks. *Environmetrics* **16**, 219–234 (2005) [1](#)
 40. Zimmermann, R., Han, Z.H.: Simplified Cross-Correlation Estimation for Multi-Fidelity Surrogate Cokriging Models. *Advances and Applications in Mathematical Sciences* **7**(2), 181–201 (2010) [3](#)