

An efficient parallel global optimization strategy based on Kriging properties suitable for material parameters identification

Émile Roux, Yannick Tillier, Salim Kraria, Pierre-Olivier Bouchard

▶ **To cite this version:**

Émile Roux, Yannick Tillier, Salim Kraria, Pierre-Olivier Bouchard. An efficient parallel global optimization strategy based on Kriging properties suitable for material parameters identification. Archive of Mechanical Engineering, 2020, 67 (2), pp.169. 10.24425/ame.2020.131689 . hal-02624246

HAL Id: hal-02624246

https://minesparis-psl.hal.science/hal-02624246

Submitted on 26 May 2020

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.

ARCHIVE OF MECHANICAL ENGINEERING

2020

DOI: 10.24425/ame.2020.131689

Key words: global optimization, parallel computation, Kriging meta-model, inverse analysis

Emile Roux [1], Yannick Tillier[2], Salim Kraria[2], Pierre-Olivier Bouchard[2]

An efficient parallel global optimization strategy based on Kriging properties suitable for material parameters identification

Material parameters identification by inverse analysis using finite element computations leads to the resolution of complex and time-consuming optimization problems. One way to deal with these complex problems is to use meta-models to limit the number of objective function computations. In this paper, the Efficient Global Optimization (EGO) algorithm is used. The EGO algorithm is applied to specific objective functions, which are representative of material parameters identification issues. Isotropic and anisotropic correlation functions are tested. For anisotropic correlation functions, it leads to a significant reduction of the computation time. Besides, they appear to be a good way to deal with the weak sensitivity of the parameters. In order to decrease the computation time, a parallel strategy is defined. It relies on a virtual enrichment of the meta-model, in order to compute q new objective functions in a parallel environment. Different methods of choosing the $q$new objective functions are presented and compared. Speed-up tests show that Kriging Believer (KB) and minimum Constant Liar (CLmin) enrichments are suitable methods for this parallel EGO (EGO-p) algorithm. However, it must be noted that the most interesting speed-ups are observed for a small number of objective functions computed in parallel. Finally, the algorithm is successfully tested on a real parameters identification problem.

1. Introduction

Numerical modeling is now well established in the material forming industry to obtain fast and accurate solutions for different kinds of manufacturing processes. However, the accuracy of numerical simulations strongly depends on

✉ Emile Roux, e-mail: emile.roux@univ-smb.fr

[1]Université Savoie Mont-Blanc, SYMME, F-74000 Annecy, France.

[2]MINES ParisTech, PSL Research University, CEMEF-Centre de mise en forme des matériaux, CNRS UMR 7635, CS 10207 rue Claude Daunesse, 06904 Sophia Antipolis Cedex, France.



© 2020. The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives License (CC BY-NC-ND 4.0, https://creativecommons.org/licenses/by-nc-nd/4.0/), which permits use, distribution, and reproduction in any medium, provided that the Article is properly cited, the use is non-commercial, and no modifications or adaptations are made.

www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

the accuracy of the material behavior law and its parameters. Due to the complexity of stress states encountered in materials forming, the use of normalized tensile or torsion tests must often be completed by more complex tests exhibiting non-homogeneous stress/strain states. The use of inverse analysis is therefore mandatory for the parameters identification stage both for material behavior and damage models.

The identification of non-linear material behavior parameters using inverse analysis leads to complex optimization problems [1–3]. One of the most common techniques for this kind of identification is the minimization of the error between observables coming from experimental tests and numerical simulations. In addition, the mix of global (load-stroke curves) and local (digital image correlation measurements) data is often recommended to improve the accuracy of the identification methodology. This approach therefore leads to high computation costs since the identification of material parameters requires a large number of objective function evaluations. Each evaluation often requires a non-linear finite element computation. This is particularly true, when dealing with large plastic strains, ductile damage and fracture. It is thus necessary to work with efficient algorithms, adapted to these kinds of costly problems.

Many approaches have already been presented to deal with parameters identification based on finite element computation. One way of classifying these approaches is to distinguish 2 groups: local optimization methods and global optimization ones.

Local optimization methods do not guarantee that the global optimum is found. Therefore, the optimum obtained with these methods strongly depends on the initialization step of the algorithm. The most common local methods are Newton-based methods like Broyden–Fletcher–Goldfarb–Shanno [4] (BFGS) or quasi-Newton methods [4]. These methods require the computation of the gradient of the cost function. Usually, in the case of parameters identification using finite element computations, this gradient is numerically computed using the perturbation method. The computation of the gradient in this framework has two main drawbacks: the high computation time (computing the gradient in d dimensions requires $d+1$ finite element computations), and the need for a continuous cost function, which is sometimes not the case for non-linear finite element computations. Good alternatives to gradient based methods are simplex-based methods such as the Nelder-Mead algorithm [5] that do not require any gradient computation. These algorithms still suffer, though, from their dependence to initialization data (local optimization methods).

The other group of algorithms, the global optimization methods, are able to explore multiple optima. Therefore, they do not suffer from being stuck in local optima like the previous methods. Multiple algorithms are available, among which one can note genetic algorithms, evolutionary strategies [6]. These methods are well established, but their main drawback regarding the addressed problem is the large number of cost function evaluations to obtain the final optimum.

To overcome these issues (computation time and global optimum research), numerous hybrid methods are reported in the literature. These methods are called hybrid because of the combined use of response surface methods and optimization algorithms. Standers et al. [7] used the Successive Response Surface Method (SRSM) coupled with an explicit finite element solver to identify non-linear material parameters while Ageno et al. [8] used a gradient method based on Sequential Quadratic Programming (SQP) and Trust Region algorithm (TR) to identify elastic-plastic parameters using free-standing foils test. Abendroth et al. [9] proposed to build a neural network on the basis of several finite element computations of a small punching test and then used this neural network as an approximation of the objective function that is to be minimized by the SQP method. Abendroth's work has identified ductile damage parameters with a limited number of finite element computations. The work of Franchi et al. [10] used a multi Island genetic optimization algorithm coupled to a Radial Basis Functions (RBF) neural network to determine constitutive parameter values of a machining process. Souto et al. [11] used a Nelder-Mead direct search algorithm to propose innovative approaches to optimize the shape of a sample to make constitutive parameters identification easier. To address parameter identification of discrete element method model, Rackl et al. [12] used a Kriging meta-model coupled to the Levenberg-Marquardt algorithm [13], and Richter et al. [14] proposed a modular procedure named "generalized surrogate modeling-based calibration". All these works are dealing with expensive objective functions, i.e., each evaluation of the objective function refers to a finite element computation.

In this paper, an efficient identification strategy to find the global minimum of finite element based cost functions is studied. The technique used here can be classified in the "expensive black box function optimization" field, as defined by Jones [15]. The finite element computation and objective function calculation are encapsulated in an independent "black box". This "black box" is solely defined by its input (optimization parameters) and its output (the objective function). The Efficient Global Optimization (EGO) algorithm defined by Jones [15] or meta-model Assisted Evolutionary Algorithms (MAEA) presented by Emmerich [16] are suitable for solving such expensive optimization problems. Meta-models, such as Kriging, Radial basis function, or other, are used to reduce the number of "black box" function calls. The main purpose on a meta-model is to give an approximation of the objective function values over unknown areas. The main steps to work with a meta-model are to (i) build the meta-model on an initial database of "black box" computations; (ii) use this meta-model to select an interesting area of investigation; and (iii) run a "black box" call in this interesting area. These three steps lead to the enrichment of the database. After several iterations of these three steps, a solution of the global optimization problem is obtained. This methodology can be classified in the field of analysis of computer experiments as defined by Santner et al. [17], or in more recent work, such as [18]. With nowadays computation environment, it seems appropriate to extend this kind of algorithms to parallel environments. The work of

www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

Ginsbourger et al. [19] is a very interesting way to introduce parallel consideration in Kriging-based optimization algorithms. They proposed a formulation called *multipoint expected improvement* criterion (*q-EI*), which is able to select q new design points simultaneously, based on a known Kriging predictor. Ginsbourger et al. concluded that this *q-EI* criterion and its different approximations provide new satisfactory set of points to improve the solution of the optimization problem. Their experimental set-up is based on a known Kriging predictor. Then, the new set of design points is evaluated thanks to the *q-EI* criterion, but without updating the Kriging predictor.

The aim of this paper is to focus on the specific case of material parameters identification problems commonly encountered in the modeling of forming processes [20]. Unlike the cases described above, the general shape of the objective function is known. In the first part, the use of an ordinary Kriging methodology is defined within an optimization strategy approach. Next, the extension to parallel optimization proposed by Ginsbourger et al. [19] is described and included in the optimization strategy. The aim of this part is to test the efficiency of these different parallel strategies. To evaluate the efficiency of the algorithm, a speed-up criterion is evaluated and discussed. In the second part, this algorithm is tested on various analytical functions representative of material parameters identification problems. An analysis of the efficiency of this parallel algorithm is conducted and the results are discussed. Finally, the last section illustrates the methodology through the identification of the parameters of an elastic-plastic material behavior law coupled to a ductile damage model.

The algorithm presented here was integrated in the MOOPI (*MOdular software for Optimization and Parameters Identification*) software developed at Cemef-Mines ParisTech. It has shown its efficiency in forming applications [20] and model calibration using digital image correlation data [21]. The software is developed in an oriented object framework (C++), which makes it easy to add extra optimization algorithms, without working on the management of the finite element computation. Moreover, all the functionalities are accessible through a dedicated user interface.

2. Optimization assisted by Kriging meta-model

2.1. The optimization problem

The goal of the method is to find the global minimum of a function f in a defined region X.

$$x_{\min} = \min_{x \in X} f(x). \tag{1}$$

This function f is named the objective function. We assume that f is a function from $\mathfrak{R}^d$ to $\mathfrak{R}$, and that X is included in $\mathfrak{R}^d$. d is the dimension of the optimization problem, i.e., the number of parameters to be identified.

The value of f for a given x can be obtained from the "black box" solver. In our case, this function is computed by a finite element software coupled to a post-processing subroutine. For these mechanical finite element calculations, the computation of f is considered to be expensive in terms of CPU time. It is thus important to reduce the number of evaluations of the objective function.

2.2. Outline of the method

The outline of the method can be summed up by the following algorithm (Algorithm 1):

Algorithm 1: Global optimization method

1:	$i = 1$	
2:	DBp = init( )	*//DBp is the database of parameter set, DBp* $\in X$
3:	$DBy = f(DBp)$	*// "black box" function call. DBy is the database objective function value associated to parameter set*
4:	While ($i < i$ max) do	
5:	Meta =MetaModeling(DBp,DBy)	*// meta-model generation and calibration*
6:	$newDBp$ =argmin(Meta)	*// exploration and exploitation*
7:	$DBp = [DBp \cup newDBp]$	*// data enrichment*
8:	$DBy = [DBy \cup f(newDBp)]$	*// "black box" function call*
9:	$i = i + 1$	
10:	end	

This global optimization method (Algorithm 1) is a general description of an optimization procedure that is assisted by a meta-model. This procedure is built based on [16, 22]. This procedure leads to finding a global minimum of the objective function f. The choice is made to map the objective function in the design space by a meta-model. This map is then used to explore the minimum area.

Other authors, Emmerich et al. [16], have included meta-modeling directly in an evolutionary strategy algorithm: the selection step is thus assisted by a meta-model. The approach of including the meta-model is suitable enough for being used in conjunction with advanced genetic algorithms like NSGA-II [23, 24]. A major difficulty in using genetic algorithms is the settings of the parameters of the algorithm. In our approach, we want to reduce this number of setting parameters and therefore to make the optimization easier to set.

2.3. Ordinary Kriging

To obtain an efficient mapping of the objective function in the design space, an ordinary Kriging technique is used. Two steps are necessary: the construction of the Kriging predictor and a calibration of the Kriging predictor parameters.

2.3.1. Kriging predictor construction

Ordinary Kriging comes from Gaussian fields process theory, and was initially used in geostatistics [25]. Ordinary Kriging assumes that the process is a sum of a constant trend $\overline{m}$ and a local deviation R_x. We assume that the R_x process results from a Gaussian random field process with a mean value equal to 0 and a variance of σ^2. Therefore, the "black box" function F_x can be written as a stochastic process:

$$F_x = \overline{m} + R_x \,. \tag{2}$$

The estimator needs a master point database *DBp*, and some associated values of the objective function *DBy* in order to be computed.

$$DBp : \mathbf{p} = [p_1, \ldots, p_n] \in \Re^{d \times n}. \tag{3}$$

The database *DBp* is a list of n sets of parameters. Each set of parameters is specified as a vector of d values. *DBy* is a vector obtained by the n evaluations of the objective function.

$$DBy : \mathbf{y} = [y_1, \ldots, y_n] = [f(p_1), \ldots, f(p_n)] \in \Re^n. \tag{4}$$

The database *DBp* is initialized by a Latin hyper cube sampling method [24].

To build the estimator $\hat{y}$ of the Gaussian process F_x, $\overline{m}$ and σ^2 are estimated by a mean least square computation using the following equations:

$$\overline{m} = \frac{\bar{\mathbf{1}}^t \mathbf{C}^{-1} \mathbf{y}}{\bar{\mathbf{1}}^t \mathbf{C}^{-1} \bar{\mathbf{1}}}, \tag{5}$$

$$\sigma = \frac{\left(\mathbf{y} - \bar{\mathbf{1}}\overline{m}\right)^t \mathbf{C}^{-1} \left(\mathbf{y} - \bar{\mathbf{1}}\overline{m}\right)}{n} \,. \tag{6}$$

$\bar{\mathbf{1}}$ is the unit vector of size n, $\bar{\mathbf{1}}^t$ the transpose unit vector, and $\mathbf{C}$ the covariance matrix defined in Eq. (7).

$$\mathbf{C} = \left[\mathrm{corr}\left(p_i, p_j\right)\right]_{i,j \in [1,n]} \,. \tag{7}$$

The covariance matrix is built using the correlation function:

$$\mathrm{corr}\,(x, x', \theta, \tau) = \exp\left(-\sqrt{\sum_{i=1}^{d} \left(\frac{x'_i - x_i}{\theta_i}\right)^2}^{\tau}\right). \tag{8}$$

The function given in Eq. (8) evaluates the correlation between 2 points x and x', which are 2 points of the space X. Gaussian functions are often used in the analysis of computer design experiments as correlation functions. The function

given in Eq. (8) introduces two calibration parameters θ and τ. If θ is a scalar, the Kriging meta-model is isotropic, whereas if θ is a vector the meta-model is anisotropic. The parameter τ defines the smoothness of the predictor: $\tau = 2$ for a smooth predictor and $1 \leqslant \tau < 2$ for non-smooth predictors. The values of the parameters θ and τ are set during the calibration phase.

Equation (8) introduces the Gaussian correlation function. This kind of correlation function is well adapted to the analysis of computer experiments. But the Gaussian functions have a non-finite derivative order, which can lead to non-invertible covariance matrix $\mathbf{C}$. Other correlation functions can be found in the literature. The Matérn functions [26] are a good alternative to the Gaussian function.

The evaluation of $\overline{m}$ and $\mathbf{C}$ leads to the computation of the Kriging predictor $\hat{y}(x)$ [17]:

$$\hat{y}(x) = \overline{m} + \left(\mathbf{y} - \bar{\mathbf{1}}\overline{m}\right)^t \mathbf{C}^{-1}\mathbf{c}_{\mathbf{v}}(x), \tag{9}$$

where $x \in \mathfrak{R}^d$, and $\mathbf{c}_{\mathbf{v}}(x) = \left[\text{corr}\,(x, p_i)\right]_{i\in[1,n]}$.

The variance $\hat{\sigma}^2(x)$ of the estimate $\hat{y}(x)$ is given by the following equation:

$$\hat{\sigma}^2(x) = \sigma^2\left[1 - \mathbf{c}_{\mathbf{v}}(x)^t\mathbf{C}^{-1}\mathbf{c}_{\mathbf{v}}(x) + \frac{\left(1 - \bar{\mathbf{1}}^t\mathbf{C}^{-1}\mathbf{c}_{\mathbf{v}}(x)\right)^2}{\bar{\mathbf{1}}^t\mathbf{C}^{-1}\bar{\mathbf{1}}}\right]. \tag{10}$$

The predictor $\hat{y}(x)$ has some interesting properties. $\hat{y}(x)$ is an interpolation function, i.e., $\hat{y}\,(p_i) = f\,(p_i)$, $i \in [1, n]$, and the variance $\hat{\sigma}^2$ is equal to 0 at each master points. This is useful for deterministic computer design experiments. The predictor gives an approximation (Eq. (9)) and an estimation of the error related to the approximation (Eq. (10)) for all points belonging to the research space X.

2.3.2. Calibration

The goal of the calibration is to set the values of θ and τ in the correlation function (Eq. (8)). For material parameters identification, a smooth predictor is more appropriate (derivable objective function), which is why τ is chosen to be equal to 2, therefore only the θ parameter needs to be calibrated.

Two main methods are presented in the literature for calibrating Kriging meta-models: Cross Validation (CV) and Maximum Likelihood Estimation (MLE). According to Martin et al. [27], CV gives better results than MLE. However, CV requires more computation time. MLE is chosen in this paper because the calibration of the meta-model requires only the determinant of the correlation matrix. The inverse computation of the correlation matrix is unnecessary. The function to maximize for the MLE calibration is:

$$\begin{aligned} \text{MLE} &= \max_{\theta}\left(\ln(L)\right), \\ \ln\left(L(\theta,\tau)\right) &= -\frac{1}{2}\left[n\left(\ln\left(\sigma^2\right) + \ln(2\pi) + 1\right) + \ln\left(|\mathbf{C}|\right)\right]. \end{aligned} \tag{11}$$

www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

The MLE is not a trivial optimization problem. As suggested by Laurenceau et al. [28], the problem is solved thanks to a gradient method with a specific starting point given by:

$$\theta_k = \frac{l_{\max}}{\sqrt{-\ln(c)}}, \quad k \in [1, d], \tag{12}$$

where $l_{\max}$ is the maximal distance between master points.

2.4. Sequential optimization strategies

To describe the optimization procedure, we assume that a master points database and the associated objective function values are given. The goal of the meta-model is to define new points that need to be computed in order to enrich the meta-model and, in the end, to find the global minimum of the optimization problem.

Using a Kriging meta-model gives two intermediary results: a mean prediction $\hat{y}(x)$ of the Gaussian process F_x (Eq. (9)) and a variance prediction $\hat{\sigma}^2$ (Eq. (10)). These two results can be used to solve the global optimization problem. The mean and variance predictors are used for exploitation and exploration stages:

- Exploitation is the improvement of a local minimum that has already been detected. The new master point to be computed is obtained by minimizing the mean predictor function:

$$p_{n+1} = \min_{x \in X} \hat{y}(x). \tag{13}$$

- Exploration is the improvement of the prediction accuracy. A new master point is preferably defined in a research area with poor sampling. The predictor variance is minimized during the exploration:

$$p_{n+1} = \min_{x \in X} \hat{\sigma}(x). \tag{14}$$

Global optimization needs these two stages to find a global optimum. The exploration stage guarantees the global aspect of the search and the exploitation stage conducts to the improvement of the solution.

In order to reach the two goals (exploration and exploitation), functions (13) and (14) have to be minimized. Therefore, a multi objective approach is needed.

Many multi-objective formulations are proposed in the literature. In this paper, the Expected Improvement [15] (*EI*) approach is used. Other criteria can also be used to minimize the lower confidence bound [16] or to maximize the probability of the improvement [29].

EI was initially presented by Jones in the EGO algorithm. *EI* takes care of Kriging properties, and is defined by using a Gaussian distribution function:

$$EI(x) = (y_{\min} - \hat{y}(x))\, \Phi\left(\frac{y_{\min} - \hat{y}(x)}{\hat{\sigma}(x)}\right) + \hat{\sigma}(x)\varphi\left(\frac{y_{\min} - \hat{y}(x)}{\hat{\sigma}(x)}\right), \tag{15}$$

where $y_{\min} = \min_{i \in [1,n]} (y_i)$, Φ is the cumulative density function of the standard normal law with mean 0 and variance 1, and φ is the probability density function of the same normal law.

This *EI* function is a multimodal function whose maximization requires a global optimization procedure. However, this maximization does not take much time compared to the time required to compute the "black box" function. The maximization of the *EI* function is achieved using an evolution strategy algorithm. More details about this algorithm can be found in Dirk et al. [30]. A $(\lambda + \mu)$ strategy is used (Algorithm 2).

Algorithm 2: $(\lambda + \mu)$ Evolution strategy

1:	$i = 1$	
2:	P_1 = population initialization	// $P_1 \in X$
3:	Evaluate(P_1)	
4:	While ($i <$ *i* max) do	
5:	C_i = generate(P_i)	// *generation of* λ *children* C_i *from the parents population* P_i
6:	Evaluate(C_i)	// *evaluation of each child*
7:	P_{i+1} = selection in ($C_i \cup P_i$)	// μ *best individuals are selected*
8:	$i = i + 1$	
9:	end	

The Global optimization method (Algorithm 1) is a sequential algorithm. The initialization step is the only parallel step, because the initial values of the master points can be computed simultaneously. Each value refers to an independent call of the "black box". The iterative stage of Algorithm 1 is sequential: the exploration and exploitation stages of the meta-model consist in maximizing an auxiliary function (*EI* function). This optimization (Algorithm 2) gives one optimum. Therefore, only one new point of interest is extracted from the meta-model at each iteration.

2.5. Extension to parallel strategy

The procedure presented above gives a global solution of the minimization problem (1). However, this algorithm is sequential, which can lead to high computation time. Parallelization of the algorithm could lead to a reduction of CPU time. It would indeed be interesting to extract q points of interest at each iteration of the procedure. These q new points of interest are independent and refer to q calls of the "black box" function. These q "black box" function calls can thus be computed simultaneously and efficiently in a parallel environment.

The parallel strategy presented here is built thanks to the meta-model properties. In the work of Ginsbourger et al. [19] direct evaluation methods of the *q-EI*

criterion can be found. In this work we chose to work with the simplest method. The *q-EI* criterion is approximate thanks to an iterative method. The main idea of the iterative *q-EI* approximation is to set the value of the interesting extracted points to a virtual value. Then, a new meta-model is built using the master point database enriched by the virtual point, which is considered as known. Finally, a new point of interest is extracted according to the virtually enriched master point database. This procedure is performed iteratively while the maximum number *imax* of "black box" function calls is not reached. This procedure is described in Algorithm 3 and in Fig. 1. Moreover, the method is applied on a simple example in Fig. 1.

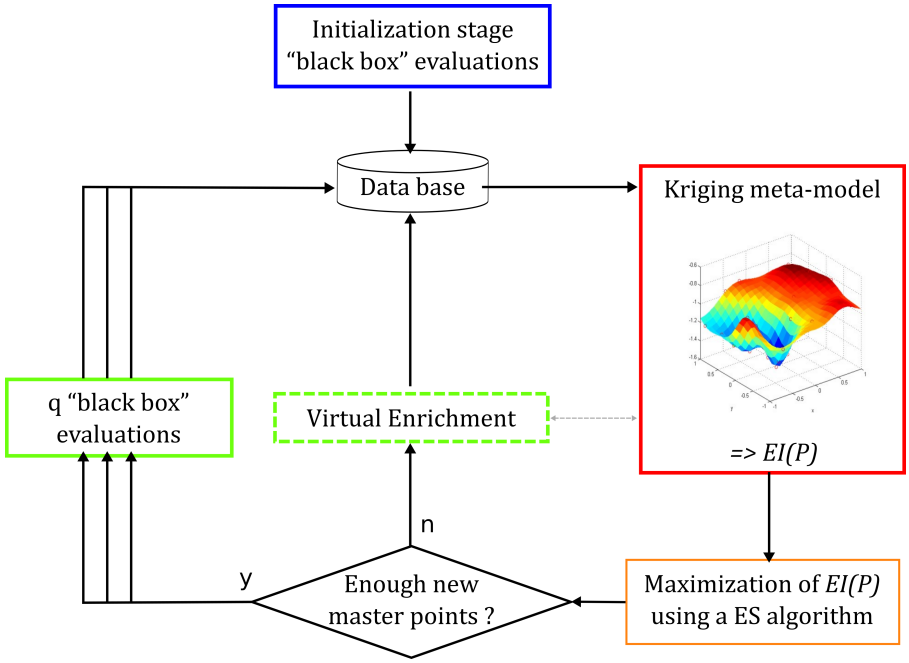


Fig. 1. Flow chart of the parallel procedure

In Algorithm 3, the q points of interest are extracted and a virtual enrichment is used. However, if one of these virtual enrichments is wrong, the following extracted point can be inaccurate. In Algorithm 3, two iterative processes are slotted together. The first one is the global optimization process (line 4 to line 17 in Algorithm 3), and the second one is the selection of the new q points of interest (line 7 to line 13 in Algorithm 3) that allows us to approach the *q-EI* criterion.

The most important step of this procedure is the virtual setting of the objective function (Algorithm 3, line 11). Different functions f^* have been presented in the literature. In this paper, two approaches are tested and compared: Kriging Believer (KB) and Constant Liar (CL).

Algorithm 3: Global optimization method – parallel extension

1:	$i = 1$	
2:	$DBp = \text{init}()$	// $DBp \in X$
3:	$DBy = f(DBp)$	// *"black box" function call*
4:	While $(i < i\,\text{max})$ do	
5:	$DBy^* = 0$	
6:	$DBp^* = 0$	
7:	For $j = 1$ to $j = q$	
8:	$\text{Meta}^* = \text{MetaModeling}(DBp \cup DBp^*, DBy \cup DBy^*)$	// *meta-model generation and calibration*
9:	$newDBp = \min(\text{Meta}^*)$	// *exploration and exploitation*
10:	$DBp^* = [DBy^* \cup newDBp]$	// *data enrichment*
11:	$y^* = f^*(newDBp)$	// *virtual setting of master point objective function*
12:	$DBy^* = [DBy^* \cup y*]$	// *virtual data enrichment*
13:	End	
14:	$DBy = [DBy \cup f(DBp^*)]$	// *"black box" function call*
15:	$DBp = [DBp \cup DBp^*]$	
16:	$i = i + q$	
17:	End	

2.5.1. Kriging Believer

The Kriging Believer (KB) approach uses the Kriging predictor (Eq. (9)). Kriging Believer sets the virtual value to the most probable value of the objective function according to the Kriging meta-model.

$$f^*(x) = \hat{y}(x). \tag{16}$$

This formulation seems to be the most appropriate, but it may be inaccurate if the meta-model predicts an objective function value that is smaller than all the objective function values that have already been computed.

2.5.2. Constant Liar

Another way to define the virtual objective function is to assign it a constant value y_L. This method is called the Constant Liar (CL).

$$f^*(x) = y_L\,. \tag{17}$$

Different choices can be made but the user has to define the value of the constant y_L:

$$y_L = \min_{i \in [1,n]} (y_i), \qquad y_L = \max_{i \in [1,n]} (y_i), \qquad y_L = \text{mean}_{i \in [1,n]} (y_i).$$

These different values are denoted as CLmin, CLmax, and CLmean, respectively, in the next part of the paper. Large values of y_L lead to a more global search, whereas smaller values lead to a more local search.

In Fig. 2, the three steps of the parallel procedure are depicted:

- Init. Step (initialization step): 4 master points are known, the Kriging meta-model has been built and calibrated (Fig. 2a) and the Expected Improvement (EI) has been computed (Fig. 2b). The maximum of the EI is identified at $x = 0.4$ (green point).
- Virtual enrichment step: a new master point is defined at $x = 0.4$ (where EI is found to be maximum at the previous step), and the value of the new master point is set to a virtual value according to the meta-model prediction (the KB approach is used here). With this new database of master points (which includes real master points and one virtual point, in green on the graph Fig. 2c), the Kriging meta-model is updated and a new maximum of the EI is found at $x = -0.75$ (green point in Fig. 2d). This point becomes a new master point.
- Final step: at this stage, two new master points have been proposed by the method, two calls of the black-box function can run in parallel to evaluate the cost function exactly for these two new points (green squares in Fig. 2e). The database of master points is now made of 6 points (Fig. 2e).

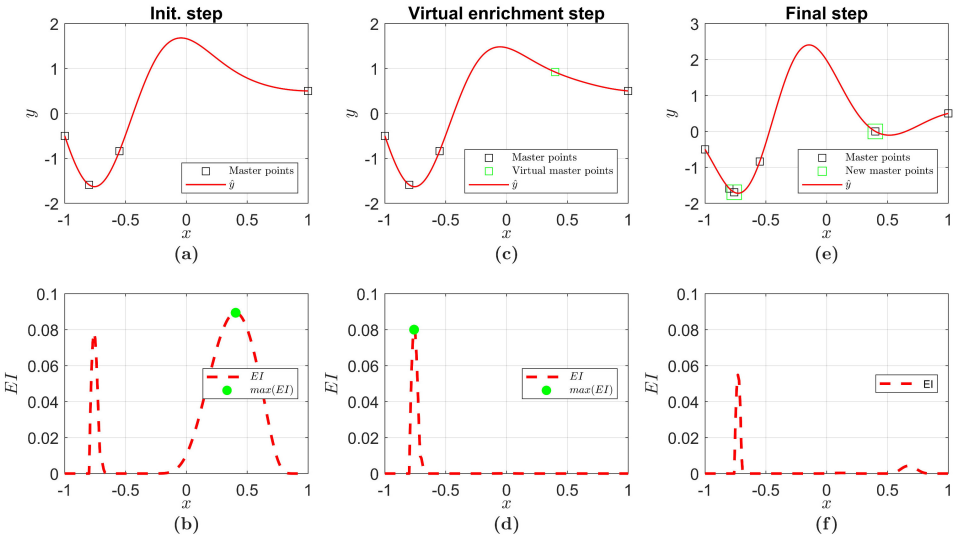


Fig. 2. Illustration of the parallel procedure on a simple one-dimensional case

Note that the virtual enrichment step can be repeated q times to propose q new points of interest to be computed in parallel, instead of two in this example.

2.6. Stopping criteria

Parameter identification problems lead to finding a minimum value of the objective function, which tends towards zero. A natural stopping criterion is to reach a minimum value set by the user. However, in some cases, when the mapping of the objective function is accurate, the use of sampling criteria (*EI*) leads to additional master points that are very close. Indeed, the correlation factor that is associated with two points too close to each other is 1 (Eq. (8)). This correlation factor has a dramatic impact on the Kriging strategy, since the correlation matrix **C** becomes singular.

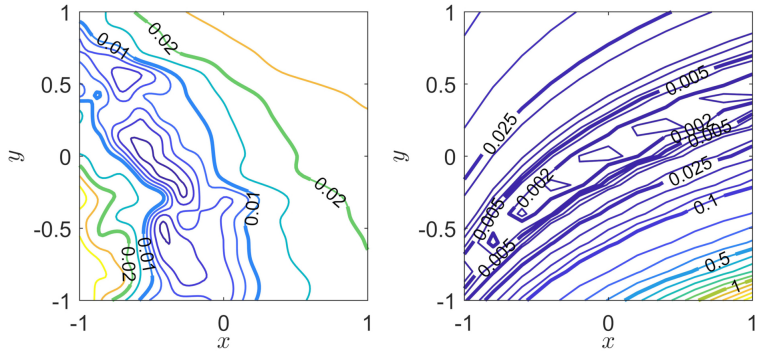
In order to build a more robust methodology, other stopping criteria must be used. Jones et al. [15] suggested stopping the EGO algorithm when *EI* is lower than 0.01. These stopping criteria do not account for the objective function value. It follows that, in order to address parameter identification issues, a local improvement may be necessary to achieve a satisfactory objective function value and to get an accurate value of the identified parameters. Such an improvement could be obtained by chaining the EGO algorithm with a secondary optimization algorithm using a classical local method such as the Nelder-Mead or BFGS algorithm [4, 31]. The starting point for this secondary optimization would be set to the EGO optimal solution. This strategy would ensure both a global and local search and thus a better accuracy of the final optimal solution.

3. Assessment of the methodology on analytical function

3.1. Definition of representative analytical functions

In order to check the efficiency of the optimization procedure, some analytical test functions are used. These test functions are chosen to be representative of parameters identification problems. In order to make this choice, two landscapes of objective functions resulting from material parameters identification are presented in Fig. 3. The first landscape is an objective function corresponding to the identification of ductile damage parameters and the second landscape is an objective function relating to the identification of hardening law parameters. These two landscapes present pathological issues commonly observed when identifying material parameters for material forming applications. The first case (Fig. 3a) is representative of a multi minima problem. Multi extrema may occur due to the softening mechanical behavior, which is a consequence of the coupling between damage growth and the mechanical material behavior. In this case, the use of a single global observable (load-displacement curve) is not sufficient to ensure the uniqueness of the minimization problem. The second case (Fig. 3b) is a typical

ill-conditioned problem. This is either due to a low sensitivity of one parameter on the observable, or to a strong correlation between the two parameters. This kind of correlation issue can typically occur between the hardening exponent and the material's consistency [32].



(a) (b)

Fig. 3. (a) Objective function 1 from damage parameters identification problems; (b) Objective function 2 from hardening parameters identification problems

In order to assess the efficiency of the procedure, instead of using finite element models that are time consuming, we have decided to use analytical function that are representative of such pathological issues. These functions will be used as the "black box" in the optimization procedure.

3.1.1. Modified Branin-Hoo function:

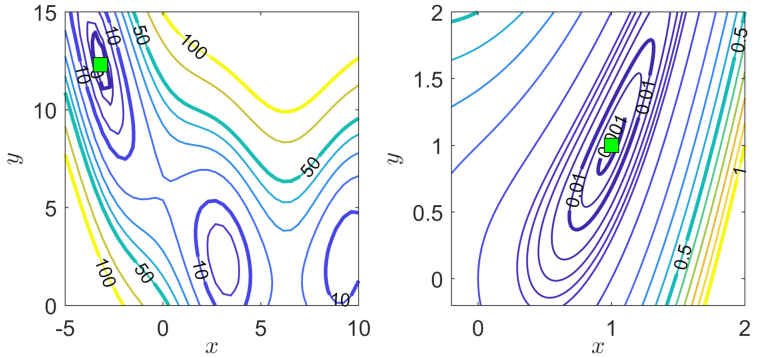
$$f(x,y) = \left(y - \frac{5.1}{4\pi^2}x^2 + \frac{5}{\pi}x - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x) + 10 + 0.5x + 1.17, \qquad (x,y) \in [-5,10] \times [0,15]\,. \tag{18}$$

This Modified Branin-Hoo function comes from Huang's work [33]. It has 2 local minima at (3.14, 2.27) and (9.42, 2.47). The global minimum is located at (−3.14, 12.27) and is equal to 0. The landscape of the function is presented in Fig. 4a. It is close to the one presented in Fig. 3a.

3.1.2. Adapted Rosenbrock function:

$$f(x) = \alpha \sum_{i=1}^{d/2} \left[(1 - x_{2i})^2 + \left(x_{2i+1} - x_{2i}^2\right)^2\right], \quad x \in [-0.2, 2]^d. \tag{19}$$

The Rosenbrock function [34] is a d dimensional function that has one minimum at $x_{i\in[1,d]} = 1$, equal to 0. This is a typical ill-conditioned problem. In order



(a) (b)

Fig. 4. (a) Modified Branin-Hoo function (Eq. (19)); (b) 2D adapted Rosenbrock function (Eq. (20))

to be close to the landscape presented in Fig. 3b, which corresponds to hardening parameters identification, the value of α, a parameter that can be uses to tune the Rosenbrock function, was set to 0.1 (Fig. 4b).

3.1.3. Hartmann 6D function

$$H_{6,4} = -\sum_{i=1}^{4} \alpha_i \exp\left[-\sum_{j=1}^{6} A_{ij}\left(x_j - P_{ij}\right)^2\right] + 3.32237, \quad x_j \in [0,1],$$

$$\alpha = \begin{bmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{bmatrix}, \quad A = \begin{bmatrix} 10 & 3 & 17 & 3.05 & 1.7 & 8 \\ 0.05 & 10 & 17 & 0.1 & 8 & 14 \\ 3 & 3.5 & 1.7 & 10 & 17 & 8 \\ 17 & 8 & 0.5 & 10 & 0.1 & 14 \end{bmatrix}, \tag{20}$$

$$P = 10^{-4} \begin{bmatrix} 1312 & 1696 & 5569 & 124 & 8283 & 5886 \\ 2329 & 4135 & 8307 & 3736 & 1004 & 9991 \\ 2348 & 1451 & 3522 & 2883 & 3047 & 6650 \\ 4047 & 8828 & 8732 & 5743 & 1091 & 381 \end{bmatrix}.$$

The Hartmann 6D function has 6 local minima and one global minimum at $x = (0.20, 0.15, 0.47, 0.27, 0.31, 0.65)$, which is equal to 0. The Hartmann 6D function is an eight dimensional problem, which is ill-conditioned. Because of its multiple minima, it is therefore an interesting function to test the efficacy of the proposed parallel EGO algorithm.

These functions will be used as "black box" functions to evaluate and discuss the robustness and efficiency of the procedure that is presented in this work.

3.2. Isotropic or anisotropic ordinary Kriging

Kriging meta-model building can be done using either isotropic or anisotropic correlation functions (8). Anisotropic correlation leads to a more complex calibration phase: an auxiliary function of dimension d needs to be minimized (Eq. (11)), instead of a 2D auxiliary function with isotropic Kriging. A test is performed to check the efficiency of anisotropic Kriging for parameters identification. Table 1 shows the results of this comparison in terms of the number of iterations for each of analytical functions presented above. The algorithm is stopped when the target value specific to each function is reached (Table 1). It can be seen that anisotropic Kriging finds the minimum with a lower number of iterations than isotropic Kriging. The benefit in terms of reduction of the number of iterations is between 14% and 26%.

Table 1.
Isotropic and anisotropic correlation – number of "black box" function calls for a fixed convergence goal

	Stopping criteria value	Number of iterations		Anisotropic benefit
		Isotropic	Anisotropic	
Branin-Hoo	10e-4	27	23	14%
Rosenbrock – 2D	10e-4	18	14	22%
Rosenbrock – 4D	10e-3	71	54	24%
Hartmann – 6D	5e-3	54	40	26%

However, the cost dedicated to calibration is slightly higher for anisotropic Kriging. This additional cost was evaluated by measuring the CPU time for the Hartman 6D function. For a database of 50 points, isotropic Kriging takes 0.031 s, while anisotropic Kriging takes 0.37 s. Therefore, there is a factor of 10 in terms of additional costs. However, the absolute duration of anisotropic Kriging remains very low compared to the time spent for the "black box" function calculation. This is particularly true when using finite element computations.

The use of an anisotropic correlation for Kriging thus induces a significant reduction of the CPU time for a given optimization problem.

3.3. Improvement of the parallel strategy

3.3.1. Speed-up test

To evaluate the improvement of the parallel strategy, a similar approach to the one used for the improvement of parallel computing is followed. The reference unit is not the computation time, as for conventional parallel computation, but the

number of "black box" function calls. The speed-up S_p is defined in Eq. (21)

$$S_p = \frac{T_s}{T_p} . \tag{21}$$

p is the number of parallel tasks. In our case, p is the number of "black box" functions evaluated simultaneously. T_s is the number of sequential iterations, i.e., the number of "black box" function calls during a sequential procedure. Finally, T_p is the total number of parallel iterations for p processors. Thus, the total number of "black box" function calls is equal to $T_p p$. The speed-up is calculated without taking into account the initialization stage (which is negligible with respect to the total duration of the optimization).

As an example, if a given problem is solved with the sequential version of the algorithm in 30 iterations ($T_s = 30$), the same problem is solved with parallel version in 15 iterations ($T_p = 15$). Therefore, the obtained speed-up is $S_p = 2$.

The number of parallel tasks p is set to 6 values $p \in [1, 2, 3, 4, 8, 16]$. Two different types of virtual enrichments are tested: KB Eq. (16), and CLmin Eq. (17). An anisotropic correlation is used because of the results of the previous section.

The results obtained with the CLmax criterion are not presented. This criterion leads to bad speed-ups at best, or at worst no results are obtained. This can be explained by the introduction of a pessimistic point in the master points database during the virtual enrichment phase. Two close points can have very different values (minimum and maximum of the database in the case of the CLmax criterion). This large difference leads to the use of small calibration parameters in the correlation function (8) and to the construction of a meta-model, which looks like a peak function around master points.

The results of the KB and CLmin criteria are presented in Fig. 5. The theoretical "goal"-speed-up, which corresponds to $S_p = p$, is represented by the large red dash line in all graphs.

Several remarks can be made regarding the speed-up results:

- For both criteria, the speed-up is always greater than one. This means that the time required to reach the optimal solution using the parallel strategy is always shorter than that of the sequential strategy.
- The analysis of the speed-up results on the different functions does not allow one to conclude that one enrichment method (KB or CLmin) is better than the other. In fact, no clear trend can be observed. Both criteria lead to results that are very close to each other.
- The CLmin criterion shows a decreasing speed-up on the Hartmann 3D function (Fig. 5d, $p = 8$). This decreasing speed-up is problematic for the overall efficiency of the method: the increase in the number of parallel tasks does not always lead to a reduction in the duration of resolution minimization. But this trend is never observed on other functions.
- In some cases, a speed-up greater than the theoretical speed-up may be observed. This means that the number of "black box" function calls in

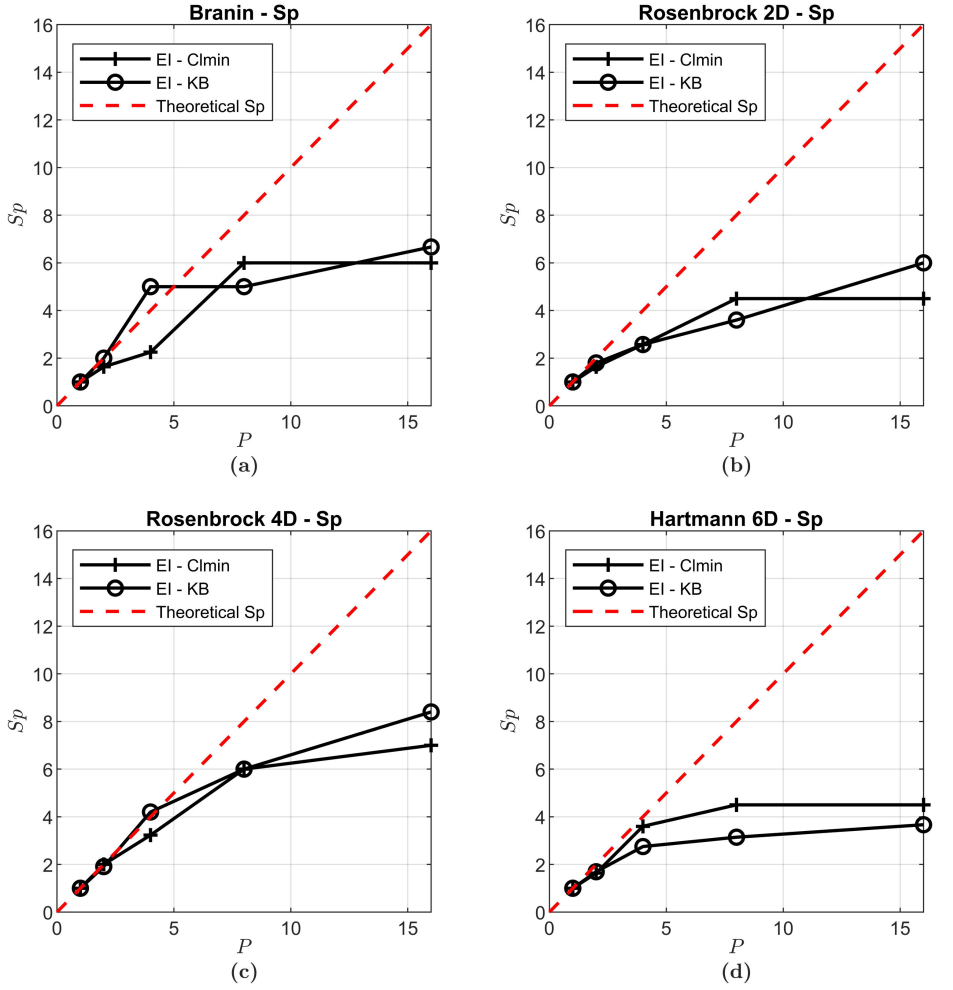


Fig. 5. Speed-up: (a) Modified Branin-Hoo function; (b) Rosenbrock 2D function; (c) Rosenbrock 4D function; (d) Hartmann 6D function

parallel strategy is lower than the number of "black box" function calls under sequential strategy. It also proves the efficiency of the developed method in some specific cases. This interesting behavior is observed for the modified Branin-Hoo function (Fig. 5a) and for the Hartmann 3D function (Fig. 5d) for low p values ($p = 2$ and $p = 4$). This behavior can be explained by the accuracy of the Kriging predictor for small dimensional problems on these functions.

- For all four functions, the increase in speed-up is very interesting for low values of p. In this cases, the measured speed-up is very close to the theoretical one. For high values of p, the difference between the measured speed-up and the theoretical one increases. The parallel enrichment method

becomes therefore less efficient. This loss of efficiency is clearly observed with the CLmin criterion in connection with the modified Branin-Hoo function (Fig. 5a), the Rosenbrock 2D function (Fig. 5b) and the Hartman 6D functions (Fig. 5d). In fact, speed-up values remain unchanged while p increases from 8 to 16. A maximum speed-up seems to be reached for these 3 functions.

3.3.2. Influence of the initial size of the data base

In previous tests, the databases of master points were always initialized with a three level full factorial design plan (i.e., a regular 3-by-3 grid of points) for the modified Branin-Hoo function. In this section, the influence of the size of the initial plan on the speed-up is studied. Two additional full factorial plans are tested [35]: a two-levels full plan (i.e., a regular 2-by-2 grid of points), and a four-levels full plan (i.e., a regular 4-by-4 grid of points). For these tests the KB virtual enrichment is used. The speed-up is calculated thanks to (Eq. (21)). The *Ts* values are presented in Table 2. It appears that the use of different initial data base sizes leads to different convergence rates for the method. The larger the size of the initial database, the smaller the number of iterations required for convergence.

Table 2.

Number of sequential iterations *Ts* to solve the minimization problem for different initial database sizes

Full plan size	*Ts*
2 levels	23
3 levels	17
4 levels	16

The speed-up results are shown in Fig. 6. Some remarks can be made:

- For all three database sizes, the parallel algorithm is always efficient. In fact, the speed-up is always greater than 1.
- The three-levels plan and two-levels plan shows good and equivalent speed-up. Therefore, using a larger initial database may not be the best option. In fact, it seems that the four-levels plan has a lower speed-up than the three-levels plan.

It can be seen that the choice of the initial data base has an impact on the speed-up. The tests on the modified Branin-Hoo function show that a full three-levels plan is a good choice for this function. The decrease in efficiency with the increase in the initial database size can be explained. Indeed, the points added during the minimization procedure are placed by maximizing the expected improvement, whereas the full plans are not optimal regarding the expected improvement criterion. Therefore, non-optimal points coming from the initial plan can have a negative influence on the speed-up.

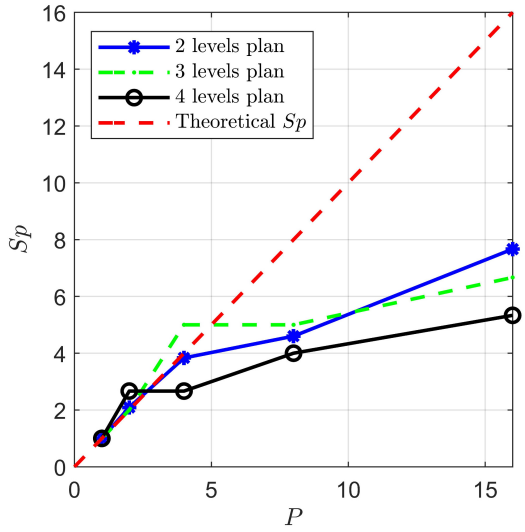


Fig. 6. Speed-up on the Modified Branin-Hoo function – influence of the initial size of the data base

No general conclusion can be drawn from this test at this stage. Each problem can lead to a different optimal initial database size, but this optimal size cannot be known without a preliminary study (which is time consuming). Nevertheless, the use of a small initial database (2 or 3-levels plan) seems to be a satisfactory option.

4. Application to a realistic identification case

In this section, the methodology presented above is applied to a real case of identification. This example deals with the identification of ductile damage parameters based on a uniaxial tensile test. An aluminum material is tested; its grade is EN AW-5774[AlMg3]. The "black box" function is composed of a finite element model of the tensile test and of an objective function. In this section, the "black box" is detailed, and results of the identification are presented and discussed.

4.1. Black box description

To build the black box function, three parts are necessary: the experimental results, the finite element model and the definition of an objective function.

4.1.1. Experimental set-up and results

To study the behavior of the aluminum material, a uniaxial tensile test is carried out on a normalized sample. The resulting load/displacement curve is shown in Fig. 7 (black cross line). Note that experimental results can also be a combination

of global and local observables such as local strains or displacement fields obtained by digital image correlation [21].

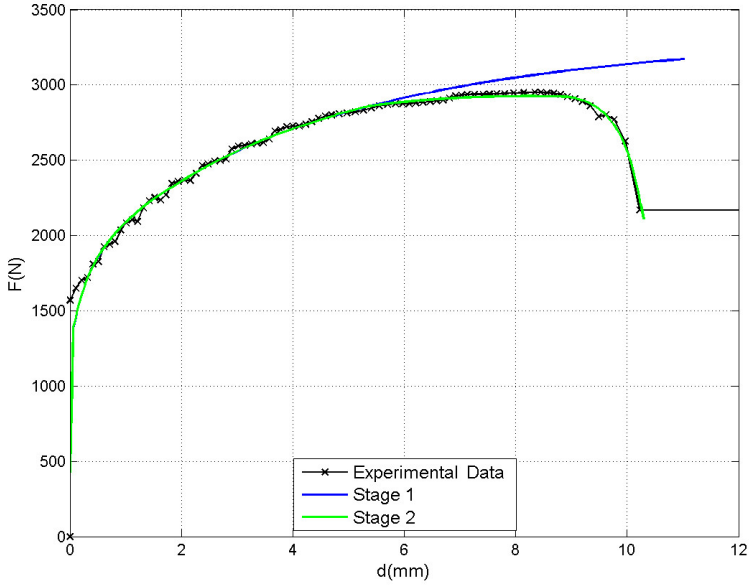


Fig. 7. Load/displacement curve – experimental data and identification results

4.1.2. Finite element model

The tensile test is modeled using the CIMLib®finite element library developed at CEMEF [36]. To describe the material behavior, an elastic-plastic law coupled with a Lemaitre ductile damage model was chosen. The main equations describing the elastic-plastic behavior and ductile Lemaitre damage model are given here:

Hardening law:

$$\sigma_0 = \sigma_y + K\bar{\varepsilon}^h. \tag{22}$$

Damage evolution law:

$$\dot{w} = \begin{cases} 0 & \text{if } \bar{\varepsilon} < \bar{\varepsilon}_d \,, \\ \dfrac{\lambda^{pl}}{1-w}\left(\dfrac{-Y}{S_0}\right)^b & \text{if } \bar{\varepsilon} \geqslant \bar{\varepsilon}_d \,. \end{cases} \tag{23}$$

The flow stress σ_0 is described by a hardening law equation (22), based on the yield stress σ_y, the consistency K, the equivalent plastic strain $\bar{\varepsilon}$, and the hardening exponent h. The evolution law for the damage parameter w is given by Eq. (23), where $\bar{\varepsilon}_d$ is the plastic strain threshold for damage growth, λ^{pl} is the plastic multiplier, Y the strain energy release rate, S_0 and b are material damage parameters. In this model, damage is coupled with the material behavior which

allows us to model damage-induced softening. More details on this model and its numerical implementation are given in [37]. The objective of the identification is to determine the values of 6 material parameters, 3 for plastic hardening (K, σ_y, h) and 3 for the ductile damage model ($\bar{\varepsilon}_d$, b, S_0). The finite element model provides a numerical load-displacement curve for a given set of material parameters P.

4.1.3. Objective function

In order to evaluate the difference between the experimental and the numerical results, an objective function has to be defined. The formula is given by Eq. (24). F^{num} and F^{exp} are respectively the numerical and experimental forces, and *disp* is the tensile tool displacement. More details on this formulation and on how to deal with the softening part of the load-displacement curve can be found in [38].

$$\begin{cases} fc_f(p) = \sqrt{\dfrac{\sum\limits_i \left[F_i^{\text{num}}(p) - F_i^{\text{exp}}\right]^2 \Delta disp_i}{\min\left(\sum\limits_i \left[F_i^{\text{num}}(p)\right]^2 \Delta disp_i,\ \sum\limits_i \left[F_i^{\text{exp}}\right]^2 \Delta disp_i\right)}}, \\ \Delta disp_i = disp_i - \Delta disp_{i-1}, \end{cases} \tag{24}$$

where i denote each point of the data set.

4.2. Identification procedure

The identification of the material's behavior takes place in 2 steps. The first one is the identification of the plastic behavior described in Eq. (22). To run this first identification, the objective function is computed on the first part of the load/displacement curve ($0 < d < 6$ mm). During this identification the damage model is disabled. For each parameter, the search range is presented in Table 3, stage 1.

The second identification deals with the identification of damage parameters (Eq. (23)). For this second identification, the hardening parameters are set to the values identified at the previous step, and the objective function is computed using the full load/displacement curve. The search ranges of each parameter are presented in Table 3, stage 2.

Both identifications are performed using the parallel version of the EGO algorithm with anisotropic correlation function. Four "black box" functions are called in parallel for each iteration.

4.3. Identification results

The results are presented in Table 3, Fig. 7 and Fig. 8. The identified parameters, for both stages, are given in Table 3. The identified load/displacement curves are

Table 3.

Realistic identification case – parameters search range and identification results

Stage	Parameters	Search range	Identified parameters	Objective function	Number of "black box" function calls	Number of iterations
1	σ_y(MPa)	[10, 400]	93	0.77%	139	27
	K (MPa)	[200, 1000]	441			
	h	[0.1, 0.6]	0.44			
2	b	[0.5, 3]	1.00	0.77%	104	18
	S_0 (MPa)	[0.1, 3]	0.77			
	$\bar{\varepsilon}_d$	[0.04, 0.20]	0.097			

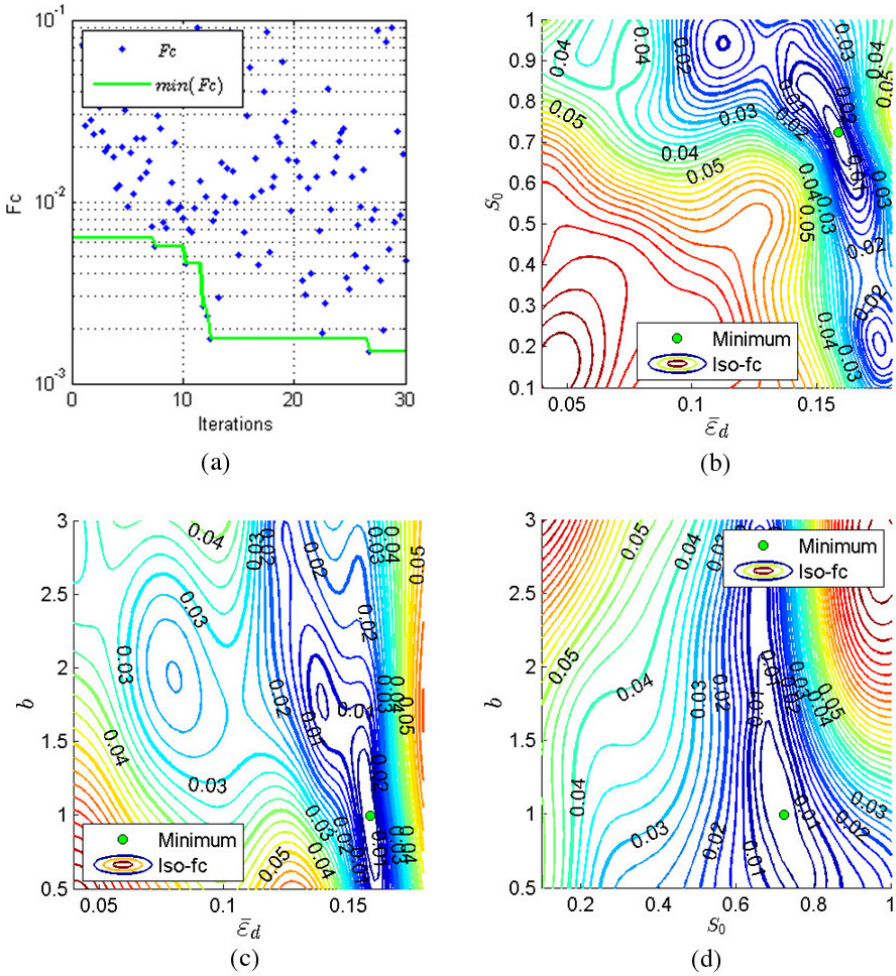


Fig. 8. Stage 2 – results of a realistic identification case: a) cost function vs. iterations diagram, b-c-d) final Kriging surface displayed on 3 cutting planes crossing the optimum solution

shown in Fig. 7, in which we observe that the experimental and identified curves match perfectly. This good result is in agreement with the final value of the objective function (Table 3). In fact, a value of 0.77% is reached for both stages.

5. Conclusion

A parallel extension of the EGO algorithm is presented to improve computation time. This parallel strategy can be used in a practical way to solve optimization problems in a limited time frame. The presented works aims at testing the *q-EI* criterion proposed in [19]. A sequential approximation of this criterion is developed in this work. The goal is to test its efficiency on parameter identification issues.

The parallel EGO optimization procedure has been investigated and tested on analytical functions (Eq. (20)–(22)), which have been chosen in order to be representative of material parameters identification issues and on a realistic case of material parameter identification.

The first tests on isotropic and anisotropic correlations show that anisotropic correlation can lead to a 20% reduction in CPU time. This result is very interesting regarding identification issues. Identification problems are often not very sensitive. The use of anisotropic correlation combined with an efficient calibration of the correlation parameters is a good way to overcome the issue of low sensitivity.

The second tests on the parallel extension of the EGO algorithm show that the Kriging Believer criterion (KB) and the minimum Constant Liar criterion (CLmin) give comparable results. For both criteria, the speed-up values are very interesting if the number of parallel objective function computations is low (lower than 4 or 8). A maximum speed-up value seems to appear on some specific functions, when the number of parallel objective function computations increases. This is probably due to the fact that the parallel procedure is based on new points, for which the exact value of the objective function has not yet been computed. In addition, these results are always validated when other correlation functions or initial database sizes are considered. In fact, the test on the modified Branin-Hoo function shows that the initial database size does not have a significant impact on the speed-up.

A third test is performed on a realistic identification case. The identified parameters give a numerical load-displacement curve in accordance with the experimental curve. The optimization algorithm shows its efficiency on a realistic application. Moreover, the Kriging surfaces are very interesting in order to evaluate the sensitivity of each identified parameter on the objective function. These surfaces can help defining more appropriate objective functions or new observable to improve the efficiency and accuracy of the identification procedure.

Manuscript received by Editorial Board, March 03, 2020;
final version, April 22, 2020.

www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

References

[1] P.A. Prates, M.C Oliveira, and J.V. Fernandes. Identification of material parameters for thin sheets from single biaxial tensile test using a sequential inverse identification strategy. *International Journal of Material Forming*, 9:547–571, 2016. doi: 10.1007/s12289-015-1241-z.

[2] M. Gruber, N. Lebaal, S. Roth, N. Harb, P. Sterionow, and F. Peyraut. Parameter identification of hardening laws for bulk metal forming using experimental and numerical approach. *International Journal of Material Forming* 9:21–33. doi: 10.1007/s12289-014-1196-5.

[3] R. Amaral, P. Teixeira, A.D. Santos, and J.C. de Sá. Assessment of different ductile damage models and experimental validation. *International Journal of Material Forming* 11, 435–444, 2018. doi: 10.1007/s12289-017-1381-4.

[4] J. Nocedal and S. Wright. *Numerical Optimization*, 2nd ed. Springer-Verlag, New York, 2006.

[5] J.A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965. doi: 10.1093/comjnl/7.4.308.

[6] K.Y. Lee and F.F. Yang. Optimal reactive power planning using evolutionalry algorithms: a comparative study for evolutionary programming, evolutionary strategy, genetic algorithm, and linear programming. *IEEE Transactions on Power Systems*, 13(1):101–108, 1998. doi: 10.1109/59.651620.

[7] N. Stander, K.J. Craig, H. Müllerschön, and R. Reichert. Material identification in structural optimization using response surfaces. *Structural and Multidisciplinary Optimization*, 29:93–102, 2005. doi: 10.1007/s00158-004-0476-y.

[8] M. Ageno, G. Bolzon, and G. Maier. An inverse analysis procedure for the material parameter identification of elastic- plastic free-standing foils. *Structural and Multidisciplinary Optimization*, 38:229–243, 2009. doi: 10.1007/s00158-008-0294-8.

[9] M. Abendroth and M. Kuna. Identification of ductile damage and fracture parameters from the small punch test using neural networks. *Engineering Fracture Mechanics*, 73(6):710–725, 2006. doi: 10.1016/j.engfracmech.2005.10.007.

[10] R. Franchi, A. Del Prete, and D. Umbrell. Inverse analysis procedure to determine flow stress and friction data for finite element modeling of machining. *International Journal of Material Forming*, 10:685–695, 2017. doi: 10.1007/s12289-016-1311-x.

[11] N. Souto, A. Andrade-Campos, and S. Thuillier. Mechanical design of a heterogeneous test for material parameters identification. *International Journal of Material Forming*, 10:353–367, 2017. doi: 10.1007/s12289-016-1284-9.

[12] M. Rackl, K.J. Hanley, and W.A. Günthner. Verification of an automated work flow for discrete element material parameter calibration. In: Li X., Feng Y., Mustoe G. (eds.), *Proceedings of the 7th International Conference on Discrete Element Methods. DEM 2016*, volume 188, pages 201–208. Springer, Singapore 2017. doi: 10.1007/978-981-10-1926-5_23.

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

[14] C. Richter, T. Rößler, G. Kunze, A. Katterfeld, and F. Will. Development of a standard calibration procedure for the DEM parameters of cohesionless bulk materials – Part II: Efficient optimization-based calibration. *Powder Technology*, 360:967–976, 2020. doi: 10.1016/j.powtec.2019.10.052.

[15] D.R. Jones, M. Schonlau, and W.J. Welch. Efficient global optimization of expensive black-box function. *Journal of Global Optimization*, 13:455–492, 1998. doi: 10.1023/A:1008306431147.

[16] M.T.M. Emmerich, K.C. Giannakoglou, and B. Naujoks. Single- and multi-objective evolutionary optimization assisted by gaussian random field metamodels. *IEEE Transactions on Evolutionary Computation*, 10(4):421–439, 2006. doi: 10.1109/TEVC.2005.859463.

[17] T.J. Santner, B.J. Williams, and W.I. Notz. *The Design and Analysis of Computer Experiments*. Springer, New York, 2018. doi: 10.1007/978-1-4939-8847-1.

www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

[18] E. Brochu, V.M. Cora, and N. de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. Technical Report TR-2009-23, Department of Computer Science, University of British Columbia, Canada, November 2009.

[19] D. Ginsbourger, R. Riche, and L. Carraro. Kriging is well-suited to parallelize optimization. In Y. Tenne, Ch-K. Goh (eds.), *Computational Intelligence in Expensive Optimization Problems*, volume 2, pages 131–162, Springer-Verlag, Berlin, 2010.

[20] E. Roux and P.-O. Bouchard. Kriging metamodel global optimization of clinching joining processes accounting for ductile damage. *Journal of Materials Processing Technology*, 213(7):1038–1047, 2013. doi: 10.1016/j.jmatprotec.2013.01.018.

[21] E. Roux and P.-O. Bouchard. On the interest of using full field measurements in ductile damage model calibration. *International Journal of Solids and Structures*, 72:50–62, 2015. doi: 10.1016/j.ijsolstr.2015.07.011.

[22] J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423, 1989.

[23] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer M. et al. (eds,), *Parallel Problem Solving from Nature PPSN VI. PPSN 2000*. Lecture Notes in Computer Science, volume 1917, pages 849–858. Springer, Berlin, Heidelberg, 2000. doi: 10.1007/3-540-45356-3_83.

[24] M. Hamdaoui, F.Z. Oujebbour, A. Habbal, P. Breitkopf, and P. Villon. Kriging surrogates for evolutionary multi-objective optimization of CPU intensive sheet metal forming applications. *International Journal of Material Forming*, 8:469–480, 2015. doi: 10.1007/s12289-014-1190-y.

[25] J.J. Droesbeke, M. Lejeune, and G. Saporta. *Statistical Analysis of Spatial Data*. Editions TECHNIP, 1997 (in French).

[26] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[27] J.D. Martin and T.W. Simpson. Use of kriging models to approximate deterministic computer models. *AIAA Journal*, 43(4):853–863, 2005. doi: 10.2514/1.8650.

[28] J. Laurenceau and P. Sagaut. Building efficient response surface of aerodynamic function with kriging and cokriging. *AIAA Journal* 46(2):498–507, 2008. doi: 10.2514/1.32308.

[29] D.R. Jones. A taxonomy of global optimization methods based on response surfaces. *Journal of Global Optimization*, 21:345–383, 2001. doi: 10.1023/A:1012771025575.

[30] V.A. Dirk and H.G. Beyer. A comparison of evolution strategies with other direct search methods in the presence of noise. *Computational Optimization and Applications*, 24:135–159, 2003. doi: 10.1023/A:1021810301763.

[31] J.A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1965. doi: 10.1093/comjnl/7.4.308.

[32] P.-O. Bouchard, J.-M. Gachet, and E. Roux. Ductile damage parameters identification for cold metal forming applications. *AIP Conference Proceedings*, 1353(1):47–52, 2011. doi: 10.1063/1.3589490.

[33] D. Huang, T.T. Allan, W.I. Notz, and N. Zeng. Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization*, 34:441–466, 2006. doi: 10.1007/s10898-005-2454-3.

[34] H.H. Rosenbrock. An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3):175–184, 1960. doi: 10.1093/comjnl/3.3.175.

[35] M. Pillet. *The Taguchi Method Experiment Plans*. Les Edition d'Organisation, 2005 (in French).

[36] H. Digonnet, L. Silva, and T. Coupez. Cimlib: A Fully Parallel Application For Numerical Simulations Based On Components Assembly. *AIP Conference Proceedings*, 908:269–274, 2007. doi: 10.1063/1.2740823.

[37] P.-O. Bouchard, L. Bourgeon, S. Fayolle, and K. Mocellin. An enhanced Lemaitre model formulation for materials processing damage computation. *International Journal of Material Forming*, 4:299–315, 2011. doi: 10.1007/s12289-010-0996-5.

[38] E. Roux, M. Thonnerieux, and P.-O. Bouchard. Ductile damage material parameter identification: numerical investigation. In *Proceedings of the Tenth International Conference on Computational Structures Technology*, paper 135, Civil-Comp Press, 2010. doi: 10.4203/ccp.93.135.