

# Cokrigage et stratégie multiparamétrique pour l'optimisation d'assemblages

Luc Laurent, Pierre-Alain Boucard, Bruno Soulier

► **To cite this version:**

Luc Laurent, Pierre-Alain Boucard, Bruno Soulier. Cokrigage et stratégie multiparamétrique pour l'optimisation d'assemblages. 20ème Congrès Français de Mécanique (CFM 2011), Aug 2011, Besançon, France. hal-01431907

**HAL Id: hal-01431907**

**<https://hal.archives-ouvertes.fr/hal-01431907>**

Submitted on 19 Jan 2017

**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.

# Cokrigage et stratégie multiparamétrique pour l'optimisation d'assemblages

L. LAURENT<sup>a</sup>, P.A. BOUCARD<sup>a</sup> & B. SOULIER<sup>a</sup>

a. LMT-Cachan, (ENS Cachan/CNRS/Université Paris 6/PRES UniverSud Paris), 61 av. du Président Wilson, F-94230 Cachan, France

## Résumé :

Lorsque l'on cherche à optimiser des assemblages de structures, le coût des calculs permettant de trouver le minimum d'une fonction objectif est très important en raison des fortes non-linéarités présentes dans les calculs (problème de frottement et/ou de contact). L'approche proposée s'appuie sur une stratégie d'optimisation multi-niveaux de modèles utilisant un métamodèle de cokrigage, construit à partir des évaluations de la fonction objectif et de ses gradients, et un modèle mécanique complet basé sur l'algorithme itératif LATIN. Le modèle mécanique exploite une stratégie multiparamétrique assurant, grâce à une procédure de réinitialisation de l'algorithme, des gains considérables en terme de coût de calcul. Ce papier se concentre sur l'étude de la mise en œuvre du métamodèle à partir des données issues du modèle mécanique.

## Abstract :

In the context of optimization of assemblies, the number of calculations to find the optimum of an objective function is important. Moreover each calculation is expensive in terms of CPU time due to the nonlinearities (friction and/or contact problems). The proposed strategy is based on a multilevel model optimization using a cokriging metamodel and a specific strategy to solve mechanical problem. The cokriging metamodel is built not only with the responses of the objective function but also with the gradients. The mechanical model is based on a multiparametric strategy using the LATIN iterative algorithm. The reinitilization of the algorithm with previous solution ensures significant CPU time reducing. This article focuses on the building of the metamodel with the data provided by the mechanical model.

**Mots clefs :** optimisation multiniveaux, cokrigage, méthode LATIN

## 1 Introduction

Les processus d'optimisation nécessitent de nombreuses évaluations pour parvenir à trouver l'optimum du système étudié. Dans un contexte d'assemblage présentant de fortes non linéarités dues aux problèmes de frottement et/ou de contact, ces évaluations peuvent être très coûteuses ; c'est pourquoi il est indispensable de développer une stratégie dédiée destinée à réduire les temps de calcul. La solution proposée se base sur une stratégie d'optimisation multiniveaux de modèles constituée de deux niveaux et faisant intervenir deux procédés : un métamodèle de cokrigage et une stratégie de résolution multiparamétrique.

## 2 Optimisation multiniveaux de modèles

La stratégie d'optimisation multiniveaux de modèles [11] se décompose en deux niveaux (cf. figure 1) : le premier est constitué d'un métamodèle construit avec un nombre limité de données et le second par le modèle mécanique complet. Le principe d'optimisation se décompose alors en deux phases : la première repose sur l'utilisation d'un algorithme exploratoire qui, en exploitant le métamodèle, permet de localiser les zones potentielles où peut(vent) se trouver le(s) minimum(as) de la fonction étudiée (fonction objectif par exemple). Ces informations alimentent ensuite le second niveau où une optimisation est réalisée directement à l'aide du modèle mécanique complet afin de localiser précisément le minimum. Lors du processus complet, les coûts de

calcul proviennent de deux aspects : la construction du métamodèle à partir des réponses du modèle mécanique complet et l'optimisation directe sur le modèle mécanique complet. La stratégie proposée ici permet de réduire le coût de calcul engendré par ces deux phases, et seule la première sera ici présentée.

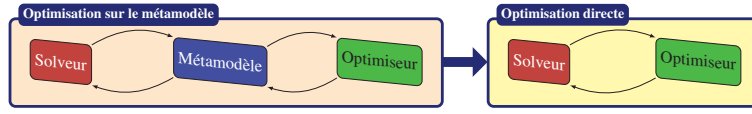


FIGURE 1 – Optimisation multiniveaux de modèles

### 3 Métamodèle de cokrigage

Le métamodèle choisi pour former le premier niveau de la stratégie d'optimisation est un métamodèle de type krigeage introduit par D. Krige [5] et développé par G. Mathéron [9, 10]. La méthode de cokrigage provient de la géostatistique multivariable [4, 12]. Elle permet de construire une approximation d'une fonction étudiée à partir de la valeur de plusieurs variables en des points échantillonnés. Dans le contexte présenté ici, le métamodèle sera construit non seulement à partir des évaluations de la fonction, mais également à partir de ses gradients. En effet, comme il sera présenté au paragraphe 3b, l'usage de la stratégie multiparamétrique permet d'obtenir à moindre coût les gradients d'une quantité d'intérêt du modèle mécanique étudié.

#### 3.1 Notations

Dans cette partie, les notations suivantes seront employées :  $\mathbf{x}^{(i)}$  et  $\mathbf{x}^{(0)}$  désignent des points de l'espace de conception  $\mathcal{D}$  respectivement un des  $n_s$  points échantillonnés et un point quelconque de  $\mathcal{D}$ ,  $x_i$  désigne la  $i^{\text{ème}}$  composante d'un point  $\mathbf{x}$ ,  $Y(\mathbf{x})$  et  $\tilde{Y}(\mathbf{x})$  désignent respectivement la réponse d'une fonction analytique ou d'un modèle mécanique et la réponse approchée fournie par le métamodèle.  $R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  est une fonction de corrélation.

#### 3.2 Principe

La formulation du métamodèle repose sur une décomposition sous la forme d'une somme de deux termes : le premier  $\mu(\mathbf{x})$  représente la tendance des données (fonction de régression déterministe classique) et la seconde  $Z(\mathbf{x})$  est un processus stochastique représentant l'erreur de prédiction du modèle de tendance.

$$\tilde{Y}(\mathbf{x}^{(0)}) = \mu(\mathbf{x}^{(0)}) + Z(\mathbf{x}^{(0)}), \quad \forall \mathbf{x}^{(0)} \in \mathcal{D} \quad (1)$$

$$\mathbb{E}(Z) = 0, \quad \text{Cov}[Z(\mathbf{x}^{(i)}), Z(\mathbf{x}^{(j)})] \neq 0, \quad \forall (\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \in \mathcal{D}^2 \quad (2)$$

où  $\mathbb{E}$  et  $\text{Cov}$  désignent les classiques espérance et covariance statistiques.

Le principe du cokrigage est de considérer une relation de covariance entre les réponses d'une fonction déterministe (on considérera dans ce papier les réponses du modèle mécanique comme déterministes) et les gradients aux différents points échantillonnés. Les relations de covariance dépendent seulement de la distance entre les échantillons. Le processus stochastique est alors représenté par un processus aléatoire stationnaire du second ordre à moyenne nulle.

La structure de covariance utilisée pour le cokrigage s'écrit sous la forme d'une relation de covariance faisant intervenir la variable primaire (la réponse du modèle mécanique ou de la fonction analytique) qui n'est autre que la structure de covariance du krigeage (équation 3) et des relations (équations 4-5-6) faisant intervenir les variables primaire et secondaires (les variables secondaires dans notre cas sont les gradients de la fonction étudiée).

$$\forall (\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \in \mathcal{D}^2, \quad \text{cov} [Z(\mathbf{x}^{(i)}), Z(\mathbf{x}^{(j)})] = \sigma^2 R(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sigma^2 c_{ij} \quad (3)$$

$$\text{Cov} \left[ \frac{\partial Z}{\partial x_k}(\mathbf{x}^{(i)}), Z(\mathbf{x}^{(j)}) \right] = -\sigma^2 \frac{\partial R}{\partial x_k}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sigma^2 c_{kij} \quad (4)$$

$$Cov \left[ Z(\mathbf{x}^{(i)}), \frac{\partial Z}{\partial x_k}(\mathbf{x}^{(j)}) \right] = -\sigma^2 \frac{\partial R}{\partial x_k}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sigma^2 c_{kij} \quad (5)$$

$$Cov \left[ \frac{\partial Z}{\partial x_k}(\mathbf{x}^{(i)}), \frac{\partial Z}{\partial x_l}(\mathbf{x}^{(j)}) \right] = -\sigma^2 \frac{\partial^2 R}{\partial x_k \partial x_l}(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \sigma^2 c_{kilj} \quad (6)$$

La construction du métamodèle de cokrigeage consiste alors à déterminer les coefficients  $\lambda_{ij}$  de la combinaison linéaire faisant intervenir les réponses et gradients (équation 7) en minimisant la variance de prédiction (écrite sous forme d'une erreur des moindres carrés, cf. équation 8) sous une condition de non biais (équation 9).

$$\tilde{Y}(\mathbf{x}^{(0)}) = \sum_{i=1}^{n_s} \lambda_{0i}(\mathbf{x}^{(0)}) Y(\mathbf{x}^{(i)}) + \sum_{j=1}^{n_d} \sum_{i=1}^{n_s} \lambda_{ij}(\mathbf{x}^{(0)}) \frac{\partial Y}{\partial x_j}(\mathbf{x}^{(i)}) \quad (7)$$

$$\text{MSE}[\tilde{Y}(\mathbf{x}^{(0)})] = \mathbb{E} \left[ \left( \tilde{Y}(\mathbf{x}^{(0)}) - Y(\mathbf{x}^{(0)}) \right)^2 \right] \quad (8)$$

$$\mathbb{E}[\tilde{Y}(\mathbf{x}^{(0)})] = \mathbb{E}[Y(\mathbf{x}^{(0)})] \quad (9)$$

Le prédicteur obtenu suivant cette démarche se nomme meilleur prédicteur linéaire sans biais (*Best Linear Unbiased Predictor* : BLUP) et s'écrit sous forme matricielle :

$$\tilde{Y}(\mathbf{x}^{(0)}) = \hat{\beta}_c + \mathbf{c}_{c0}^T \mathbf{C}_c^{-1} (\mathbf{Y}_{sc} - \mathbf{X}_c \hat{\beta}_c) \quad (10)$$

$$\begin{aligned} \hat{\beta}_c &= +(\mathbf{X}_c^T \mathbf{C}_c^{-1} \mathbf{X}_c)^{-1} \mathbf{X}_c^T \mathbf{C}_c^{-1} \mathbf{Y}_{sc} & \mathbf{c}_{c0} &= [c_{01} \quad c_{02} \quad \cdots \quad c_{0n_s} \quad c_{110} \quad c_{210} \quad \cdots \quad c_{n_d n_s 0}] \\ \text{où } \mathbf{X}_c &= [1, 1, \dots, 1, 0, 0, \dots, 0]^T & \mathbf{C}_c &= \begin{bmatrix} \mathbf{C} & \mathbf{C}_d \\ \mathbf{C}_d^T & \mathbf{C}_{dd} \end{bmatrix} \end{aligned}$$

Le terme  $\hat{\beta}_c$  est déterminé par une méthode de moindre carrés généralisés. Les vecteur et matrice  $\mathbf{c}_{c0}$  et  $\mathbf{C}_c$  sont obtenus respectivement en calculant les corrélations du point  $\mathbf{x}^{(0)}$  aux points échantillonnés et en calculant les corrélations entre les sites pour les variables primaire et secondaire. La matrice  $\mathbf{C}$  est la matrice de corrélation classiquement utilisée pour le krigeage, la matrice  $\mathbf{C}_d$  est la matrice de corrélation liant les réponses et les gradients aux sites échantillonnés et la matrice  $\mathbf{C}_{dd}$  est la matrice de corrélation liant les gradients aux sites échantillonnés. Le vecteur  $\mathbf{Y}_{sc}$  contient quant à lui les réponses et gradients calculés aux sites échantillonnés.

Les paramètres de la fonction de corrélation (fonction gaussienne ou Matérn dans notre cas) et la variance du processus stochastique sont estimés à l'aide d'une méthode de maximisation de la vraisemblance [8].

### 3.3 Exemple

Pour illustrer la qualité du cokrigeage, nous l'appliquons à une fonction analytique d'une variable<sup>1</sup>. A partir de 5 valeurs de la fonction et 5 gradients calculés en des sites générés grâce à une méthode LHS (*Latin Hypercube Sampling*), un métamodèle de krigeage et un de cokrigeage sont construits. La fonction de corrélation utilisée est la fonction Matérn. Les métamodèles construits et leurs dérivées sont présentés en figure 2. Cette figure montre la capacité du métamodèle de cokrigeage à interpoler non seulement les réponses mais également les dérivées (et donc les gradients). Pour un nombre réduit de points, le métamodèle de cokrigeage présente une meilleure qualité que celui de krigeage.

## 4 Couplage stratégie multiparamétrique - métamodèle

### 4.1 Stratégie de résolution

Dans cette partie, nous considérons des structures assemblées où les pièces seront considérées élastiques linéaires sous l'hypothèse des petites perturbations. Les seules non linéarités présentes, proviennent des problèmes de contact et de frottements entre les pièces. Le problème est résolu en utilisant l'algorithme itératif LATIN introduit par P. Ladevèze [6, 7].

1.  $y(x) = \exp(-x/10) \cos(x) + x/10$

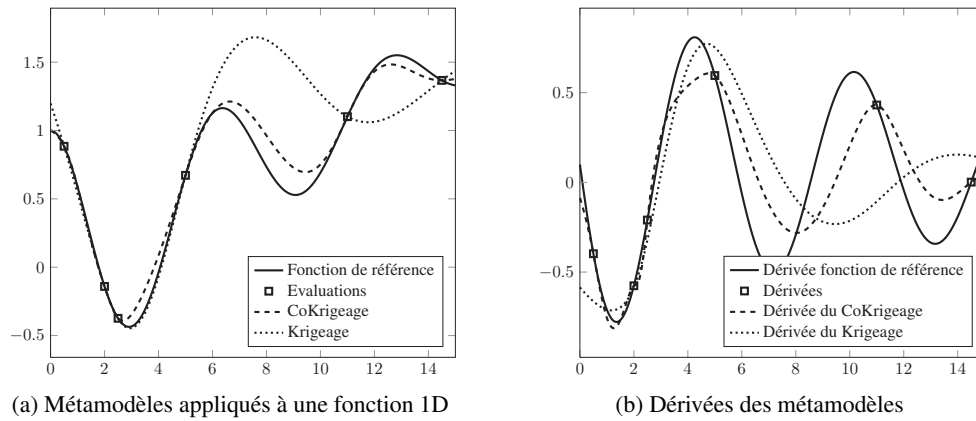


FIGURE 2 – Krigage et cokrigage ordinaire, et leur dérivées

La démarche s’articule autour de trois points : la structure étudiée est décomposée en *interfaces* et *sous-structures*, un algorithme itératif dédié est utilisé pour résoudre le problème mécanique, et au cours des itérations, les opérateurs de la méthode restent constants (ils ne dépendent ni du chargement, ni des paramètres des interfaces). Dans le cas des assemblages élastiques, la stratégie de résolution LATIN consiste à résoudre successivement deux groupes d’équations : le premier est constitué des *équations locales*  $\Gamma$  (qui peuvent être non linéaires) concernant les interfaces et le second est constitué des *équations linéaires*  $A_d$  traduisant le comportement des sous-structures. Ces résolutions successives conduisent, grâce à des directions de recherche (paramètres de la méthode), à trouver la solution à l’intersection des deux espaces  $\Gamma$  et  $A_d$  (cf. figure 3a).

### 4.2 Stratégie multiparamétrique

Dans le contexte de l’optimisation, il est nécessaire de réaliser un nombre important de calculs pour trouver le minimum de la fonction objective étudiée. Ces calculs sont réalisés avec de plus ou moins faibles variations des paramètres (variables de conception telles que des jeux, précharges, coefficients de frottement...). Les différents problèmes à résoudre associés chacun à un jeu de paramètres restent des problèmes très similaires. La stratégie multiparamétrique utilisée ici va permettre d’accélérer la résolution de ces problèmes similaires. Cette stratégie a été introduite par P.A. Boucard [3] et utilisée et étudiée dans [1, 2].

Le principe de la stratégie multiparamétrique réside dans la réinitialisation de l’algorithme itératif LATIN : lorsqu’un problème associé à un jeu de paramètres doit être résolu, l’algorithme est réinitialisé avec la solution convergée obtenue précédemment pour un autre jeu de paramètres. La figure 3b illustre schématiquement le principe de la stratégie multiparamétrique.

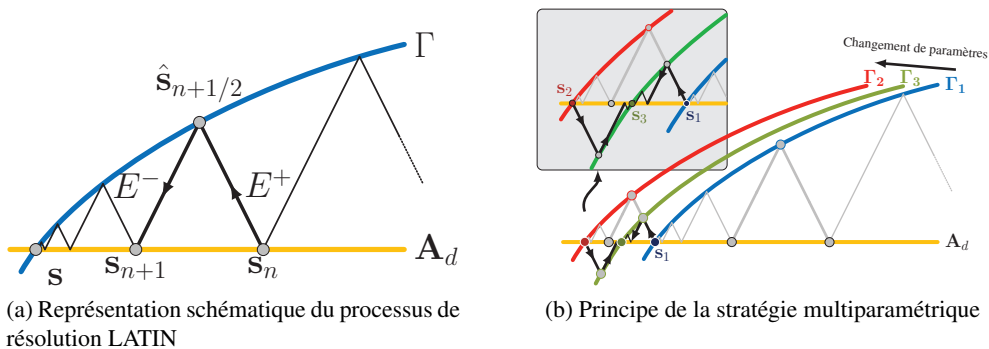


FIGURE 3 – Principe de l’algorithme itératif LATIN et de la stratégie multiparamétrique

L’emploi de cette stratégie permet d’obtenir la solution du problème avec un nombre réduit d’itérations et par conséquent en réduisant de façon significative le temps de calcul.

### 4.3 Illustration de la méthode

Dans cette partie, la méthode de construction du métamodèle est mise en œuvre sur un exemple académique quasi-statique. La figure 4a présente la géométrie du problème : trois carrés ( $h = 50\text{mm}$ ,  $E = 200\text{GPa}$  et  $\nu = 0.3$ ) constituant chacun une sous-structure sont en contact avec frottement les uns avec les autres. Le chargement est réalisé en deux temps : une précharge de la sous-structure 3 puis une pression appliquée sur la sous-structure 2. Pour ce cas test, les deux variables de conception prises en compte seront les deux coefficients de frottements et la fonction étudiée sera l'effort exercé par la sous-structure 2 sur la paroi. En appliquant la stratégie de résolution, en faisant varier les coefficients et en relevant la valeur de l'effort associé à chaque jeu de paramètres, on obtient la surface présentée en figure 4b.

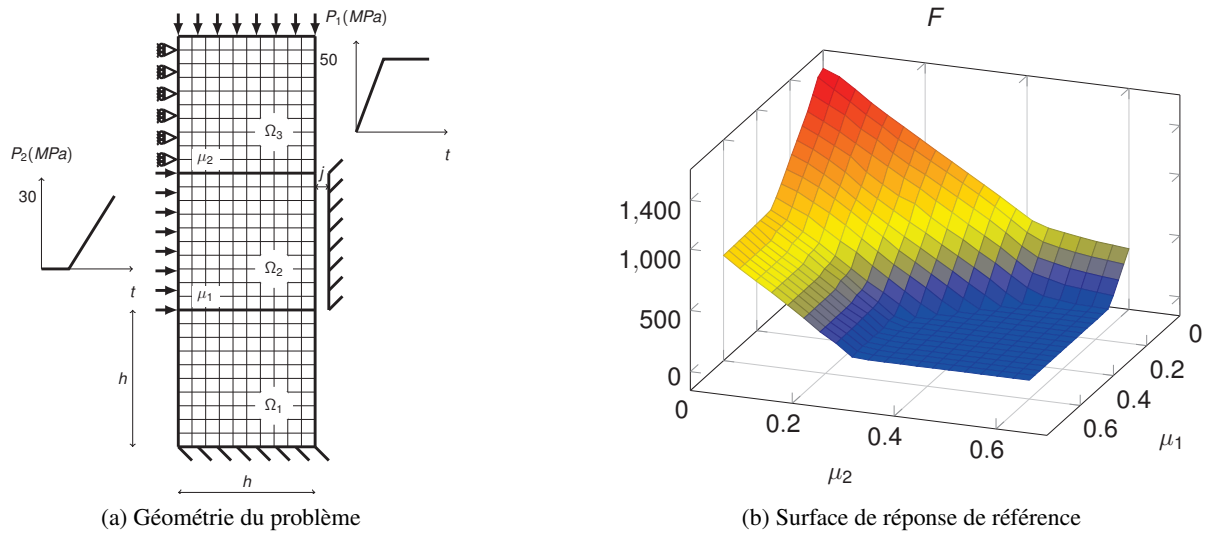


FIGURE 4 – Géométrie du problème et effort exercé par la sous-structure 2 sur la paroi en fonction des coefficients de frottement  $\mu_1$  et  $\mu_2$

La stratégie multiparamétrique et le métamodèle sont alors couplés. Pour cela, on considère un nombre constant (15) d'appels au solveur et l'on construit un métamodèle de krigeage et un de cokrigeage afin de les comparer. Dans ces conditions, le métamodèle de krigeage sera construit à partir de la réponse du modèle mécanique en 15 points générés par méthode LHS<sup>2</sup> tandis que le métamodèle de cokrigeage sera construit à partir de 5 réponses et 5 gradients (les gradients sont calculés par différences finies). La figure 5 présente les surfaces de réponse obtenues.

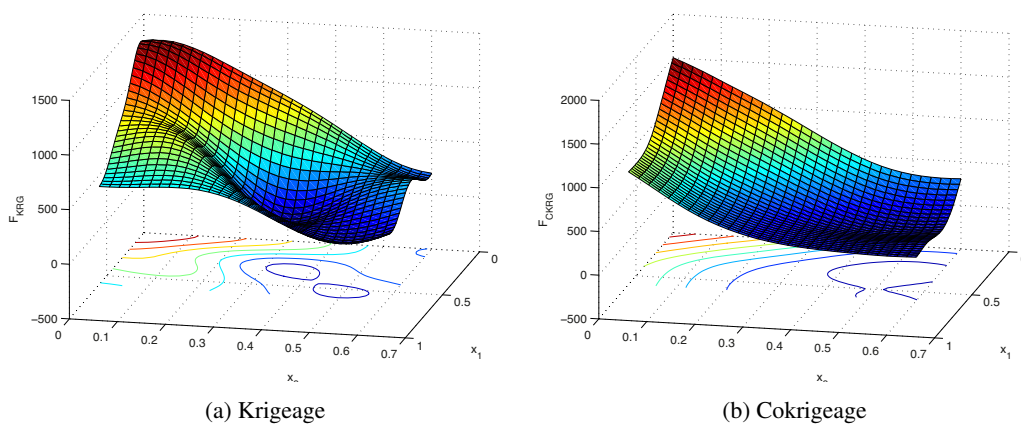


FIGURE 5 – Métamodèles de krigeage et cokrigeage obtenus avec 15 appels au solveur

La stratégie multiparamétrique montre sur ce cas test sa capacité à réduire de façon importante les temps de calcul : dans le cas des 15 évaluations sans calcul des gradients, le gain<sup>3</sup> apporté par la stratégie multiparamétrique est de 2.97 avec un temps cpu total de 44.3s alors que pour calculer 5 réponses et 5 gradients, le gain est de 3.40 pour un temps CPU de 36.4s. Ceci se justifie par la capacité de la stratégie multiparamétrique à accélérer la résolution lorsque l'on réinitialise l'algorithme avec une solution obtenue en un point très proche de celui calculé. Le métamodèle de cokrigeage est donc dans ce cas moins coûteux à construire que celui de krigeage. Ce résultat est d'autant plus intéressant que le cokrigeage possède une qualité similaire à celle du krigeage : la moyenne<sup>4</sup> des carrés des écarts du métamodèle de krigeage est de  $8.463 \cdot 10^{-4}$  et de  $1.110 \cdot 10^{-3}$  pour le cokrigeage.

## 5 Conclusions

La stratégie d'optimisation multiniveaux de modèles présentée s'appuie sur deux aspects pour réduire les coûts de calcul : une stratégie d'optimisation en deux étapes s'appuyant sur un métamodèle obtenu avec un faible coût de calcul et une stratégie multiparamétrique tirant partie des calculs déjà réalisés pour accélérer la résolution de problèmes similaires. Le métamodèle de cokrigeage permet lorsqu'il est couplé avec la stratégie multiparamétrique de fournir une approximation fidèle de la fonction étudiée avec un coût de calcul modéré. Le déploiement de la stratégie complète fera l'objet de développements futurs. La stratégie multiparamétrique permettra également de diminuer les temps de calcul nécessaires à la réalisation de l'optimisation du second niveau en l'employant conjointement avec un algorithme d'optimisation utilisant les gradients.

**Remerciements** : Ces travaux ont été effectués dans le cadre du projet ANR-08-COSI-007-10, OMD2 financé par l'Agence Nationale de la Recherche.

## Références

- [1] Allix, O. and Vidal P., A new multi-solution approach suitable for structural identification problems, *Computer Methods in Applied Mechanics and Engineering*, 191(25-26) :2727–2758, 2002.
- [2] Boucard, P.A. and Champaney, L., A suitable computational strategy for the parametric analysis of problems with multiple contact, *International Journal for Numerical Methods in Engineering*, 57(9) :1259–1281, 2003.
- [3] Boucard, P.A. and Ladeveze P., A multiple solution method for non-linear structural mechanics. *Mechanical Engineering*, 50(5) :317–328, 1999.
- [4] Isaaks E.H. and Srivastava R.M., An introduction to applied geostatistics, *Oxford University Press*, New York, 1989.
- [5] Krige D.G., A statistical approach to some mine valuation and allied problems on the Witwatersrand, Master's thesis, 1951.
- [6] Ladevèze P., Sur une famille d'algorithmes en mécanique des structures, *Compte rendu de l'académie des Sciences*, 300(2) :41–44, 1985.
- [7] Ladevèze P., Nonlinear computational structural mechanics : new approaches and non-incremental methods of calculation, *Springer Verlag*, 1999.
- [8] Mardia, K.V. and Marshall R.J., Maximum likelihood estimation of models for residual covariance in spatial regression, *Biometrika*, 71(1) :135, 1984.
- [9] G. Matheron. Traité de géostatistique appliquée. *Memoires du Bureau de Recherches Geologiques et Minieres*, 14, 1962.
- [10] Matheron G., Principles of geostatistics, *Economic geology*, 58(8) :1246, 1963.
- [11] Robinson G.M. and Keane A.J., A case for multi-level optimisation in aeronautical design. *Aeronautical Journal*, 103(1028) :481–485, 1999.
- [12] Wackernagel H., Multivariate geostatistics : an introduction with applications, *Springer Verlag*, 1995.

3. Le gain calculé ici correspond au gain de la stratégie en terme de temps de calcul et se calcule de la manière suivante :  $\text{Gain} = \frac{\text{Nombre calcul} \times \text{Temps CPU du premier calcul}}{\text{Temps CPU total avec la stratégie multiparamétrique}}$  (le temps du premier calcul est pris comme référence).

4. calculée sur un nombre d'échantillons plus important que le nombre de sites échantillonnés pour construire le métamodèle ; les écarts sont normés par la plus grande valeur de la fonction de référence soit environ  $1500N$