



HAL
open science

Stratégie rationnelle pour la résolution des problèmes paramétrés dans le cadre de la Proper Generalized Decomposition

David Néron, Pierre-Alain Boucard, Christophe Heyberger

► **To cite this version:**

David Néron, Pierre-Alain Boucard, Christophe Heyberger. Stratégie rationnelle pour la résolution des problèmes paramétrés dans le cadre de la Proper Generalized Decomposition. 11e colloque national en calcul des structures, CSMA, May 2013, Giens, France. hal-01722080

HAL Id: hal-01722080

<https://hal.science/hal-01722080>

Submitted on 2 Mar 2018

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.

Public Domain

Stratégie rationnelle pour la résolution des problèmes paramétrés dans le cadre de la Proper Generalized Decomposition

David NÉRON, Pierre-Alain BOUCARD, Christophe HEYBERGER

LMT-Cachan (ENS Cachan/CNRS/UPMC/PRES UniverSud Paris, France)
{neron,boucard,heyberger}@lmt.ens-cachan.fr

Résumé — Dans ce travail, on propose une stratégie de calcul pour la résolution d'un problème paramétré sur un espace de conception donné en vue, par exemple, de la construction d'un métamodèle. Cette stratégie est basée sur la méthode de réduction de modèle *Proper Generalized Decomposition* (PGD) et permet une diminution importante du temps CPU. Pour cela, elle se base sur la similarité des calculs à réaliser pour des jeux de paramètres différents, afin de tirer parti des simulations déjà réalisées. Le nouvel aspect introduit ici est une technique de sélection des jeux de paramètres les plus pertinents à traiter pour enrichir la connaissance globale de la solution et construire, pour coût minimum, une base réduite permettant de représenter l'ensemble des solutions dans l'espace de conception.

Mots clés — PGD, réduction de modèles, problème paramétré, LATIN

1 Introduction

Ce travail est issu du projet de recherche OMD², financé en partie par l'ANR et piloté par Renault, qui a pour objectif le développement de méthodes numériques pour l'optimisation multidisciplinaire. Parmi les verrous scientifiques qui freinent l'essor, voire l'intégration, de ce type de méthodes dans le processus de conception des véhicules au niveau industriel, un point majeur est la lourdeur des simulations numériques nécessaires. En effet, l'optimisation d'une structure passe par la recherche du minimum d'une fonction coût dans un espace de conception donné et les algorithmes de recherche (que ce soit par analyse directe ou construction préalable d'un métamodèle) conduisent tous à évaluer la réponse de la structure pour une série de jeux de paramètres. Chacune de ces évaluations peut, en soi, nécessiter un coût de calcul important, mais la multiplication de ces résolutions mène, d'autant plus souvent, à des temps prohibitifs en l'absence de technique dédiée. L'objectif de cette étude est donc de développer une stratégie de calcul pour la résolution de problèmes paramétrés afin de réduire le temps de calcul lorsqu'un grand nombre d'appels à la solution sont requis pour des jeux de paramètres différents. Une approche directe, consistant à résoudre indépendamment les problèmes pour chacun des jeux de paramètres, ne tire pas parti de la similarité des problèmes à résoudre et conduit à un coût de calcul rédhibitoire.

Une autre voie est d'avoir recours à des techniques de réduction de modèles. Le fondement de ces techniques est la recherche de la solution d'un problème donné dans une base de dimension réduite par rapport à celle de l'espace des solutions original. Cette base, dite réduite, qui doit permettre d'approximer la solution avec une précision suffisante, peut être construite au préalable, que ce soit à partir de la POD de la solution d'un problème « de remplacement », plus simple que le problème initial [1, 2, 3, 4], ou avec des techniques de sélection qui permettent de bâtir un indicateur d'erreur afin de maîtriser l'erreur commise (e.g. [5, 6] pour les méthodes dites *Reduced Basis*). Dans le cas de la *Proper Generalized Decomposition* (PGD) qui est utilisée ici, la base réduite est générée à la volée, en ne cherchant à construire que les vecteurs représentatifs du problème traité.

2 PGD pour les problèmes paramétrés

La PGD a fait l'objet de nombreux travaux récents, avec des variantes en terme d'algorithmie, mais surtout d'applications. À titre d'exemples, elle a été utilisée pour traiter des problèmes dits multidimensionnels [7, 8], des problèmes stochastiques [9]. Dans le projet OMD², dont cet article fait l'objet, la PGD a été utilisée comme une représentation à variables séparées temps-espace de la solution et l'algorithme utilisé est la méthode itérative LATIN [10, 11, 12]. Dans celle-ci, à une itération donnée, la base réduite générée lors des itérations précédentes est utilisée pour construire un modèle réduit du problème et trouver une nouvelle approximation de la solution. Si la qualité de celle-ci n'est pas suffisante, la base est enrichie en recherchant un nouveau produit de fonctions temps-espace grâce à un algorithme glouton.

Dans le cas où on s'intéresse à la génération de la solution d'un problème paramétré pour l'ensemble des jeux paramètres appartenant à un espace de conception donné, une première approche est présentée dans [13], en générant *offline* une PGD de la solution dans laquelle les paramètres sont de nouvelles coordonnées, ce qui permet d'évaluer *online* de façon quasi-instantanée la solution pour toutes les valeurs de ceux-ci. Ce type de vision est particulièrement intéressante lorsque la rapidité de la réponse *online* est une priorité, au prix éventuel d'un temps de calcul *offline* important.

L'approche utilisée ici, introduite dans [12], a pour but de réduire le temps CPU global de l'étude. Lorsque la solution pour un nouveau jeu de paramètres est requise, l'algorithme est initialisé en réutilisant la base réduite générée pour les jeux de paramètres traités précédemment. Dans bien des cas, cette base est suffisante pour obtenir une approximation initiale de la solution de bonne qualité, et peu de vecteurs doivent être générés et ajoutés à la base pour affiner la solution. Ceci permet une réduction très importante du coût de calcul total par rapport à l'utilisation d'une approche directe lorsque la solution pour une succession de jeux de paramètres est nécessaire. Dans [12], cette méthode avait été utilisée en traitant, un par un, chacun des jeux, sans rationaliser l'ordre dans lequel chaque problème était pris en compte. L'évolution qui est présentée ici consiste à mettre en place une technique, inspirée de la méthode *Reduced Basis*, qui permette d'optimiser le choix des jeux de paramètres pour lesquels la résolution à l'aide de la PGD conduit à enrichir rapidement la base réduite globale. Le but est de pouvoir représenter toutes les solutions restantes de l'espace de conception avec une précision suffisante sans avoir à ajouter de nouveaux vecteurs.

3 Problème de référence et étude paramétrique

Le problème qui sert d'illustration est celui de diffusion dans un domaine Ω , initialement à une température T_{init} . On s'intéresse à l'évolution de la température T dans Ω et au cours d'un intervalle de temps $\mathcal{I} = [0, t_{final}]$. Cette structure est soumise à une production volumique de chaleur propre r sur Ω , une température T_d sur la partie $\partial_T\Omega$ de la frontière, une densité de flux de chaleur $-y_d$ sur la partie complémentaire $\partial_Y\Omega$ (cf. Figure 1). On désigne la masse volumique par ρ , la capacité thermique massique par c , la conductivité thermique par λ , la densité de flux de chaleur par \underline{q} (mais on choisit de manipuler $\underline{Y} = -\underline{q}$, ce qui permet de réécrire la loi de comportement sous une forme pour laquelle la convergence de la méthode LATIN est assurée [10]). Le problème de diffusion consiste à trouver $\mathbf{s} = (T, \underline{Y})$ définie sur $\mathcal{I} \times \Omega$ qui vérifie :

- l'équation d'équilibre :

$$\rho c \frac{\partial T}{\partial t} = r + \text{div } \underline{Y} \quad (1)$$

- la loi de Fourier :

$$\underline{Y} = \lambda \text{ grad } T \quad (2)$$

- les conditions aux limites (où \underline{n} est la normale extérieure à $\partial_Y\Omega$) :

$$T = T_d \quad \text{sur } \partial_T\Omega \quad \text{et} \quad \underline{Y} \cdot \underline{n} = y_d \quad \text{sur } \partial_Y\Omega \quad (3)$$

- la condition initiale :

$$T = T_{init} \quad \text{à } t = 0 \quad (4)$$

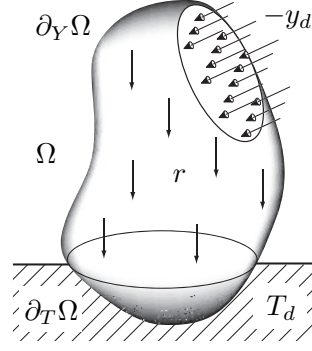


Fig. 1 – Le problème de référence

On souhaite étudier la réponse de cette structure lorsque certains paramètres décrivent un espace de conception. On peut envisager ici des variations sur différents types de paramètres — chargement (r , flux imposé), conditions aux bords (conditions limites et initiales), paramètres matériau (ρ , c , λ) et de l'étude (discrétisation, durée, longueur du domaine) — mais, pour simplifier la présentation, on se restreint ici aux variations de λ et ρc . Pour explorer l'espace de conception correspondant au jeu de paramètres $p = [\lambda, \rho c]$, on échantillonne les intervalles de variation de ceux-ci, ce qui conduit à déterminer la solution du problème pour $N \times M$ jeux de paramètres $p^{jk} = [\lambda_j, \rho c_k]$ avec $j = 1, \dots, N$ et $k = 1, \dots, M$. L'espace discret correspondant à ces jeux est noté \mathcal{P} . L'étude consiste donc à rechercher la solution de $N \times M$ problèmes similaires de la forme (après discrétisation en espace) :

$$\mathbf{M}^k \mathbf{T}^{jk} + \mathbf{K}^j \mathbf{T}^{jk} = \mathbf{Q} \quad (5)$$

4 Méthode LATIN et stratégie multiparamétrique

Dans la présentation qui suit, seules les grandes idées sont présentées. Le détail des développements peut être trouvé dans [12].

4.1 Approche itérative LATIN pour un jeu de paramètres fixé

Pour résoudre le problème associé à un jeu de paramètres fixé, on introduit les espaces suivants (ainsi que les espaces homogènes associés, notés $*$) :

- l'espace \mathcal{T} des champs T admissibles :

$$\mathcal{T} \equiv \{ T \mid T|_{\partial_T \Omega} = T_d \text{ et } T|_{(t=0)} = T_{init} \} \quad (6)$$

- l'espace \mathbf{A}_d des champs solution (T, \underline{Y}) admissibles :

$$\mathbf{A}_d \equiv \{ \mathbf{s} = (T, \underline{Y}) \mid T \in \mathcal{T} \text{ et } \text{div} \underline{Y} + r = \rho c \frac{\partial T}{\partial t} \text{ avec } \underline{Y} \cdot \underline{n}|_{\partial_Y \Omega} = y_d \} \quad (7)$$

- l'espace $\mathbf{\Gamma}$ des champs vérifiant la loi de Fourier :

$$\mathbf{\Gamma} \equiv \{ \mathbf{s} = (T, \underline{Y}) \mid \underline{Y} = \lambda \underline{\text{grad}} T \} \quad (8)$$

La solution du problème, qui s'écrit $\mathbf{s} = (T, \underline{Y}) = \mathbf{\Gamma} \cap \mathbf{A}_d$, est construite en utilisant la méthode LATIN [10], c'est-à-dire en générant itérativement un élément de $\mathbf{\Gamma}$ puis un élément de \mathbf{A}_d comme le montre la figure 2. L'algorithme peut être initialisé par une approximation grossière $\mathbf{s}_0 \in \mathbf{A}_d$, cette approximation s'affinera automatiquement au fil des itérations. Afin de passer d'un espace à l'autre, on introduit les directions de recherche \mathbf{E}^+ pour l'étape locale et \mathbf{E}^- pour l'étape linéaire :

$$\begin{cases} \mathbf{E}^+ : & h(\hat{\underline{Y}}_n - \underline{Y}_n) + (\underline{\text{grad}} \hat{T}_n - \underline{\text{grad}} T_n) = 0 \\ \mathbf{E}^- : & h(\underline{Y}_{n+1} - \hat{\underline{Y}}_n) - (\underline{\text{grad}} T_{n+1} - \underline{\text{grad}} \hat{T}_n) = 0 \end{cases} \quad (9)$$

$$\mathbf{s}_0 \in \mathbf{A}_d \cdots \longrightarrow \mathbf{s}_n \in \mathbf{A}_d \cap \mathbf{E}^- \longrightarrow \underbrace{\hat{\mathbf{s}}_n \in \Gamma \cap \mathbf{E}^+ \longrightarrow \mathbf{s}_{n+1} \in \mathbf{A}_d \cap \mathbf{E}^-}_{\text{Itération } n+1} \longrightarrow \hat{\mathbf{s}}_{n+1} \in \Gamma \cap \mathbf{E}^+ \longrightarrow \cdots$$

Fig. 2 – Méthode itérative LATIN

La convergence de la méthode peut être prouvée sous des conditions peu restrictives qui sont satisfaites ici en prenant $h = 1/\lambda$ [10]. À l'itération $n+1$, on peut construire un indicateur de convergence de la méthode en calculant, par exemple, l'erreur en résidu. T_{n+1} étant discrétisé sous la forme \mathbf{T}_{n+1} , celle-ci peut s'écrire (avec $\|\bullet\| = \int_{\mathcal{I}} \bullet^2 dt$) :

$$e_{n+1} = \frac{\|\mathbf{K}^{-1}(\mathbf{M}\dot{\mathbf{T}}_{n+1} + \mathbf{K}\mathbf{T}_{n+1} - \mathbf{Q})\|_2}{\|\mathbf{T}_{n+1}\|_2} \quad (10)$$

4.2 Proper Generalized Decomposition

La solution du problème est recherchée sous une forme à variables temps-espace séparées. Le champ de température T , défini sur $\mathcal{I} \times \Omega$, est ainsi recherché sous la forme :

$$T(t, \underline{M}) \approx T_0(t, \underline{M}) + \sum_i \tau_i(t) \mathbb{T}_i(\underline{M}) \quad (11)$$

où $T_0 \in \mathcal{T}$ et $\{\mathbb{T}_i\}$ forment une base réduite de fonctions de l'espace dont les vecteurs $\mathbb{T}_i \in \mathcal{T}^*$ sont pondérés par les fonctions du temps τ_i . Le même type de décomposition peut être utilisée pour \underline{Y} . Pour expliciter la construction de la décomposition PGD, supposons qu'une base réduite $\{\mathbb{T}_i\}_{i=1, \dots, m}$ ait été construite au cours des n premières itérations, ce qui a conduit à une approximation du champ de température $T_n = T_0(t, \underline{M}) + \sum_{i=1}^m \tau_i(t) \mathbb{T}_i(\underline{M})$. À l'itération $n+1$, la nouvelle approximation $T_{n+1} = T_n + \Delta T$ de la température est générée en apportant une correction $\Delta T \in \mathcal{T}^*$ à $T_n \in \mathcal{T}$. Différentes approches sont envisageables pour générer cette correction (cf. [14] pour une synthèse de celles-ci). Celle utilisée ici est une formulation de type Galerkin de l'équation d'admissibilité à zéro de ΔT , dans laquelle la forme de ΔT est imposée.

Dans un premier temps, la base réduite $\{\mathbb{T}_i\}_{i=1, \dots, m}$ est réutilisée en l'état, sans chercher à l'enrichir. On cherche ainsi une correction ΔT sous la forme :

$$\Delta T = \sum_{i=1}^m \bar{\tau}_i(t) \mathbb{T}_i(\underline{M}) \quad (12)$$

où les \mathbb{T}_i sont fixés et seules des corrections $\bar{\tau}_i$ sont à déterminer. Un critère portant sur la variation de l'erreur en résidu e_{n+1} permet de décider si cette phase de réutilisation de la base réduite a été suffisante. Si ce n'est pas le cas, la base est enrichie dans un second temps en recherchant une nouvelle correction ΔT sous la forme :

$$\Delta T = \tau_{m+1}(t) \mathbb{T}_{m+1}(\underline{M}) \quad (13)$$

où cette fois ni \mathbb{T}_{m+1} , ni τ_{m+1} ne sont pas fixés. Ce type de technique d'exploration afin d'enrichir la base réduite s'apparente à un algorithme dit « glouton ». La réutilisation préliminaire de la base réduite, qui dans bien des cas est suffisante pour faire progresser la solution, fait qu'à l'itération n , la taille m de cette base est bien inférieure n .

4.3 Stratégie multiparamétrique

Dans le cadre d'une étude paramétrique, l'utilisation de la stratégie LATIN permet d'envisager plusieurs techniques qui tirent parti des calculs déjà réalisés pour les jeux de paramètres précédents. La technique la plus basique est d'initialiser la méthode LATIN (\mathbf{s}_0 dans la figure 2) avec la solution du problème associé au jeu de paramètres déjà traité le plus proche de celui qu'on veut aborder maintenant. Cette idée a déjà montré ses performances dans [15] et notamment dans le cadre de problèmes de contact [16].

Une approche plus efficace dans le cadre de la PGD est de réutiliser la base réduite construite pour le jeu de paramètres le plus proche dans l'espace de conception. Cette base est tout d'abord recombinaison pour représenter au mieux la solution du nouveau problème, et enrichie si cela s'avère nécessaire pour satisfaire le niveau de précision requis. Cette technique a été utilisée dans [12] avec d'importants gains en terme de coût CPU pour la génération de la solution d'un problème pour l'ensemble des paramètres d'un espace de conception.

Cependant, à l'issue du traitement d'un certain nombre de jeux de paramètres, il s'avère généralement que la base réduite déjà calculée n'a plus besoin d'être enrichie. Sa seule recombinaison est suffisante pour générer la solution des nouveaux jeux étudiés. On qualifie une telle base de « complète » (notée \mathbb{B}_c dans la suite), dans la mesure où elle permet de représenter l'ensemble des solutions de l'espace de conception avec un minimum de vecteurs.

Deux situations peuvent être envisagées dans le cadre d'une étude d'optimisation.

- dans la première, cette étude n'est réalisée qu'une fois et seule compte la limitation du temps global de recherche de la solution pour les jeux de paramètres requis par le processus d'optimisation ;
- dans la seconde, le temps *online* doit être limité au maximum afin d'accélérer le processus de conception. C'est le cas où l'étude paramétrique est récurrente et mérite un traitement *offline* en amont. La possession d'une base complète est alors particulièrement intéressante.

La problématique qui est abordée dans la suite est la génération d'une base complète à moindre coût, c'est-à-dire la mise en place d'une technique qui permette de sélectionner les jeux de paramètres les plus à-même d'enrichir la base sans apporter d'information redondante et donc de réaliser au final un minimum de calculs.

Remarquons qu'une technique pragmatique pour générer une base complète est de résoudre le problème pour l'ensemble des jeux de paramètres (à l'aide d'une approche directe ou d'une approche PGD) et d'appliquer une SVD à l'ensemble des solutions obtenues avant de tronquer la base qui en résulte au niveau d'erreur souhaité. Une telle approche est évidemment très coûteuse et n'a que peu d'intérêt dans la pratique.

4.4 Stratégie de sélection

La stratégie de sélection est inspirée des techniques *Reduced Basis* (e.g. [5, 6]). On s'intéresse à la génération d'une base complète \mathbb{B}_c représentative des solutions d'un problème pour un espace paramétrique \mathcal{P} donné. La base \mathbb{B}_c est initialisée à partir de la base réduite obtenue en résolvant, à l'aide de la PGD, le problème pour le jeu de paramètre « moyen » dans \mathcal{P} . Dans un second temps, on utilise cette base pour y projeter les problèmes correspondants aux autres jeux de paramètres, ce qui permet de calculer la meilleure approximation du problème (associée à \mathbb{B}_c) pour chacun de ces jeux pour un coût modique. Une erreur en résidu e (cf. équation (10)) est alors calculée pour chacune de ces solutions et le jeu de paramètres associé à la plus grande erreur est ensuite résolu par l'algorithme PGD en enrichissant la base initiale \mathbb{B}_c . La base \mathbb{B}_c se complète donc progressivement, jusqu'à satisfaire le niveau d'erreur souhaité. Dans la pratique, un critère est introduit pour limiter les problèmes projetés dans la base réduite à ceux dont l'erreur en résidu était la plus importante, ce qui permet de réduire encore le temps CPU.

5 Exemple numérique

5.1 Description

Pour montrer la faisabilité de la méthode, on s'intéresse au problème unidimensionnel de la figure 3. La durée de l'étude est fixée à 30 secondes, discrétisée en 100 pas de temps. La longueur du domaine est de 1 mètre avec une discrétisation de 1000 éléments. Lors de la résolution, le critère de convergence utilisé (à la fois pour résoudre un problème par PGD et pour déclarer la base commune complète) est une erreur en résidu $e < 10^{-3}$. Dans cette présentation on se limite à une étude à 2 paramètres $p^{jk} = [\lambda_j, \rho c_k]$, où les deux coefficients matériaux sont paramétrés par j et k :

- la variation du paramètre λ_j est donnée sur la figure 4(a). On suppose $f(j) = 1 + \frac{j-1}{10}$ et on prend $\lambda = 0,5$ comme donnée.
- la variation du paramètre ρc_k est donnée sur la figure 4(b). On considère les valeurs ρc , ℓ (taille de la zone hétérogène), x_0 (centre de la première zone), x_f (centre de la dernière zone) comme des données qui sont respectivement prises égales à 15; 0,15; 0,14; 0,8 pour cette étude. De plus, on note $d = \frac{x_f - x_0}{10}$.

j et k , qui ne sont pas forcément des entiers, varient tous deux dans l'intervalle $[1; 11]$. Cet intervalle est échantillonné en 33 valeurs régulières. L'espace de conception \mathcal{P} correspond donc à 1089 jeux de paramètres. Des exemples avec un plus grand nombre de paramètres ont aussi été traités, avec des résultats tout à fait similaires.

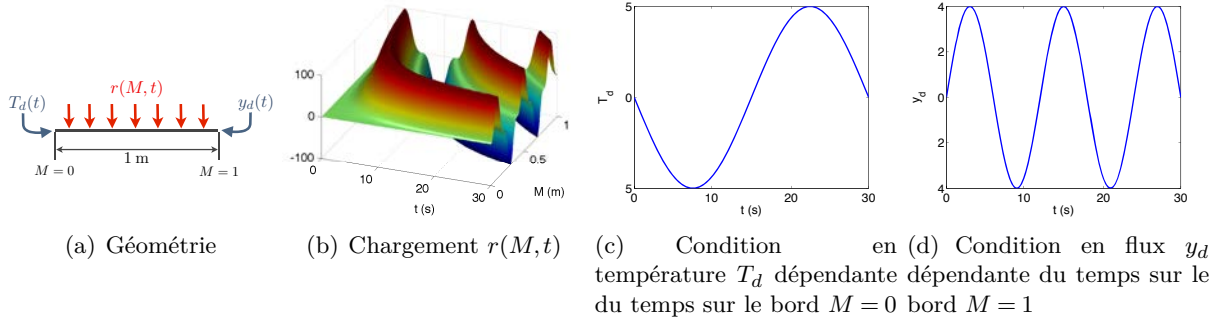


Fig. 3 – Problème monodimensionnel traité

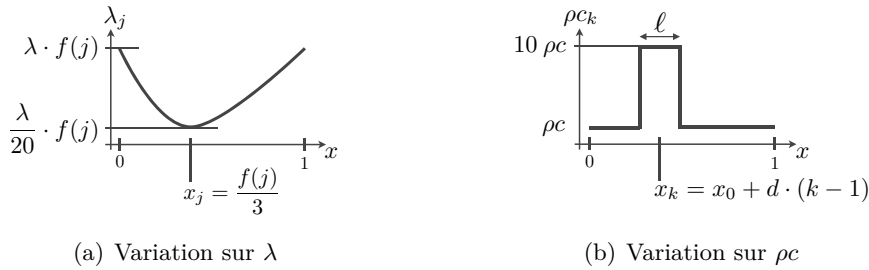


Fig. 4 – Jeux de paramètres considérés

5.2 Génération d'une base complète

Une résolution par PGD pour chacun des 1089 jeux de paramètres conduit en moyenne à des bases réduites de 26 fonctions spatiales par jeu si on traite chaque problème de manière indépendante. Cet ensemble de $1089 \times 26 = 28314$ fonctions spatiales peut être réduit en appliquant une SVD, ce qui conduit à une base complète de 72 fonctions pour le niveau d'erreur retenu. L'utilisation de la technique de sélection décrite dans la section 4.4 permet de générer une base complète comportant un nombre de fonctions sensiblement identique en explorant seulement 11 problèmes. La figure 5 montre les jeux de paramètres sélectionnés ainsi que l'ordre dans lequel ils ont été utilisés pour construire la base.

5.3 Application à la résolution d'une étude d'optimisation

On simule maintenant une étude d'optimisation fictive comme la demande de résolutions successives d'un certain nombre de problèmes pour des valeurs aléatoires des jeux de paramètres. On considère 4 études d'optimisation demandant la résolution de respectivement 20, 50, 100 et 200 problèmes dans l'espace de conception. Pour chaque étude, les résultats présentés ont été moyennés sur 6 différentes tirages aléatoires afin de quantifier la robustesse de la méthode.

Le tableau 1 regroupe les gains, calculés par rapport au temps CPU de l'approche directe, qui consiste à générer la solution de chacun des problèmes à l'aide de la PGD mais de manière indépendante des autres jeux de paramètres. On s'intéresse aux deux stratégies suivantes :

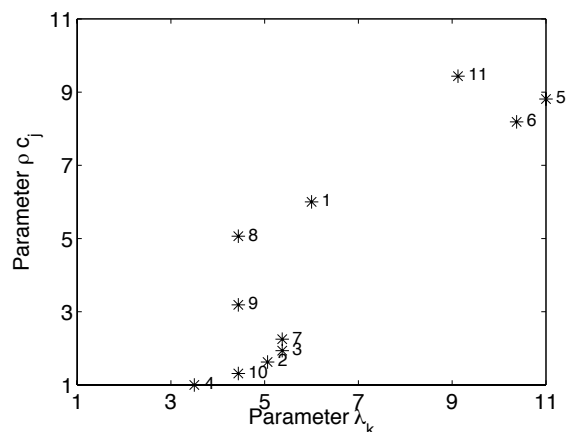


Fig. 5 – Chemin sélectionné pour construire la base complète

- **stratégie 1** : utilisation de la méthode multiparamétrique introduite dans [12], où l’ensemble du calcul est réalisé de manière *online* et les problèmes sont traités au gré des besoins de l’algorithme d’optimisation en partageant une même base réduite qu’on enrichit progressivement si besoin (le gain est noté G_1) ;
- **stratégie 2** : utilisation de la technique de sélection introduite ici pour générer *offline* une base complète et l’utiliser *online* pour résoudre rapidement les problèmes sollicités par l’optimiseur (on distingue alors le gain *online* noté $G_{2online}$ et le gain total G_{2total}).

	stratégie 1	stratégie 2	
	G_1	$G_{2online}$	G_{2total}
20 pbs	8,54	73,37	0,80
50 pbs	18,08	82,98	1,98
100 pbs	27,65	80,36	3,87
200 pbs	43,25	88,20	7,44

Tableau 1 – Gains pour l’étude paramétrique

Si seul le temps total est important, dans le cadre d’une étude d’optimisation unique, l’intérêt d’avoir à notre disposition une base initiale est réduit. Le gain penche largement en faveur de la stratégie 1, d’autant plus que le nombre de problèmes à évaluer est important. En revanche, dans le cas d’une étude récurrente où priorité est donnée au temps *online*, l’avantage d’une base réduite préalablement construite, comme c’est le cas dans la stratégie 2, est clair. Il permet une réponse rapide et quasi-indépendante du nombre de solutions sollicitées par l’algorithme d’optimisation.

6 Conclusion

Nous avons proposé ici une technique de génération en amont d’une base dite « complète » permettant, par une sélection de quelques calculs préliminaires, de n’ajouter aucune nouvelle fonction à la base lors de la réalisation de nouveaux calculs lors d’une étude paramétrique.

Sur les exemples étudiés, le coût de calcul de cette base reste non négligeable, car il nécessite une exploration de l’espace de conception. Cependant, si cette base est générée de manière *offline*, les temps associés aux nouveaux calculs effectués *online* sont réduits de façon drastique. Cette première méthode est donc à privilégier lorsque les études paramétriques doivent menées de façon récurrente et que le temps *online* doit être réduit. En revanche, si on prend en compte les coûts de génération de cette base complète, les gains sont moins importants et une stratégie de génération progressive de la base est plus efficace numériquement.

Les exploitations ont été faites sur des cas unidimensionnels et linéaire de thermique. Néanmoins la stratégie proposée est totalement générique et son extension au cadre non-linéaire dans

le cadre de la méthode LATIN ne pose pas de difficulté particulière. Des travaux sont en cours pour étendre les résultats présentés à des problèmes plus complexes.

Remerciements

Ce travail a été réalisé dans le cadre du projet ANR-08-COSI-007-10 (Optimisation Multi-Disciplinaire 2, OMD²), financé par l'ANR.

Références

- [1] Y. Maday and E.M. Ronquist. The reduced-basis element method : application to a thermal fin problem. *Journal on Scientific Computing*, 26(1) :240–258, 2004.
- [2] K. Kunish and L. Xie. Pod-based feedback control of the burgers equation by solving the evolutionary hjb equation. *Computers & Mathematics with Applications*, 49(7-8) :1113–1126, 2005.
- [3] T. Lieu, C. Farhat, and A. Lesoinne. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Computer Methods In Applied Mechanics and Engineering*, 195(41-43) :5730–5742, 2006.
- [4] M.D. Gunzburger, J.S. Peterson, and J.N. Shadid. Reduced-order modeling of time-dependent pdes with multiple parameters in the boundary data. *Computer Methods In Applied Mechanics and Engineering*, 196(4-6) :1030–1047, 2007.
- [5] Y. Maday, A.T. Patera, and G. Turnici. A priori convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations. *Journal of Scientific Computing*, 17(1-4) :437–446, 2002.
- [6] G. Rozza. Reduced basis approximation and error bounds for potential flows in parametrized geometries. *Communications in Computational Physics*, 9(1) :1–48, 2011.
- [7] A. Ammar, B. Mokdad, F. Chinesta, and R. Keunings. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. *Journal of Non-Newtonian Fluid Mechanics*, 139(3) :153–176, 2006.
- [8] F. Chinesta, A. Ammar, F. Lemarchand, P. Beauchene, and F. Boust. Alleviating mesh constraints : Model reduction, parallel time integration and high resolution homogenization. *Computer Methods in Applied Mechanics and Engineering*, 197 :400–413, 2008.
- [9] A. Nouy. Recent developments in spectral stochastic methods for the numerical solution of stochastic partial differential equations. *Archives of Computational Methods in Engineering*, 16(3) :251–285, 2009.
- [10] P. Ladevèze. Nonlinear computational structural mechanics - new approaches and non-incremental methods of calculation. *Springer Verlag*, 1999.
- [11] P. Ladevèze, J-C. Passieux, and D. Néron. The latin multiscale computational method and the proper generalized decomposition. *Computer Methods in Applied Mechanics & Engineering*, 199, num. 21-22 :1287–1296, 2010.
- [12] C. Heyberger, P-A. Boucard, and D. Néron. Multiparametric analysis within the proper generalized decomposition framework. *Computational Mechanics*, 49(3) :277–289, 2011.
- [13] B. Bognet, F. Bordeu, F. Chinesta, A. Leygue, and A. Poitou. Advanced simulation of models defined in plate geometries : 3D solutions with 2D computational complexity. *Computational Methods in Applied Mechanical Engineering*, 201 :1–12, 2012.
- [14] A. Nouy. 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.
- [15] P-A. Boucard and P. Ladevèze. Une application de la méthode latin au calcul multirésolution de structures non linéaires. *Revue Européenne des Eléments Finis*, 8 :903–920, 1999.
- [16] P-A. Boucard and L. Champaney. A suitable computational strategy for the parametric analysis of problems with multiple contact. *International Journal for Numerical Methods in Engineering*, 57 :1259–1282, 2003.