



HAL
open science

Méthode algébrique pour l'optimisation multiniveau - Application à l'optimisation d'un avion d'affaires supersonique

Abderrahmane Benzaoui, Régis Duvigneau

► **To cite this version:**

Abderrahmane Benzaoui, Régis Duvigneau. Méthode algébrique pour l'optimisation multiniveau - Application à l'optimisation d'un avion d'affaires supersonique. 9e colloque national en calcul des structures, CSMA, May 2009, Giens, France. hal-01419957

HAL Id: hal-01419957

<https://hal.science/hal-01419957>

Submitted on 20 Dec 2016

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.



Distributed under a Creative Commons Attribution 4.0 International License

Méthode algébrique pour l'optimisation multiniveau - Application à l'optimisation d'un avion d'affaires supersonique.

A. Benzaoui¹, R. Duvigneau¹

¹ Projet OPALE - INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis (France)
Abderrahmane.Benzaoui@sophia.inria.fr
Regis.Duvigneau@sophia.inria.fr

Résumé — Dans cet article, nous proposons une approche multiniveau permettant d'accélérer les algorithmes d'optimisation classiques. Cette approche utilise les vecteurs propres de la matrice Hessienne de la fonction coût pour définir des sous-espaces de conception. La recherche de l'optimum est alors effectuée successivement dans ces sous-espaces de dimension variable. La méthode proposée est validée sur un problème de reconstruction de forme, puis appliquée à l'optimisation d'un avant projet d'avion d'affaires supersonique.

Mots clés — optimisation multiniveau, décomposition du Hessien, accélération de la convergence.

1 Introduction

Les algorithmes d'optimisation multidisciplinaires sont souvent confrontés au problème de dimensionnalité de l'espace de conception. Plus le nombre de paramètres est élevé, plus la recherche de l'optimum est coûteuse. Cette augmentation du coût n'est pas due uniquement à la taille du problème, mais aussi à sa raideur, c'est à dire au faible taux de convergence.

Des études précédentes ont été menées pour accélérer la convergence de ces algorithmes lorsque le nombre de paramètres est élevé en utilisant des méthodes hiérarchiques. Ces études s'inspirent de la méthode multigrille, utilisée pour résoudre des équations aux dérivées partielles. Plus particulièrement, Jameson et al [5] ont généralisé une méthode multigrille pour l'optimisation aérodynamique en résolvant simultanément les équations de l'écoulement et celles du problème d'optimisation correspondant. De même, Lewis et Nash [6, 7] ont utilisé le concept multigrille pour l'optimisation des systèmes gouvernés par des équations différentielles. Par ailleurs, Désidéri et al [4, 1] ont développé un algorithme hiérarchique pour l'optimisation de forme dans le cas d'une paramétrisation par une courbe de Bézier ou par la méthode Free-Form Deformation (FFD) [9]. Cet algorithme utilise la propriété d'élevation du degré des courbes de Bézier pour assurer le transfert d'informations d'un niveau grossier à un niveau plus fin.

Dans les études précédentes, le choix des sous-espaces de conception est lié à la paramétrisation des formes et peut ne pas être efficace [3]. Dans ce travail, nous proposons une méthode multiniveau plus générale permettant l'accélération de la convergence des algorithmes d'optimisation. Contrairement aux études précédentes, cette technique ne repose pas sur la paramétrisation pour définir les sous-espaces de conception. Elle peut donc être employée pour d'autres problèmes

que l'optimisation de forme. L'idée principale de cette méthode est de construire les sous-espaces de conception à partir des modes propres du Hessien de la fonctionnelle. En effet, par analogie avec la théorie multigrille, le taux de convergence dépend des valeurs propres de la matrice Hessienne. Ainsi, avec une stratégie multiniveau adéquate, on sélectionne les modes de faible taux de convergence pour y affiner l'optimisation avec plus d'efficacité.

Afin de valider la méthode que nous avons développée, cette dernière est testée sur un problème de reconstruction de forme géométrique dont la solution est connue. Il s'agit de la recherche de la meilleure approximation d'une forme donnée, par une courbe de Bézier. Dans ce problème, la fonction coût à minimiser est l'erreur d'approximation et les paramètres d'optimisation sont les coordonnées des points de contrôle de la courbe de Bézier. On montre qu'il est possible d'atteindre la solution théorique avec un coût plus faible qu'une méthode d'optimisation classique.

Après l'avoir validée, la méthode multiniveau est appliquée à un problème d'optimisation d'un avion d'affaires supersonique (SSBJ). Il s'agit d'un problème multidisciplinaire qui fait intervenir l'aérodynamique, la structure et la propulsion [8]. L'objectif est de minimiser la masse totale de l'avion au décollage tout en respectant des contraintes liées au rayon d'action, à la distance de décollage et à la vitesse d'approche. Bien que ce problème soit difficile à cause de la multimodalité et de la complexité des contraintes, l'approche proposée permet d'atteindre de meilleures performances aérodynamiques avec un moindre coût de calcul.

Dans toutes les applications présentées dans cet article, l'optimisation est réalisée grâce à la méthode Multidirectional Search Algorithm (MSA) de Torczon [2]. Cependant, d'autres méthodes peuvent être employées. Plus particulièrement, nous travaillons actuellement sur l'implémentation d'une approche de type Particle Swarm Optimization (PSO) dans l'algorithme multiniveau.

2 Optimisation multiniveau

Soit f la fonction coût à minimiser, E l'espace de conception et $X \in E$ le vecteur de conception (de n paramètres) de cette fonction. Soit $X_0 \in E$ le vecteur initial. Dans la méthode multiniveau, afin de définir les sous-espaces de recherche, on commence par calculer la matrice Hessienne $H(X_0)$ (de façon exacte ou approchée) de la fonction coût au point X_0 et déterminer ses vecteurs propres $V = (v_1, \dots, v_n)$. Ces derniers sont classés par ordre de valeur propre croissante (ou par valeur absolue de valeur propre croissante si le Hessien n'est pas défini positif en ce point). En effet, les directions correspondant aux petites valeurs propres sont des directions de faibles variations de la fonction f , dans lesquelles les algorithmes d'optimisation peinent à converger. En revanche, dans les directions correspondant aux grandes valeurs propres, la convergence est rapide. L'idée de notre approche multiniveau est alors de consacrer davantage d'efforts dans les directions de faibles taux de convergence. Supposons que le processus d'optimisation soit arrivé à un niveau l où le vecteur de paramètres obtenu est $X^{(l)} \in E$. L'algorithme multiniveau calcule la solution $X^{(l+1)} \in E$ au niveau $l+1$ en ajoutant un terme de correction permettant de minimiser la fonction coût. Le paramètre recherché est donc le terme de correction qu'on détermine de la façon suivante :

Supposons que le niveau $l+1$ est caractérisé par $m = m_{l+1}$ paramètres ($m \leq n$) et considérons la base V_m et le sous-espace E_m définis par :

$$V_m = (v_1, \dots, v_m)$$

$$E_m = \{z = V_m y \mid y \in \mathfrak{R}^m\}$$

On choisit le vecteur de correction comme un vecteur de E_m . Ainsi, pour obtenir la meilleure correction, l'algorithme d'optimisation recherche un vecteur $y \in \mathfrak{R}^m$ tel que la fonction :

$$g(y) = f(X^{(l)} + V_m y) \quad (1)$$

est minimisée. La minimisation de la fonction g est effectuée par un algorithme d'optimisation classique (l'algorithme MSA dans notre cas). Notons que pour une méthode comme la méthode MSA, le coût d'optimisation de la fonction g dépend de la dimension du vecteur y . Comme ce dernier est inférieur ou égal à n , le coût d'optimisation des sous-niveaux est inférieur à celui du niveau le plus fin. Si y^* est le minimum de la fonction g , le vecteur de conception au niveau $l + 1$ s'écrit alors :

$$X^{(l+1)} = X^{(l)} + V_m y^*$$

Cette écriture nous permet à la fois de passer d'un niveau grossier à un niveau fin ou inversement sans aucune perte d'information. La méthode multiniveau emploie plusieurs niveaux, ordonnés en cycles, par analogie avec la terminologie de la méthode multigrille. Ainsi, un cycle est défini par une séquence de niveaux dont les nombres de paramètres (dimensions des sous-espaces) sont respectivement m_1, m_2, \dots, m_{nl} et dont les nombres d'itérations correspondant sont $it_1, it_2, \dots, it_{nl}$, avec nl le nombre de niveaux dans le cycle (cf. tableau (1)). Ce cycle peut être répété autant de fois que nécessaire. A chaque nouveau cycle, le Hessien est ré-évalué au dernier point obtenu par l'algorithme d'optimisation et les sous-espaces de recherches sont mis à jours. Pour récapituler, l'algorithme multiniveau est résumé ci-dessous :

1. Initialiser le vecteur de conception X_0 ;
2. initialiser le compteur de cycles à $k = 0$;
3. tant que $k < nc$ et le critère d'arrêt non atteint, effectuer l'optimisation sur un cycle multiniveau :
 - (a) évaluer la matrice Hessienne $H(X_k)$ et déterminer ses vecteurs propres ;
 - (b) initialiser le compteur de niveaux à $l = 1$;
 - (c) prendre $X^{(l)} = X_k$;
 - (d) tant que $l \leq nl$ optimiser sur un niveau :
 - i. choisir une base V_m constituée de $m = m_l$ vecteurs propres,
 - ii. utiliser un algorithme d'optimisation pour déterminer un vecteur de correction $y^* \in \mathfrak{R}^m$ qui minimise la fonction $g(y) = f(X^{(l)} + V_m y)$;
 - iii. $X^{(l+1)} \leftarrow X^{(l)} + V_m y^*$;
 - iv. $l \leftarrow l + 1$ et aller à (3d) ;
 - (e) $X_{k+1} \leftarrow X^{(l)}$;
 - (f) $k \leftarrow k + 1$ et aller à (3) ;
4. $X^* \leftarrow X_k$.

Le critère d'arrêt que nous utilisons pour l'étape (3) de l'algorithme précédent est la diminution relative de la fonction coût. Tant que cette diminution est supérieure à une valeur donnée et que le nombre de cycle est inférieur à une limite prédéfinie nc , le processus d'optimisation continue.

3 Application au problème de reconstruction de forme

Le problème de reconstruction de forme est un exemple simple permettant la validation de l'algorithme multiniveau. Il présente l'avantage de facilité de calcul de la fonction coût (qui est quadratique) ainsi que de son Hessien. Dans ce problème on souhaite approcher une fonction

niveau	itérations correspondantes
<i>fin</i>	it_1 it_1
<i>grossier</i>	it_2

Tableau 1 – schéma descriptif d'un V-cycle à deux niveaux

cible donnée, $F_0(t)$, par une courbe de Bézier de degré n , $F(t, X)$, où $X = (x_1, \dots, x_n)^T$ représente les coordonnées des points de contrôle. Dans notre cas, la fonction cible est, elle aussi, une courbe de Bézier de degré $n_0 > n$. Ces deux fonctions sont données par :

$$F_0(t) = \sum_{k=0}^{n_0-1} B_{n_0-1}^k(t)x_{0k+1} \quad (2)$$

et

$$F(t, X) = \sum_{k=0}^{n-1} B_{n-1}^k(t)x_{k+1} \quad (3)$$

pour $t \in [0, 1]$. Dans l'équation (2), x_{01}, \dots, x_{0n_0} sont les coordonnées des points de contrôle de la fonction cible. De même, dans les équations (2) et (3), B_{n-1}^k représentent les polynômes de Bernstein donnés par :

$$B_n^k(t) = C_n^k t^k (1-t)^{n-k}$$

avec

$$C_n^k = \frac{n!}{k!(n-k)!}$$

En se basant sur la méthode des moindres carrés, le problème d'approximation est équivalent à un problème d'optimisation dont la fonction coût à minimiser représente l'écart entre les deux courbes. Soit :

$$f(X) = \int_0^1 (F(t, X) - F_0(t))^2 dt \quad (4)$$

Dans cette étude, la fonction cible est de degré 14 tandis que la fonction approchée est de degré 8. Par ailleurs, le Hessien est calculé analytiquement en dérivant l'expression (4).

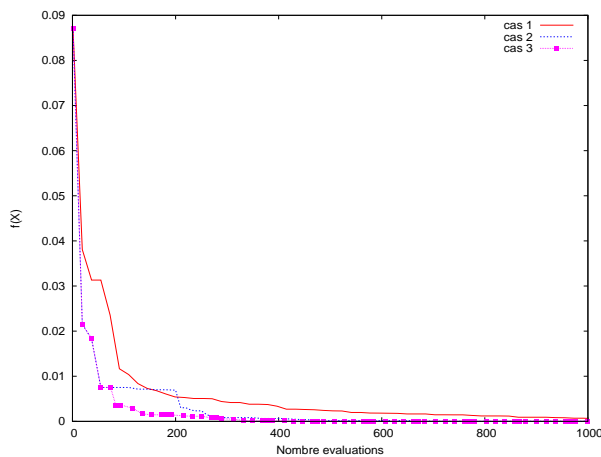


Figure 1 – Evolution de la fonction coût en fonction du nombre d'évaluations pour le problème de reconstruction de forme.

La figure (1) représente l'évolution de la fonctionnelle en fonction du nombre d'évaluation et ce pour trois cas : en utilisant une méthode MSA classique avec un seul niveau (cas 1), en utilisant

une méthode MSA avec un seul niveau mais en travaillant dans une base propre du Hessien (cas 2) et en utilisant l'algorithme multiniveau (cas 3). Les cycles du cas 3 sont définis par la séquence des niveaux (8/4) avec 3 itérations par niveau. Cela constitue un V-cycle à deux niveaux, selon la terminologie des méthodes multigrille. Notons que, dans la figure (1), le choix de la représentation de l'évolution en fonction du nombre d'évaluations se justifie par le fait que, dans un problème industriel, la plus grande partie du coût d'optimisation est due à l'évaluation de la fonctionnelle.

On constate sur cette figure que le fait d'utiliser la décomposition spectrale du Hessien permet de réduire considérablement le coût de calcul et d'accélérer l'algorithme d'optimisation. Ce gain est encore meilleur lorsque on utilise une approche multiniveau avec un faible nombre d'itérations par niveau.

4 Optimisation d'un avant projet d'avion d'affaires supersonique

L'optimisation d'un avion d'affaires supersonique (SSBJ) est un cas test intéressant vu son caractère industriel multidisciplinaire et complexe. Ce problème, qui fait intervenir l'aérodynamique, la propulsion et la structure de l'avion, a été utilisé par Sobieszcanski-Sobieski et al dans leur étude sur les méthodes de type "BLISS" [8]. Dans notre étude, nous nous en servons pour montrer l'efficacité de la méthode multiniveau et sa capacité à traiter des problèmes complexes. L'objectif est de minimiser la masse de l'avion au décollage, tout en respectant des contraintes sur le rayon d'action R_{act} , sur la distance de décollage d_{dec} et sur la vitesse d'approche v_{app} . Ces contraintes sont :

$$\begin{cases} R_{act} > 6500 \text{ km} \\ d_{dec} < 1828 \text{ m} \\ v_{app} < 70 \text{ m/s} \end{cases}$$

Le dimensionnement et le calcul des performances de l'avion sont effectués à l'aide de programmes fournis par Dassault Aviation qui font intervenir un jeu de quinze variables de conception. Par ailleurs, les contraintes sur les performances de l'avion et sur les variables de conception sont prises en compte dans notre algorithme, par l'intermédiaire de termes de pénalité.

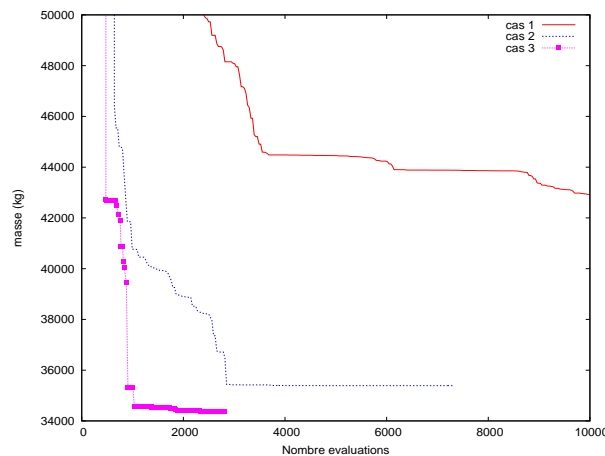


Figure 2 – Evolution de la masse de l'avion en fonction du nombre d'évaluations.

La figure (2) montre l'évolution de la fonction coût (masse de l'avion) en fonction du nombre d'évaluations et ce pour trois cas comme précédemment. Les cycles utilisés dans le cas 3 sont définis par la séquence des niveaux (15/7) avec 5 itérations par cycle pour le niveau 15 et 6 itérations par cycle pour le niveau 7. On constate sur cette figure que l'utilisation d'une base propre du Hessien et de l'approche multiniveau permettent non seulement d'accélérer la convergence, mais

d'atteindre de meilleures performances. Cependant, vu la multimodalité du problème et la complexité des contraintes, la méthode MSA peut facilement être piégé par un minimum local, ce qui rend le réglage des paramètres de calcul très délicat car une petite modification peut dégrader les résultats. Par ailleurs, il convient de signaler que pour les résultats présentés ici, le calcul du Hessien est réalisé par différences finies, ce qui est très coûteux et peu précis. Nous travaillons actuellement sur une évaluation approchée par l'utilisation d'un métamodèle ou d'un modèle quadratique global.

5 Conclusion

Nous avons présenté dans cet article une approche d'optimisation multiniveau basée sur la décomposition spectrale du Hessien. Cette approche est très générale et peut s'appliquer à une grande variété de problèmes ou utiliser différentes méthodes d'optimisation. Dans cette approche, la recherche de l'optimum d'une fonction se fait dans des sous-espaces engendrés par les vecteurs propres du Hessien, ce qui permet d'affiner l'optimisation dans les directions de faible taux de convergence.

La méthode que nous avons présentée est validée sur un problème de reconstruction de forme et montre une véritable efficacité en terme d'accélération de convergence. Notons que cette efficacité dépend du nombre de conditionnement du Hessien. Plus ce nombre est grand et plus la méthode multiniveau est efficace.

La méthode est ensuite appliquée à l'optimisation de la conception d'un avion d'affaires supersonique. Là encore, l'approche multiniveau permet de réduire considérablement le coût de résolution et en plus, d'atteindre de meilleures performances, à condition de choisir judicieusement les paramètres de calcul.

Références

- [1] B. Abou El Majd, R. Duvigneau, J. A. Désideri, Aerodynamic Shape Optimization using a Full Adaptive Multilevel Algorithm, *ERCOFTAC Conference Design Optimization : Methods and Applications*, Iles Canaries, Avril 2006.
- [2] J. E. Dennis, V. Torczon, Direct Search Method on Parallel Machines, *SIAM Journal on Optimization*, 1, 448–474, 1991.
- [3] J. A. Désideri, Two-level ideal algorithm for shape optimization, *International Conference on Advances in Numerical Mathematics*, Moscow, septembre 2005.
- [4] J. A. Désideri, B. Abou El Majd, A. Janka, Nested and self-adaptive Bézier parameterization for shape optimization, *International Conference on Control, Partial Differential Equation and Scientific Computing*, Pékin, septembre 2004.
- [5] A. Jameson, L. Martinelli, Optimum Aerodynamic Design using the Navier-Stokes Equations, *Theoretical and Computational Fluid Dynamics*, 10, 213–277, 1998.
- [6] R. Lewis, S. Nash, A Multigrid Approach to the Optimization of Systems Governed by Differential Equations, *American Institute of Aeronautics and Astronautics*, Reston, VA, AIAA-2000-4890, 2000.
- [7] R. Lewis, S. Nash, Model Problems for Multigrid Optimization of Systems Governed by Differential Equations, *SIAM J. Sci Comput.*, 26, 1811–1837, 2005.
- [8] J. Sobieszczanski-Sobieski, J. S. Agte, R. R. Sandusky, Bilevel Integrated System Synthesis, *AIAA Journal*, 38 (1), 164–172, Janvier 2000.
- [9] T. Sederberg, S. Parry, Free-form deformation of solid geometric models, *Computer Graphics*, 20 (4), 151–160, 1986.