Modèles réduits pour l'optimisation robuste multiobjectifs

Vincent Baudoui

17 décembre 2008

1 Introduction

La conception des systèmes aéronautiques passe aujourd'hui par la mise en œuvre de simulations numériques complexes à même de représenter les différents phénomènes physiques mis en jeu. Ces phénomènes étant de mieux en mieux connus, les modèles se complexifient et les codes de calcul deviennent de plus en plus coûteux... L'optimisation numérique est alors confrontée au problème de la durée importante des calculs. Une approche consiste à utiliser un modèle de substitution réalisé à partir d'un nombre réduit d'évaluations du code de calcul, et possédant un temps de réponse négligeable.

Dans le contexte industriel, les solutions optimales recherchées doivent de plus satisfaire à des contraintes de robustesse liées aux incertitudes du problème. La sensibilité des solutions optimales peut alors être analysée a posteriori, ou bien être directement prise en compte dans le processus d'optimisation.

L'application visée ici est l'optimisation multiobjectifs de la chambre de combustion d'un moteur d'avion, dans le but de minimiser les émissions polluantes et d'assurer la stabilité du foyer.

2 Modèles réduits

Face à un code de calcul coûteux, les méthodes d'optimisation directes (c'est-à-dire qui travaillent directement sur le code) deviennent impraticables, car elles nécessitent de réaliser un grand nombre d'évaluations. On utilise alors un modèle de substitution construit à partir d'un nombre d'évaluations plus limité. Les valeurs fournies par un tel modèle restent bien sûr approchées par rapport au code de calcul de départ, mais elles permettent néanmoins d'obtenir de bons résultats d'optimisation.

La construction d'un modèle réduit est basée sur un plan d'expériences qui définit les points à échantillonner. Partant par exemple sur la base d'un hypercube latin (LHS), le plan d'expériences peut être construit de manière adaptative : à chaque itération, on construit un modèle réduit puis on détermine les points les plus intéressants à calculer pour l'itération suivante, dans les zones qui semblent mal décrites par le modèle ou qui paraissent prometteuses pour la recherche d'optima. Le plan d'expériences ainsi généré permettra de réaliser un modèle réduit de meilleure qualité et qui conduira au point optimal recherché.

Différents types de modèles réduits peuvent être utilisés afin d'obtenir un modèle global du code de calcul [JCS01] : modèles polynômiaux, krigeage, modèles à base de splines (MARS), réseaux de neurones, réseaux à fonctions radiales de base (RBF), etc.

3 Robustesse

L'optimisation robuste cherche à optimiser un système en prenant en compte les différentes sources d'incertitude du problème.

En optimisation "classique", on cherche à résoudre un problème du type :

$$\underset{x \in \mathbb{R}^n}{\operatorname{argmin}} \; f(x),$$

où x représente les n paramètres d'entrée qui permettent de décrire le système, et f l'objectif que l'on souhaite optimiser. Dans l'application qui nous concerne, on va par exemple chercher le positionnement d'un injecteur qui minimise l'émission de polluants du moteur.

Du fait des incertitudes de fabrication, on ne peut pas être certain de réaliser en pratique le système optimal issu de la résolution de ce problème. Les paramètres vont en effet varier dans des intervalles de tolérance autour des valeurs optimales souhaitées. Selon l'emplacement réel de l'injecteur dans le système, les performances pourront donc différer des résultats optimaux calculés. C'est pourquoi on s'oriente désormais vers la recherche de solutions dites "robustes" : on souhaite obtenir des solutions optimales peu sensibles à des variations dans le système. Il faut donc considérer $f(x+\delta)$ à la place de $f(x)$ dans le problème d'optimisation, avec δ la perturbation due à la fabrication [BS07]. Le but de l'optimisation robuste est alors de trouver une solution dont les performances ne se dégradent pas trop lorsque δ varie. Sur l'exemple de la figure 1, la solution 2 sera ainsi préférée à la solution 1 car elle est plus robuste.

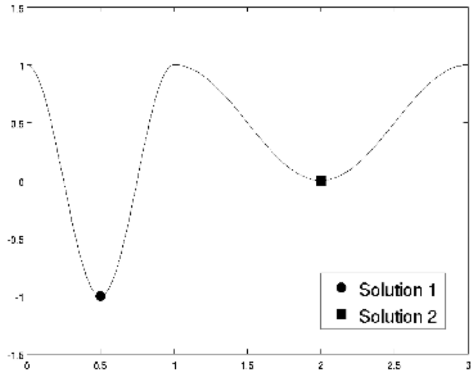

Fig. 1 – Optimum robuste (solution 2)

On introduit une mesure de la robustesse d'un point de l'espace de recherche basée sur la notion de voisinage : on va observer le comportement de la fonction objectif autour du point considéré.

Plusieurs formulations de la robustesse ont été imaginées. La plus évidente consiste à prendre en compte la plus mauvaise valeur de la fonction objectif dans le voisinage du point. Le problème d'optimisation devient alors :

$$\underset{x \in \mathbb{R}^n}{\text{argmin}} \sup_{\xi \in V(x)} f(\xi),$$

avec $V(x)$ un voisinage du point x. La résolution de ce problème permet d'obtenir la solution pour laquelle la valeur maximale de la fonction dans le voisinage est la plus petite. On s'assure ainsi que tout point du voisinage de cette solution robuste aura au pire une performance égale à ce maximum.

On peut aussi considérer la robustesse relativement à la moyenne et à la variance de la fonction objectif dans le voisinage, ce qui permet de prendre en compte la distribution de probabilité des erreurs sur les paramètres d'entrée si celle-ci est connue. Enfin, la robustesse peut être étudiée vis-à-vis des contraintes : on ne souhaite pas que le voisinage de l'optimum dépasse les limites imposées par les contraintes du problème.

La prise en compte de la robustesse dans un problème d'optimisation classique peut se faire de deux manières : il est possible comme on l'a vu plus haut de transformer le problème initial en un nouveau problème d'optimisation classique, mais on peut aussi considérer que la robustesse est un objectif qui entre en contradiction avec l'objectif de performance (comme illustré sur la figure 1). On est alors amené à considérer un problème d'optimisation multiobjectifs, qui permettra de choisir a posteriori le degré de robustesse que l'on souhaite avoir vis-à-vis des performances qui en découlent.

4 Optimisation multiobjectifs

L'optimisation multiobjectifs cherche à optimiser simultanément plusieurs objectifs interdépendants qui peuvent être contradictoires. Un tel problème n'admet généralement pas une solution unique mais un ensemble de solutions équivalentes appelé "front de Pareto" [CS03].

La résolution d'un problème d'optimisation multiobjectifs peut se faire de différentes façons. La solution la plus simple consiste à se ramener à un problème mono-objectif par une agrégation des différents objectifs, mais on n'obtient alors qu'une seule solution du front optimal. Des algorithmes ont donc été développés afin de pouvoir calculer le front de Pareto complet. Ce sont généralement des heuristiques, car un tel problème est considéré comme "difficile". On peut notamment citer l'algorithme d'agrégation dynamique, la programmation par but ou la méthode ϵ-contrainte, ainsi que tous les algorithmes de type évolutionnaire (avec notamment les algorithmes génétiques et les essaims particulaires). Des méthodes issues de l'optimisation mono-objectif ont aussi été étendues pour prendre en compte plusieurs objectifs (recuit simulé, recherche tabou, etc.). Enfin, des recherches sont effectuées aujourd'hui sur des méthodes dites "hybrides" mettant en jeu différents algorithmes de manière séquentielle ou parallèle.

5 Conclusion

Les multiples approches abordées ici vont être mises en œuvre au cours de la thèse, dans le but d'optimiser la chambre de combustion d'un moteur d'avion selon plusieurs objectifs. L'optimisation se basera sur un modèle réduit enrichi grâce à un plan d'expériences adaptatif, et les incertitudes de fabrication du système seront prises en compte au cours du processus par une étude du voisinage des solutions envisagées.

Références

[BS07] Hans-Georg Beyer and Bernhard Sendhoff. Robust optimization - a comprehensive survey. *Computer Methods in Applied Mechanics and Engineering*, 196(33-34) :3190–3218, July 2007.

[CS03] Yann Collette and Patrick Siarry. *Multiobjective Optimization : Principles and Case Studies.* 2003.

[JCS01] R. Jin, W. Chen, and T. W. Simpson. Comparative studies of metamodelling techniques under multiple modelling criteria. *Structural and Multidisciplinary Optimization*, 23(1) :1–13, December 2001.