

Optimisation convexe pour l'allocation de commandes pour une baie multi-moteurs d'un lanceur réutilisable

Leonie Gallois¹, Louis Thioulouse², Florian Dietrich²

¹ INSA Toulouse, France

gallois@insa-toulouse.fr

² ONERA, France

{louis.thioulouse,florian.dietrich}@onera.fr

Mots-clés : *recherche opérationnelle, allocation, optimisation convexe, multi-objectif, système sur-actionné, embarqué, baie multi-moteurs, lanceur réutilisable, angle de braquage, problème statique, ensemble admissible, couple, moment, poussée, cône de second ordre, ECOS, CVX, Matlab*

1 Raisons

Des acteurs publics et privés dans la construction de lanceurs s'orientent désormais vers des lanceurs réutilisables pour des raisons économiques. Un lanceur réutilisable a pour mission d'envoyer une charge utile en orbite comme un satellite et de faire revenir son étage inférieur sur Terre sur un site prédéfini. Ainsi il est possible de l'utiliser de nouveau lors d'une future mission. La partie propulsive du 1er étage d'un lanceur réutilisable comporte plusieurs moteurs qu'on nomme baie multi-moteurs. Un étage de lanceur peut ne disposer que d'un moteur. Cependant le choix technologique d'une baie de plusieurs moteurs augmente la fiabilité du lanceur.

2 Problématique

La baie multi-moteurs d'un lanceur réutilisable est un système sur-actionné puisqu'elle possède plus de degrés de liberté physiques que d'actionneurs. Elle nécessite donc un module d'allocation afin d'être contrôlée. Le module a pour tâche de répartir de façon statique des consignes de poussées globales entre les différents moteurs en respectant les limites physiques des actionneurs et en limitant l'usure des composants. L'algorithme d'allocation est appliqué à une baie multi-moteurs, il doit donc satisfaire des contraintes de convergence, de rapidité et de précision afin d'être embarquable. C'est pourquoi l'on cherche à appliquer l'optimisation convexe pour résoudre ce problème d'allocation.

3 Réalisations

Élaboration d'un état de l'art sur les différentes techniques d'allocation des systèmes sur-actionnés. Modélisation de la baie multi-moteurs dans le repère lanceur en coordonnées cartésiennes. Linéarisation du problème d'allocation grâce au passage du produit vectoriel en produit matriciel. Formulation des contraintes d'optimisation comprenant les contraintes d'allocation ainsi que les contraintes de saturation physique. Analyse de convexité des contraintes et relaxation d'une des contraintes. Construction et tests de plusieurs fonctions multi-objectif pour minimiser l'erreur d'allocation ainsi que l'usure des composants. Implémentation du problème d'optimisation convexe avec l'interface CVX sous Matlab. Tests du problème avec le solveur ECOS sur différentes architectures et pour des consignes attestant de la robustesse

de l'algorithme. Proposition d'une deuxième convexification par restriction du domaine admissible. Quantification des solutions admissibles non atteignables et analyse de la consigne pire cas. Caractérisation des consignes réalisables dans le but d'améliorer les performances de l'algorithme par la suite.

4 Méthodologie de la phase de tests

Les travaux ont été codés sous Matlab puisque le cahier des charges a été pensé pour Matlab. Le choix de l'outil d'optimisation s'est porté sur l'interface CVX. Il s'agit d'une interface d'optimisation convexe qui suit des règles d'implémentation pour assurer la convexité du problème à résoudre. Elle a été développée par Micheal C. Grant et Stephen P. Boyd et exclusivement conçue pour l'optimisation convexe.

L'outil CVX interface différents solveurs comme le solveur ECOS qui a été sélectionné pour ces travaux. Il s'agit d'un solveur libre de droit qui se base sur les cônes de second ordre. ECOS constitue le solveur le plus rapide¹ pour des problèmes avec peu de variables et de paramètres pouvant être utilisés avec une licence libre de CVX. Les instances testées sont issues de commandes réelles simulant l'atterrissage du lanceur. Elles contiennent aussi bien des commandes réalisables venant saturer les actionneurs que non réalisables pour tester la robustesse des solutions proposées par l'algorithme d'allocation.

Les différentes architectures testées se basent sur des baies multi-moteurs physiquement réalistes. Les fonctions multi-objectif testées sont issues de techniques classiques de la littérature. La sélection de la fonction multi-objectif s'est faite selon la robustesse de la fonction appliquée à différents poids, le respect du cahier des charges ainsi que la continuité des solutions en braquage et poussée pour les différents moteurs.

5 Résultats et interprétation

L'algorithme d'allocation respecte le cahier des charges élaboré en interne pour chaque commande réalisable. En effet, le temps d'exécution de l'algorithme sous Matlab est inférieur à une seconde. L'erreur sur la norme de poussée et l'erreur sur le couple restent inférieures ou égales à celles attendues. Enfin, les courbes des braquages et de la poussée des différents moteurs en fonction du temps sont toutes lisses. La superposition de leurs courbes montre que la commande a été équitablement répartie entre les moteurs.

6 Conclusion

Ce stage met à disposition de l'ONERA un algorithme d'allocation des commandes d'une baie multi-moteurs d'un lanceur réutilisable. En plus des contraintes d'allocation, le système est soumis à des contraintes physiques. On cherche ici à minimiser l'erreur d'allocation ainsi que l'usure des composants. Ce stage aura été l'occasion de montrer l'efficacité de l'optimisation convexe pour résoudre ce type de problème. L'algorithme a été testé et validé par un cahier des charges sur différentes architectures. La diversité des consignes de tests montre la robustesse de l'algorithme. Ce rendu est accompagné de la caractérisation de l'ensemble des consignes réalisables. Ainsi, l'algorithme pourra encore être amélioré par la suite.

Références

- [1] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598) :671–680, 1983.
- [2] Paolo Toth and Daniele Vigo. *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, 2002.