



Resolution of mixed-variables problems in LocalSolver

Olivier Rigal

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

February 7, 2020

Résolution de problèmes d'optimisation à variables mixtes dans LocalSolver

Olivier Rigal

LocalSolver
36 Avenue Hoche, 75008 PARIS
origal@localsolver.com

Mots-clés : *recherche locale, variables mixtes, solveur*

1 Contexte

LocalSolver est un solveur d'optimisation mathématique de type *model-and-run* combinant différentes techniques de recherche opérationnelle [2]. Celui-ci utilise notamment une heuristique de recherche locale, mais également la programmation linéaire et non-linéaire.

Nous étudions ici la résolution des problèmes d'optimisation à variables mixtes. Ces problèmes sont ceux utilisant des produits de variables combinatoires et continues. Ceux-ci peuvent apparaître, par exemple, dans les problèmes de planification de production où une variable booléenne active la fabrication d'un produit donné et une variable continue décide de la quantité produite.

Nous nous limitons aux problèmes pour lesquels, lorsque l'affectation combinatoire est fixée, le problème continu résiduel est linéaire. La recherche locale classique est actuellement mise en difficulté par ce type de problème, car peu adaptée au traitement de sous-problème utilisant des variables continues.

2 Technique de résolution

La technique de résolution implémentée au sein de LocalSolver consiste à ajouter une phase de réparation après chaque mouvement de la recherche locale. Plus précisément, lors de chaque itération, un mouvement de recherche locale modifie l'affectation combinatoire dans la solution courante. Ensuite, les variables combinatoires sont fixées et une tentative de réparation de la partie continue du problème est effectuée. Cette réparation consiste à résoudre à l'optimum le sous-problème résiduel continu et linéaire grâce à l'algorithme du simplexe. L'ensemble des modifications est alors évalué et accepté ou rejeté par l'heuristique de LocalSolver.

Cette technique de résolution permet d'allier la puissance de la recherche locale qui donne d'excellents résultats sur les problèmes purement combinatoires et l'algorithme du simplexe qui permet de résoudre des problèmes linéaires continus dans des temps courts. Cette méthode a déjà été mise en place sur des problèmes spécifiques comme celui d'*Unit Commitment* [3] et celui d'*Inventory Routing* [1]. L'implémentation de la technique au sein de LocalSolver a ainsi permis de rendre générique cette méthode pour tout type de problèmes à variables mixtes.

3 Amélioration de la technique

3.1 Accélération de la descente

Avec une implémentation naïve, nous pouvons remarquer qu'une proportion importante du temps de résolution est consacrée à résoudre les sous-problèmes résiduels continus. Afin de

limiter ce phénomène, un ensemble de vérification a été mis en place afin d'identifier rapidement certains sous-problèmes qui ne permettront pas d'améliorer la solution courante. Nous utilisons par exemple le lemme de Farkas qui permet d'obtenir un certificat d'infaisabilité d'un problème linéaire continue. Ces certificats sont exploités afin de détecter des sous-problèmes résiduels infaisable et qu'il sera donc impossible de réparer.

3.2 Ciblage des variables à modifier

La seconde amélioration apportée à la technique consiste en l'amélioration du ciblage lors du choix des variables à modifier au cours d'un mouvement de recherche locale touchant l'affectation combinatoire. Pour cela, nous utilisons les informations fournies par l'algorithme du simplexe. Lorsque la solution courante est infaisable, nous exploitons le certificat d'infaisabilité du lemme de Farkas afin de se diriger vers une solution faisable. Lorsque la solution courante est faisable, nous nous servons de l'information duale afin d'améliorer la valeur optimale du sous-problème résiduel.

4 Résultats

L'implémentation de cette technique au sein de LocalSolver a permis d'améliorer significativement ses performances sur différentes classes de problèmes. Ces améliorations sont particulièrement importantes sur le problème de *Network Design* pour lequel nous avons généré un ensemble d'instances de tests et sur le problème d'*Inventory Routing* où les tests ont été effectués sur les instances *Inventory Routing over a finite periodic planning horizon* de l'OR Library¹. Par exemple, sur ces dernières instances, un gain de plus de 25% est observé avec un temps d'optimisation de 5 minutes grâce à l'implémentation de la méthode par rapport à la version précédente de LocalSolver.

Références

- [1] T. Benoist et F. Gardi *Randomized Local Search for Real-Life Inventory Routing*, Transportation Science, 2011
- [2] F. Gardi, T. Benoist, J. Darlay, B. Estellon, et R. Megel. *Mathematical Programming Solver Based on Local Search*, Wiley, 2014.
- [3] F. Gardi et K. Nouioua. *Local Search for Mixed-Integer Nonlinear Optimization : A Methodology and an Application*, Springer Berlin Heidelberg, 2010

1. <http://people.brunel.ac.uk/~mastjjb/jeb/info.html>