

Optimisation continue dans LocalSolver

Simon BOULMIER

Innovation 24, Paris (France),
Université Grenoble Alpes, Grenoble (France)

Mots-clés : *LocalSolver*, *optimisation continue*, *recherche locale*

Contexte

LocalSolver est un solveur de programmation mathématique, développé depuis une dizaine d'années par l'équipe d'Innovation 24¹, entreprise d'édition, de service et de conseil en optimisation. Initialement conçu pour traiter les problèmes combinatoires en variables binaires, son modèle de résolution est basé sur des heuristiques et méta-heuristiques de recherche locale.

Les excellents résultats de la recherche locale, l'hybridation avec des méthodes avancées - telles que le simplexe primal-dual - et l'utilisation de structures de modélisation performantes lui ont permis de s'imposer dans le catalogue des solveurs industriels. L'objectif est aujourd'hui d'avancer vers un outil universel, capable de résoudre n'importe quel problème exprimable dans un langage de modélisation, et ce, sans avoir à se confiner à des sous-classes de problèmes, telles que la programmation linéaire, quadratique, convexe, etc.

Objectifs

Ce travail marque le début d'une phase de développement qui a pour objectif de préparer une arrivée sur le marché de l'optimisation continue non-linéaire, ainsi que d'améliorer les performances du logiciel sur des problèmes d'optimisation en variables mixtes. Dans cette présentation, on s'intéressera aux problèmes continus non linéaires sous contraintes de bornes :

$$\begin{array}{l} \min f(x) \\ \text{sc. } \left\{ \begin{array}{l} x \in \mathbb{R}^n, \\ l \leq x \leq u. \end{array} \right. \end{array}$$

Un critère classique pour un algorithme qui résout ce problème est sa capacité à converger vers un optimum local du problème, ainsi que sa vitesse de convergence asymptotique. L'approche de LocalSolver est différente : il s'agit de fournir des solutions de bonne qualité en un temps raisonnable. Ainsi, le critère retenu pour un algorithme sera par exemple la qualité des solutions qu'il fournit, à temps de calcul fixé.

L'objectif de cette présentation est de montrer que des approches analogues à la recherche locale par voisinage variables, telles que celles qui composent le module combinatoire de LocalSolver [1], peuvent être étendues au cas de l'optimisation continue, et apporter un gain par rapport aux algorithmes classiques.

Contenu technique

Après une brève description du fonctionnement de LocalSolver sur les problèmes combinatoires, on expliquera comment les algorithmes classiques de l'optimisation continue peuvent être

1. <http://www.localsolver.com/about.html>

vus comme une recherche locale, en le sens qu'ils sont définis par un voisinage - un sous-ensemble de l'espace des solutions - couplé à une stratégie d'exploration et de sélection dans ce voisinage.

Nous détaillerons ensuite deux algorithmes élémentaires que sont la méthode de relaxation cyclique et la descente de gradient, voir par exemple [2]. Nous expliquerons comment les voisinages et stratégies d'exploration qui les définissent peuvent être rendus variables, améliorant ainsi les performances des algorithmes initiaux.

Nous présenterons enfin quelques résultats numériques, sur des fonctions test diverses et fortement linéaires issues de [3], dans le but d'illustrer les gains apportés par les approches proposées.

Références

- [1] F. Gardi. Towards a mathematical programming solver based on local search. *UPMC Sorbonne Universités*, 2013.
- [2] M. Minoux. *Programmation Mathématique*. Springer edition, 2008.
- [3] S. Finck, N. Hansen, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2010 : Presentation of the noiseless functions. *INRIA research report RR-6829*.