Projet de fin d'études / Stage recherche niveau master 2

Generic primal heuristics for VRPSolver

Equipe-projet Inria RealOpt
Université de Bordeaux

Recently, we have developed VRPSolver [8, 9], which is a generic solver for a certain class of combinatorial optimization problems. Problems in this class contain substructure(s), which can be described by a set of resource constrained paths in a given graph. Such structure is characteristic for many vehicle routing, scheduling, packing, resource allocation, and other problems.

Problems in this class can be formulated as (Mixed) Integer Linear Programs (MILP) and solved by one of many available MILP solvers which are based on the Branch-and-Cut method [7]. Nevertheless, even modern MILP solvers are often inefficient for these problems.

VRPSolver is based on Branch-Price-and-Cut method [3] and has a state-of-the-art performance for solving to optimality some classic vehicle routing and packing problems. However, to achieve the desired performance, one often should provide a very good initial solution of the problem in hand. Thus, although our solver is relatively generic (it can be applied to many problems), it often requires a problem-specific knowledge to generate good feasible solutions.

VRPSolver contains generic primal heuristics such as restricted master heuristic and diving heuristic [11]. However, these heuristics are usually slow and cannot be used to obtain rapidly an initial solution.

MILP solvers have several embedded primal heuristics, reviewed in [1, 4]. The main heuristics are *Relaxation Induced Neighbourhood Search* [2], *Local Branching* [6], *Feasibility Pump* [5], and *Polishing Heuristic* [10]. By taking inspiration from these works, the aim of the master internship is to **develop a fast generic heuristic for combinatorial optimization problems with resource constrained path structure**, in particular for vehicle routing problems. Such a heuristic should exploit the structure of this class of problems.

Références

[1] Timo Berthold. Primal heuristics for mixed integer programs. Master's thesis, Technischen Universität Berlin, 2006.

[2] Emilie Danna, Edward Rothberg, and Claude Le Pape. Exploring relaxation induced neighborhoods to improve mip solutions. *Mathematical Programming*, 102(1) :71–90, Jan 2005.

[3] Jacques Desrosiers and Marco E. Lübbecke. Branch-price-and-cut algorithms. In *Wiley Encyclopedia of Operations Research and Management Science*. American Cancer Society, 2011.

[4] Martina Fischetti and Matteo Fischetti. Matheuristics. In Rafael Martí, Pardalos Panos, and Mauricio G.C. Resende, editors, *Handbook of Heuristics*, pages 1–33. Springer International Publishing, Cham, 2016.

[5] Matteo Fischetti, Fred Glover, and Andrea Lodi. The feasibility pump. *Mathematical Programming*, 104(1) :91–104, Sep 2005.

[6] Matteo Fischetti and Andrea Lodi. Local branching. *Mathematical Programming*, 98(1) :23–47, Sep 2003.

[7] John E. Mitchell. Branch-and-cut algorithms for combinatorial optimization problems. In Panos M. Pardalos and Mauricio G. C. Resende, editors, *Handbook of applied optimization*. Oxford University Press, 2002.

[8] Artur Pessoa, Ruslan Sadykov, Eduardo Uchoa, and François Vanderbeck. A generic exact solver for vehicle routing and related problems. In Andrea Lodi and Viswanath Nagarajan, editors, *Integer Programming and Combinatorial Optimization*, volume 11480 of *Lecture Notes in Computer Science*, pages 354–369, Cham, 2019. Springer International Publishing.

[9] Artur Pessoa, Ruslan Sadykov, Eduardo Uchoa, and François Vanderbeck. A generic exact solver for vehicle routing and related problems. Cadernos do LOGIS 2019/2, Universidade Federal Fluminense, 2019.

[10] Edward Rothberg. An evolutionary algorithm for polishing mixed integer programming solutions. *INFORMS Journal on Computing*, 19(4) :534–541, 2007.

[11] Ruslan Sadykov, François Vanderbeck, Artur Pessoa, Issam Tahiri, and Eduardo Uchoa. Primal heuristics for branch-and-price : the assets of diving methods. *INFORMS Journal on Computing*, 31(2) :251–267, 2019.

Profil recherché

Etudiant en Master 2 ou école d'ingénieur.

Connaissances requises :
— Programmation linéaire (en nombres entiers)
— Programmation C++

Connaissances souhaitées dans un ou plusieurs de ces domaines :
— Programmation Julia
— Méthode de génération de colonnes
— Problèmes de tournées de véhicules
— Programmation dynamique

Le stage s'effectuera au sein de l'équipe Inria ReAlOpt localisée à l'Institut de Mathématiques de Bordeaux.

Contacts

ruslan.sadykov@inria.fr