Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 108C (2017) 715–724

Procedia Computer Science

International Conference on Computational Science, ICCS 2017, 12-14 June 2017, Zurich, Switzerland

Data Mining Approach for Feature Based Parameter Tunning for Mixed-Integer Programming Solvers

Matheus G. Vilas Boas[1], Haroldo G. Santos[1], Rafael de S. O. Martins[2], and Luiz H. C. Merschmann[3]

[1] Federal University of Ouro Preto, Department of Computing, Ouro Preto, Brazil
matheusgueedes91@gmail.com, haroldo.santos@gmail.com

[2] Federal University of Ouro Preto, Department of Information Systems and Computing, João Monlevade, Brazil
martins.rso@gmail.com

[3] Federal University of Lavras, Department of Computer Science, Lavras, Brazil
luiz.hcm@dcc.ufla.br

Abstract

Integer Programming (IP) is the most successful technique for solving hard combinatorial optimization problems. Modern IP solvers are very complex programs composed of many different procedures whose execution is embedded in the generic Branch & Bound framework. The activation of these procedures as well the definition of exploration strategies for the search tree can be done by setting different parameters. Since the success of these procedures and strategies in improving the performance of IP solvers varies widely depending on the problem being solved, the usual approach for discovering a good set of parameters considering average results is not ideal. In this work we propose a comprehensive approach for the automatic tuning of Integer Programming solvers where the characteristics of instances are considered. Computational experiments in a diverse set of 308 benchmark instances using the open source COIN-OR CBC solver were performed with different parameter sets and the results were processed by data mining algorithms. The results were encouraging: when trained with a portion of the database the algorithms were able to predict better parameters for the remaining instances in 84% of the cases. The selection of a single best parameter setting would provide an improvement in only 56% of instances, showing that great improvements can be obtained with our approach.

Keywords: feature based parameter tunning, mixed-integer programming, data mining, regression algorithms, coin-or branch and cut

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the scientific committee of the International Conference on Computational Science
10.1016/j.procs.2017.05.286

1 Introduction

A mixed-integer linear program (MIP) is an optimization problem in which a linear objective function is minimized subject to linear constraints over real and integer valued variables. A large number of relevant problems can be modeled in this format, from production planning to prediction of protein structures. Integer programming problems are notoriously hard to solve[8]. In fact, no efficient general algorithm is known for their solution. Nevertheless, the dramatic solver performance improvements occurred in the last decades[6, 10], allowed operations research practitioners to compute the exact solution of large scale models.

Modern MIP solvers are complex programs: performance improvements were obtained by enhancing the classical Branch & Bound[12] algorithm with the inclusion of many additional procedures which can be executed in the search tree. These procedures are basically cutting planes, devised to improve the linear programming relaxation and heuristics, created to speedup the production of high quality feasible solutions. Since these procedures involve additional computations, their activation does not always improves the overall solver's performance. Thus, MIP solvers typically have more than 50 parameters which can be tuned to improve their performance when solving different problems. A possible approach for this problem is the use of automated techniques to discover a good parameter setting considering average results in a representative set of problems. As we show later, this approach is severely limited due to the large diversity of problem types that can be modeled as MIPs: different composition of variables (e.g.: integer/continuous), different constraint types (e.g.: knapsack, cardinality and flow) and coefficient matrix characteristics (e.g. density of non-zeros and range of coefficients). Thus, smarter approaches can benefit from this problem diversity and perform Feature Based Parameter Tuning (FBPT), i.e. different best parameter settings can be defined for different classes of problems. These classes can be defined previously or discovered in the tuning process.

A successful FBPT is described in [5] in the context of heuristics for one specific application. Authors approached the curriculum-based course timetabling problem using Simulated Annealing. Parameter selection is performed using data mining (regression methods). Promising results were obtained, reaching 95.5% of accuracy in the classification algorithm. Similar application specific works are [1, 9, 11] and [3].

Considering MIP solvers, FBPT tools are scarce. To the best of our knowledge it was only attempted in [4] where machine learning methods were used to predict the best parameter settings for a small set of instances. In [4], the metric used to evaluate the performance of different parameters settings is the total time used to prove the optimality. This metric is insensitive to solver executions which finished with different optimality gaps. It is not sensitive also to executions were the first feasible solution was produced in very different times. A much better metric is employed in [14] to tune the SCIP MIP solver: the whole history of the search process is considered, i.e. every improvement in the the incumbent solution and its time is computed so that the objective is to discover a parameter setting which is better than others at *any time*. Our metric is equivalent to this last one, since all relevant events in the search process (improvement in lower or upper bounds) and their times are considered.

In this paper we address the following questions: can instance features of Mixed Integer Programs be used to determine better parameter settings for groups of instances considering a comprehensive performance evaluation? Additionally, can these groups be determined automatically using machine learning? To answer these questions we built a dataset with 308 problem instances and evaluated the open source MIP solver COIN-OR CBC[13, 7] using 49 different parameter settings. Three regression methods were trained in a subset of instances and evaluated for predicting the best parameter settings for new instances considering its features.