Models and algorithms for configuring and testing prototype cars

François Clautiaux[1], Siham Essodaigui[2], Alain Nguyen[2], Ruslan Sadykov[3], Nawel Younes[2]

[1] Université de Bordeaux, CNRS, Inria, IMB
francois.clautiaux@math.u-bordeaux.fr, ruslan.sadykov@inria.fr

[2] Renault S.A.
{siham.essodaigui ,alain.nguyen, nawel.younes}@renault.com

[3] Atoptima
Ruslan.Sadykov@atoptima.com

Mots-clés : *Scheduling, Mixed integer linear programming, Column and row generation, Diving heuristic, Automobile industry*

1 Problem description and industrial context

In this presentation, we study an optimization problem that arises in the context of the automobile industry. When a new model of vehicle has just been conceived, it needs to undergo a series of tests before being released on the market. This phase is costly, since specific vehicles called *prototypes* are built exclusively for test purposes. The validation tests carried out in the various facilities are numerous, and are repeated on the different mechanical parts of the prototypes (thermal, acoustic, driving, sound insulation, safety or endurance tests for example). The goal of the company is to minimize the number of tests realized after their due dates, and the number of prototypes to manufacture.

A difficulty met by the company is related to the notion of *configuration*. The same model of vehicle is sold in different configurations (with or without a sunroof, different colors, engine, sound system, ...). Some tests can only be realized if the prototype has a certain configuration. As an example, the waterproofing tests are not the same depending on whether the prototype has a sunroof or not. The choice of the configuration of the prototypes is a key decision when one needs to minimize their number. With increasing international competition, car manufacturers need to reduce both their costs and the time of the test phase.

From an operations research point of view, the problem can be seen as a two-step optimization problem. In the first step, one decides how many prototypes are built, and which configuration to assign to each prototype. In the second step, a sequence of tests is assigned to each prototype. A test can only be assigned to a prototype if the configuration of the latter is compatible. Since prototypes are produced and tested in different facilities, the test assignment problem can be either seen as a vehicle routing problem, or a scheduling problem with time lags. As time lags are short compared to the test durations in our instances, the scheduling vocabulary is adopted in the paper : we use the term *machine* for the prototypes, and *jobs* for the tests. Our problem generalizes multi-purpose scheduling problems [2], and differs from machine qualification problems [3], by including conflicts between possible configurations.

2 Contribution of the talk

We introduce formally and characterize our new industrial problem : we propose a formal definition and the proof that finding a feasible solution for the problem is already NP-complete. Then we discuss some useful properties, which allow to reduce the size of the search space in optimization algorithms, and characterize the cases where the configuration constraints become equivalent to job incompatibilities, as defined in [1]. We then propose two MILP models, a compact model, and a model with an exponential number of variables, which is solved by column generation. We also propose a matheuristic based on the latter model. The solutions obtained by these methods are compared with those obtained by a constraint programming model previously used by the industrial partner.

An originality of our work is to address the problem by the means of a path-flow formulation. In our path-flow model, the sequence of jobs assigned to a machine corresponds with a path in a network, where nodes are related to states representing a partial solution for a machine, and arcs correspond to decisions of assigning a job to a machine in this state. Our contribution is to propose an efficient algorithm to produce columns in the column generation process. In our approach, configurations are thus only considered implicitly, *i.e.* they are a byproduct of the set of jobs assigned to each machine. To satisfy the compatibility constraints, we have to embed additional information into the states to avoid assigning a set of job to a machine, for which no possible configuration is possible. These constraints are embedded into a labelling algorithm, which is used to produce feasible assignments of jobs to a machine. From an algorithmic point of view, this calls for a new specific labelling process that we propose in this paper. In our method, new specific resource consumption and dominance checks have to be designed to account for this specificity.

To obtain primal solutions even for the largest instances of the benchmark, we propose a matheuristic, which is devised by restricting the set of solutions to those where jobs are sequenced in increasing order of due date on each machine. Because of the sequence-dependent lags, this restriction may exclude all optimal solutions, but it is in line with the objective of minimizing the number of late jobs. This restriction spares us from imposing elementarity constraints in the labelling algorithm, thus reducing drastically the number of non-dominated labels.

3 Computational experiments

We conducted computational experiments on real-life instances used by the industrial partner. For each test case, we tested our compact MILP model, and our diving heuristic. We also compute dual bounds using a column-and-cut generation algorithm, to evaluate the quality of the heuristic. We compared our results to those obtained by a constraint programming (CP) model used by the industrial partner. The path-flow reformulation always finds solutions that are at least as good as the CP solver (except for the smallest instance), improve the results for the large instances, and is able to prove the optimality of several solutions.

Références

[1] Hans L. Bodlaender, Klaus Jansen, and Gerhard J. Woeginger. Scheduling with incompatible jobs. *Discrete Applied Mathematics*, 55(3) :219–232, 1994.

[2] Rosiane de Freitas Rodrigues, Mitre Costa Dourado, and Jayme Luiz Szwarcfiter. Scheduling problem with multi-purpose parallel machines. *Discrete Applied Mathematics*, 164 :313–319, 2014. Combinatorial Optimization.

[3] Mengying Fu, Moeed Haghnevis, Ronald Askin, John Fowler, and Muhong Zhang. Machine qualification management for a semiconductor back-end facility. In *Proceedings of the 2010 Winter Simulation Conference*, pages 2486–2492, 2010.