A hybrid learning and combinatorial optimization approach for automotive maintenance scheduling

Tony Wauters[a], Jannes Verstichel[a], Katja Verbeeck[a],
Greet Vanden Berghe[a,b]
[a]KaHo Sint-Lieven, CoDes
{ name.surname} @kahosl.be
[b]Katholieke Universiteit Leuven
Computer Science and Information Technology

In this abstract we present methods for solving a real world task scheduling problem, in the automotive maintenance sector. Tasks have to be assigned to personnel taking into account their skills, available working periods and other constraints. We present a model for this complex and highly constrained scheduling problem, and an approach for solving it.

In garages, tasks are scheduled by human planners with a lot of expertise. They solve the complex problem of assigning maintenance tasks to the personnel. These tasks have to be executed in one of their predefined modes. A mode defines the skilled resources such as employees, a workplace, a replacement vehicle, a waiting room for the customer, ... that are required to perform the task within a certain, mode dependent timespan. Examples of skills are: knowledge of a certain type of engine for an employee; the availability of a lift for a workplace and an automatic gearbox for a replacement vehicle. A resource may also have a certain level for its skills like expert and novice. Examples of modes are: 2 employees who need 60 minutes to perform a periodic maintenance, while 4 employees and one additional car lift can finish the job in only 20 minutes.

Tasks must preferably be scheduled within their time windows and according to their precedence relations (a task can only start when its predecessor tasks are finished). The tasks are preemptive. They can be interrupted in case of temporary unavailability of resources. This preemption, however is also subject to a number of constraints.

A task must not be interrupted immediately after its start or immediately before its end. Examples of these interruptions are: staff pauses, lunch breaks and the end of a working day.

In these real-life problems, time is usually assumed discrete with a problem-dependent granularity, resulting in the use of time slots. From discussions with practitioners, we concluded that time slots of 15 minutes are most common. When customers bring their car to the garage, they can wait in the waiting room until it is finished or they can use a replacement vehicle. The waiting room and the replacement vehicles stay occupied while the personnel is having a break. In these problems we can distinguish planned and unplanned repair and maintenance tasks. As unplanned tasks are often prioritized, we must schedule all planned tasks as good as possible while keeping a certain percentage of the available time free.

The objective function of this problem includes minimizing timewindow violations (total weighted tardiness), and minimizing resource idle times, while respecting the precedence constraints and the resource capacities.

We developed a hybrid approach for solving this task scheduling problem. It includes methods inspired by activity list based generation (adopted from the project scheduling literature [1]), learning automata (LA), and genetic algorithms (GA). We make use of a two step approach. First we build an activity list (i.e. an ordered list of tasks) and the corresponding mode assignment, and then we build a schedule handling the tasks in the order defined by the activity list. For the generation of the activity lists, we use simple heuristics like earliest due date first together with a genetic algorithm. The generation of good quality schedules from a given activity list is done by a learning tree search. This search procedure uses learning automata methods at each depth in the tree to decide which resources to use for each task. Learning automata [2] are simple single state reinforcement learning methods. The objective of an automaton is to learn an optimal action based on past actions and environmental feedback.

When we compare our method to manually created schedules by practitioners, we notice a much better use of the resources and their capacities. We are also able to schedule the tasks of an active day in a few milliseconds, and up to 15 working days in about 4 seconds.

Acknowledgment This research was carried out within the IWT SBO DiCoMAS project.

References

[1] R. Kolisch and S. Hartmann. Heuristic algorithms for solving the resource-constrained project-scheduling problem: Classification and computational analysis. *Handbook on recent advances in project scheduling*, 1998.

[2] M.A.L. Thathachar and P.S. Sastry. *Networks of Learning Automata Techniques for Online Stochastic Optimization.* Kluwer Academic Publishers, 2004.