Reinforcement learning approaches for the parallel machines job shop scheduling problem

Tony Wauters[2], Yailen Martinez[1,4], Patrick De Causmaecker[3], Ann Nowé[1], and Katja Verbeeck[2]

[1] CoMo, Vrije Universiteit Brussel, Belgium
[2] CoDes, KaHo Sint-Lieven, Gent, Belgium
[3] CoDes, K.U. Leuven Campus Kortrijk, Belgium
[4] AI-Lab, Central University of Las Villas, Cuba

This paper addresses the application of AI techniques in a practical OR problem, i.e. scheduling. Scheduling is a scientific domain concerning the allocation of tasks to a limited set of resources over time. The goal of scheduling is to maximize (or minimize) different optimization criteria such as the makespan (i.e. the completion time of the last operation in the schedule), the occupation rate of a machine or the total tardiness. In this area, the scientific community usually classifies the problems according to the characteristics of the systems studied. Important characteristics are: the number of machines available (one machine, parallel machines), the shop type (Job Shop, Open Shop or Flow Shop), the job characteristics (such as pre-emption allowed or not, equal processing times or not) and so on [1]. In this work we present two Reinforcement Learning approaches for the Parallel Machines Job Shop Scheduling Problem (JSP-PM).

The job-shop scheduling problem with parallel machines also known as the flexible job shop scheduling problem, represents an important problem encountered in current practice of manufacturing scheduling systems. It consists of assigning any operation of each job to a resource, i.e. one of the machines in a pool of identical parallel machines, in order to minimize a certain objective [2]. The pool of identical parallel machines, is sometimes called a machine type, a workcenter or also a flexible manufacturing cell [2]. The difference with the classic Job-Shop (JSSP) is that instead of having a single resource for each machine type, in flexible manufacturing systems a number of parallel machines are available in order to both increase the throughput rate and avoid production stop when machines fail or maintenance occurs. The objective we consider here is the minimization of the schedule makespan.

Literature on job shop scheduling with parallel machines is not rare, but approaches using learning based methods are. In the literature we find different (meta-)heuristic approaches for this problem. In [3] a tabu-search method which was originally introduced for the classic JSSP, is applied. In [4] a variable neighborhood genetic algorithm is used and in [2] a hybrid method combining a genetic algorithm and an ant colony optimization method is proposed. We will use the latter reference to compare our results with.

Reinforcement Learning is the problem faced by an agent that must learn behavior through trial-and-error interactions with a dynamic environment. Each time the agent performs an action in its environment, a trainer may provide a

reward or penalty to indicate the desirability of the resulting state. For example, when training an agent to play a game, the trainer might provide a positive reward when the game is won, negative when it is lost and zero in all other states. The task of the agent is to learn from this indirect, possibly delayed reward, to choose sequences of actions that produce the greatest cumulative reward [6]. Only just recently, Thomas Gabel and Martin Riedmiller [5] suggested and analyzed the application of reinforcement learning techniques to solve the task of job shop scheduling problems. They demonstrated that interpreting and solving this kind of problems as a multi-agent learning problem is beneficial for obtaining near-optimal solutions and can very well compete with alternative solution approaches.

The goal of this paper is therefore to study the capabilities of using different learning based methods for the JSP-PM. We will focus on value based reinforcement learning versus policy based approaches. Moreover, two different architectural viewpoints are taken, one in which resources are intelligent agents that have to choose which operation to process next, and an other in which operations themselves are seen as intelligent beings that have to choose their mutual scheduling order. As Reinforcement Learning methods we use a value iteration method (Q-Learning) and a policy iteration method (Learning Automata).

Our results show that both learning methods perform better then the recent hybrid-heuristic method reported in literature [2]. On average the mean relative errors are more than 1% better. Furthermore, our methods are capable of finding the optimal solutions more frequently.

References

1. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA. (1998)
2. Rossi, A., Boschi, E.: A hybrid heuristic to solve the parallel machines job-shop scheduling problem. Advances in Engineering Software **40** (2009) 118–127
3. Chambers, J.: Flexible job shop scheduling by tabu search (1996)
4. Variable Neighborhood Genetic Algorithm for the Flexible Job Shop Scheduling Problems. Volume 5315 of Lecture Notes In Artificial Intelligence. (2008)
5. Gabel, T., Riedmiller, M.: On a successful application of multi-agent reinforcement learning to operations research benchmarks. IEEE Transactions on Systems, Man, and Cybernetics. (2009)
6. Mitchell, T.: Machine Learning. McGraw-Hill Science/Engineering/Math (1997)