Integrating exact and heuristic methods to efficiently solve the ScheLoc problem

Arthur Kramer[1], Raphael Kramer[2]

[1] Department of Production Engineering, Federal University of Rio Grande do Norte - UFRN, Natal 59077-080, Brazil
arthur.kramer@ufrn.br

[2] Department of Production Engineering, Federal University of Pernambuco - UFPE, Recife 50740-550, Brazil
raphael.kramer@ufpe.br

Keywords : *ScheLoc problem, exact framework, job scheduling, facility location.*

1 Introduction

In this work, we deal with the Discrete Parallel Machine Makespan Scheduling and Location Problem (DPMMSLP), also known as ScheLoc problem. This problem integrates two well studied combinatorial optimization problems: (i) the parallel machines scheduling problem with the objective of minimizing the makespan, referred to as $P||C_{\max}$ in the three-field $\alpha|\beta|\gamma$ scheduling notation by [1]; and the (ii) facility location problem. In the DPMMSLP, the objective is to define where locate the available machines and define the job sequencing on these machines in order to minimize the total time for processing all jobs, i.e., the makespan.

The motivation for solving scheduling and facility location problems rely on the practical application in real-world problems (industry, healthcare, logistics) as well as on the challenge in efficiently solving them, since many of these problems are classified as NP-hard problems. In the literature, there many relevant works addressing these problems, but only few of them focusses on solving both problems in an integrated mode. Among the recent works that address the DPMMSLP, we highlight the recent contribution by [2, 4]. To the best of our knowledge, Hessler and Deghdak [2] were the first authors to deal with the DPMMSLP, by proposing a mathematical model, some heuristics and lower bounds. Recently, Wang et *al.* [4] proposed a network flow-based formulation and three heuristics to solve the DPMMSLP.

2 Problem description and exact framework

In the DPMMSLP, we are given a set $J = \{1, \dots, n\}$ of jobs to be processed in at most p identical parallel machines. The location of each of the p machines must be chosen from a discrete set $M = \{1, \dots, m\}$ of possible locations, with $p < m$. Each job $j \in J$ must be processed by p_j units of time on exactly one machine, without preemption and each selected location $k \in M$ can host at most one machine. In addition, a job j can only be processed on a machine located in $k \in M$ after a given release date r_{jk}. The release dates represent, for example, a transportation time from job storage location to the machine location. The objective is to locate the p machines and to sequence the n jobs on these machines in order to minimize the makespan.

To solve the DPMMSLP, we present an exact framework that involves the use of: (i) lower bound schemes; (ii) an Iterated Local Search (ILS) metaheuristic; (iii) a mathematical model; (iv) Column Generation (CG) technique; and (v) two Integer Programming Based Heuristics (IPBH). The considered lower bound schemes are the ones shown in [2]. Concerning the ILS algorithm, we considered a multi-start ILS that generates initial solutions based on a sequential approach, i.e., we first define the machine locations and then schedule the jobs. The local search are based on swap moves and consider auxiliary data structures to speed-up its execution. The mathematical model we present is an Arc-Flow (AF) formulation that is known for providing tight relaxations. Since AF models are also known by its pseudo-polynomial size, we apply CG to solve the linear relaxation and obtain a valid lower bound. The first IPBH consists in, after the execution of the CG algorithm, solving the AF model with the subset of variables generated during the CG execution. The second IPBH solves the AF with only a subset $\bar{M} \subseteq M$ of the possible locations, with $|\bar{M}| \geq p$.

The idea of the framework is to combine the five main components listed above to improve the bounds (lower and upper bounds) after each individual execution of the components. It stars by the execution of a simple routine to compute an initial Lower Bound (LB) (see [2]). Next, the ILS is executed aiming at obtaining an Upper Bound (UB). In the third step, the AF linear relaxation is solved by means of a CG method, the LB is updated and an AF model, with the subset of variables generated by the CG, is solved to obtain an UB. Then another AF, but only considering a subset of locations, is solved, aiming at improve the UB. At this point, if the optimality is not yet proven, the full AF model is then invoked to close the gap. Each time a LB or UB is updated, the optimality is checked and if the gap between UB and LB is zero, an optimal solution is found and the execution is finished. In addition, each time an UB is obtained, a local search procedure (the same considered in the ILS) is executed to further improve the UB. For more details on the method, we address the reader to [3].

3 Conclusions

We presented an exact framework to solve the discrete parallel machine makespan scheduling and location problem, that involves job scheduling and location decisions in an integrated fashion. The components of the framework considers a mathematical model, a column generation approach, an iterated local search and two integer programming-based heuristic. The framework was evaluated on two set of instances, the first proposed in [2] and the second in [3]. The obtained results show the good performance of the method. Optimal solutions were found for all instances proposed in [2], most of them for the first time. In addition, 92 out of 283 large instances proposed in [3] were also solved to the proven optimality.

Future researches can be carried out to develop tailored exact methods, such as branch-and-cut and branch-and-price algorithms, to deal with the unsolved instances of the DPMMSLP or even on the adaptation of the presented procedures to solve variants of the problem, such as the problem with unrelated machines of different objective functions, such as the minimization of the total weighted completion time.

References

[1] R.L. Graham, E.L. Lawler, J.K. Lenstra, and A.H.G.Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. In P.L. Hammer, E.L. Johnson, and B.H. Korte, editors, *Discrete Optimization II*, volume 5 of *Annals of Discrete Mathematics*, pages 287–326. Elsevier, 1979.

[2] Corinna Hessler and Kaouthar Deghdak. Discrete parallel machine makespan ScheLoc problem. *Journal of Combinatorial Optimization*, 34(4):1159–1186, 2017.

[3] Raphael Kramer and Arthur Kramer. An exact framework for the discrete parallel machine scheduling location problem. *Computers & Operations Research*, 132:105318, 2021.

[4] Shijin Wang, Ruochen Wu, Feng Chu, Jianbo Yu, and Xin Liu. An improved formulation and efficient heuristics for the discrete parallel-machine makespan ScheLoc problem. *Computers & Industrial Engineering*, 140:1–9, 2020.