

Joint Scheduling of Production and Transport with Alternative Job Routing in Flexible Manufacturing Systems

Seyed Mahdi Homayouni^{1, a)} and Dalila B.M.M. Fontes^{1, 2, b)}

¹ *LIAAD - INESC TEC, Porto, Portugal.*

² *Faculdade de Economia da Universidade do Porto, Porto, Portugal.*

^a Corresponding author: smh@inesctec.pt

^b dfontes@inesctec.pt

Abstract. This work proposes a mathematical programming model for jointly scheduling of production and transport in flexible manufacturing systems considering alternative job routing. Although production scheduling and transport scheduling have been vastly researched, most of the works address them independently. In addition, the few that consider their simultaneous scheduling assume job routes as an input, i.e., the machine-operation allocation is previously determined. However, in flexible manufacturing systems, this is an important source of flexibility that should not be ignored. The results show the model efficiency in solving small-sized instances.

INTRODUCTION

This work addresses the joint scheduling of production and transport in flexible manufacturing systems (FMSs), where each manufacturing operation can be processed in a set of alternative unrelated machines. Therefore, machine-operation allocation must also be concurrently determined. The higher flexibility and faster response times are two essential features of FMSs; however, its performance is highly dependent on the scheduling policy, which determines the allocation over time of scarce resources.

Conceptually, the job shop problem (JSP) and scheduling in FMSs are similar; however, usually, JSPs do not consider material handling issues. The JSP determines a sequence of manufacturing operations of a set of jobs on a set of machines while minimizing the makespan (the maximal completion time amongst all jobs). In the standard version, job operations need to be processed in a specific order and each operation is processed on one specific machine that can process one operation at one time. The JSP is a well-known NP-hard problem [1] and has been extensively studied. Over the years many extensions to the JSP have been proposed; one of the most popular being the flexible job shop problem (FJSP). The FJSP no longer assumes that each operation is processed in one specific machine, but rather that it can be processed on any machine from a given set of alternative machines. This problem, which is NP-hard in the strong sense [1], has been the subject of much research in the last three decades; a recent survey on the various techniques used to solve it and on the different objective functions considered can be found in [2]. Job shop problems (JSP and FJSP) usually assume job transport time between machines to be included in the processing time, which is an unrealistic assumption, since it must be, at least, route dependent, i.e., the transport time must depend on the origin and destination. Therefore, considering transport issues within these problems seems to be a natural extension, that indeed occurs in practice. Some researchers have already realized that and some works have been, recently, published, but mainly for the JSP. A solution for the JSP with transport considerations includes scheduling the manufacturing operations, as well as the transport operations, thus it is a combination of two NP-hard problems.

Scheduling in FMS environments is harder than the JSP due to the additional complexity resulting from the transport of jobs usually considered. Nevertheless, it has been vastly researched in the past few decades; however, just some research has involved the simultaneous scheduling of manufacturing and transport operations. The first such work is usually attributed to [3]. Over the years, a few dispatch rules and other heuristic approaches have been proposed, see, e.g., [4], [5]; but only recently an exact solution approach has been proposed [6]. Although scheduling in FMSs usually assumes a predetermined job route, it is easy to see that even if processing times are larger than

transport times, the latter may contribute to an increase in machine idle time as machines may have to wait for the jobs. Thus, considering alternative machines to process each operation provides the additional flexibility that can be used to reduce machine idle time and thus, makespan. In addition, in FMSs machines are usually flexible and thus, the possibility of allowing for alternative job routing exists in practice. Thus, including machine-operation allocation within this problem leads to the problem that is obtained by including transport scheduling in the FJSP.

Therefore, the problem being considered can be seen both as an extension of the FJSP by including job transportation and of the joint manufacturing and transport scheduling in FMSs, by including alternative job routes; thus, it is NP-hard as it generalizes known NP-hard problems.

This problem has not been the subject of much research. Indeed, as far as we are aware of, solution approaches have been reported only by six works and only heuristic approaches have been proposed. Deroussi & Norre [7] seem to have been the first to address this problem. They propose an iterated local search, in which perturbations consist of three feasibility preserving random moves. Neighbor solutions are obtained through classical exchange and insert moves. A simulated annealing procedure is used to manage the acceptance of neighbor solutions. The authors propose five sets of problem instances obtained by modifying known benchmark instances. The authors propose a set of problem instances derived from the known benchmark instances initially proposed by Bilge & Ulusoy [3] by considering additional machines.

Kumar et al. [8] propose a differential evolution (DE) that embed a machine selection heuristic and a vehicle assignment heuristic. The DE finds sequences of manufacturing operations by evolving permutation vectors. An available machine is then assigned to each operation. For each transport operation, a vehicle is chosen such that the vehicle total time is the lowest. The method was tested on a modified version of the 82 instances proposed by Bilge & Ulusoy [3], but different from that of Deroussi & Norre [7]. The method effectiveness was inferred from the solution to the original 82 instances, which was compared with literature heuristic results; however, some of the reported results cannot be correct as the makespan is below the optimal one.

Zhang et al. [9], [10] address the problem through decomposition methods. In [9] a genetic algorithm (GA) is used to assign each operation to a machine and each transport to an automated guided vehicle (AGV). Then a Tabu search (TS) finds and improves a sequence of manufacturing and transport operations for each machine and AGV, respectively. This approach is improved in [10] by including a shifting bottleneck (SBN) procedure. As before, the GA is used to evolve the allocation of the processing and transport resources. At each generation, the SBN finds a sequence of operations, both for manufacturing and transport, which then the TS attempts to improve. Computational experiments were performed using the problem instances proposed by Deroussi & Norre [7]. For six instances both methods found the same makespan; while each was better in two of the four remaining ones.

Recently, Deroussi [11] proposed a hybridization of a particle swarm optimization (PSO) with a stochastic local search. The PSO is used to solve the AGV assignment and sequencing problem and then each operation is assigned to the first machine available, finally, the sequencing of operations for each machine is obtained by choosing the operations according to their arrival time at the machine buffer. A local search procedure based on the classical exchange and insert moves is then applied to attempt to improve the incumbent solution. The authors only reported results for the average performance over the ten problem instances, which were of less quality than those of [9], [10].

More recently, Nouri et al. [12] proposed a hybrid metaheuristic that uses a GA to evolve solutions and a TS to improve them. The GA has a two-part chromosome, the first part represents the assignment problem; while the second one represents the sequencing problem. At the end of each GA generation the solutions obtained are clustered by proximity, that is, the population is divided into a predetermined number of groups. Then, TS is applied to the best solution within each group to search for a better solution in its neighborhood. At the end of the TS phase, the best solution found in each group is shared with the other groups. The authors report results for the instances used in [9], [10], although just for nine of them. The results reported are better for all nine instances. The improvements reported range from 13.83% to 31.15%. However, we believe the results not to be comparable. Indeed, for some of the instances, it is easy to see that the makespan has to be larger than the one they report.

As far as we are aware of, no optimal solutions are known for any of these instances. Previous mathematical models can only be found in [9] and [13]. The former only provides some formalism, but not really a model. The latter provides two modeling approaches: a sequence-based model and a position-based model. However, the authors assume an infinite number of AGVs; thus, jobs are transported without any delay and vehicles do not need to travel to pick up a job (empty travel), which amounts to consider that each job has a vehicle associated with it and accompanying it throughout the whole manufacturing process. Therefore, no transport scheduling decisions exist. Nevertheless, it should be noticed that the models could only be used to solve instances with sizes ranging from two jobs, three, operations, and two machines to six jobs, five, operations, and four machines; while our model solves instances with up to seven jobs, nineteen operations, and eight machines.

PROBLEM DEFINITION AND FORMULATION

An FMS is composed of a set of computer numeric control (CNC) machines connected by a set of AGVs under the control of a computer system. There is a set of jobs, each consisting of a predefined set of (manufacturing) operations. Each operation can be processed by any machine in a given set with a known processing time. A vehicle picks up the job from the load/unload (LU) area and moves it to the machine chosen to perform its first operation. Initially, the vehicles are parked at the LU area. For the following operations, the job is picked up by a vehicle from the machine where the last operation was performed, which waits for operation completion if necessary and taken to the machine chosen to process the operation. Once all operations of a job are completed the job is ready to be delivered to the customer. Note that, jobs arriving at a machine are delivered to the buffer area and the AGV can pursue its next assignment immediately. Machines layout and AGVs guide paths machines are known.

The mixed integer linear programming (MILP) model developed is based on that of [6], however, it extends it since machine-operation allocation is also determined. The objective function of the MILP model minimizes the production makespan, which is the longest completion time amongst the jobs. As in [6] the model uses two chains of precedence decisions: one for the manufacturing operations and another for the transport tasks. In addition, machine-operation allocation decisions are also considered. Further, the constraints can be described in three groups: i) the sequence of manufacturing operations, ii) the sequence of the transport tasks, and iii) the completion time constraints. More specifically, the constraints impose the sequence of operations on each machine, while satisfying precedence constraints, and assure that each machine has at most one first operation and as many last operations as first operations. Note that a machine may not be selected to process any operation. Moreover, this group of constraints guarantees that operations that are done consecutively in a machine, are allocated to the same machine; and the first and last operation of each machine is allocated to the corresponding machine. Finally, it also guarantees that each operation is assigned to exactly one machine. In the second group of constraints, it is ensured that the number of first and last transport tasks is equal to the number of available AGVs and impose the sequence of transport tasks on the AGVs, without any explicit assignment of AGVs to the transport tasks. Further, in the last group, the decision chains for the sequence of manufacturing operations and the transport tasks are interconnected by the completion time constraints that ensure that operations are processed only after the jobs are delivered to the machines and that jobs are picked up at the machines only after being processed by it. The model will be provided at the workshop.

RESULTS AND CONCLUSIONS

Computational experiments were carried out using the set of instances proposed in [7], [11]. Each problem instance is composed of eight machines, one LU area from which jobs enter and leave the system, and two AGVs. Each operation may be processed in one of two alternative machines, both having the same processing time. (Problem instances data is available in <https://fastmanufacturingproject.wordpress.com>). The model was implemented in and solved by Gurobi® software.

Table 1 reports the characteristics of the problem instances (number of machines M , jobs J , and operations O), the optimal makespan (except for instance 7, for which we were not able to find one, and it was stopped after 86000s.), and the computational time (in seconds) required to find it. The following columns provide the makespan value previously obtained by the best heuristic approaches in the literature: GATS-genetic algorithm with tabu search [9] and GTSB- GATS+shifting bottleneck [10]. The percentage optimality gap has been computed as $(UB-Opt)/Opt (\%)$.

Although [12] say that they consider the same instances, their results are not reported here since they cannot be correct. This, can easily be seen from the instances data as for some instances the makespan reported is less than the minimum longest job processing and transport time. This latter value was computed considering that machines and vehicles are always available whenever needed and that the best possible combination of processing and loaded transport time is chosen. For example, in FJSPT1 the total processing time for job 2 is 96 and the minimum total loaded transport time for the job is 16; thus job 2 cannot be ready in less than 112. However, [12] report a makespan of 110. (Similar calculations show that instances FJSPT3, FJSPT4, FJSPT5, and FJSPT7 have a minimum possible longest processing plus transport time of 116, 102, 94, and 90, respectively; however, the authors report makespan values of 104, 89, 81, and 84, respectively.)

The computational results obtained show that the proposed modeling approach, in addition to being novel, is capable of quickly finding optimal solutions for small-sized problem instances. This work is particularly relevant as no exact approaches exist for the joint production and transport scheduling in FMSs with alternative job routings.

TABLE 1. Results for the benchmark problem instances.

Problem Instances	Characteristics			MILP		GTSB [10]		GATS [9]	
	M	J	O	C_{max}	Time	Best	GAP	Best	GAP
FJSPT1	8	7	19	134	258.28	146	8.96	144	7.46
FJSPT2	8	6	15	112	27.61	118	5.36	118	5.36
FJSPT3	8	6	16	124	5.98	124	0.00	124	0.00
FJSPT4	8	5	19	114	579.48	124	8.77	124	8.77
FJSPT5	8	5	13	94	3.44	94	0.00	94	0.00
FJSPT6	8	6	18	138	45.81	144	4.35	144	4.35
FJSPT7	8	8	19	112 ^a	5312.05	122	8.93	124	10.71
FJSPT8	8	6	20	179	57.87	181	1.12	180	0.56
FJSPT9	8	5	17	140	178.17	146	4.29	150	7.14
FJSPT10	8	6	21	176	595.39	178	1.14	178	1.14
Max							8.96		10.71
Mean							4.29		4.55
Min							0.00		0.00

ACKNOWLEDGMENTS

We acknowledge the support of FEDER/COMPETE2020/NORTE2020/POCI/PIDDAC/MCTES/FCT funds through grants NORTE-01- 0145-FEDER-000020", PTDC/EEIAUT2933/2014, and 02/SAICT/2017-31821/POCI-01-0145-FEDER-031821.

REFERENCES

1. M. R. Garey, D. S. Johnson, and R. Sethi, "The Complexity of Flowshop and Jobshop Scheduling," *Math. Oper. Res.*, **1**, 117–129, (1976).
2. I. A. Chaudhry and A. A. Khan, "A research survey: Review of flexible job shop scheduling techniques," *Int. Trans. Oper. Res.*, **23**, 551–591, (2016).
3. Ü. Bilge and G. Ulusoy, "A Time Window Approach to Simultaneous Scheduling of Machines and Material Handling System in an FMS," *Oper. Res.*, **43**, 1058–1070, (1995).
4. T. F. Abdelmaguid, A. O. Nassef, B. A. Kamal, and M. F. Hassan, "A hybrid GA/heuristic approach to the simultaneous scheduling of machines and automated guided vehicles," *Int. J. Prod. Res.*, **42**, 267–281, (2004).
5. Y. Zheng, Y. Xiao, and Y. Seo, "A tabu search algorithm for simultaneous machine/AGV scheduling problem," *Int. J. Prod. Res.*, **52**, 5748–5763, (2014).
6. D. B. M. M. Fontes and S. M. Homayouni, "Joint Production & Transportation Scheduling in Flexible Manufacturing Systems," *J. Glob. Optim. Manusc.*, (2018).
7. L. Deroussi and S. Norre, "Simultaneous scheduling of machines and vehicles for the flexible job shop problem Solution approach : basic ideas," in *International Conference on Metaheuristics and Nature Inspired Computing*, 2010, 2–3.
8. M. V. S. Kumar, R. Janardhana, and C. S. P. Rao, "Simultaneous scheduling of machines and vehicles in an FMS environment with alternative routing," *Int. J. Adv. Manuf. Technol.*, **53**, 339–351, (2011).
9. Q. Zhang, H. Manier, and M. A. Manier, "A genetic algorithm with tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times," *Comput. Oper. Res.*, **39**, 1713–1723, (2012).
10. Q. Zhang, H. Manier, and M.-A. Manier, "Metaheuristics for Job Shop Scheduling with Transportation," in *Metaheuristics for Production Scheduling*, Hoboken, NJ USA: John Wiley & Sons, Inc., 2013, 465–493.
11. L. Deroussi, "A Hybrid PSO Applied to the Flexible Job Shop with Transport," in *Lecture Notes in Computer Science*, **8472**, Springer, Cham, 2014, 115–122.
12. H. E. Nouri, O. Belkahlia Driss, and K. Ghédira, "Simultaneous scheduling of machines and transport robots in flexible job shop environment using hybrid metaheuristics based on clustered holonic multiagent model," *Comput. Ind. Eng.*, **102**, 488–501, (2016).
13. S. Karimi, Z. Ardalan, B. Naderi, and M. Mohammadi, "Scheduling flexible job-shops with transportation times: Mathematical models and a hybrid imperialist competitive algorithm," *Appl. Math. Model.*, **41**, 667–682, (2017).