Hybrid System for Simultaneous Job Shop Scheduling and Layout Optimization Based on Multi-agents and Genetic Algorithm

Filipe Alves[1,2], M. Leonilde R. Varela[3(✉)], Ana Maria A. C. Rocha[3], Ana I. Pereira[1,2], José Barbosa[1], and Paulo Leitão[1]

[1] Research Centre in Digitalization and Intelligent Robotics (CeDRI), Instituto Politécnico de Bragança, Bragança, Portugal
{filipealves,apereira,jbarbosa,pleitao}@ipb.pt
[2] Algoritmi R&D Centre, University of Minho, Braga, Portugal
[3] Department of Production and Systems, Algoritmi Research Centre, University of Minho, Braga, Portugal
{leonilde,arocha}@dps.uminho.pt

Abstract. A challenge is emerging in the design of scheduling support systems and facility layout planning, both for manufacturing environments where dynamic adaptation and optimization become increasingly important on the efficiency and productivity. Focusing on the interactions between these two problems, this work combines two paradigms in sequential manner, optimization techniques and multi-agent systems, to better reflect practical manufacturing scenarios. This approach, in addition to significantly improve the quality of the solutions, enables fast reaction to condition changes. In such stochastic and very volatile environments, the manufacturing industries, the fast rescheduling, or planning, are crucial to maintain the system in operation. The proposed architecture was codified in MatLab® and NetLogo and applied to a real-world job shop case study. The experimental results achieved optimized solutions, as well as in the responsiveness to achieve dynamic results for disruptions and simultaneously layout optimization.

Keywords: Multi-agent systems · Meta-heuristics · Scheduling · Optimization

1 Introduction

In order to survive in today's turbulent market environment, which is characterized by dynamic fluctuations of demand patterns across product mix and increasing rates of new products introduction, organizations need to respond agilely in this kind of environment. Since there is no co-operation between the planning and the scheduling functions, production schedule generation processes become inflexible and are restricted by the pre-generated process plan [13].

Generally, scheduling problem consist in the identification and optimization of solutions for organizing the operations set execution under certain time constraints as well as capacity constraints in resources [8,14,20]. Thus, in scheduling area there are techniques used that are synthesized with the modern directions referring to applying artificial intelligence methods or optimization algorithms based on meta-heuristics.

Although optimization features are required, the real manufacturing scheduling problems are quite dynamic, considering new orders, delays, failures and unexpected events. Hence, it is hard to handle using only centralized optimization methods [5], since they have a high response time and do not provide autonomous and dynamic behavior. However, Multi-Agent Systems (MAS) [22] in agent-based modeling platforms, offer an alternative way to design, simulate and control systems, with capabilities to adapt to emergence or disruptions through fast local decisions with dynamic behaviors [2,23].

Regarding the literature in this field, there are some issues that present reviews and surveys related to different strategies and techniques of artificial intelligence for job shop scheduling in manufacturing systems, such as the use of meta-heuristics and multi-agent systems [3,6]. In this context some authors address, for example, a flexible job shop scheduling using meta-heuristics based on multi-agents [11,12]. It is also important to note that few studies present the job shop scheduling problem for efficient simulation layouts [17]. On the other hand, some authors integrate in a collaborative way the job shop scheduling with layout planning, that is, presenting a hybrid system capable of optimizing multiple solutions [15,16].

In this sense, this work dealing with a multi-agent system combined with genetic algorithm, proposed to simultaneously perform a job shop scheduling in a flexible layout under a manufacturing dynamic environment. The idea of this system approach, regarding to the existing works, is to consider and enable a cooperation between optimization and agents negotiation by providing dynamic and autonomous simulations that involve disruptions or unexpected events.

The rest of the paper is organized as follows: Sect. 2 describes the system architecture to improve the scheduling solutions in a dynamic and optimized context. Section 3 presents the scheduling and layout tool, with brief descriptions of the Genetic Algorithm and how the layout can be affected and optimized. On the other hand, Sect. 4 describes the agent-based model, developed in NetLogo, for the dynamic decentralized solutions. Section 5 presents the simulation tool applied to a case study, and consequently the discussion of the results. Finally, Sect. 6 rounds up the paper with the conclusions and future work.

2   System Architecture

The information trade-off between the timing of the works arrival, the creation of setups, the assignment of priorities, the testing of disruptions and the obtaining of the solutions for simulation of a real and dynamic job shop environment, will be according to an architecture that encompasses all these parameters in a distributed system. In this context, the system architecture for the manufacturing

industry scheduling system integrates off-line and on-line modules as presented in Fig. 1. So, it is possible to deal sequentially with two sub-processes: optimized planning using optimization methods and real-time responsiveness solutions using MAS. The two modules are able to exchange information, balancing the decision-making.
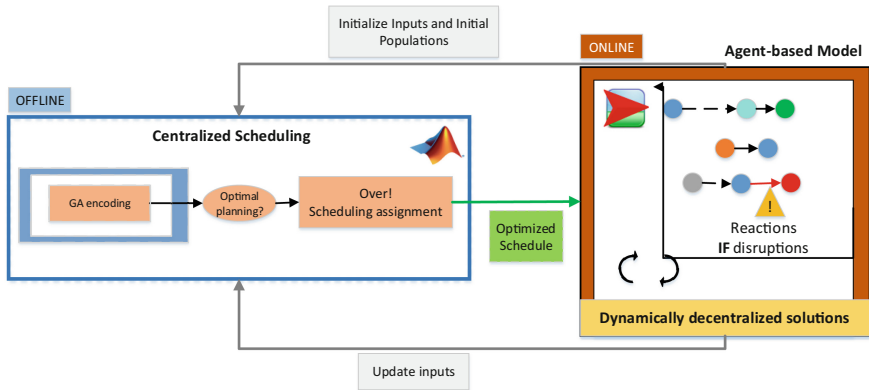

Fig. 1. Developed system architecture.

The module on the left performs the optimized scheduling for the jobs, running off-line, for a given situation under study. The module on the right concerns to the dynamic re-scheduling, in response to disruptions or condition changes, e.g. a broken machine in some situation or a failure in some position. In this module, the re-scheduling is obtained by the interaction of individual entities, each one reasoning about its own schedule. The individual entities (jobs and machines) can share information and have the ability to negotiate in order to overcome disruptions or failures, that is, they can obtain dynamic re-arranging in the layout, with minimal human intervention.

3 Scheduling and Layout Tool

The scheduling activity seeks to make efficient use of its resources to ensure the fast execution of the work so that it can be delivered within the agreed deadlines. Therefore, the scheduling contains varied objectives, evaluated by performance measures, such as the average number of jobs, production profits, minimization of time spent, etc. The present work aims to formulate a dynamic and flexible manufacturing scheduling system, integrated in a layout feature for further process optimization. The possibility of creating different products, requiring different types of processes (jobs) and which in turn require a set of operations, can be performed in certain positions/machines of a particular layout arrangement.

In this context, it is essential to develop efficient decision support methods to solve job shop scheduling, because the operators need to test several scenarios, which makes time requirements crucial.

The off-line module describes the approach to perform the optimized scheduling using Genetic Algorithm (GA). Initially proposed by Holland [9], GA uses a population of individuals to apply genetic procedures: crossover between two different individuals or/and mutation in one individual. The algorithm repeats the crossover and mutation procedures in new populations until the desired diversity of solutions is performed [7]. The optimization method is summarized by the GA presented in Alves et al. [1]. The iterative procedure ends after a maximum number of iterations (NI) or after a maximum number of function evaluations (NFE).

3.1   Layout Optimization

Layout is a feature inherent in any operation. In this way, it is necessary to continuously optimize the layout arrangement, not only for the scheduling of inputs but also to improve the profitability of the outputs (Fig. 2).
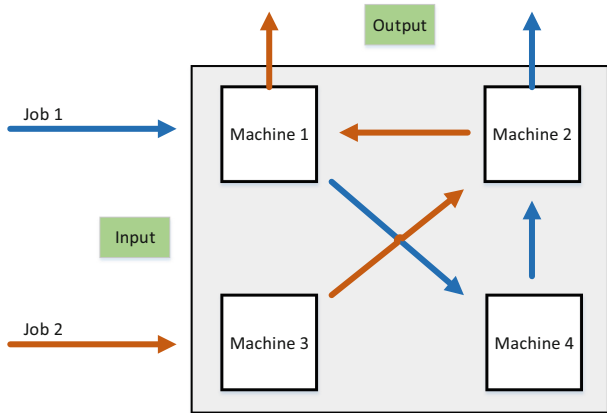

Fig. 2. Facility layout problems and variants.

Through the form and scheduling, appearance and the way materials, information and customers flow through the layout, the fullness of functions will be triggered [18]. The layout optimization problems are quite complex and in general are NP-hard, which requires a high computational complexity. The layout problem focuses on the flexible scheduling, reconfigurable, and agile manufacturing environments where the demand is affected (by machine disruption for example) [18]. With dynamics layout models embedded in optimization models and combined with multi-agent systems, it will be possible to minimize the sum of material handling costs. On the other hand, it will be possible guarantee the

best disposition of human and material resources, allowing efficiency, but also dynamic and autonomous re-arranging from the agent-based model.

4   Agent-Based Model

The on-line module considers the use of MAS to implement the dynamic re-scheduling in case of disruption. In addition, it will be possible to interact, simulate and visualize the scheduling performed by the off-line module.

The agent-based model considers three types of agents:

– **Machine agent:** These agents represent the work machines, which will allow the set of manufacturing operations in order to obtain the products. They are immobile, totally passive, and are subject to interruptions or failures.
– **Operation (job) agent:** These agents represent the "jobs" (a job represents a set of operations) that move around to provide the manufacturing of the products according to the scheduling and location of the machines agents. Operation agent only interacts with their own machines.
– **Product agent:** These agents represent the final product to be obtained. After the complete set of operations they require, they send a warning message informing the user of their conclusion.

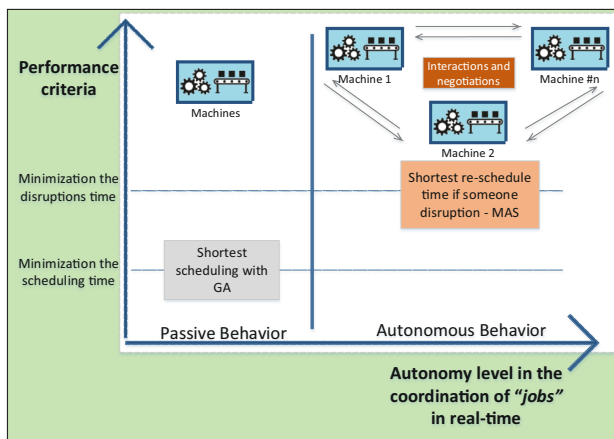The Fig. 3 presents the two main categories of global process behaviour: the passive and autonomous behaviours.

Fig. 3. Dynamic "machines" schedule.

In the passive behavior, the operations follow carefully the planned optimized schedule provided by the off-line module, i.e. using the GA coded in MatLab® [10], without taking into account the disruptions or failures. In the

autonomous behavior, the operation agent follows the planned route but is able to dynamically adapt the schedule in case of disruptions through the interaction with other machines, which may be available and with operations in common, to re-arrange the schedule that was previously allocated to the broken machine.

5   Simulation Tool

The described agent-based model was implemented in NetLogo [21], a simulation platform that allows to rapidly instantiate agent-based models to observe the behavior of systems. It provides an intuitive user interface where can be add buttons and control widgets to easily manipulate a model to view different scenarios [4]. The agent-based model developed in NetLogo is connected with MatLab® to allow the exchange of the optimized scheduling solution, through an extension that allows data passing between Netlogo and MatLab® or vice versa, facilitating dynamic integration of these software platforms.

In each simulation, some parameters are used to vary the populations of the agents according to the database. The simulation protocol involved three crucial steps, and it can be applied to many other simulations within the database, thus:

- **Step 1:** The user will have at his disposal the specifications of the problem. Is possible to select the different products to carry out and also to define the quantities of operations (jobs) from the selection already existing in the database. On the other hand, the set of available machines will also be assigned and their coordinates remained unchanged during the experiments.
- **Step 2:** The scheduling is performed by the user. The information will be transmitted and the overall schedule of the optimization method for the jobs planning (output viewer - interface) will be received.
- **Step 3:** It is displayed the jobs schedule sequence and if there is disruption, it can be updated the dynamic reschedule by agents negotiation.

5.1   Case Study

This work presents a Job Shop Scheduling Problem (JSSP) formulated from the Flexible Manufacturing System (FMS).

This data is based on the AIP-PRIMECA cell at the University of Valenciennes which offers the possibility to create different products [19], but in this study a product called "AI" will be created. The operation list also defines the processing times and in this case, there are eight different manufacturing operation types, as shown in the Table 1.

Besides this, the FMS is composed by seven workstations, each one being able to perform a set of operations. The machines are responsible for the completion of manufacturing operations to do the jobs. Some machines are able to complete the same manufacturing operation, while some operations can only be completed on a single machine. Additionally, each machine is continuously available as the system start and each machine can process only one operation at a time. The cell is composed with seven machines and the processing times of the eight manufacturing operations are described on Table 2.

Table 1. Production sequence for each type of job.

	A	I
#1	Plate loading	Plate loading
#2	Axis mounting	Axis mounting
#3	Axis mounting	Axis mounting
#4	Axis mounting	I_comp mounting
#5	r_comp mounting	Screw_comp mounting
#6	L_comp mounting	Inspection
#7	I_comp mounting	Plate loading
#8	Screw_comp mounting	
#9	Inspection	
#10	Plate loading	

Table 2. Processing time of each manufacturing operation (in seconds).

Operation	M1	M2	M3	M4	M5	M6	M7
Plate loading	10						
Plate unloading	10						
Axis mounting		20	20				
r_comp mounting		20		20			
I_comp mounting					20		
L_comp mounting				20		20	
Screw_comp mounting			20	20			
Inspection							10

5.2 Results Discussion

For case study, simulations were carried out on a PC Intel(R) Core(TM) i7 CPU 2.2 GHz with 6.0 GB of RAM. Since the GA is a stochastic method, 10 runs were carried out with random initial points. The values of the control parameters used in GA were adjusted to a suitable experience of the problem, i.e. it was considered a population size ($Ps = 30$) and concerning the probability of the procedures (crossover and mutation), 50% rate was selected. In turn, the function evaluation was fixed at $NFE = 5000$ and the maximum number of iterations as $NI = 100$.

In the present case, one "AI" and two jobs are created. The objective of this round of experiments (i.e., optimized scheduling with passive behavior and disruptions tests with autonomous/dynamic behavior) was to highlight the efficiency of the proposed approach defined by the ability of the latter to minimize the time and automatically optimize disruptions or failures in machines.

The scheduling tool allowed to obtain 100% of successful rate since they found a feasible solution in all runs. In turn, the scheduling is obtained extremely fast,
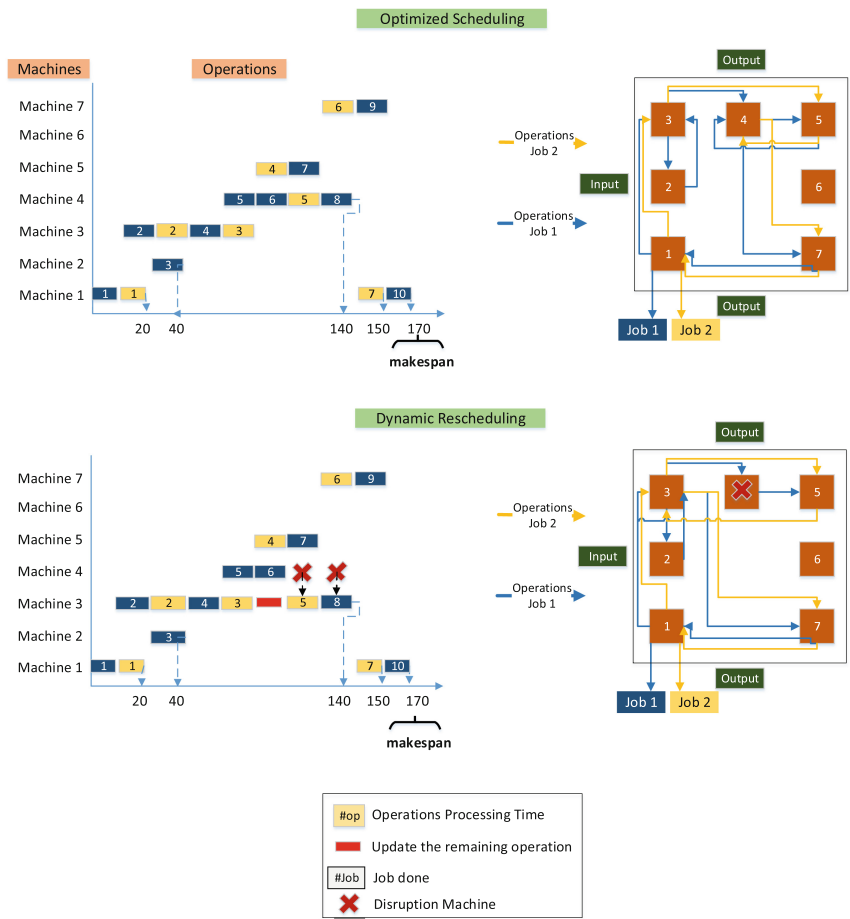
Fig. 4. Optimized solution simulation and dynamic layout rescheduling.

i.e, it only took 25 s. On the other hand, allowed to obtain the solution that characterizes the optimal time of execution of the jobs in question, tested for unprecedented scheduling and another with machine disruptions.

The Fig. 4 presents the optimized solution of the problem under study as well as the subject dynamic rescheduling in tests of machine disruptions.

It is important to note that the scheduling obtained by the GA has always ensured a feasible solution as well as it allows to identify the sets of operations to their respective machines. At the top of the Fig. 4 it is possible to verify optimized scheduling (not subject to disruption testing), which enabled the set of operations to identify their respective machines and allocate themselves properly in the layout for the production of the jobs. Consequently, it is also possible to analyze that the time required to create job 1 and job 2 is respectively 170 and 150 s. After this, using the interface in NetLogo, the same scheduling of the case

under study was subjected to a random disruption of a machine after a certain time (in this case machine 4), leaving two operations, one of each job without conclusion. Thus, through the agent-based model, the dynamic and autonomous rescheduling protocol was started to allocate the remaining operations again in the layout. They were allocated to the machine 3 without causing any disturbance and in the same order as they were left. This solution enabled not only the minimal human interaction but also allowed to finish the jobs in the time previously expected, without affecting the end of the remaining operations or priorities imposed.

6   Conclusions and Future Work

This paper addresses a hybrid system for job shop scheduling problems and layout optimization in a dynamic context with numerous constraints, uncertainties and random events. The system architecture has been proposed to solve the two sub-problems presented in a sequential manner. An experimental case study was proposed based on FMS. Thus, through the results presented, it was possible to obtain a fast and optimized scheduling by the off-line module and at the same time interact with the NetLogo platform. In this sense, the same scheduling was subject to random interruption of a machine, which autonomously and with the use of a negotiation protocol between agents, allowed the dynamic rescheduling of the missing operations. In turn, machine 3 allocated the operations and ensured that the final time for completion of the two jobs was not exceeded. This approach not only enables optimized solutions but also ensures minimal human intervention in real-time disruption testing for layout re-arranging.

The main contribution of this paper it was to provide a new way of solving a job shop scheduling problems in simultaneous with layout optimization, using the machines ability to dynamically adapt and design his own schedule in case of external disruptions. Thus, we can improve both flexibility and efficiency in today's competitive manufacturing environments, such as flexible manufacturing systems and just-in-time production, as well as add strategies and dynamics to improve quality and increase profits.

For future work, it is possible to reformulate and increase the complexity of the problem. Another approach to the future could be to use other meta-heuristics and implement dynamic scheduling alternatives in case of disruptions according to products and jobs priorities.

References

1. Alves, F., Pereira, A.I., Fernandes, A., Leitão, P.: Optimization of home care visits schedule by genetic algorithm. In: International Conference on Bioinspired Methods and Their Applications, pp. 1–12. Springer (2018)
2. Barbosa, J., Leitão, P.: Simulation of multi-agent manufacturing systems using agent-based modelling platforms. In: 2011 9th IEEE International Conference on Industrial Informatics (INDIN), pp. 477–482. IEEE (2011)
3. Çalış, B., Bulkan, S.: A research survey: review of AI solution strategies of job shop scheduling problem. J. Intell. Manufact. **26**(5), 961–973 (2015)
4. Chiacchio, F., Pennisi, M., Russo, G., Motta, S., Pappalardo, F.: Agent-based modeling of the immune system: NetLogo, a promising framework. BioMed Res. Int. **2014** (2014)
5. Eremia, M., Liu, C.C., Edris, A.A.: Advanced Solutions in Power Systems: HVDC, FACTS, and Artificial Intelligence, vol. 52. Wiley, Hoboken (2016)
6. Gen, M., Lin, L., Zhang, H.: Evolutionary techniques for optimization problems in integrated manufacturing system: state-of-the-art-survey. Comput. Ind. Eng. **56**(3), 779–808 (2009)
7. Ghaheri, A., Shoar, S., Naderan, M., Hoseini, S.S.: The applications of genetic algorithms in medicine. Oman Med. J. **30**(6), 406 (2015)
8. Groover, M.P.: Automation, Production Systems, and Computer-Integrated Manufacturing. Prentice Hall Press, Upper Saddle River (2007)
9. Holland, J.H.: Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. MIT press, Cambridge (1992)
10. MATLAB: version 8.6.0 (R2015b). The MathWorks Inc., Natick, Massachusetts (2015)
11. Nouiri, M., Bekrar, A., Jemai, A., Niar, S., Ammari, A.C.: An effective and distributed particle swarm optimization algorithm for flexible job-shop scheduling problem. J. Intell. Manufact. **29**(3), 603–615 (2018)
12. Nouri, H.E., Driss, O.B., Ghédira, K.: 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. Ouelhadj, D., Petrovic, S.: A survey of dynamic scheduling in manufacturing systems. J. Sched. **12**(4), 417 (2009)
14. Rewers, P., Trojanowska, J., Diakun, J., Rocha, A., Reis, L.P.: A study of priority rules for a levelled production plan. In: Advances in Manufacturing, pp. 111–120. Springer (2018)
15. Ripon, K.S.N., Torresen, J.: Integrated job shop scheduling and layout planning: a hybrid evolutionary method for optimizing multiple objectives. Evol. Syst. **5**(2), 121–132 (2014)
16. Sahin, C., Demirtas, M., Erol, R., Baykasoğlu, A., Kaplanoğlu, V.: A multi-agent based approach to dynamic scheduling with flexible processing capabilities. J. Intell. Manufact. **28**(8), 1827–1845 (2017)
17. Sudo, Y., Matsuda, M.: Agent based manufacturing simulation for efficient assembly operations. Procedia CIRP **7**, 437–442 (2013)
18. Tompkins, J.A., White, J.A., Bozer, Y.A., Tanchoco, J.M.A.: Facilities Planning. Wiley, Hoboken (2010)
19. Trentesaux, D., Pach, C., Bekrar, A., Sallez, Y., Berger, T., Bonte, T., Leitão, P., Barbosa, J.: Benchmarking flexible job-shop scheduling and control systems. Control Eng. Pract. **21**(9), 1204–1225 (2013)

20. Varela, M., Putnik, G., Manupati, V., Rajyalakshmi, G., Trojanowska, J., Machado, J., et al.: Collaborative manufacturing based on cloud, and on other I4.0 oriented principles and technologies: a systematic literature review and reflections. Manag. Prod. Eng. Rev. **9**(3), 90–99 (2018)
21. Wilensky, U.: NetLogo: center for connected learning and computer-based modeling. Northwestern University, Evanston, IL, 1999 (2014)
22. Wooldridge, M.: An Introduction to Multiagent Systems. Wiley, Chichester (2009)
23. Xie, J., Liu, C.C.: Multi-agent systems and their applications. J. Int. Counc. Electr. Eng. **7**(1), 188–197 (2017)