Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 107 (2022) 1323–1328

www.elsevier.com/locate/procedia

55th CIRP Conference on Manufacturing Systems

Towards an Automated Application for Order Release

Günther Schuh[a], Andreas Gützlaff[a], Matthias Schmidhuber[a], Judith Fulterer[a], Tim Janke[a,]*

[a]Laboratory for Machine Tools and Production Engineering (WZL) of the RWTH Aachen University, Campus-Boulevard 30, 52074 Aachen, Germany

* Corresponding author. Tel.: +49 175 430 07 11; fax: +49 241 80 222 93. *E-mail address:* t.janke@wzl.rwth-aachen.de

Abstract

The demand for customer-specific product variants leads to a shift to job shop production, in which order scheduling takes a mayor role to increase logistical target values without ad-hoc interventions into the running production. Agent applications, e.g. combinations of reinforcement learning (RL) and simulation, are promising solutions to solve the scheduling problem. This paper designs a methodology for automating the order release decision of real production scenarios by applying a RL agent, which has been trained on an application-specific simulation model. By an integrated validation unit the performance can be measured against known order release strategies.

Keywords: Reinforcement learning; order release; simulation; job shop production; scheduling

1. Introduction

Today's production management faces the challenge to meet changing customer requirements regarding a growing individualization level and general sensitivity for logistical service levels [1]. In production, the increasing number of individual product variants results in smaller batch sizes and a more flexible job shop production [2]. By using a job shop production, a high flexibility with regard to the quality of specific customer requirements as well as the varying quantity caused by order fluctuations can be fulfilled [3]. This change leads to increased demands concerning production planning and control (PPC) [4] to achieve the logistical target values, e.g. short throughput times and high utilization rates [5], and at the same time to keep margin for disposition [6].

A mayor lever is the order release, assigning available resources and capacities in the production to specific orders and form a short-term production schedule based on determined rules [7]. Key elements to be considered are a list of generated orders and work plans including corresponding machines and lead times as well as batch sizes of each order [5]. In order to reduce ad-hoc interventions in dynamic systems, production changes must be constantly considered [8]. To meet these requirements in production environments that are reacting to the ongoing mass customization trend, new dynamic methods of scheduling are required [9]. Scheduling methods based on reinforcement learning (RL) that allow a constant reconsideration of the current status and deciding on the orders to be released next might be a promising field of research [10].

Therefore, this paper proposes a methodology to automate the order release process by using a RL agent and integrating near real-time feedback data from the production. The focus on order release will reduce efforts for ad-hoc control interventions and using the real production status as planning base will increase the planning capability. This paper is structured as follows. The next section provides background information on the area of consideration. Section 3 gives an overview of related research approaches and discusses them on the basis of currently existing challenges in order release. Based on that, section 4 proposes the approach for an automated order release methodology and the conclusion as well as a further research direction is given in section 5.

10.1016/j.procir.2022.05.152

2. Background

In this section, the foundations necessary for understanding the role of order release within the tasks of PPC and methods used in the proposed approach for solving the scheduling problem are established.

2.1. Role of order release in production control

The correct application of production control strategies determines the fulfillment of the logistical target values, shortening throughput time, reducing inventory, increasing capacity utilization and increasing adherence to delivery date [11]. However, due to a high interdependency of these targets, it is not possible to define a simultaneous optimum for all target values [12]. For example, reducing inventory may lead to a reduction in capacity utilization and therefore to a decreased performance level. Production control contains four specific tasks (see Fig. 1).

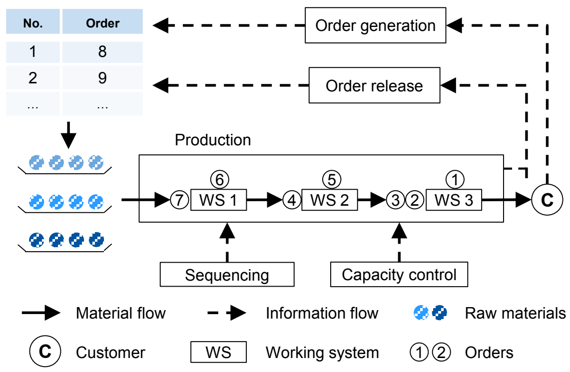

Fig. 1. Task of production control [12]

The start is a production plan filled by the *order generation*. Based on the *capacity control*, the task of *order release* feeds orders to be produced into the production. For each working system, the task of *sequencing* determines the sequence in which waiting orders are processed.[12].

Within production control, the order release marks the point from which an order is ready to be produced [6]. Therefore, it is considered as a "critical decision point" [13] at which production control at the shop floor starts. Order release consists of the two parameters order timing and selection decision [14]. Basically, four different criteria can be distinguished to approach the task of order release [12]:

- Immediate order release directly after order generation
- Order release as soon as the planned start date is reached
- Inventory-controlling order release with a certain value
- Work system load order release with an expected load

To assist the production planner, a wide range of software systems have been introduced using and combining different approaches [11]. These systems build a production schedule on basis of several input data, e.g. master production schedule, bill of materials, material inventory levels, lead times and batch sizes [15]. Leading systems for this purpose are enterprise resource planning (ERP), manufacturing execution systems (MES) and advanced planning and scheduling systems (APS). APS systems can reduce the gap between planned and actual production schedule, but the procedure is still based on software-specific assumptions and the degree of integration into the planning processes [16].

2.2. Introduction to the job shop scheduling problem (JSP)

In literature, solving the described production control tasks of order release and sequencing in a job shop production is well known as the job shop scheduling problem (JSP) [9, 17]. The JSP is considered to be part of the hardest combinatorial optimization problems and has proven to be NP-hard [18]. Typically, the following assumptions are made [19]:

- One machine of each type in the shop
- One operation at a time on each machine and on any job
- An operation of a job can be performed by just one machine
- An already started job can't be interrupted
- The next operation of a job can be started after completing its preceding operations
- No alternative routings for a job
- Each machine is continuously available for production
- No restriction on queue length before any machine

2.3. Use of reinforcement learning in the JSP

Reinforcement learning is a machine learning method that has proven to be a suitable approach for determining a solution strategy for highly demanding applications without requiring expert knowledge – simply by direct interaction with the environment and decision making through "trial and error" [20, 21]. In recent years, a lot of attention has been paid to apply RL to PPC in order to obtain more robust solutions in an environment where unpredictable and unplanned machine downtime is common [22].

RL is based on the interaction of an agent with its environment. For each time step, the agent performs an action based on the state of the environment and depending on the changed state receives a specific reward [23]. In addition to a dynamic environment that must be observable by the agent, RL problems require a reinforcement and value function defining the policy of actions to perform and the value of rewards to be received [24]. With this problem formulation, the agent tries to maximize the sum of rewards gotten from its actions in order to solve the stated problem [25]. To apply this principle, the problem needs to be modeled as Markov Decision Problem assuming that the achievement of future states only depends on the present state and action [26].

The decision process requires a simulation model that represents the behavior of the production sufficiently accurately and that can interact with the agent, because it is not applicable to perform this in a real production environment due to economic and time constraints [27]. Discrete-event simulation is used to model the production environment, which depicts a sequence of events representing actions performed on orders [28]. Simulation-based approaches can increase prediction accuracy by including production-specific characteristics as well as realistic processes [29].

3. Related work

This section reviews current work on the job shop scheduling problem by especially considering the task of order release. The related work is classified based on the methods used with a special focus on meta-heuristics, because the proposed approach in this paper using a RL agent relates to it.

3.1. Order release in the job shop scheduling problem (JSP)

In recent years, many authors addressed the JSP with focus on either order release, sequencing or both [10]. A mayor research direction focuses on the task of order sequencing (or dispatching), where decentralized decisions are made to optimize the system according to each arising queue at each machine [8, 9, 30]. Here, different sequencing rules are used and compared to each other [19]. At the same time, the lever of an advanced order release strategy is addressed in fewer recent approaches, but becomes especially useful for a high-variety and low-volume production [31, 32].

A specific challenge in order release is the nonlinear relationship between resource utilization and throughput time, which leads to the fact that it is hardly possible to specify an exact completion date at the moment of order release [33]. By dynamically optimizing the task of order release, the necessity for ad-hoc interventions through dispatching rules to meet the logistic target values decreases [34]. Some approaches especially focus on the order release to achieve a small overall make span or high adherence to delivery dates [10, 33, 35].

To overcome challenges from industrial practice by applying a new order release strategy, the proposed approach in this paper cope with four mayor challenges:

- Comprehensive consideration of the relevant inter-relationships and influencing factors of a real production
- Acquisition of feedback data for dynamic reaction to malfunctions and order changes
- Enabling a multidimensional objective function and adherence to delivery date as boundary constraint
- Applicability of the methodology to the order release of a practical job shop production

Three different main methods can be classified for solving scheduling problems, which are presented in the following paragraphs [9]: Exact methods with mathematical modeling, heuristics as well as meta-heuristics involving artificial intelligence (AI) [36].

3.2. Exact methods

Since the problem has got a broad attention in research, mathematical modeling has been applied [37], e.g. integer programming [38] and mixed integer programming [39]. The use of exact methods is limited by computational requirements as the problem counts as NP-hard. However, this can be overcome by more advanced approaches, e.g. decomposing the problem into smaller instances, branch-and-bound, Lagrangian relaxation and by making use of modern computational performance [37]. Therefore, research is being conducted using exact methods until today [9, 40]. Nevertheless, finding exact solutions for optimization problems is limited to a certain complexity level and therefore practical production sizes are difficult to depict.

3.3. Heuristics

Instead of computing an exact solution upfront or release orders into the production directly after their generation, several heuristics have been developed to effectively improve the production performance [34]. Heuristics allow for a significant reduction in complexity and hence, on a basic level, compete with generalized software solutions and mathematical models [37].

Two simple and very common heuristics for order release used in practice are constant work in process (Conwip) and load-oriented order release [12]. Conwip is based on the inventory regulation by linking the production input and output. Orders are released whenever the production inventory level falls below a planned value. In the load-oriented approach, an order is released when the inventory limit has not been exceeded at any of the work systems through which the order will pass. In contrast to Conwip, the procedure does not always include the orders with the full order time to the stock accounts of the work systems. [12]

With regard to the formulated challenge of comprehensively considering the relevant interrelationships and influencing factors of a real production, heuristics try to find general rules rather than considering a specific production environment and hence are not further considered in this paper.

3.4. Meta-heuristics including learning-based systems

The already presented methods have been extended by meta-heuristics, genetic algorithms and learning-based systems in order to solve the complex problems more effectively [37]. Many current research approaches already extend the known meta-heuristics by adding AI to their methods [36].

[33] propose an adaptive order release mechanism to specify order release times using RL. The approach meets the focus of applying RL to the task of order release, but aims for solving the problem in a flow shop with different constraints compared to these of a job shop problem, e.g. alternative routes for each specific product. For solving the scheduling problem in a job shop regarding more than one objective, [41] developed a variable neighborhood search algorithm, that uses decomposition techniques to obtain near-optimal sequences of performing the operations and generating a schedule. This approach does not include, re-scheduling based on a constantly changing situation in production and hence does not meet the requirement of including feedback data from the production.

A real-time scheduling of a large number of tasks has been introduced in the course of the corona crisis by [35] using an end-to-end neural network trained by reinforcement learning using negative total tardiness in the reward function. In this approach, real-time scheduling is achieved for a practical production size, but just one optimization objective is considered.

A solution for handling unforeseen events was introduced by [42] as a basic approach towards real-time disruption management strategies, that is able to manage dynamic conditions while keeping cost effectiveness, product quality and adherence to delivery date. The authors integrate a schedule state simulator with rescheduling knowledge stored in a deep Q-network, that is trained with reward to a goal schedule state. Here, it remains unclear if the method can be transferred to a job shop production. A hybrid Deep Q Learning (DQN) training method with real production data is proposed by [10] in order to solve practical production problems combining an initial python-based training instance in SimPy with a further training instance using an interface to the commercial software Plant Simulation. This approach shows how to combine performant and scalable training methods using python with detailed status information using advanced simulation software. While the practical reference becomes clear, it is not yet scaled-up to a representative problem instance and the second training step proposed in the hybrid approach has not yet been validated.

The review of the literature presented shows that newly introduced approaches that combine mathematical techniques and simulation with learning-based systems such as RL are able to solve the JSP in an effective way. However, the authors agree that space for improvement remains in different dimensions: A demand for a strengthened collaboration and integration of different PPC methodologies into one application as well as a more practical guide for implementing current research approaches into a real production are prospected [11]. Still not solved to a sufficient grade is the need for quick decision support in the PPC with respect to dynamic production state changes [8]. Therefore, the requisite for a higher integration of near real-time feedback data from the production into the planning and control continues to gain importance [6]. Also, many approaches to solve the JSP focus on dispatching strategies, while fewer make use of the advantages lying at the beginning of production at the time of order release. Also, the applicability to a real-sized job shop production has not been shown in many cases.

4. Application of automated order release

This section presents a methodology for automating the order release process by using a RL agent and near real-time production data to solve the JSP (see Fig. 2).

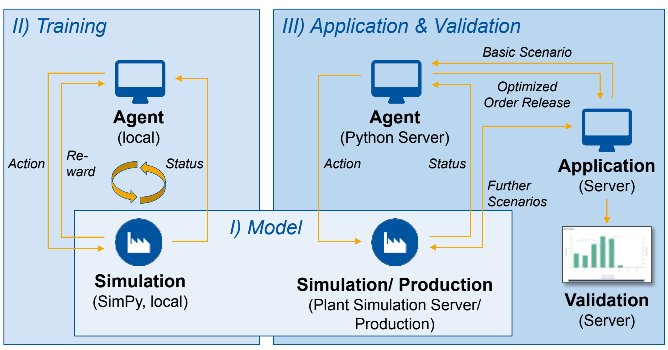

Fig. 2. Setup for the implementation of an automated order release system

In a first modeling step, a *simulation model* (I) is used to dynamically represent the *status* of a production system by modeling the production process and output feedback data. Second, in the *training* phase (II) a RL *agent* is trained by interacting with the simulation model to derive a control strategy for order release. In the third *application and validation* phase (III), the trained *agent* is implemented on a server to control the order release process of a real production process via *simulation* interface. This setting in a final step can be used for comparing and *validating* different scenarios against each other.

4.1. Model

The model used in the training phase is implemented in SimPy as a custom-made python simulation to quickly communicate with the agent implemented in python as well. Although this is a simplified simulation model, specially designed for this application, the basic requirements are to depict the available production workstations, the shift model, the machine status and capacities and order information e.g. work plans, remaining operation times and order progress. In addition, influencing factors of a real production such as downtimes can be considered.

During the training phase, a simplified python simulation on a local computer is sufficiently precise and fast, in the application phase the simulation model must be synchronizable with real production data and hence must be accessible on a server. In addition, to guarantee modeling the production functionality as real as possible, a more detailed model in Plant Simulation is used. This simulation software constitutes an industry standard with advanced modeling and automation possibilities, which is important for an effective use in practice especially when the synchronization with a real production is planned. This is done by a set of feedback data that is exchanged with the model by using automated file updates or a shared data base of the ERP/ MES and the simulation model.

Based on this setting, if there is a change of the considered production system both simulation models – in SimPy and Plant Simulation – have to be adapted, because an agent can only perform correct strategies, if it has been trained with the system at hand.

4.2. Training

The described simulation model serves as the training environment which a RL algorithm interacts with. The exact RL method must be chosen on basis of the problem size and the considered scheduling horizon. Related works show the DQN training method has proven to be a suitable approach [10, 43].

The training requires historical data that in combination with the simulation model represents the functionality and common product program of the production. In the step-by-step training phase, the RL algorithm releases an order from the inserted list of generated orders into the simulated production environment and gets a reward based on the reinforcement and value function. Developing these exact functions will be a key component in terms of speed and scalability of the proposed approach. Through a multi-dimensional objective function the adherence to delivery date as well as the throughput time will be optimized. For each time step of the interaction, the agent receives a reward and takes the next action based on the observed status of the system. During the method development, the parameters of the algorithm have to be iteratively adapted to create suitable results and to improve the practicability to be implemented in a real production environment.

4.3. Application and validation

Finally, the described two steps of modeling and training are combined into an automated methodology able to efficiently control the complex task of order release in a job shop production. Therefore, the simulation model in Plant Simulation serves as the correspondent production environment for the then trained agent. The communication protocol is based on the state message logic proposed by [10]. The simulation can be used alone or integrated with near-real time data from a real production. In this case, the already mentioned data is required, e.g. work plans, current machine information, current order information and current list of generated orders.

In the application phase, the RL agent acts on basis of the strategy derived from training in order to meet the required logistical target values addressed in the objective function adherence to delivery date and throughput time. According to this strategy, the agent releases the most suitable orders from a constantly updated list of orders with specific information on the delivery date and order type.

To achieve a fully automated process of order release the developed communication interface between agent and simulation is not sufficient but needs to be complemented by an online application, which calls the agent and triggers the task. Moreover, with this instance different scenarios can be compared and validated according to the logistic target values. By setting up the required simulation parameters, further scenarios can be created in the application instance, which are then executed either through the agent’s release strategy or by choosing a common heuristic such as Conwip.

5. Conclusion and further research

This paper elaborates on the importance of order release and compares current approaches to the JSP with four established challenges that keep remaining in that domain: Consideration of specific production parameters, integration of feedback data, enabling a multidimensional objective function and applicability to a practical job shop production.

The proposed approach includes the automation of the order release task by combining industrial-grade simulation with a RL agent. In the training phase, the RL agent interacts with a simplified discrete-event simulation in order to quickly derive a strategy optimizing the order selection and timing. This trained agent is then applied to a more detailed production simulation to release orders based on the learned policy that tries to optimize the adherence to delivery date and throughput time. It is automated by the overlying online application.

In order to implement this automated order release application, the RL algorithm as well as the simulation model have to be developed based on the requirements of the order release process. Therefore, parameters and the value function defining the RL algorithm have to be specified in order to fulfill the specified logistical target values able to outperform current approaches. To apply the method to a real production environment, especially the feedback data requirements need to be investigated in order to integrate these into the corresponding simulation model. Finally, as a production environment dynamically changes over time, at a certain point it might be required to reiterate the training phase in order to cope with the current production and keep the agent up to date.

Acknowledgements

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany‘s Excellence Strategy – EXC-2023 Internet of Production – 390621612

References

[1] Schmidtke N, Behrendt F, Thater L, Meixner S. Technical potentials and challenges within internal logistics 4.0. In: *2018 4th International Conference on Logistics Operations Management (GOL)*: IEEE; 2018 - 2018. p. 1.

[2] Duffie N, Bendul J, Knollmann M, 2017. An analytical approach to improving due-date and lead-time dynamics in production systems *45*, p. 273.

[3] Thommen, J.-P., Achleitner, A.-K., Gilbert, D.U., Hachmeister, D., Kaiser, G., Editors; 2017. *Allgemeine Betriebswirtschaftslehre*. Springer Fachmedien Wiesbaden.

[4] Haeussler S, Stampfer C, Missbauer H, 2020. Comparison of two optimization based order release models with fixed and variable lead times *227*, p. 107682.

[5] Zijm H, Regattieri A. Manufacturing Planning and Control Systems. In: Zijm H, Klumpp M, Regattieri A, Heragu S (eds.). *Operations, Logistics and Supply Chain Management*. Cham: Springer International Publishing; 2019. p. 251.

[6] Schuh, G; 2012. *Produktionsplanung Und -Steuerung 1: Grundlagen der PPS,* 4th edn. Springer Berlin / Heidelberg, Berlin, Heidelberg.

[7] Pürgstaller P, Missbauer H, 2012. Rule-based vs. optimisation-based order release in workload control: A simulation study of a MTO manufacturer *140*, p. 670.

[8] Lang S, Schenk M, Reggelin T, 2019. Towards Learning- and Knowledge-Based Methods of Artificial Intelligence for Short-Term Operative Planning Tasks in Production and Logistics: Research Idea and Framework *52*, p. 2716.

[9] Xie J, Gao L, Peng K, Li X *et al.*, 2019. Review on flexible job shop scheduling *1*, p. 67.

[10] Kemmerling M, Samsonov V, Lütticke D, Schuh G *et al.* Towards Production-Ready Reinforcement Learning Scheduling Agents: A Hybrid Two-Step Training Approach Based on Discrete-Event Simulations. In: Franke J, Schuderer P (eds.). *Simulation in Produktion und Logistik 2021*. Göttingen: Cuvillier Verlag; 2021. p. 325.

[11] Stevenson M, Hendry LC, Kingsman † BG, 2005. A review of production planning and control: the applicability of key concepts to the make-to-order industry *43*, p. 869.
[12] Lödding, H; 2016. *Verfahren der Fertigungssteuerung: Grundlagen, Beschreibung, Konfiguration,* 3rd edn. Springer Berlin Heidelberg, Berlin, Heidelberg.
[13] RODERICK LM, PHILLIPS DTON, HOGG GL, 1992. A comparison of order release strategies in production control systems *30*, p. 611.
[14] Lingayat S, Mittenthal J, O'Keefe RM, 1995. Order Release in Automated Manufacturing Systems *26*, p. 175.
[15] Zijm, H., Klumpp, M., Regattieri, A., Heragu, S., Editors; 2019. *Operations, Logistics and Supply Chain Management*. Springer International Publishing.
[16] Schuh G, Stich V, Reuter C, Blum M *et al.* Cyber Physical Production Control. In: Jeschke S, Brecher C, Song H, Rawat DB (eds.). *Industrial Internet of Things: Cybermanufacturing Systems*. Cham: Springer International Publishing; 2018. p. 519.
[17] Garey MR, Johnson DS, Sethi R, 1976. The Complexity of Flowshop and Jobshop Scheduling *1*, p. 117.
[18] Gonçalves JF, Magalhães Mendes JJ de, Resende MGC, 2005. A hybrid genetic algorithm for the job shop scheduling problem *167*, p. 77.
[19] Vinod V, Sridharan R, 2011. Simulation modeling and analysis of due-date assignment methods and scheduling decision rules in a dynamic job shop production system *129*, p. 127.
[20] Silver D, Schrittwieser J, Simonyan K, Antonoglou I *et al.*, 2017. Mastering the game of Go without human knowledge *550*, p. 354.
[21] Mnih V, Kavukcuoglu K, Silver D, Rusu AA *et al.*, 2015. Human-level control through deep reinforcement learning *518*, p. 529.
[22] Panzer, M., Bender, B., Gronau, N; 2021. *Deep Reinforcement Learning In Production Planning And Control: A Systematic Literature Review*. Hannover Institutionelles Repositorium der Leibniz Universität Hannover.
[23] Sutton, R.S., Barto, A; 2018. *Reinforcement learning: An introduction,* 2nd edn. MIT Press, Cambridge.
[24] Harmon, M.E., Harmon, S.S; 1997. *Reinforcement Learning: A Tutorial*. Defense Technical Information Center, Fort Belvoir, VA.
[25] Overbeck L, Hugues A, May MC, Kuhnle A *et al.*, 2021. Reinforcement Learning Based Production Control of Semi-automated Manufacturing Systems *103*, p. 170.
[26] Hubbs CD, Li C, Sahinidis NV, Grossmann IE *et al.*, 2020. A deep reinforcement learning approach for chemical production scheduling *141*, p. 1.
[27] Waschneck, B; 2020. *Autonome Entscheidungsfindung in der Produktionssteuerung komplexer Werkstattfertigungen*.
[28] Gutenschwager, K., Rabe, M., Spieckermann, S., Wenzel, S; 2017. *Simulation in Produktion und Logistik: Grundlagen und Anwendungen*. Springer Vieweg, Berlin.
[29] Frazzon EM, Kück M, Freitag M, 2018. Data-driven production control for complex and dynamic manufacturing systems *67*, p. 515.
[30] Zhang T, Xie S, Rose O. Real-time job shop scheduling based on simulation and Markov decision processes. In: Chan WK, D'Ambrogio A, Zacharewicz G, Mustafee N, Wainer G, Page EH (eds.). *WSC'17: 2017 Winter Simulation Conference WSC turns 50: simulation everywhere! December 3-6, 2017, Red Rock Casino Resort & Spa, Las Vegas, NV*. Piscataway, NJ: IEEE; 2017. p. 3899.
[31] Land MJ, Gaalman GJC, 1998. The performance of workload control concepts in job shops: Improving the release method *56-57*, p. 347.
[32] Hutter T, Haeussler S, Missbauer H, 2018. Successful implementation of an order release mechanism based on workload control: a case study of a make-to-stock manufacturer *56*, p. 1565.
[33] Schneckenreither M, Haeussler S. Reinforcement Learning Methods for Operations Research Applications: The Order Release Problem. In: Nicosia G, Pardalos P, Giuffrida G, Umeton R, Sciacca V (eds.). *Machine Learning, Optimization, and Data Science*. Cham: Springer International Publishing; 2019. p. 545.
[34] Kundu K, Staudacher AP. Order review and release methods in internal logistics of a MTO company. In: *2017 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*: IEEE; 2017 - 2017. p. 133.
[35] Wu C-X, Liao M-H, Karatas M, Chen S-Y *et al.*, 2020. Real-time neural network scheduling of emergency medical mask production during COVID-19. Appl Soft Comput *97*, p. 106790.
[36] Cakar T, Yildirim MB, Barut M, 2005. A neuro-genetic approach to design and planning of a manufacturing cell *16*, p. 453.
[37] Jones A, Rabelo LC, Sharawi AT. Survey of Job Shop Scheduling Techniques. In: Webster JG (ed.). *Wiley Encyclopedia of Electrical and Electronics Engineering*. Hoboken, NJ, USA: John Wiley & Sons, Inc; 2001.
[38] Balas E, 1965. An Additive Algorithm for Solving Linear Programs with Zero-One Variables *13*, p. 517.
[39] Balas E, 1969. Machine Sequencing Via Disjunctive Graphs: An Implicit Enumeration Algorithm *17*, p. 941.
[40] Pinedo, M.L; 2016. *Scheduling*. Springer International Publishing, Cham.
[41] Ahmadian MM, Salehipour A, Cheng TCE, 2021. A meta-heuristic to solve the just-in-time job-shop scheduling problem *288*, p. 14.
[42] Palombarini JA, Martinez EC. Automatic Generation of Rescheduling Knowledge in Socio-technical Manufacturing Systems using Deep Reinforcement Learning. In: *IEEE ARGENCON 2018: Proceeding*. Piscataway, NJ: IEEE; 2018. p. 1.
[43] Samsonov V, Kemmerling M, Paegert M, Lütticke D *et al.* Manufacturing Control in Job Shop Environments with Reinforcement Learning. In: *Proceedings of the 13th International Conference on Agents and Artificial Intelligence*: SCITEPRESS - Science and Technology Publications; 2021 - 2021. p. 589.