COMBINED USE OF AI TECHNIQUES AND SIMULATION TO SUPPORT PRODUCTION SCHEDULING: EVIDENCE FROM EMPIRICAL RESEARCH

Romeo Bandinelli[1] and Virginia Fani[1]

[1]Department of Industrial Engineering, University of Florence, e-mail: romeo.bandinelli@unifi.it

KEYWORDS

Discrete Event Simulation, DES, Agent Based Simulation, ABS, Artificial Intelligence, Scheduling

ABSTRACT

This paper reports a literature review on the Artificial Intelligence (AI) techniques applied in the field of production planning and scheduling and explores the synergistic integration of AI with simulation for enhancing production scheduling. Leveraging Microsoft Project Bonsai and AnyLogic, we present a case study that demonstrates the effectiveness of AI-driven simulation models in optimizing scheduling tasks. Our research highlights the potential of combining AI with traditional simulation methods. The results offer insights into the practical applications and future potential of AI in industrial automation and production management.

INTRODUCTION

In the dynamic landscape of industrial production, the efficacy of production scheduling emerges as a pivotal factor driving efficiency and competitiveness Fani et al. (2017). This paper delves into the integration of Artificial Intelligence (AI) with simulation to revolutionize production scheduling practices. The advent of AI has opened new avenues for addressing complex scheduling challenges, traditionally reliant on heuristic and deterministic methods. By harnessing AI's predictive and adaptive capabilities, we aim to transcend traditional limitations, offering more resilient and efficient scheduling solutions. Our research not only underscores the importance of innovative scheduling in the context of Industry 4.0 but also provides empirical evidence of its practical application and benefits. The integration of AI with simulation, as demonstrated through a case study using Microsoft® Project Bonsai and AnyLogic®, demonstrates the feasibility of this type of approach beginning with commercial solutions and its possible application in industrial plants.

As previously mentioned, the focus of the study concerns the use of AI techniques to support the specific operational level problem of the production scheduling in the industrial field. This is one of the most critical manufacturing tasks and has been extensively studied by the scientific community.

Although there have been many intelligent scheduling algorithms, it is still an important goal to seek more efficient and practical approaches. As stated by Spanos et al. (2022), from an optimization standpoint, detailed production scheduling is an extraordinarily complicated issue, with most situations being classified as being non-deterministic polynomial-time NP-hard. It means that it's very difficult to find an exact and optimal solution in a reasonable time, especially in the case of complex problems involving a large number of resources and constraints. For that reason, in recent years there has been growing attention on the use of AI techniques, as they can provide an efficient and flexible alternative to support these kinds of activities.

In this context, the following research questions have been selected:

RQ1: Which AI techniques does the scientific literature propose to support production scheduling?

RQ2: What opportunities does the combined use of AI techniques and simulation open up?

LITERATURE REVIEW

In order to answer the RQ1, a literature review has been carried out and numerous algorithms for solving scheduling problems have been selected and studied. This broad range of approaches offers a large number of options and methods for addressing specific production scheduling challenges. For this reason, in order to evaluate the adaptability to specific problems of all the methods analyzed in the previous sections, it is necessary to provide a classification dividing them into different categories based on their key characteristics. In particular, the above classification is presented in table form. Scheduling algorithms are shown on the rows, while methodologies to which they belong, classes of the scheduling problem, scheduling strategies, objective functions and bibliographic references characterize the columns.

Scheduling strategy refers to whether the algorithm belongs to static (S) or dynamic (D) scheduling. Here, static scheduling refers to a schedule that is established in advance and remains unchanged during execution. In other words, all parameters and constraints related to the production system are known and constant over time. On

the other hand, dynamic scheduling refers to a schedule with the ability to deal with uncertain disturbance in real time.
The notation used in the scheduling techniques classification (Table 2) is shown in Table 1.

Table 1: Scheduling techniques classification notation

	Notation	Description
Scheduling Problem	FSSP	Flow Shop Scheduling Problem
	PFSSP	Permutation Flow Shop Scheduling Problem
	JSSP	Job Shop Scheduling Problem
	FJSSP	Flexible Job Shop Scheduling Problem
	SMSP	Single Machine Scheduling Problem
Scheduling strategy	S	Static Scheduling
	D	Dynamic Scheduling
Objective Function	Min TC	Minimize Total Cost
	Min PC	Minimize Production Cost
	Min TEC	Minimize Total Energy Consumption
	Max MU	Maximize Average Machine Utilization
	Min VMW	Minimize Variance of Machine Workload
	Min DT	Minimize Delay Time
	Min LT	Minimize Lead Time
	Min TST	Minimize Total Setup Time
	Min MCT	Minimize Maximum Completion Time
	Min CT	Minimize Mean Cycle Time
	Min TT	Minimize Total Tardiness
	Min MT	Minimize Mean Tardiness
	Min TWT	Minimize Total Weighted Tardiness
	Min LJ	Minimize Number of Late Jobs
	Max OJ	Maximize Number of Jobs Completed On Time

Table 2: Scheduling techniques classification

Algorithm	Methodology	Scheduling Problem	Scheduling strategy	Objective Function	Source
MILP	MP	FSSP	S	Min TC	(Vahedi-Nouri et al., 2021)
MILP	MP	JSSP	S	Min Makespan	(Mourtos et al., 2021)
CP	MP	FJSSP	S	Min Makespan	(Schweitzer et al., 2023)
CP	MP	JSSP	S	Min DT	(Kovács et al., 2021)
ACO	MH	FJSSP	S	Min Makespan	(Torres-Tapia et al., 2022)
AH	MH	JSSP	S	Min TT	(Elmenreich et al., 2021)
GA	MH	FJSSP	S	Min Makespan	(Govi et al., 2021)
GA	MH	JSSP	S	Min Makespan	(Chen and Zhan, 2022)
GA	MH	JSSP	S	Min TC	(Ghasemi et al., 2022)
GA	MH	JSSP	S	Min Makespan	(Dai et al., 2022)
GA	MH	FSSP	S	Min LT, Min PC	(Fülöp et al., 2022)
GA	MH	SMSP	S	Min LJ, Min TST	(Z. Zhao et al., 2021)
GA	MH	JSSP	S	Min Makespan, Min MT	(Salama et al., 2022)
PSO	MH	JSSP	S	Min Makespan	(Xu and Wu, 2021)
ANN	AI	JSSP	S	Min TC	(Antons and Arlinghaus, 2022b)
ANN	AI	JSSP	S	Min TC	(Antons and Arlinghaus, 2022a)
ANN	AI	FSSP	D	Min Makespan	(Azab et al., 2021)
DRL	AI	JSSP	S	Min Makespan	(Elsayed et al., 2022)
DRL	AI	FJSSP	S	Min Makespan	(Popper and Ruskowski, 2022)
DRL	AI	JSSP	D	Min Makespan	(Wang et al., 2021)
DRL	AI	FJSSP	D	Min LT	(Burggräf et al., 2022)
DRL	AI	PFSSP	S	Min LJ	(Dong et al., 2022)
DRL	AI	FSSP	D	Min Makespan	(Grumbach et al., 2022)
DRL	AI	FJSSP	S	Min Makespan	(Elsayed et al., 2021)
DRL	AI	JSSP	D	Min Makespan, Min PC	(Zhou et al., 2021)
DRL	AI	FJSSP	D	Min TWT, Max MU, Min VMW	(Luo et al., 2022)
QL	AI	PFSSP	S	Min Makespan	(Vijayan and Parameshwaran Pillai, 2022)
QL	AI	JSSP	S	Min Makespan	(Dasbach et al., 2022)
QL	AI	FSSP	S	Min TEC	(Wilk-Kołodziejczyk et al., 2022)
QL	AI	FJSSP	D	Min Makespan	(Said et al., 2021)
QL	AI	JSSP	D	Max Throughput, Min CT, Min TWT	(Ghaleb et al., 2021)
DQN	AI	JSSP	D	Min DT	(Y. Zhao et al., 2021)
DQN	AI	PFSSP	S	Min MCT	(Yang et al., 2022)

DQN	AI	FJSSP	D	Min Makespan, Min TEC	(Li et al., 2022)
DQN	AI	FSSP	D	Max Throughput, Max OJ	(Marchesano et al., 2022)
DQN	AI	JSSP	S	Min TT	(Liu et al., 2021)
DQN	AI	JSSP	S	Min Makespan, Min TEC	(Eriksson et al., 2022)
DQN	AI	JSSP	D	Min TWT	(Zhang et al., 2022)

CASE STUDY ON AI TECHNIQUES AND SIMULATION

In order to answer to RQ2, a case study on the application of a commercial AI solution, the Microsoft® Bonsai platform, and the commercial AnyLogic® simulation software has been carried out.
This research adopts a two-pronged methodology, combining AI techniques with simulation. The AI component involves training a reinforcement learning agent using Microsoft® Project Bonsai. This agent learns optimal scheduling strategies through iterative simulations, adjusting to varying production scenarios. The simulation aspect employs AnyLogic®, a versatile platform for modeling complex systems. Here, we simulate a production environment to test and refine the AI's scheduling decisions. The methodology focuses on real-time data integration and adaptive learning, ensuring that the AI agent can respond effectively to dynamic changes in the production environment. This approach bridges the gap between theoretical AI models and practical, real-world applications in industrial scheduling.
To elaborate further on the methodology, the following tools have been adopted.

AI Agent Training

We utilized Microsoft Project Bonsai for training an AI agent in reinforcement learning. This involved setting up scenarios where the AI could learn from different scheduling challenges, iteratively improving its decision-making abilities.

Simulation Environment

AnyLogic was employed to create a detailed simulation of the production environment. This included modeling various production elements like machinery, labor, and workflow, to accurately reflect real-world conditions.

Integration of AI with Simulation

The AI agent's decisions were tested and refined within the AnyLogic simulation environment. This allowed for real-time evaluation and adaptation of the AI's strategies, ensuring they were practical and effective in a simulated production context.

Data Handling and Processing

A crucial part of our methodology was the handling of data, both for training the AI and for running simulations. This involved ensuring data accuracy, relevance, and timeliness.
Feedback Loop for Continuous Improvement: The methodology incorporated a feedback loop where the results from the simulation were used to further train and refine the AI agent, making the system increasingly efficient over time.
By combining AI's analytical power with the practical insights from simulation, this methodology aimed to create a robust and adaptable scheduling tool suitable for the complexities of modern production environments.

RESULTS

From the results of the previous section, it emerges how the field of RL is a promising solution to meet the demands of the Smart Manufacturing paradigm. The application of RL in manufacturing production enables the automation of complex processes, allowing for the autonomous execution of repetitive and routine tasks. Additionally, it addresses the variability of production demands by adapting scheduling strategies and resource allocation in real-time.
In order to take advantage of the implementation of an RL agent in the production system, it is essential to consider the importance of simulation as a complementary tool. Indeed, simulation provides a controlled and reproducible virtual environment in which ML agents can be trained and evaluated efficiently.
In this context, the research question RQ2, defined as "*What opportunities does the combined use of AI techniques and simulation open up?*", has been developed. In relation to RQ2, the following results will be reported:

- First, the most promising areas for the development of simulation and AI combined simulations are reported.
- Next step, it is described the procedure to convert the regular AnyLogic testcase model, into a Microsoft Project Bonsai ready simulator.
- Finally, a brain on the Bonsai platform is created and subsequently trained in the simulation environment developed in the previous step.

AnyLogic testcase model connectable with Bonsai

This section describe how the an AnyLogic model has been deployed into an RL-ready simulation model and executing the wrapping process, which consists of incorporating the RL-ready version of the testcase into a second model named "Wrapper" model. This second model already includes all the needed dependencies to make the proper connection to the Bonsai platform.
Initially, in order to modify the starting testcase model into an RL-ready simulation model, there are two main changes that are directly related to the RL training.
First, the *BonsaiEvent* pauses the simulation every six months, and this would be the moment in time that the

model triggers the reinforcement learning loop. It would be used if each episode or each simulation is run for more than six months with a non-stationary arrival rate, and the brain is expected to learn how adaptively change the four parameters to keep the production cost to a minimum.

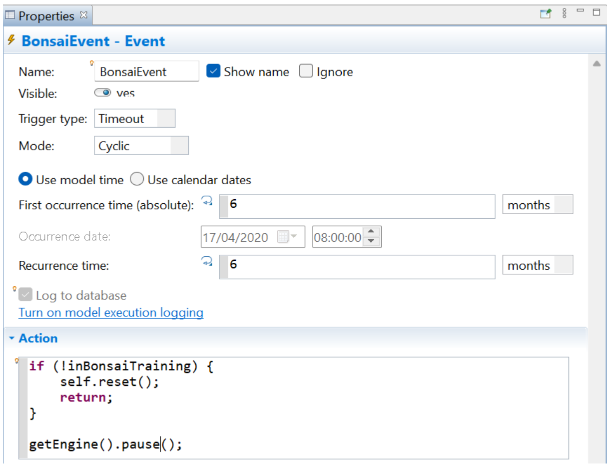

Figure 1. BonsaiEvent properties

The second important modification in the model is the addition of a boolean variable named *exceededCapacity*. In the “On enter” field of *auxQueueA* the variable *exceededCapacity* is set to be “true” if the current size of *auxQueueA* has its capacity reached (Figure 2). The *exceededCapacity* variable is used by the brain as a flag that the current input variables fail to keep the production flow at a desirable level and resulted in accumulation of unproduced orders in *auxQueueA*.

On enter:

```
// if the queue is overflowed
if (auxQueueA.size() >= auxQueueA.capacity) {
    // set the flag
    exceededCapacity = true;
    if (!inBonsaiTraining) {
        finishSimulation();
    }
}
```

Figure 2. Definition of the boolean variable exceededCapacity

To start the wrapping process, there are two necessary assets that are available for download from the AnyLogic website:

- *Bonsai Connector Library*: A custom AnyLogic library that, under the hood, is used to communicate between the RL-ready model and the Bonsai platform.
- *Wrapper Model*: It is similar to any other AnyLogic model, but it is not intended to be used by itself. Instead, the RL-ready model will get incorporated into it and this is what will be used as Bonsai simulator.

Then, the RL-ready testcase model is open alongside the Wrapper model and the top-level agent, named “Main”, is dragged and released from the Projects panel into the Wrapper agent. While agent types are typically instantiated from within one model, in this case it’s being done across separate models. For this reason, from this point on, all the animation from the testcase model’s root agent will appear inside the Wrapper agent. Furthermore, running experiments from the Wrapper model will also run the original model.

Training the AI agent with Bonsai

Bonsai allows to manage these two phases directly through its graphical interface, which presents a teach tab and a train tab. The teach tab consists of a coding panel, in which it is possible to write the code underlying the functioning of the brain, and a graphing panel, which represents the iterative learning process defined by the written code. On the other hand, the train tab replaces the coding panel with an empty data panel and shows an updated teaching graph. When the training process is starting, Bonsai automatically starts up a fleet of simulator instances. The fleet appears in the updated graph.
With each iteration, the brain earns a performance score based on how well it solved the problem. Bonsai reports training progress for the brain in the data panel as a Goal Satisfaction plot. Individual goal satisfaction values indicate how close the brain got to achieving the related goal for a given iteration. The latest overall goal satisfaction value is also reported in the concept node of the teaching graph.
Subsequently, Bonsai service creates container instances inside the personal Azure subscription to be able to run the training. In this case, the number of instances is configured to 50. At this point, the training process can begin, and it can be followed in real-time on the train tab of the Bonsai platform. Indeed, within the data panel of the train tab there are charts that can be configured directly by the user, and which are updated whenever the brain finds an improved policy during training.
Bonsai automatically stops training when either of the following occurs: the overall goal satisfaction value reaches 100% or the graph lines become horizontal lines for a predefined number of training episodes.
A 100% satisfaction value means that the brain has fully learned the current curriculum. A horizontal plot line means the brain is no longer improving. When a brain fails to improve after a given number of episodes, Bonsai terminates the training to avoid wasting computational resources. It is also possible to stop the training process early by clicking the red Stop Training button at the top of the graphing panel.
In the test case the training process had a duration of 1 hour and 32 minutes, and the latest mean reward is equal to -45.78. The number of iterations produced has been approximately 1,2 million.

AnyLogic-Bonsai AI agent application

A problem underlying the RL is that there are opportunities that the algorithm could unlearn what it has learned before, and never really be able to make it back. Considering this, the Bonsai platform uses the champion

challenger approach. This method consists in comparing a reference neural network, called “champion”, with an alternative neural network, called "challenger", keeping always the best one of the two solutions.
The next phase is the one called assessment. It refers to the process of evaluating the performance of the brain once it has been trained. During this phase, various evaluations and tests are performed to measure the effectiveness of the brain in addressing the specific problem for which it is designed.
The process with which it is possible to carry out this phase on the Bonsai platform is very similar to the one used for training. In other words, it is necessary to connect the AnyLogic testcase model to the AI platform and then start the assessment phase by clicking on the appropriate field. Assessing will allow to see what the final neural network looks like and if it is capable to bring the desired benefits.
Once the brain is validated, it can be exported and deployed as an Azure app. This process consists in taking the neural network, with all the components that are necessary to run that neural network and packaging it up into a container registry inside the personal Azure subscription. Then, it is possible to get access to the web app and hosted brain via URL.
To see the trained brain controlling the simulation, it is necessary to first check the “playback” checkbox into the properties of the Bonsai connector object of the testcase model, and then paste the web app URL into the field named “Exported brain address”. The result of this operation is shown in Figure 3.

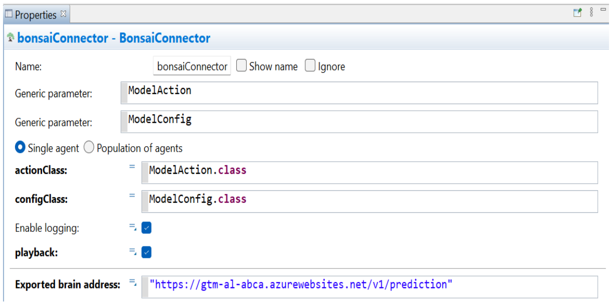

Figure 3: Bonsai connector properties

Finally, it is possible to run the *AnimatedExperiment* and observe the model run with the trained brain in control.

CONCLUSION

By following the methodological steps proposed in this study, three significant outcomes were achieved.
Regarding the first research question (RQ1), a systematic literature review was conducted to provide an overview of production scheduling algorithms studied in the literature. A notable finding emerged, which categorizes these methods into three distinct groups: Mathematical Programming, Metaheuristics, and Artificial Intelligence techniques. Additionally, the results of the review revealed a growing interest among researchers in the field of Artificial Intelligence, particularly in the sub-field of Reinforcement Learning. Alongside presenting the various algorithms belonging to these three categories, the review also described the most frequently encountered classes of scheduling problems: Job Shop Scheduling Problem, Flexible Job Shop Scheduling Problem, Flow Shop Scheduling Problem, and Permutation Flow Shop Scheduling Problem.
Interpreting the results of the systematic literature review, some limitations need to be considered. Specifically, the research activity of the papers was conducted exclusively on the Scopus database, and the search query string was defined considering the English-language documents published in the period from January 2021 to January 2023 inclusive.
Next, in order to answer to the second research question (RQ2) with focus on RL-based techniques, a case study was conducted concerning the training on the AnyLogic® virtual environment of an AI agent created directly on the Microsoft® Project Bonsai platform. In particular, consistently the main objective of the analysis was to evaluate the potential deriving from the combined use of these two solutions.
The first experimental step is based on the conversion of a regular AnyLogic model, already validated and taken from the AnyLogic example repository, into a Microsoft Project Bonsai ready simulator. Then, the brain was created on Bonsai with the use of inkling code, and subsequently it was trained using the AnyLogic environment. Once the brain Antons has been validated, it was finally exported and deployed as an Azure app.
Ultimately, several key factors emerged that could influence the implementation of a RL-based APS system within enterprises. These factors, known as drivers and barriers, play a decisive role in shaping the decision to adopt or not such a system.

REFERENCES

Antons, O., Arlinghaus, J.C., 2022a. Machine Learning and Autonomous Control—A Synergy for Manufacturing. Studies in Computational Intelligence 1034, 417–428.
Antons, O., Arlinghaus, J.C., 2022b. Data-driven and autonomous manufacturing control in cyber-physical production systems. Computers in Industry 141.
Azab, E., Nafea, M., Shihata, L.A., Mashaly, M., 2021. A machine-learning-assisted simulation approach for incorporating predictive maintenance in dynamic flow-shop scheduling. Applied Sciences (Switzerland) 11.
Burggräf, P., Wagner, J., Saßmannshausen, T., Ohrndorf, D., Subramani, K., 2022. Multi-agent-based deep reinforcement learning for dynamic flexible job shop scheduling. Procedia CIRP, pp. 57–62.
Chen, Q., Zhan, P., 2022. Research on Job Shop Scheduling Algorithm of Intelligent Manufacturing Based on Machine Learning. ACM International Conference Proceeding Series, pp. 942–945.
Dai, L., Lu, H., Hua, D., Liu, X., Chen, H., Glowacz, A., Królczyk, G., Li, Z., 2022. A Novel Production

Scheduling Approach Based on Improved Hybrid Genetic Algorithm. Sustainability (Switzerland) 14.

Dasbach, T., Olbort, J., Wenk, F., Ander, R., 2022. Sequencing Through a Global Decision Instance Based on a Neural Network. IFIP Advances in Information and Communication Technology 639 IFIP, 334–344.

Dong, Z., Ren, T., Weng, J., Qi, F., Wang, X., 2022. Minimizing the Late Work of the Flow Shop Scheduling Problem with a Deep Reinforcement Learning Based Approach. Applied Sciences (Switzerland) 12.

Elmenreich, W., Schnabl, A., Schranz, M., 2021. An artificial hormone-based algorithm for production scheduling from the bottom-up. ICAART 2021 - Proceedings of the 13th International Conference on Agents and Artificial Intelligence, pp. 296–303.

Elsayed, A.K., Elsayed, E.K., Eldahshan, K.A., 2021. Deep Reinforcement Learning based Actor-Critic Framework for Decision-Making Actions in Production Scheduling. Presented at the Proceedings - 2021 IEEE 10th International Conference on Intelligent Computing and Information Systems, ICICIS 2021, pp. 32–40.

Elsayed, E.K., Elsayed, A.K., Eldahshan, K.A., 2022. Deep Reinforcement Learning-Based Job Shop Scheduling of Smart Manufacturing. Computers, Materials and Continua 73, 5103–5120.

Eriksson, K., Ramasamy, S., Zhang, X., Wang, Z., Danielsson, F., 2022. Conceptual framework of scheduling applying discrete event simulation as an environment for deep reinforcement learning. Procedia CIRP, pp. 955–960.

Fani, V., Bandinelli R., Rinaldi R., 2017, A simulation optimization tool for the metal accessory suppliers in the fashion industry: A case study, Proceedings - 31st European Conference on Modelling and Simulation, ECMS 2017Pages 240 - 2462017 31st European Conference on Modelling and Simulation, ECMS 2017, Budapest, 23 May 2017, 26 May 2017

Fülöp, M.T., Gubán, M., Gubán, Á., Avornicului, M., 2022. Application Research of Soft Computing Based on Machine Learning Production Scheduling. Processes 10.

Ghaleb, M., Namoura, H.A., Taghipour, S., 2021. Reinforcement Learning-based Real-time Scheduling under Random Machine Breakdowns and Other Disturbances: A Case Study. Proceedings - Annual Reliability and Maintainability Symposium.

Ghasemi, A., Ashoori, A., Heavey, C., 2021. Evolutionary Learning Based Simulation Optimization for Stochastic Job Shop Scheduling Problems. Applied Soft Computing 106.

Ghasemi, A., Kabak, K.E., Heavey, C., 2022. Demonstration of the Feasibility of Real Time Application of Machine Learning to Production Scheduling. Presented at the Proceedings - Winter Simulation Conference, pp. 3406–3417.

Govi, D., Rizzuto, A., Schipani, F., Lazzeri, A., 2021. A Two-stage Genetic Algorithm for a Novel FJSP with Working Centers in a Real-world Industrial Application. Proceedings of the 2nd International Conference on Innovative Intelligent Industrial Production and Logistics, IN4PL 2021, pp. 75–83.

Grumbach, F., Müller, A., Reusch, P., Trojahn, S., 2022. Robust-stable scheduling in dynamic flow shops based on deep reinforcement learning. Journal of Intelligent Manufacturing.

Kovács, B., Tassel, P., Kohlenbrein, W., Schrott-Kostwein, P., Gebser, M., 2021. Utilizing constraint optimization for industrial machine workload balancing. Presented at the Leibniz International Proceedings in Informatics, LIPIcs.

Li, Y., Gu, W., Yuan, M., Tang, Y., 2022. Real-time data-driven dynamic scheduling for flexible job shop with insufficient transportation resources using hybrid deep Q network. Robotics and Computer-Integrated Manufacturing 74.

Liu, W., Wu, S., Zhu, H., Zhang, H., 2021. An Integration Method of Heterogeneous Models for Process Scheduling Based on Deep Q-Learning Integration Agent. Presented at the Proceedings of the 16th IEEE Conference on Industrial Electronics and Applications, ICIEA 2021, 1966–1971.

Luo, S., Zhang, L., Fan, Y., 2022. Real-Time Scheduling for Dynamic Partial-No-Wait Multiobjective Flexible Job Shop by Deep Reinforcement Learning. IEEE Transactions on Automation Science and Engineering 19, 3020–3038.

Marchesano, M.G., Guizzi, G., Popolo, V., Converso, G., 2022. Dynamic scheduling of a due date constrained flow shop with Deep Reinforcement Learning. Presented at the IFAC-PapersOnLine, 2932–2937.

Mourtos, I., Vatikiotis, S., Zois, G., 2021. Scheduling Jobs on Unrelated Machines with Job Splitting and Setup Resource Constraints for Weaving in Textile Manufacturing. IFIP Advances in Information and Communication Technology 630 IFIP, 424–434.

Popper, J., Ruskowski, M., 2022. Using Multi-Agent Deep Reinforcement Learning for Flexible Job Shop Scheduling Problems. Procedia CIRP, 63–67.

Said, N.E.-D.A., Samaha, Y., Azab, E., Shihata, L.A., Mashaly, M., 2021. An Online Reinforcement Learning Approach for Solving the Dynamic Flexible Job-Shop Scheduling Problem for Multiple Products and Constraints. Proceedings - 2021 International Conference on Computational Science and Computational Intelligence, CSCI 2021, pp. 134–139.

Salama, S., Kaihara, T., Fujii, N., Kokuryo, D., 2022. Multi-Objective Approach with a Distance Metric in Genetic Programming for Job Shop Scheduling. International Journal of Automation Technology 16, 296–308.

Schweitzer, F., Bitsch, G., Louw, L., 2023. Choosing Solution Strategies for Scheduling Automated Guided Vehicles in Production Using Machine Learning. Applied Sciences (Switzerland) 13.

Spanos, A.C., Gayialis, S.P., Kechagias, E.P., Papadopoulos, G.A., 2022. An Application of a

Decision Support System Enabled by a Hybrid Algorithmic Framework for Production Scheduling in an SME Manufacturer. Algorithms 15.

Torres-Tapia, W., Montoya-Torres, J.R., Ruiz-Meza, J., 2022. A Hybrid Algorithm Based on Ant Colony System for Flexible Job Shop. Communications in Computer and Information Science 1685 CCIS, 198–209.

Vahedi-Nouri, B., Tavakkoli-Moghaddam, R., Hanzalek, Z., Dolgui, A., 2021. Integrated Workforce Allocation and Scheduling in a Reconfigurable Manufacturing System Considering Cloud Manufacturing. IFIP Advances in Information and Communication Technology 631 IFIP, 535–543.

Vijayan, S., Parameshwaran Pillai, T., 2022. Application Of A Machine Learning Algorithm In A Multi Stage Production System. Transactions of Famena 46, 91–102.

Wang, L., Hu, X., Wang, Y., Xu, S., Ma, S., Yang, K., Liu, Z., Wang, W., 2021. Dynamic job-shop scheduling in smart manufacturing using deep reinforcement learning. Computer Networks 190.

Wilk-Kołodziejczyk, D., Chrzan, K., Jaśkowiec, K., Pirowski, Z., Żuczek, R., Bitka, A., Machulec, D., 2022. Comparison of the Classical Algorithm with the Training Algorithm in Scheduling Problem ADI Production. Archives of Foundry Engineering 22, 5–12.

Xu, G., Wu, T., 2021. Application Research of Improved Particle Swarm Optimization Algorithm Used on Job Shop Scheduling Problem. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 12736 LNCS, 53–65.

Zhang, L., Yang, C., Yan, Y., Hu, Y., 2022. Distributed Real-Time Scheduling in Cloud Manufacturing by Deep Reinforcement Learning. IEEE Transactions on Industrial Informatics 18, 8999–9007.

Zhao, Y., Wang, Y., Tan, Y., Zhang, J., Yu, H., 2021. Dynamic Jobshop Scheduling Algorithm Based on Deep Q Network. IEEE Access 9, 122995–123011.

Zhao, Z., Liu, S., Zhou, M., Abusorrah, A., 2021. Dual-Objective Mixed Integer Linear Program and Memetic Algorithm for an Industrial Group Scheduling Problem. IEEE/CAA Journal of Automatica Sinica 8, 1199–1209.

Zhou, T., Tang, D., Zhu, H., Wang, L., 2021. Reinforcement Learning with Composite Rewards for Production Scheduling in a Smart Factory. IEEE Access 9, 752–766.

AUTHOR BIOGRAPHIES

ROMEO BANDINELLI is an associate professor at the University of Florence, Department of Industrial Engineering. He is member of the IFIP 5.1 "Global Product development for the whole lifecycle". His e-mail address is romeo.bandinelli@unifi.it

VIRGINIA FANI is a Post Doc Research Fellow at the Industrial Engineering Department of the University of Florence. The issues she is dealing with are related to production processes optimization along the supply chain, with particular focus on the peculiarities that the fashion companies have to face. Her e-mail address is virginia.fani@unifi.it