3rd Conference on Production Systems and Logistics

Explainable Deep Reinforcement Learning For Production Control

Philipp Theumer[1], Florian Edenhofner[1], Roland Zimmermann[1], Alexander Zipfel[1]
[1]*Fraunhofer IGCV, Fraunhofer Institute for Casting, Composite and Processing Technology IGCV, Augsburg, Germany*

Abstract

Due to the growing number of variants and smaller batch sizes manufacturing companies have to cope with increasing material flow complexity. Thus, increasing the difficulty for production planning and control (PPC) to create a feasible and economic production plan. Despite significant advances in PPC research, current PPC systems do not yet sufficiently meet the industry's requirements (e.g., decision quality, reaction time, user trust). However, recent progress in the digitalization of production systems results in an increased amount of data being collected, thus enabling the use of data-intensive applications technologies, e.g., machine learning (ML). ML provides new possibilities for PPC to handle increasing complexity caused by rising numbers of product variants paired with smaller lot sizes. At the same time, ML can increase the decision quality and reduce the reaction time to disturbances in the production system, e.g., machine breakdowns. Partly, ML models, e.g., artificial neural networks (ANN), are perceived as black-box models, resulting in reduced user's trust in the decision proposed by an ML-based PPC system. The approach presented in this publication aims at a more functional and user-friendly PPC system by leveraging multi-agent reinforcement learning (MARL), an accomplished approach within the field of ML-based production control, and approaches for explaining decisions made by reinforcement learning (RL) algorithms. With the help of MARL, short reaction time and high decision quality can be realized. Subsequently, the developed MARL system is combined with methods from the field of explainable Artificial Intelligence (XAI) to increase the users' trust. The use case results show that with the help of the developed system, rule-based controls, which are often used in industry, can be outperformed while providing explainable decisions.

Keywords

Production Control; Machine Learning; Deep Reinforcement Learning; Multi-Agent Reinforcement Learning; Explainable AI

1. Introduction

Manufacturing companies are facing an increasing material flow complexity because of a rising number of variants and a decrease in batch size [1]. More individualized production processes result in a rising production complexity and thus, in challenges (e.g., reaction to disturbances) for efficient production management [2–4]. For example, to determine a cost-optimal sequence, different set-up times, processing times, and necessary process steps for each variant have to be taken into account. With each newly introduced variant, the solution space increases.

In this context, complexity can be differentiated into static and dynamic complexity. While static complexity focuses on the long-term design of production systems, dynamic complexity results from short-term changes in production structures as well as material and information flows triggered by unpredictable disruptions (e.g., machine breakdowns) [2,5].

The dynamic complexity results in decision-making situations in the context of production control, in which the employee's experience is no longer sufficient to react optimally near real-time while considering the economic effects [6]. With the help of decision support systems, the user can be supported and thus be enabled to react appropriately and avoid adverse effects on the production system like downtime due to a material flow break [6].

Despite these potentials, the decision support of PPC systems is only partially accepted in practice. An indication of this can be seen in the frequency with which manual rescheduling is carried out. In the study conducted by LÖDDING ET AL., only 19 % of respondents stated that the planning of PPC systems is accepted and not overridden [7]. This can be caused by a lack of acceptance of the employees' proposed decisions or a low quality of the systemic proposals due to the production system's high complexity. The lack of trust is also shown in just under 35 % of respondents, who rated their confidence in the PPC system's results [7]. However, KLETTI [6] identifies employee acceptance as crucial for decision support systems in manufacturing. In particular, the user-oriented presentation of information needs to be improved in 71 % of the companies surveyed [8].

In the context of ML, the already existing problem of lack of trust in decision proposals of systems is further aggravated. ML methods, enabled by the increase of data collected, provide new possibilities for PPC to handle the rising complexity. Especially with regard to the dynamic complexity, ML methods allow the increase of decision quality while reducing the reaction time [9]. An accomplished approach for ML-based PPC is using multi-agent systems (MAS) based on deep reinforcement learning (DRL) [10–12]. For this approach, ANNs are used for choosing actions [13]. On the downside, ML models like ANNs are perceived as black-box models [14]. Therefore, the user's trust in decisions proposed by the system can be even further diminished [14].

Within the scope of this paper, an approach for a more functional and user-friendly PPC system is developed. Therefore, a MAS based on DRL is developed to realize short reaction time and high decision quality. For tackling low trust in the proposed decision, approaches from the field of explainable ML are used for realizing the explainability of the system's decisions.

2. State of research

This section focuses on a brief introduction to different approaches leveraging deep reinforcement learning in production control (2.1) and methods for explainable decision finding (2.2).

2.1 Deep reinforcement learning in production control

PPC encompasses an organization's entire materials, time, and production management, as a holistic concept [15]. The target of a PPC system is to increase the logistic performance while maintaining or reducing the logistic costs. High logistic performance is characterized by high delivery reliability and short delivery time. Low inventory and high utilization of machines are influencing the logistic costs beneficially. Due to the competing targets, these must be prioritized on a company-specific basis. PPC play a key role in achieving efficient and economical production [16]. In recent years, approaches leveraging RL have gained much attention within production control, due to the high potential of solving complex problems [10–12].

In RL, an agent interacts with its environment and learns a strategy—also called policy, $\pi(S_t)$—to maximize its reward (R_{t+1}). With the help of the policy, the agent performs an action (A_t) depending on its state (S_t). Based on S_t and A_t the agent receives R_{t+1} [13]. Within the field of RL, deep learning (DL), which uses ANNs, can be used to determine the policy for chosen agents. The combination of RL and DL is called DRL and is a subfield of ML [17]. DRL enables the agents to approximate a function to learn how to behave optimally in an environment and reach the given goals (e.g., high delivery reliability). The agent learns the

optimal strategy by choosing actions based on the current state of the environment, with the goal of maximizing a numerical reward [13]. DRL is a promising way to solve problems, which cannot or only with much effort be solved analytically [18].

A MAS consists of a set of agents interacting with the environment to perform one or more tasks jointly. The agents need information about their respective environment to achieve this optimization goal. The agents must obtain information from the environment, evaluate it with respect to the goals, and then select suitable actions [19]. WASCHNECK ET AL. [12] combined a MAS with DRL, realizing a decentralized autonomous approach for a dispatching heuristic, which was successfully tested for a production system producing semiconductors. The agents choose the best possible actions by using a policy based on a Deep Q-Network (DQN) [12]. DQN is based on Q-learning, but an ANN approximates the Q-values instead of the Q-table [20]. The risk of local optimization of the MAS was reduced by using a global reward function [12]. RÖSCH ET AL. [21] implemented a MAS using proximal policy optimization (PPO), a DRL approach, for energy-oriented production control. Agents who had the ability to control the electricity level had to cooperate with electricity-consuming agents to maximize a common reward. Therefore, jobs had to be scheduled to be completed within a given period of time while avoiding a violation of a given energy threshold.

The approaches presented were able to increase decision quality—represented by the improvements of production control—while reducing the reaction time significantly. Neither of the approaches was focusing the explainability of the decisions made by the MAS.

2.2 Explainable AI

Different approaches can realize the explainability of decisions in the context of ML. On the one hand, transparent models can be used; e.g., the parameters used for classification can be read out directly in a decision tree. Thus, the entire decision process is comprehensible, and the model does not require further processing [22]. These models are also called ante-hoc models [23]. However, ante-hoc models are disadvantageous in terms of the model's accuracy compared to opaque models (e.g., ANN) [22].

On the other hand, post-hoc methods can be used to subsequently explain decisions made by opaque models[22,23]. With the help of the post-hoc methods, the decisions of ML algorithms can be explained while levering the advantages concerning the model's accuracy [24]. There are two categories of explainability. Firstly, global explainability describes relationships learned by the model and its general behavior. Secondly, local explainability determines the influences of specific features leading towards a specific prediction [22]. Frequently used post-hoc methods include Local Interpretable Model-agnostic Explanations (LIME) [25] and SHapley Additive exPlanations (SHAP). SHAP is an approach based on the Shapley Value [26] and decomposes a model's prediction into each attribute's contribution to that prediction [27].

REHSE ET AL. [28] use an approach of explainable ML in the context of a model fabric. For this approach, a recurrent neural network is used to make predictions about the further course of the production processes. These predictions are subsequently explained by using LIME. Here, both local (individual decisions) and global explainability (general model's behavior) are realized and subsequently visualized to the user [28]. KUHNLE ET AL. [29] investigate the decision logic of a single agent with DRL using a decision tree. However, the comprehensive explainability of the agents' decisions needs to be further analyzed to overcome the black-box problem [29]. Local and global explainability was used by HUBER ET AL. [30] to explain the behavior of a DQN-agent in a single agent environment. It was investigated that the combination of a local and global explanation helped to achieve a higher performance in the tests conducted [30]. Thus, the combination of MAS with DRL and explainable Artificial Intelligence (XAI) within the production control field promises great potential for improved performance and user-oriented information visualization.

3. Approach

The approach presented within this paper—based on a MAS with DRL— intends to realize short reaction time and high decision quality without neglecting the user's trust in decisions proposed by an agent-based decision support system. Therefore, ANNs are being used to select the best possible action based on the system's current status. Due to the black-box nature of ANNs, additional steps are needed to realize the explainability of decisions. The developed approach is subdivided into two phases. (1) A user-specific observation and action space is defined based on specific user roles within a production system. Subsequently, the multi-agent framework is developed, which enables choosing the best possible action based on the system's current status. (2) Lastly, a method for the explainability of decisions made by the agent and the specific ANN is implemented. Thus, the developed system is enabled to propose explanations.

3.1 Multi-agent system with DRL

This section defines the observation space as well as the action space, which are indispensable for the use of a MAS. The observation space determines the data, which each agent receives for selecting an available action from the action space. For determining the action space for each agent, potential actions (e.g., selection of the following order, short-term capacity increase) are identified for individual user groups in PPC (e.g., production controllers, foremen). Thus, representing their ability to influence the material flow within their user-specific scope in the context of production control. To define these company-specific measures, expert interviews have to be conducted. By defining these possible actions, the decision support system can propose user role-specific measures to the users. Thus, users only receive suggestions for action and information within their scope.

Based on the different types of agents, a multi-agent framework is deducted. Figure 1 depicts a general set-up for a hierarchical MAS. The agents' action space represents measures, which the corresponding user or user group for each production resource can initiate within the scope of production control. In the example given, a second agent (e.g., a foreman) supervises two agents (e.g., machine operator). Between those two types of agents, the available action space and the observation space and, therefore, the available actions might differ. Therefore, an agent can be given a different responsibility depending on their respective abilities. For example, the agent representing a foreman might be able to change the volume of an order if necessary. In contrast, the machine operators might only determine the following order to be produced from a small number of orders.

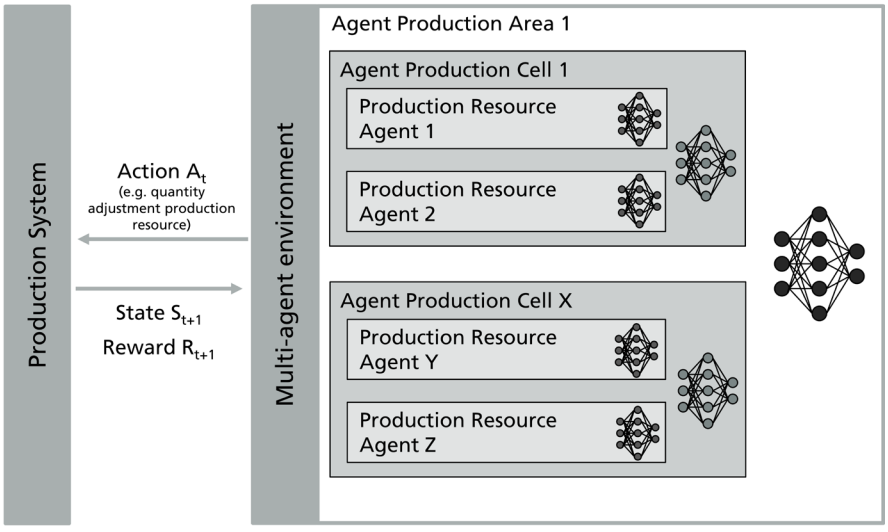

Figure 1: System architecture of the developed system with different types of agents

With the help of DRL, complex environments can be abstracted to select a suitable action given the situation [18]. Thus, enabling the determination of a policy for each separate agent, with regard to an individual observation space and action space. The agents' training occurs by interactions between the production system's simulation model and the MAS. Based on the reward given at the end of every episode (e.g., one week), the agents learn to choose the best possible action based on the system's current status. The trained agents can react near real-time on disruptions. Within the scope of this paper, PPO is used as a DRL algorithm because of its high robustness while exhibiting good learning behavior [31,32]. PPO uses the actor-critic approach, in which π is learned based on a value function [31]. This allows generalizing complex decision situations well [32]. Based on the multi-agent approach, it is possible to fulfill production control tasks while optimizing the logistic costs.

3.2 Explainability

The MAS with DRL presented in Section 3.1 focuses on improving decision-making concerning decision quality and reaction time. However, PPO is a black-box method using ANNs. In order to determine the influencing parameters on decisions, post-hoc methods are used in this paper. Thus, through DRL, a high decision quality can be maintained, and at the same time, the decision-making can be made explainable. To increase the users' confidence in the system, global explainability and local explainability are used to avoid unnecessary overrides. Post-hoc analysis requires the evaluation of the model as a whole. Therefore, it is necessary to evaluate every agent's ANN. For this purpose, all influencing factors are examined concerning their effects on selecting a particular action. Thus, an explainer can be generated, which shows the influencing factors for a given initial state and the resulting decision. Therefore, it is possible to equip a trained system with an explainer once, use them continuously and obtain a post-hoc explanation. In order to achieve this, the influence of the observations on the choice of action is calculated using SHAP values [27].

These SHAP values provide the results determined by the system. Starting from a base value (e.g., planned quantity), the influence of the individual input characteristics can be calculated to determine the resulting value [27]. SHAP enables both global and local explainability. With the help of global explainability, it is possible to present the decision-making process in a generally comprehensible way. Thus, relevant influencing parameters can be identified in general. On the one hand, this helps the user to understand the influences on the system's decision process. On the other hand, by determining the influence of different parameters, the agents' observation spaces can be adapted. This has a positive impact on the learning behavior of the agents and their ANN. The local explainability allows the determination of single influencing factors and their contribution to single decisions. Figure 2 schematically shows a so-called force plot. At the top, the specific action being explained can be seen. The features visualized in black led to a reduction from 70 to 40. The features in gray color show the features that stopped the reduction at 40. Thus, showing the influencing parameters' impact on the decision and to which extent a system built in a user-centric manner

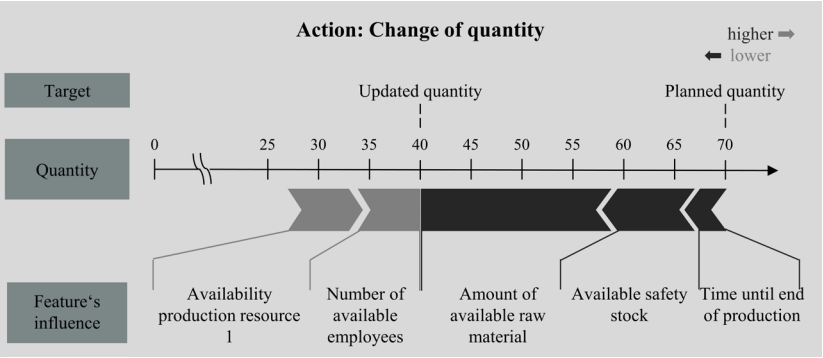

Figure 2: Decomposition of an individual action of one agents provides the influences for why this action was chosen in the given situation.

using MAS and leveraging advanced techniques such as DRL can use XAI to increase the users' trust in the proposed decisions while maintaining high decision quality and short reaction time.

4. Use Case

For the experiments being conducted within the scope of this paper, a simulation model of a production system has been used. The simulation model was built using Python as a programming language. Thus, realizing the short running times of the simulation model is beneficial for the number of iterations needed for the agents' learning process.

4.1 Simulation model

The simulation model is based on a real-life production system and consists of eight interlinked machines. Those eight machines represent a multi-stage production process. In total, the production process has four stages and parallel machines in two of them. Machines within the same stage have different (but sometimes overlapping) abilities regarding products, which can be processed as well as different processing times for specific products. Each product has to be processed once within each stage, and no skipping of process steps is allowed. Each machine can have four states (processing, waiting, set up, disturbed). Machine breakdowns are stochastically distributed and cannot be avoided. Orders for each episode (representing one week) are created based on a historic distribution at the start of each episode. Each order is characterized by product type and quantity. One episode contains, on average, 35 differing orders. All orders can be picked at the beginning of the episode. The set-up time needed for the production of an order is mainly based on the product type of the preceding order on the specific machine.

4.2 Multi-agent system

For the experiments conducted in the context of this paper, the focus has been on the improvement of order sequencing based on the system state. For this purpose, each production resource agent has up to five orders and the associated properties of the order (e.g., quantity, product type) in its observation space. Each unique order in the observation space is represented by an action in the agent's action space. Additionally to these actions, the agent can choose not to pick an order and wait if it is beneficial. If the previous action has been completed, agents can select a new action. The selection of actions from the agent's action space determines the sequence of production orders. During sequencing, further restrictions such as employee capacities, different processing times on different machines, and machine conditions (e.g., machine failures) have to be taken into account.

All agents receive a global reward at the end of the episode, determining how successful the past episode was. This approach has proven to be beneficial for the agents' learning behavior. Furthermore, local optima can be avoided while maximizing the reward [12]. Thus, resulting in better handling of the complexity of the simulation model. The reward consists of two parts; one part represents the logistical performance (e.g., lead time), and the other represents the logistical costs (e.g., inventory cost) that arise during production. If the lead time decreases, the logistical performance reward increases. If the inventory costs for an episode decrease, the logistical cost reward increases. A company-specific prioritization of the reward can be achieved by scaling the rewards.

The agents were trained by using PPO. The resulting ANNs were passed to SHAP Deep Explainer. Hereby, an explainer model could be built, allowing the decisions' explainability. This makes it possible to show both the individual decisions of the respective agents and the superordinate factors influencing the agent's decisions (global explainability). Particularly, the findings of global explainability were used to adjust the observation space and thereby improve the reward iteratively.

4.3 Results

In order to evaluate the performance of the developed MARL system, two conventional methods are used for comparison. In many companies, heuristics are common for sequencing orders within production control [33]. A widespread heuristic being used due to its simplicity is FIFO (First in - First out) [34]. The second heuristic used for the evaluation is "shortest set-up time next (SSTN)".

For comparing FIFO, SSTN, and MARL, 52 episodes were simulated. All three approaches used the same initial production program. Figure 3 compares the average total reward and the corresponding components, which consist in this use case of lead time reward and inventory reward. For all episodes, the average total reward of MARL compared to FIFO was 22 % higher. Hereby, improvements were achieved to the same extent through increased logistical performance, represented by the lead time reward (by 24 %) as well as logistical cost, represented by the inventory reward (by 37 %). The average total reward of MARL compared to SSTN was 15 % higher. Thereby, the improvements stem from an improvement of the logistical cost (by 40 %), as well as improvements of the logistical performance (4 %). Local explainability is primarily intended for realizing user-centered information visualization within specific decision situations (e.g., selecting the following order). By identifying the importance of different features with the help of global explainability, the observation space can be adjusted. Thus, the MAS's resulting reward can be increased.

MARL was able to increase the reward gained for production time as well as capital commitment costs compared to FIFO and SSTN, resulting in a higher overall reward. One challenge in optimizing logistic targets is to improve several metrics simultaneously. This arises because different objectives (e.g., short lead times, low inventory costs) interfere with each other. Optimizing those multiple objectives is complex. However, the developed approach, based on MARL, showed promising results of achieving a multi-objective optimization.

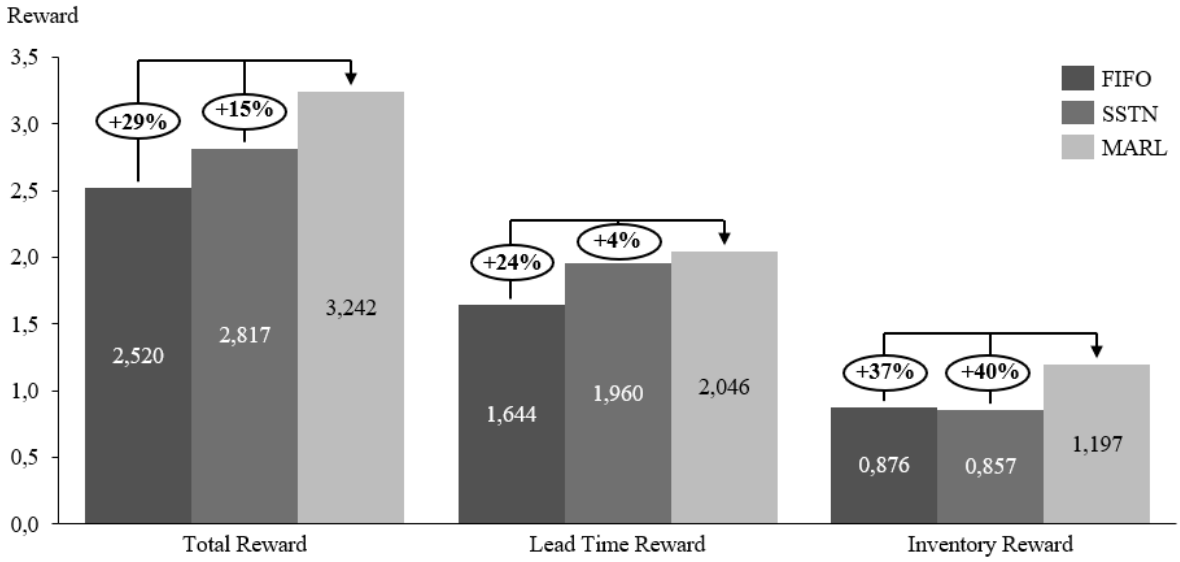

Figure 3: Comparison of the resulting rewards

5. Summary

With the help of the proposed approach, a production control based on MARL with explainable decisions can be realized. Deducted from different user roles, a MAS has been set up. With this work, it could be shown that a MAS with DRL offers the possibility to improve production control with regard to the defined reward. Combining a MAS, DRL, and XAI can improve decision quality and reaction time while also explaining the decisions being made. The focus on user-centricity is an essential component for the

applicability. The current state of the XAI component indicates that it can enable increased trust in the AI system. Further investigations, especially concerning the use of local explainability, are necessary. Furthermore, the behavior of different DRL algorithms, besides PPO, will be tested. For further validation, the number of machines and products—and therefore the complexity of the production system— will be increased.

Acknowledgements

The project REIF – Resource-efficient, Economic, and Intelligent Food chain is funded by the German federal Ministry for Economic Affairs and Climate Action (BMWK).

References

[1] Reinhart, G., Zühlke, D., 2017. Von CIM zu Industrie 4.0, in: Reinhart, G. (Ed.), Handbuch Industrie 4.0. Carl Hanser Verlag GmbH & Co. KG, München, pp. XXXI–XXXIV.

[2] Schuh, G., Reinhart, G., Prote, J.-P., Sauermann, F., Horsthofer, J., Oppolzer, F., Knoll, D., 2019. Data Mining Definitions and Applications for the Management of Production Complexity. Procedia CIRP 81, 874–879.

[3] Cheng, Y., Chen, K., Sun, H., Zhang, Y., Tao, F., 2018. Data and knowledge mining with big data towards smart production. Journal of Industrial Information Integration 9, 1–13.

[4] Engelhardt, P.R., 2015. System für die RFID-gestützte situationsbasierte Produktionssteuerung in der auftragsbezogenen Fertigung und Montage. Diss. Techn. Univ. München, 2015. Utz, München.

[5] Blunck, H., Windt, K., 2013. Komplexität schafft Spielraum für Selbststeuerung. wt-online 2-2013 2, 109–113.

[6] Kletti, J., 2015. MES - Manufacturing Execution System. Springer Berlin Heidelberg, Berlin, Heidelberg.

[7] Lödding, H., Mundt, C., Winter, M., Heuer, T., Hübner, M., Seitz, M., Schmidhuber, M., Maibaum, J., Bank, L., Roth, S., Scherwitz, P., Theumer, P., 2020. PPS-Report 2019. TEWISS Verlag.

[8] Scherwitz, P., Bank, L., Roth, S., Theumer, P., Mundt, C., Winter, M., Heuer, T., Hübner, M., Seitz, M., Schmidhuber, M., Maibaum, J., 2020. Digitale Transformation in der Produktionsplanung und -steuerung. VT (4), 252–256.

[9] Usuga Cadavid, J.P., Lamouri, S., Grabot, B., Pellerin, R., Fortin, A., 2020. Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0. Journal of Intelligent Manufacturing 31 (6), 1531–1558.

[10] Dittrich, M.-A., Fohlmeister, S., 2020. Cooperative multi-agent system for production control using reinforcement learning. CIRP Annals 69 (1), 389–392.

[11] Altenmüller, T., Stüker, T., Waschneck, B., Kuhnle, A., Lanza, G., 2020. Reinforcement learning for an intelligent and autonomous production control of complex job-shops under time constraints. Prod. Eng. Res. Devel. 14 (3), 319–328.

[12] Waschneck, B., 2020. Autonome Entscheidungsfindung in der Produktionssteuerung komplexer Werkstattfertigungen. Diss. Universität Stuttgart, 2020.

[13] Sutton, R.S., Barto, A.G., Barto, A., 2018. Reinforcement Learning: An introduction, 2nd ed. The MIT Press, Cambridge, MA, London, 526 pp.

[14] Burkart, N., Huber, M.F., 2021. A Survey on the Explainability of Supervised Machine Learning. jair 70, 245–317.

[15] Schuh, G., Stich, V. (Eds.), 2012. Produktionsplanung und -steuerung. 1: Grundlagen der PPS, 4th ed. Springer Vieweg, Berlin Heidelberg.

[16] Wiendahl, H.-P., 2014. Betriebsorganisation für Ingenieure, 8th ed. Hanser, München.

[17] François-Lavet, V., Henderson, P., Islam, R., Bellemare, M.G., Pineau, J., 2018. An Introduction to Deep Reinforcement Learning. FNT in Machine Learning 11 (3-4), 219–354.

[18] Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D., 2016. Mastering the game of Go with deep neural networks and tree search. Nature 529 (7587), 484–489.

[19] VDI/VDE Richtlinie 2653-1, 2018. Agentensysteme in der Automatisierungstechnik: Grundlagen. Beuth, Berlin, 24 pp.

[20] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., 2013. Playing Atari with Deep Reinforcement Learning, 9 pp. http://arxiv.org/pdf/1312.5602v1.

[21] Roesch, M., Linder, C., Zimmermann, R., Rudolf, A., Hohmann, A., Reinhart, G., 2020. Smart Grid for Industry Using Multi-Agent Reinforcement Learning. Applied Sciences 10 (19), 6900.

[22] Murdoch, W.J., Singh, C., Kumbier, K., Abbasi-Asl, R., Yu, B., 2019. Definitions, methods, and applications in interpretable machine learning. Proceedings of the National Academy of Sciences of the United States of America 116 (44), 22071–22080.

[23] Holzinger, A., 2018. From Machine Learning to Explainable AI, in: 2018 World Symposium on Digital Intelligence for Systems and Machines (DISA), Kosice. IEEE, pp. 55–66.

[24] Gerlings, J., Shollo, A., Constantiou, I., 2021. Reviewing the Need for Explainable Artificial Intelligence (xAI), in: Proceedings of the 54th Hawaii International Conference on System Sciences.

[25] Ribeiro, M.T., Singh, S., Guestrin, C., 2016. Why Should I Trust You?, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco California USA. ACM, New York, NY, USA, pp. 1135–1144.

[26] Shapley, L.S., 1953. 17. A value for n-Person Games, in: Arrow, K., Gale, D., Kuhn, H.W., Tucker, A.W. (Eds.), Contributions to the Theory of Games (AM-28), Volume II. Princeton University Press, Princeton, pp. 307–318.

[27] Lundberg, S., Lee, S.-I., 2017. A Unified Approach to Interpreting Model Predictions. http://arxiv.org/pdf/1705.07874v2.

[28] Rehse, J.-R., Mehdiyev, N., Fettke, P., 2019. Towards Explainable Process Predictions for Industry 4.0 in the DFKI-Smart-Lego-Factory. Künstl Intell 33, 181–187.

[29] Kuhnle, A., May, M.C., Schäfer, L., Lanza, G., 2021. Explainable reinforcement learning in production control of job shop manufacturing system. International Journal of Production Research 24 (4), 1–23.

[30] Huber, T., Weitz, K., André, E., Amir, O., 2021. Local and global explanations of agent behavior: Integrating strategy summaries with saliency maps. Artificial Intelligence 301, 103571.

[31] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal Policy Optimization Algorithms, 12 pp. http://arxiv.org/pdf/1707.06347v2.

[32] Mayer, S., Classen, T., Endisch, C., 2021. Modular production control using deep reinforcement learning: proximal policy optimization. J Intell Manuf 32 (8), 2335–2351.

[33] Mönch, L. (Ed.), 2006. Agentenbasierte Produktionssteuerung komplexer Produktionssysteme, 1st ed. DUV Deutscher Universitäts-Verlag, s.l., 300 pp.

[34] Lödding, H., 2016. Verfahren der Fertigungssteuerung: Grundlagen, Beschreibung, Konfiguration, 3rd ed. Springer Vieweg, Berlin, Heidelberg, 664 pp..

Biography

Philipp Theumer (*1991) holds a Master's degree in Mechanical Engineering and Management from the Technical University of Munich. Since 2018 he is a research assistant at the Fraunhofer IGCV in the department of Production Management.

Florian Edenhofner (*1994) holds a Master's degree in Food Technology and Biotechnology from the Technical University of Munich. In 2021 he wrote his Master's Thesis at the Fraunhofer IGCV in the department of Production Management.

Roland Zimmermann (*1998) holds a B.Sc. in Computer Science from the University of Applied Sciences Augsburg. Since 2021 he has been enrolled at the University Augsburg for a Master's degree in Computer Science and is working at the Fraunhofer IGCV in the department of Production Management.

Alexander Zipfel (*1992) holds a Master's degree in Mechanical Engineering and Management from the Technical University of Munich. He leads the group for Production Planning and Control at the Fraunhofer IGCV in the department of Production Management.