Proceedings of the 2025 Winter Simulation Conference
E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.

OPTIMIZING PRODUCTION PLANNING AND CONTROL: REWARD FUNCTION DESIGN IN REINFORCEMENT LEARNING

Marc Wegmann[1], Benedikt Gruenhag[1], Michael Zaeh[1], and Christina Reuter[1]

[1]Institute for Machine Tools and Industrial Management, TUM School of Engineering and Design, Technical University of Munich, Boltzmannstrasse 15, Garching Near Munich, 85748, GERMANY

ABSTRACT

Production planning and control (PPC) is challenged by the complex and volatile environment manufacturers face. One promising approach in PPC is the application of Reinforcement Learning (RL). In RL, an intelligent agent is trained in a simulation environment based on its experiences. The behavior of the agent is trained by defining a reward function that provides positive feedback if the agent performs well and negative feedback if it does not. Accordingly, the design of the reward function determines the impact RL can have. This article deals with the challenge of how to design a suitable reward function. To do so, 8 design principles and 21 design parameters were identified based on a structured literature review. The principles and parameters were utilized to systematically derive reward function alternatives for a given PPC task. These alternatives were applied to a use case in rough production scheduling being a sub task of PPC.

1 INTRODUCTION

More than ever, the environment producing companies are exposed to can be described as volatile, uncertain, complex, and ambiguous (VUCA) (Heesen 2024). Globally, economic tensions and upcoming tariffs are forcing manufacturers to rethink their supply chains and production networks. The wars in the Ukraine and in the Middle East cause uncertainty and disrupt supply chains regularly. Additionally, increased and unpredictable customer requirements demand more complex production systems that can adapt to the needs, as the automotive industry currently demonstrates in their transformation process from combustion engines to electrified vehicles. Moreover, some important industries naturally have complex production systems, such as the semiconductor industry or participants in the circular economy. Accordingly, manufacturers must make complex planning decisions with growing frequency. Manufacturing companies must find answers in this tense situation to maintain market competitiveness and guarantee economic sustainability.

One of the key enablers to reach a flexible and adaptable production system that can handle the VUCA world is a suitable production planning and control (PPC) (Esteso et al. 2023). PPC encompasses a range of strategic, tactical, and operational activities essential for managing production environments. Strategic PPC activities include tasks such as sales planning and the configuration of the production network (Schuh et al. 2012). They have a long-term planning horizon. Tactical activities include, e.g., process planning or production system planning and have a mid-term planning horizon (Schuh et al. 2012). Operational activities focus on short-term planning horizons, addressing specific customer orders, like order management and order dispatching (Schuh et al. 2012). PPC generally aims to ensure optimal capacity utilization with high adherence to delivery dates, minimal tied capital, and low procurement costs (Kellner et al. 2022).

The challenges of PPC in a dynamic world lie in carrying out the tasks fast, adaptively, and with good results (Wegmann and Zaeh 2023). To do so, the application of Reinforcement Learning (RL) from the field of Artificial Intelligence shows promising results in several industries (e.g., Gros et al. (2020) in the automotive industry and Wegmann et al. (2025) in the special machinery industry). In the context of this publication, RL can be described as an autonomous agent interacting with a simulation environment to learn a specified behavior (van Otterlo and Wiering 2012). The agent perceives the current state of the

environment and performs actions according to a so-called policy (Sutton and Barto 2018). The process of identifying an optimized policy for a given problem involves the agent receiving rewards or penalties based on its actions (Sutton and Barto 2018). In PPC, an agent is, for example, rewarded for low inventories, short throughput times, and a high delivery reliability. The rewards and penalties in the feedback loops are defined within the reward function (Sutton and Barto 2018). The reward function plays an important role in RL, as it crucially affects the effectiveness and efficiency of the overall approach. However, for a specific task with a specific goal, there is an infinite number of possibilities to construct the reward function. The function can incorporate and combine multiple KPIs and it can be mathematically formulated in accordance with the designer's preferences. To enable adaptive and flexible production systems that meet the challenges of the VUCA world, this publication focuses on the application of RL in PPC and specifically focuses on the design process of the corresponding reward function. This publication accordingly states the central research question: **How can the process of reward function design for RL in PPC be structured and methodologically supported?**

2 STATE OF THE RESEARCH

2.1 Challenges in Reward Function Design

The research question can be broken down into four challenges (CH) in the design of a reward function. First, it is unclear which **principles** must be taken into account in reward function design, i.e., which decisions must be addressed (**CH1**). While it seems straightforward that the architect of a reward function has to decide which KPIs to incorporate in the function, other decisions also play an important role, e.g., the mathematical formulation or how to deal with unauthorized actions of the agent. Second, it is unclear how the decision space for these principles can be structured (**CH2**), i.e., which **parameters** a certain principle consists of. For the mathematical formulation, for example, multiple options exist, such as using a discrete or a continuous function design. Also, the concrete expression of it must be defined, e.g., as a linear function compared to an exponential one. Third, it is unclear how the structure of design principles and parameters can be translated into concrete reward functions. Accordingly, the design phase must be **methodologically supported** (**CH3**) (Jaensch et al. 2022). The solution space resulting from CH1 and CH2 can become quite large and practitioners as well as scientists need a procedure to navigate through it in a structured way. Last, it is unclear how different reward function design alternatives **impact** the PPC task's performance (**CH4**). On the one hand, an analysis should be carried out to derive generally valid statements on the design procedure. Initial studies, for example, already showed that exponential reward function definitions could be superior when gradient-based RL optimization algorithms are applied (Kuhnle 2020). On the other hand, the structured identification and evaluation of possible reward functions for a certain use case should lead to the identification of a reward function that provides good results.

2.2 Related Work

A structured literature review was carried out to identify relevant publications in the field of reward function design for RL in PPC. The procedure of the literature search is based on the methodology presented by **?** for the systematic identification and classification of relevant literature. After defining the search framework, literature sources were selected, which are supplemented by further works using backward and forward searches (**?**). The search was conducted in Scopus in April 2025 and combined the domains of PPC (by using the keywords "production", "manufacturing", and "assembly"), RL (by using the keyword "Reinforcement Learning"), and reward function design (by using the keywords "reward" and "design"). The results were restricted to the subject areas of computer science and engineering. Only publications in English language and with open access availability were observed. Additionally, the results were restricted to the years 2015 to 2025. The search resulted in 357 documents overall for which a title and abstract screening was performed. Publications that could be assigned to the domains mentioned above (PPC, RL, and reward function design) after the first scan were included. Explicitly, publications that focused on application fields

outside of production (e.g., control of smart energy grids), outside of PPC (e.g., application in robotics or computer-aided design), and on the elaboration and comparison of specific optimization algorithms (e.g., advancements in Q-learning), were excluded. Additionally, articles that were not published yet and still in press were excluded. This first screening procedure resulted in 63 relevant publications for which a full paper analysis was performed. Five of the 63 publications were not accessible in full. If a publication addressed at least one of the four challenges mentioned above, it was included. Otherwise it was excluded. This resulted in 12 relevant publications. They are briefly described below and characterized according to the challenges CH1 to CH4.

Kuhnle et al. (2019) stated generally applicable modeling guidelines for reward function design, e.g., that multiple targets can be integrated by linear combinations (CH3). Ou et al. (2019) evaluated the impact of five different reward functions based on a simulation study (CH4). They mainly focused on different KPIs to be incorporated, e.g., the end-of-line output and the machines' production efficiency. Hillebrand et al. (2020) highlighted the importance of a reward function design methodology and roughly presented aspects to be considered. These aspects include, e.g., the sparsity of the reward (CH1). Jeong et al. (2021) utilized RL in material handling task assignment and route planning. They roughly considered different reward principles (CH1) but did not specify their parameters or investigate their influences on the overall performances of the tasks. Kuhnle et al. (2021) applied RL for the adaptive control of production systems. In their reward module, multiple possible reward functions were defined and they were applied to a use case in the semiconductor industry (CH4). The authors mainly distinguished between the definitions of sparse and dense reward functions and provided some examples (CH1 and CH2). Lamprecht et al. (2021) optimized maintenance scheduling by applying an RL approach. They defined a trivial and a more sophisticated reward function and compared them experimentally (CH4). Zhou et al. (2021) investigated the influence of different reward function designs on the performance of a production scheduling system. Specifically, the authors defined four KPIs to be optimized based on experiments: makespan, utilization, energy consumption, and balanced workloads. However, while a first glimpse on the design principles (CH1) was provided, other principles and their parameters were not investigated. Jaensch et al. (2022) addressed the challenge that reward functions are often defined from scratch without a standardized procedure. They proposed a test-driven development of RL, and reward functions specifically. The central idea of this approach is to validate reward function designs in small iterations by successfully applying test cases (CH3 and CH4). Rinciog and Meyer (2022) proposed a framework for standardizing RL approaches. The authors elaborated on some design principles and their parameters (CH1 and CH2), e.g., on the timing of reward distribution and on the shape of the function itself. Tang et al. (2023) proposed reward function categories in their approach to support the design process. The authors mapped certain scenarios, e.g., the fulfillment of an order, to fuzzy reward formulations, e.g., a "huge positive" reward (CH3). The fuzzy formulations were then translated to a discrete reward function design. Panzer et al. (2024) investigated the application of RL to select appropriate heuristics in production control. They discussed some aspects in reward function design that could affect the performance of the production control tasks, e.g., the normalization of rewards (CH1). Additionally, they provided some methodological insights on how to shape the reward function, e.g., by conducting a sensitivity analysis (CH3). Serrano-Ruiz et al. (2024) applied RL to a job shop scheduling problem. The authors discussed how design decisions regarding the reward function could influence the RL agent's behavior, e.g., the combination of multiple objectives or the importance of local rewards (CH1 and CH3). The evaluation of the publications in regard to the fulfillment of CH1 to CH4 is summarized in Figure 1.

2.3 Need for Action

Regarding CH1, the publications mentioned above show that several works already emphasize the importance of principles for reward function design. However, they are only partially discussed and are never summarized holistically or in a structured way. Most of the works focus on reducing the sparsity of the rewards, aiming for a dense reward function. Also, the next challenge, detailing what the solution space of these principles

Publication	CH1: principles in reward function design	CH2: parameter space of these principles	CH3: methodological support	CH4: impact on PPC task performance
Kuhnle et. al (2019)	○	○	◑	○
Ou et al. (2019)	○	○	○	●
Hillebrand et al. (2020)	◑	○	○	○
Jeong et al. (2021)	◑	○	○	○
Kuhnle et. al (2021)	◑	◑	○	●
Lamprecht et al. (2021)	○	○	○	◑
Zhou et. al (2021)	◑	○	○	●
Jaensch et al. (2022)	○	○	●	◑
Rinciog & Meyer (2022)	◑	◑	○	○
Tang et al. (2023)	○	○	◑	○
Panzer et al. (2024)	◑	○	◑	○
Serrano-Ruiz et al. (2024)	◑	○	○	○

○ Not fulfilled ◑ Partially fulfilled ● Completely fulfilled

Figure 1: Fulfillment of the challenges CH1 to CH4 in current literature.

looks like (design parameters, CH2), is rarely elaborated. Only one publication exists that provides a methodology to design a good reward function (CH3). However, Jaensch et al. (2022) do not incorporate design elements and their ranges specifically. Some researchers already conducted experimental comparative studies (CH4), but never in combination with a structured approach resulting from CH1 to CH3. To close the research gap, this publication aims to adress CH1 to CH4 holistically.

3 REWARD FUNCTION DESIGN

3.1 Scientific Approach for Reward Function Design

To meet the challenges and the need for action mentioned above, this publication pursues a five-step research approach of a morphological method (Zwicky 1967). The approach aims at the identification and investigation of the total set of possible configurations that solve a given problem statement. This set of configurations is referred to as a morphological box. Accordingly, the morphological box in this publication identifies possible configurations in reward function design and supports in identifying good reward functions. The approach of Zwicky (1967) starts with the formulation of the problem under consideration, which is the definition of possible reward function designs in RL in PPC and the selection of good ones among them. The second step lies in the identification and definition of the dimensions and corresponding value ranges of the problem. The dimensions are referred to as design principles and the value ranges as design parameters in this publication. For the identification of these, a literature analysis according to Blessing and Chakrabarti (2009) was carried out using similar key words and selection criteria as in Section 2.2. For each publication, the design principles and parameters were extracted. In the third step, the morphological box was constructed, consolidating all of the design principles and parameters from the individual contributions and, thus, summarizing all the potential design approaches in reward function design. In the fourth step, all of the solutions, i.e., design principles and parameters, which are contained in the morphological box, were closely analyzed and evaluated regarding the impact on the initial problem statement. In step 5, the possible solutions were selected and operationalized. To do so, an application guide was constructed based on Design of Experiment (DoE) approaches. (Zwicky 1967) The corresponding key result (R), namely the morphological box (R1) and the application guide (R2), are described in the following section.

3.2 R1: Morphological Box for Reward Function Design in PPC

The first scientific result is represented by the morphological box for reward function design in PPC based on step 1 to step 4. As mentioned, the morphological box is structured into design principles and design parameters. A design principle corresponds to the element of a reward function that can be varied in the process of constructing the reward function. One example is the timing of the reward distribution. A design parameter corresponds to the expression of a certain design principle. The timing of the reward distribution can, for example, be dense (reward is provided after each action of the agent), sparse (reward is provided after multiple actions or at the end of a training episode), or a combination of both. Table 1 gives an overview of the results. The classification was carried out based on the work of Janssen and Gray (2012) and extended by the findings of the literature search. For each design principle and parameter, exemplary references are provided. The design principles and corresponding parameters are described in detail in the following. The design principles are highlighted in **bold** and the design parameters in *italics.*

Table 1: Morphological box for reward function design in PPC.

Principle (P)	Parameter	Exemplary References
P1 Logistical KPIs	Order-related	(Liu et al. 2020), (Park et al. 2020), (Nasuta et al. 2024)
	Resource-related	(Waschneck et al. 2018), (Nasuta et al. 2024)
	Global	(Ou et al. 2019), (Liu et al. 2020), (Samsonov et al. 2022)
	Combined	(Zhou et al. 2021), (Kuhnle et al. 2022), (Valet et al. 2022)
P2 Timing	Dense	(Samsonov et al. 2022), (Valet et al. 2022), (Nasuta et al. 2024)
	Sparse	(Kuhnle et al. 2021), (Samsonov et al. 2022), (Nasuta et al. 2024)
	Combined	(Lang et al. 2020), (Liu et al. 2020)
P3 Function definition	Continuous	(Zhou et al. 2021), (Samsonov et al. 2022), (Nasuta et al. 2024)
	Categorical	(Kuhnle et al. 2022), (Samsonov et al. 2022), (Valet et al. 2022)
P4 Function progress	Exponential	(Kuhnle et al. 2022), (Samsonov et al. 2022), (Nasuta et al. 2024)
	Polynomial	(Park et al. 2020), (Samsonov et al. 2022), (Valet et al. 2022)
P5 Value range	Positive	(Kuhnle et al. 2021), (Zhou et al. 2021), (Kuhnle et al. 2022)
	Negative	(Park et al. 2020)
	Combined	(Samsonov et al. 2022), (Valet et al. 2022), (Nasuta et al. 2024)
P6 Normalization	Yes	(Kuhnle et al. 2021), (Zhou et al. 2021), (Kuhnle et al. 2022)
	No	(Park et al. 2020), (Samsonov et al. 2022), (Valet et al. 2022)
P7 Dealing with illegal actions	Penalization	(Kuhnle et al. 2021), (Kuhnle et al. 2022), (Valet et al. 2022)
	Restricted action space	(Lang et al. 2020), (Liu et al. 2020), (Park et al. 2020)
	Action Masking	(Samsonov et al. 2022), (Nasuta et al. 2024)
P8 Penalization of targets not achieved	Yes	(Kardos et al. 2021), (Lang et al. 2020), (Valet et al. 2022)
	No	(Ou et al. 2019), (Park et al. 2020), (Samsonov et al. 2022)

The first design principle is the selection of the relevant **logistical KPIs** to be incorporated into the reward function. Logistical KPIs in the context of PPC are diverse. In this publication, they are categorized into order-related, resource-related, global, and aggregated KPIs. An *order-related* KPI can be assigned to a certain customer order and is, for example, the throughput time. A *resource-related* KPI can be assigned to a certain production resource and is, for example, the utilization. A *global* KPI targets the overall production system and is, for example, the delivery reliability. Finally, different KPIs can also be *combined*. The second principle is the **timing** of the reward distribution. A *dense* reward is provided after every action or time step. A *sparse* reward is only provided after each episode (Rinciog and Meyer 2022). Some works utilize a *combination* of the two approaches, e.g., Lang et al. (2020) or Liu et al. (2020). The **function definition** represents the third principle. A function can either be defined to be *continuous* or *categorical*. Similarly, the **function progress** is either *exponential* or *polynomial.* The next design principle deals with the **value range** of the reward function. A reward function is defined in a *positive* value range (rewards are larger than 0), in a *negative* value range (rewards are smaller than 0), or covers

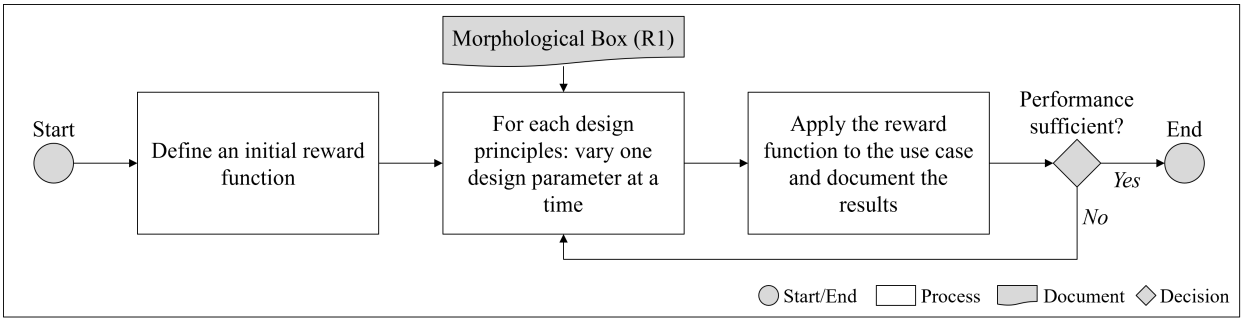

Figure 2: Flow chart of the application guide.

both ranges in a *combined* approach. Some approaches utilize **normalization**, which represents another design principle. Zhou et al. (2021) and Kuhnle et al. (2022) are two examples for this. This principle can either be *applied* or *not*. How to **deal with illegal actions** represents an additional design principle. Utilizing the reward function to restrict an illegal behavior of the agent results in a *penalization* of the corresponding actions. Other approaches either directly *restrict the action space* to only legal actions or apply *action masking* methods. The last design principle deals with the **penalization of targets that are not achieved**. Lang et al. (2020) and Valet et al. (2022) are two examples for that. Again, this principle can either be *applied* or *not*.

3.3 R2: Application Guide for Reward Function Design in PPC

After the morphological box was presented in Section 3.2, this section deals with the application of it to support the process of reward function design, which refers to the fifth step of the morphological method. The morphological box opens up a parameter space. From a user's perspective, for each design principle, one design parameter can be selected. The combination of all design parameters results in a concrete reward function. However, there are some adjustments that are not covered in the morphological box. For example, a linear reward function can have different characteristics with unlimited possibilities. Thus, the application guide must combine a methodological approach with an experimental one, as most approaches in the fields of Machine Learning and Artificial Intelligence do. To do so, methods from the DoE are applied to the morphological box. The morphological box consists of 8 design principles with overall 21 design parameters. A full factorial experiment design would lead to 1728 possible combinations, not including the integration of different characteristics, e.g., of different linear functions. Thus, an experiment design that varies one factor at a time (OFAT) is suggested in this publication according to (Tanco et al. 2009). Additionally, the idea of reward shaping is incorporated. Reward shaping refers to adjusting the reward function with the help of domain knowledge by adding additional, small rewards. The resulting guide is summarized as a flow chart in Figure 2 and described in the following.

To begin the procedure, an initial reward function must be defined as a starting point. This reward function is iteratively adapted according to the morphological box (R1). When applying the idea of varying OFAT, only one design principle from R1 is taken into consideration and the corresponding design parameters are adjusted, leading to a new reward function. While the morphological box provides guidelines and structures the solution space in reward function design, the configuration of a concrete reward function is up to the user. For example, the user must decide which concrete polynomial function to select. The above-mentioned idea of reward shaping can also be incorporated. In the next step, the constructed reward function is applied to the use case under consideration and the results are documented. If the application is successful and the performance of the RL approach is sufficiently good, the procedure of reward function design can be finished. Otherwise, it is iteratively optimized according to the morphological box.

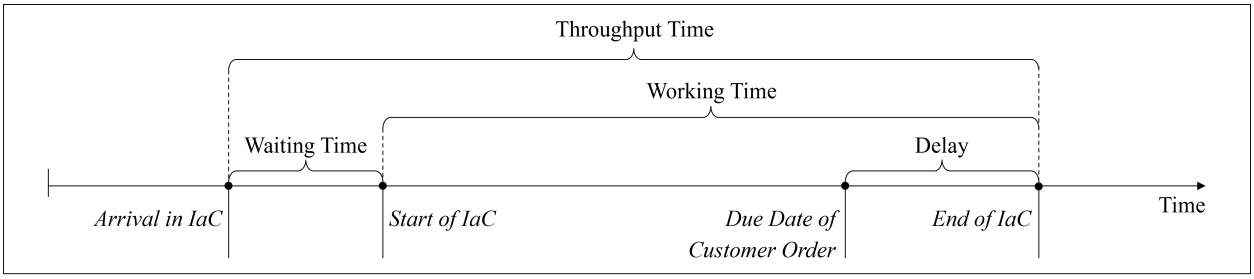

Figure 3: Variables in the IaC process of the use case.

4 APPLICATION TO A USE CASE IN ROUGH SCHEDULING

4.1 Description of the Use Case

To demonstrate and evaluate the scientific results R1 and R2, they were applied to a use case. The use case lies in rough production scheduling as part of order management in a company in the special machinery industry. In order management, the machines must be scheduled efficiently through the steps of product design and adjustment, manufacturing and procurement, assembly, as well as the initial installation and commissioning (IaC) of the machines. The last step, IaC, is the focus of this application. In IaC, the machines are placed on the shop floor and block a production area for the whole IaC process. The limited availability of floor space during IaC leads to a scheduling problem. The relevant variables in the context of IaC are described and summarized in Figure 3. Due to varying processing times in the previous steps, the individual *arrival times* of customer orders in IaC are assumed to be stochastic. When arriving, an area on the shop floor as well as the *start* and the *end date* must be defined. Some time can lie in between the arrival of the machine and the start of IaC, which is called the *waiting time*. The overall processing time between the start and end in IaC is the *working time*. Adding the initially described waiting time to that results in the *throughput time*. If the *customer due date* lies before the end of IaC, that would result in a *delay*. The production areas in IaC are assumed to be simplified as chessboard areas. They are divided into modules, which are again divided into fields. In the use case, the available floor space consists of six modules in total, with four fields each. The customer orders to be scheduled on the floor space have an area demand of 1, 2, or 4 fields, as well as individual working times. Additionally, customer orders have an orientation that determines whether an order with an area demand of 2 fields occupies the module horizontally (H) or vertically (V). Figure 4 shows an example of the production areas for 5 modules with 4 fields each and scheduled orders with different floor space requirements and orientations.

4.2 Description of the RL approach

In this section, the RL approach to support decision-making in rough scheduling of orders in IaC is described. This section especially focuses on the modeling of the action space, the state space, and the agent with the optimization algorithm. The reward function is constructed afterward and evaluated utilizing R1 and R2.

The **action space** represents the agent's room for maneuvers. The approach chosen runs through the list of orders to be scheduled in ascending order and assigns each order to necessary fields within a module and a start date within the planning horizon. Some actions are specifically defined as forbidden: the order is scheduled before its arrival date in commissioning; the order can no longer be processed within the planning horizon; the order is assigned to a field that is already occupied by another order during the processing time. If the agent selects a forbidden action, there is no scheduling and no updating of the state space. Instead, the agent must select a different action for this order. If necessary, the agent receives a penalty for selecting a forbidden action, which is defined later in the approach. The **state space** provides the agent with information about the environment and the consequences of its action. According to Samsonov et al. (2022), it makes sense to use variables in the state space that are related to the indicators in the reward

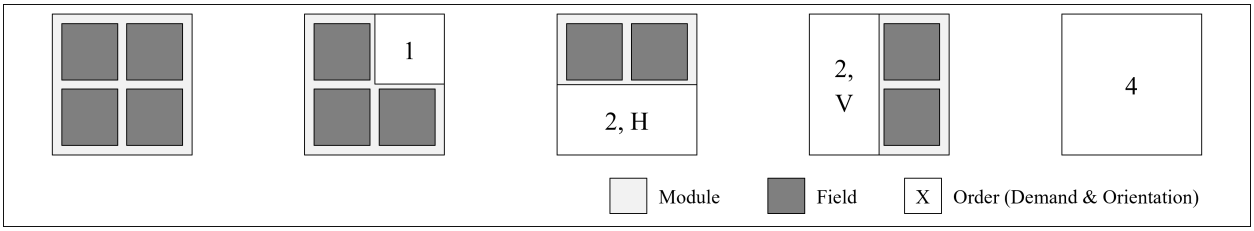

Figure 4: Scheduling problem in IaC modeled as chess board fields according to the production area supply and demand.

function. Since multiple reward functions with different indicators were constructed and tested (Section 4.3), the state space is therefore adjusted slightly depending on the target variables in the reward function. Some basic variables are part of the state space for all experiments and reward functions. These are the *occupancy* of each field and module at each point in time within the planning horizon, the *remaining number of orders*, and the *specifications* of all remaining orders in one episode such as arrival dates, processing times, space requirements, and orientation requirements. Reward functions that only incorporate the waiting time are additionally assigned the state variables *mean waiting time of already scheduled orders* and the *waiting time of the last scheduled order*. Reward functions that only incorporate the delay or on-time delivery are assigned the state variables *delay of the last scheduled order* and *due date of the next order to be scheduled*. Reward functions that combine the two optimization goals are assigned all of the state variables from above combined. The individual components of the state space are scaled to the interval [0, 1] using the min-max normalization function (García et al. 2015). The **optimization algorithm** used in this approach was a Proximal Policy Optimization (PPO) algorithm implemented in the Python library *Stable Baselines3*. For the hyper parameters, the mini batch size was set to 32 and the number of epochs when optimizing the loss was set to 3. The other hyper parameters were set to a default value. Each training episode consisted of 30 subsequent orders that were randomly drawn from a data base of 10,971 orders. To evaluate the performance of the system, five benchmark data sets from different time horizons in the overall data set were extracted. Each of them consisted of 30 consecutive orders that were placed between January 2014 and March 2021. Thus, various scenarios of high and low production volumes as well as of different product variants during that time interval were covered in the evaluation process. For each of the benchmark data sets, 10 schedules were planned by the trained RL agent to level for stochastic effects in choosing an action. Consequently, 50 schedules were created by each RL model that served as a basis to evaluate the key performance indicators. From an application perspective, the overall goal was to **minimize the waiting time** which directly affects the overall throughput time. Additionally, the scheduling aimed to **maximize the on-time delivery rate**.

4.3 Application of the Methodology

According to the application guide R2, an initial reward function was chosen to serve as a basis for the upcoming methodological steps:

$$r_1 = \begin{cases} 1, & \text{if waiting time} = 0 \\ 0.5, & \text{if } 0 < \text{waiting time} <= 0.3 \\ 0, & \text{if waiting time} > 0.3 \end{cases}$$

According to the morphological box R1, it only incorporates the waiting time as a logistical KPI (P1). The timing can be defined as dense, since the reward is distributed for each scheduled order (P2). The function is defined categorical (P3), making a distinction into a polynomial or a exponential function obsolete (P4). The value range is positive (P5) and the function is normalized (P6). Last, illegal actions (P7) or missing a target KPI (P8) are not penalized. From hereon, the reward function was adjusted utilizing the

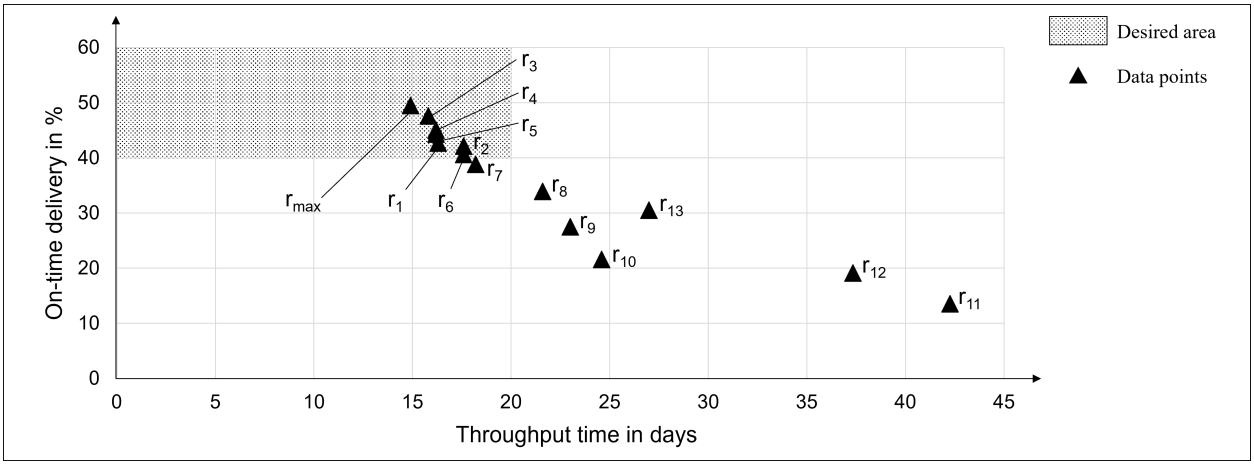

Figure 5: Excerpt from the results of the application of different reward functions to the use case.

design principles and parameters from R1 step by step. To keep the example easy to understand, the first two iterations are described in detail. These addressed the principles of function definition and progress (P3 and P4) as well as the timing of the reward distribution (P2). The other principles and parameter variations are summarized afterward and the final results are presented.

Function definition and progress. First, the design principles **P3** and **P4** were addressed by introducing a linear function (r_2), a squared function (r_3), and an exponential function (r_4). They are specifically defined as:

$$r_2 = 1 - \text{waiting time}$$
$$r_3 = (1 - \text{waiting time})^2$$
$$r_4 = -1 + e^{\ln(2)\cdot(1-\text{waiting time})}$$

According to the idea of OFAT in R2 and the morphological box in R1, the logistical KPI to be considered is still just the waiting time (P1), the timing remains dense (P2), the value range of the reward remains positive as the waiting time ranges from 0 to 1 (P5), the values remain normalized (P6), and no penalization is carried out (P7 and P8). The application of these reward functions to the use case showed that r_3 provided the best results compared to r_1, r_2, and r_4. r_3 resulted in a mean on-time delivery of around 48 % and a mean waiting time of 16 days. Accordingly, r_3 serves as a candidate for further optimization.

Timing. In the next iteration, **P2** was varied. Since the candidate r_3 from before is defined as a dense reward function, sparse reward elements are integrated into the function that only reward the agent at the end of an episode. The sparse reward element uses the average waiting time in the created schedule as a logistic KPI and is integrated into the function with different weights. The sparse element and the resulting reward functions are defined as:

$$r_{sparse} = (1 - \text{mean waiting time})^2$$
$$r_5 = 0.5 * r_3 + 0.5 * r_{sparse}$$
$$r_6 = 0.25 * r_3 + 0.75 * r_{sparse}$$
$$r_7 = 0.75 * r_3 + 0.25 * r_{sparse}$$

The waiting time is still the only KPI to be incorporated (P1), the function definition remains squared (P3 and P4), the values of the reward function are still positive (P5) and are normalized (P6). Also, no penalization is taking place (P7 and P8). The application to the use case showed that the introduction of the

sparse rewards did not led to an improvement of the overall performance. All three new reward functions led to worse results compared to r_3.

Following the same procedure, also the influence of the value range (P5) and thus normalization (P6) as well as the penalization of illegal actions was observed in experiments (P7). An overall target dimension to be achieved could not be provided in the use case and was, thus, disregarded (P8). The same structure was then first applied to reward functions that incorporate the on-time delivery in the reward function and second to reward functions that combine the waiting time and the on-time delivery in the reward function (P1). Running through the methodology and carrying out all the experiments resulted in 117 reward functions overall that were applied to the use case. An excerpt of the results is presented in Figure 5.

Different findings can be derived from the results. First, it can be seen that the function definition has only a minor influence on the overall performance of the agent, as r_1, r_2, r_3, and r_4 are very close together. The introduction of a sparse element did not lead to better results, as r_5, r_6, and r_7 show. All of them perform worse than their solely dense variant r_3. Additional experiments were carried out with reward functions that had no dense component at all (r_8, r_9, and r_{10}) leading to even worse results. Furthermore, the introduction of strong penalties on wrong actions did not improve the quality of the results but affected them strongly negative, as r_{11} and r_{12} show. Finally, the value range of the reward function was also varied. Small variations did not affect the performance while strong increases (from a value range of [0,1] to [0,16] of the reward function in r_{13}) also led to worse results. Overall, the best value for the KPIs waiting time and on-time delivery is provided by the cubical defined reward function without punishment of impermissible actions $r_{\mathbf{m}ax} = (1 - \textbf{mean waiting time})^3$**.**

Figure 5 illustrates that in the course of the iterative process, reward functions were created that lead to an improvement in the logistical KPIs. The difference between good and bad reward function designs is shown by a reduction in waiting time from 42.3 to 12.8 days and by an increase in delivery reliability from 13.5 % to 49.5 % when comparing the $r_{\mathrm{m}ax}$ to the worst reward functions in the exemplary experiments of Figure 5. When being compared to the initial reward function r_1, $r_{\mathrm{m}ax}$ still shows a reduction in waiting time from 16.3 to 12.8 days and an increase in on-time delivery from 42.7 % to 49.5 %. These results emphasizes the importance of a methodological approach to reward function design.

5 SUMMARY, DISCUSSION, AND OUTLOOK

This publication aimed to support the design of reward functions in RL approaches applied to PPC tasks. By identifying and structuring design principles and corresponding parameters within a morphological box, challenge (CH) 1 and CH2 were successfully fulfilled. Additionally, a methodology to apply the morphological box in the form of a user guide was provided, addressing CH3. Various reward functions resulting from the approach were then applied to a use case, thereby fulfilling CH4. However, some limitations were identified. First, the application was limited to a single use case in rough production scheduling. Other PPC tasks, e.g., order dispatching, with even more complex scenarios, should also be considered in future works. The use of OFAT as the experimental design does not necessarily yield fast and optimal results and could be improved by employing more advanced approaches, such as factorial designs according to Fisher or Taguchi (Tanco et al. 2009). Furthermore, the design principles and parameters in the morphological box were based solely on a literature review, which means that innovative design choices not published were not incorporated. Thus, expert knowledge could enhance the results in future works. Expert knowledge also remains essential when applying the approach, e.g., for deriving specific exponential functions or defining the penalization of false actions. Additionally, the interplay of reward function design with other RL design elements (action space, state space, and the optimization algorithm) was not addressed. As the effectiveness of the optimization algorithm can be affected by the function progress (Kuhnle 2020), for example, these intercorrelations should be investigated. Moreover, the interplay of these design elements with a validated simulation environment should be further examined, as proposed by the works of Belsare et al. (2022) and Hua et al. (2022). Finally, the experimental approach involved high computational costs and significant manual effort. These aspects scale with the complexity of the use

case. A sensitivity analysis could be carried out to identify the parameters that have the greatest impact on the performance of the approach. Variation of these parameters should then be the focus of the analysis. By incorporating these future research directions, RL in PPC can be taken to the next level and could potentially lead to more robust and resilient manufacturing organizations.

REFERENCES

Belsare, S., E. D. Badilla, and M. Dehghanimohammadabadi. 2022. "Reinforcement Learning with Discrete Event Simulation: The Premise, Reality, and Promise". In *2022 Winter Simulation Conference (WSC)*, 2724–2735 https://doi.org/10.1109/WSC57314.2022.10015503.

Blessing, L. T., and A. Chakrabarti. 2009. *DRM, a Design Research Methodology*. 1 ed. London: Springer.

Esteso, A., D. Peidro, J. Mula, and M. Díaz-Madroñero. 2023. "Reinforcement Learning Applied to Production Planning and Control". *International Journal of Production Research* 61(16):5772–5789.

García, S., J. Luengo, and F. Herrera. 2015. "Data Preparation Basic Models". In *Data Preprocessing in Data Mining*, edited by S. García, J. Luengo, and F. Herrera, 39–57. Cham: Springer.

Gros, T. P., J. Groß, and V. Wolf. 2020. "Real-Time Decision Making for a Car Manufacturing Process Using Deep Reinforcement Learning". In *2020 Winter Simulation Conference (WSC)*, 3032–3044 https://doi.org/10.1109/WSC48552.2020.9383884.

Heesen, B. 2024. "Challenges in a VUCA World". In *Effective Strategy Execution: Business Intelligence Using Microsoft Power BI*, edited by B. Heesen, 1–35. Berlin, Heidelberg: Springer.

Hillebrand, M., M. Lakhani, and R. Dumitrescu. 2020. "A Design Methodology for Deep Reinforcement Learning in Autonomous Systems". *Procedia Manufacturing* 52:266–271.

Hua, E. Y., S. Lazarova-Molnar, and D. P. Francis. 2022. "Validation of Digital Twins: Challenges and Opportunities". In *2022 Winter Simulation Conference (WSC)*, 2900–2911 https://doi.org/10.1109/WSC57314.2022.10015420.

Jaensch, F., K. Kuebler, E. Schwarz, and A. Verl. 2022. "Test-Driven Reward Function for Reinforcement Learning: A Contribution towards Applicable Machine Learning Algorithms for Production Systems". *Procedia CIRP* 112:103–108.

Janssen, C. P., and W. D. Gray. 2012. "When, What, and How Much to Reward in Reinforcement Learning-Based Models of Cognition". *Cognitive Science* 36(2):333–358.

Jeong, Y., T. K. Agrawal, E. Flores-García, and M. Wiktorsson. 2021. "A Reinforcement Learning Model for Material Handling Task Assignment and Route Planning in Dynamic Production Logistics Environment". *Procedia CIRP* 104:1807–1812.

Kardos, C., C. Laflamme, V. Gallina, and W. Sihn. 2021. "Dynamic Scheduling in a Job-Shop Production System with Reinforcement Learning". *Procedia CIRP* 97:104–109.

Kellner, F., B. Lienland, and M. Lukesch. 2022. "Produktionsplanung und -steuerung (PPS) (engl.: Production Planning and Control (PPC))". In *Produktionswirtschaft: Planung, Steuerung und Industrie 4.0 (engl.: Production Management: Planning, Control, and Industry 4.0)*, edited by F. Kellner, B. Lienland, and M. Lukesch, 159–342. Berlin, Heidelberg: Springer.

Kuhnle, A. 2020. *Adaptive Order Dispatching based on Reinforcement Learning: Application in a Complex Job Shop in the Semiconductor Industry*. Ph. D. thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany.

Kuhnle, A., J.-P. Kaiser, F. Theiß, N. Stricker, and G. Lanza. 2021. "Designing an Adaptive Production Control System Using Reinforcement Learning". *Journal of Intelligent Manufacturing* 32(3):855–876.

Kuhnle, A., M. C. May, L. Schaefer, and G. Lanza. 2022. "Explainable Reinforcement Learning in Production Control of Job Shop Manufacturing System". *International Journal of Production Research* 60(19):5812–5834.

Kuhnle, A., L. Schäfer, N. Stricker, and G. Lanza. 2019. "Design, Implementation and Evaluation of Reinforcement Learning for an Adaptive Order Dispatching in Job Shop Manufacturing Systems". *Procedia CIRP* 81:234–239.

Lamprecht, R., F. Wurst, and M. F. Huber. 2021. "Reinforcement Learning based Condition-oriented Maintenance Scheduling for Flow Line Systems". In *2021 IEEE 19th International Conference on Industrial Informatics*. July 21st-23rd, Palma de Mallorca, Spain, 1-7.

Lang, S., F. Behrendt, N. Lanzerath, T. Reggelin, and M. Mueller. 2020. "Integration of Deep Reinforcement Learning and Discrete-Event Simulation for Real-Time Scheduling of a Flexible Job Shop Production". In *2020 Winter Simulation Conference (WSC)*, 3057–3068 https://doi.org/10.1109/WSC48552.2020.9383997.

Liu, C.-L., C.-C. Chang, and C.-J. Tseng. 2020. "Actor-Critic Deep Reinforcement Learning for Solving Job Shop Scheduling Problems". *IEEE Access* 8:71752–71762.

Nasuta, A., M. Kemmerling, D. Lütticke, and R. H. Schmitt. 2024. "Reward Shaping for Job Shop Scheduling". *Machine Learning, Optimization, and Data Science* 14505:197–211.

Ou, X., Q. Chang, and N. Chakraborty. 2019. "Simulation Study on Reward Function of Reinforcement Learning in Gantry Work Cell Scheduling". *Journal of Manufacturing Systems* 50:1–8.

Panzer, M., B. Bender, and N. Gronau. 2024. "A deep reinforcement learning based hyper-heuristic for modular production control". *International Journal of Production Research* 62:2747–2768.

Park, I.-B., J. Huh, J. Kim, and J. Park. 2020. "A Reinforcement Learning Approach to Robust Scheduling of Semiconductor Manufacturing Facilities". *IEEE Transactions on Automation Science and Engineering* 17(3):1–12.

Rinciog, A., and A. Meyer. 2022. "Towards Standardising Reinforcement Learning Approaches for Production Scheduling Problems". *Procedia CIRP* 107(1):1112–1119.

Samsonov, V., K. Ben Hicham, and T. Meisen. 2022. "Reinforcement Learning in Manufacturing Control: Baselines, Challenges and Ways Forward". *Engineering Applications of Artificial Intelligence* 112:104868.

Schuh, G., T. Brosze, U. Brandenburg, S. Cuber, M. Schenk, J. Quick, *et al*. 2012. "Grundlagen der Produktionsplanung und -steuerung (engl.: Basics of Production Planning and Control)". In *Produktionsplanung und -steuerung 1: Grundlagen der PPS (engl.: Production Planning and Control 1: Basics of PPC)*, edited by G. Schuh and V. Stich, 11–293. Berlin, Heidelberg: Springer.

Serrano-Ruiz, J. C., J. Mula, and R. Poler. 2024. "Job Shop Smart Manufacturing Scheduling by Deep Reinforcement Learning". *Journal of Industrial Information Integration* 38:100582.

Sutton, R. S., and A. G. Barto. 2018. *Reinforcement Learning*. 2 ed. Cambridge, Massachusetts: The MIT Press.

Tanco, M., E. Viles, and L. Pozueta. 2009. "Comparing Different Approaches for Design of Experiments (DoE)". In *Advances in Electrical Engineering and Computational Science*, edited by S.-I. Ao and L. Gelman, 611–621. Dordrecht: Springer Netherlands.

Tang, J., Y. Haddad, J. Patsavellas, and K. Salonitis. 2023. "Multi-Bbjective Reconfigurable Manufacturing System Scheduling Optimisation: A Deep Reinforcement Learning Approach". *IFAC-PapersOnLine* 56(2):11082–11087.

Valet, A., T. Altenmüller, B. Waschneck, M. C. May, A. Kuhnle, and G. Lanza. 2022. "Opportunistic Maintenance Scheduling with Deep Reinforcement Learning". *Journal of Manufacturing Systems* 64:518–534.

van Otterlo, M., and M. Wiering. 2012. "Reinforcement Learning and Markov Decision Processes". In *Reinforcement Learning: State-of-the-Art*, edited by Springer, 3–42. Berlin, Heidelberg: Wiering, Marco and van Otterlo, Martijn.

Waschneck, B., A. Reichstaller, L. Belzner, T. Altenmüller, T. Bauernhansl, A. Knapp *et al*. 2018. "Optimization of Global Production Scheduling with Deep Reinforcement Learning". *Procedia CIRP* 72:1264–1269.

Wegmann, M., L. Steinmassl, S. B. Wagner, and M. F. Zaeh. 2025. "Optimizing production planning and control: A potential analysis of reinforcement learning". *Production Engineering*.

Wegmann, M., and M. F. Zaeh. 2023. "Towards a Methodology for Production Scheduling Using Reinforcement Learning Under Consideration of a Company's Individual Tasks and Goals". *Procedia CIRP* 120:416–421.

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

Zwicky, F. 1967. "The Morphological Approach to Discovery, Invention, Research and Construction". In *New Methods of Thought and Procedure*, edited by F. Zwicky and A. G. Wilson, 273–297. Berlin, Heidelberg: Springer.

AUTHOR BIOGRAPHIES

MARC WEGMANN joined the Institute for Machine Tools and Industrial Management at Technical University of Munich (TUM) in 2021 where he is the head of the production management and logistics department. His research interests are reinforcement learning, discrete event simulation, and production planning and control. His e-mail address is marc.wegmann@iwb.tum.de.

BENEDIKT GRUENHAG received his M.Sc. in mechanical engineering from Technical University of Munich (TUM) in 2024. He wrote his master thesis under the supervision of Marc Wegmann. His e-mail address is benedikt.gruenhag@tum.de.

PROF. DR.-ING. MICHAEL F. ZAEH has held the Chair of Machine Tools and Manufacturing Technology at the Technical University of Munich (TUM) since 2002. He completed his doctorate under Prof. Dr.-Ing. Joachim Milberg. From 1996 to 2002, he held several positions at a machine tool manufacturer, most recently as a member of the extended management board. His e-mail address is michael.zaeh@iwb.tum.de.

PROF. DR.-ING. CHRISTINA REUTER has held the Chair of Sustainable Production Systems at the Technical University of Munich (TUM) since 2025. She completed her doctorate under Prof. Dr.-Ing. Guenther Schuh. From 2017 to 2024, she worked at Airbus in various management positions. Her e-mail address is christina.reuter@iwb.tum.de.