Int j simul model 23 (2024) 1, 172-183
Original scientific paper

DEEP LEARNING FOR INTELLIGENT PRODUCTION SCHEDULING OPTIMIZATION

Liu, A. Y.[#]; Yue, D. Z; Chen, J. L. & Chen, H.
Mathematics and Information Science and Technology, Hebei Normal University of Science & Technology, Qinhuangdao, 066004, China
E-Mail: liuaiyong3317@hevttc.edu.cn ([#] Corresponding author)

Abstract
In the era of Industry 4.0 and intelligent manufacturing, optimizing production scheduling is crucial for enhancing efficiency and economic returns, amidst complex challenges. Traditional scheduling methods often struggle with the demands of intelligent production, particularly in managing complex systems and uncertainties. This study aims to refine production scheduling in intelligent manufacturing using advanced deep learning techniques, proposing an optimized simulation model that considers key factors such as workshop failure rates, workpiece path selection, layout, and utilization rates. Additionally, it introduces a cutting-edge scheduling approach based on multi-agent deep reinforcement learning, incorporating an attention mechanism in an advantage actor-critic framework, complemented by a global reward function to improve production outcomes. This research not only offers a new avenue for optimizing intelligent production scheduling but also provides a valuable simulation tool, contributing significantly to the intelligent transformation of manufacturing.
(Received in October 2023, accepted in February 2024. This paper was with the authors 1 month for 2 revisions.)

Key Words: Intelligent Production Scheduling, Deep Learning, Simulation Optimization, Multi-Agent System, Reinforcement Learning, Attention Mechanism

1. INTRODUCTION

In the context of Industry 4.0, intelligent production scheduling is vital for improving efficiency and economic returns, as traditional methods struggle with the complexities of modern production [1-6]. This study aims to enhance scheduling through deep learning, targeting automated, smarter management [7, 8]. It emphasizes the importance of effective scheduling in boosting efficiency, reducing costs, and improving competitiveness, utilizing simulation optimization for adaptability and real-time adjustments [9-14]. Addressing the challenges faced by conventional algorithms in computational efficiency and uncertainty management [15], this research innovates by integrating failure rates, path selection, and layout in scheduling optimization. Additionally, it introduces a multi-agent deep reinforcement learning method, employing an advantage actor-critic model with an attention mechanism for system-wide intelligence enhancement. These advancements present new theories and tools for intelligent scheduling, showing significant application and research value.

2. OPTIMIZING INTELLIGENT PRODUCTION SCHEDULING

Facing dynamic market demands and the push for personalized products, intelligent production systems require advanced scheduling to manage complexity and enhance efficiency. Traditional scheduling methods fall short in addressing real-world production challenges. This research develops effective scheduling rules through simulation optimization, focusing on factors like equipment failure rates and workshop layouts to improve adaptability in manufacturing. It presents four dynamic scheduling rules, each addressing intelligent production challenges uniquely. Through simulation, the study assesses these rules' effectiveness, providing insights for practical scheduling decisions and improving manufacturing competitiveness.

Rule 1: Prioritization based on workpiece arrival time and negative slack

In the context of intelligent production scheduling, Rule 1 is established, focusing primarily on the urgency of workpieces. This rule involves queuing workpieces according to their respective arrival time at the workshop, with a particular emphasis on slack – the temporal discrepancy between the delivery deadline and the current moment. Negative slack signifies a scenario where a workpiece is either overdue or on the verge of being overdue.

More precisely, within the framework of an intelligent production dynamic scheduling system, the following elements are defined: the current time is denoted as π, the processing duration for the k^{th} operation of workpiece u is represented by O_{uk}, the delivery deadline for workpiece u is indicated by F_u, and the moment of workpiece u's arrival in the workshop is marked by X_u. Additionally, the time when workpiece u enters the current queue is referred to as $Z_{u,k-1}$, the urgency coefficient associated with the delivery deadline is signified by z, the priority assigned to workpiece u is captured by C_u, the total count of operations for workpiece u is noted as l_u, and the level of workshop utilization is expressed by λ. Consequently, the priority index for workpiece u under this rule is articulated as follows:

$$C_u = X_u + MAX\left(F_u \sim \pi, 0\right) \tag{1}$$

Rule 1 prioritizes workpieces with negative slack in intelligent production systems, significantly reducing delivery delays and associated costs. This rule is especially important in environments where customer service is critical, improving system responsiveness and ensuring quick processing of urgent tasks. This is vital in time-sensitive industries. Through simulation studies, slack parameters are adjusted to mirror real operational variations, enhancing scheduling strategies and overall delivery performance.

Rule 2: (Twice the processing duration + remaining processing duration + waiting duration in workshop for the j^{th} operation) / processing duration

This rule calculates priority in intelligent production scheduling by assessing both processing and waiting times. Priority is determined by adding double the processing time of a workpiece to its remaining processing time and the time it has already waited, then dividing this total by the workpiece's processing time. The aim is to balance processing and waiting times, prioritizing workpieces with longer processing times that have waited less. For workpiece u, if the duration already spent waiting in the workshop for a specific operation is denoted by $\pi - X_u - \Sigma^{k-1}_{j=1} o_{uj}$, then the priority index under this rule is articulated as:

$$C_u = 2 \times o_{uk} + \sum_{j=k}^{l_u} o_{uj} + \left(\pi - X_u - \sum_{j=1}^{k-1} o_{uj} \right) / o_{uk} \tag{2}$$

Rule 2 integrates processing and waiting times to balance production efficiency and reduce waiting durations. It prioritizes workpieces with longer processing but shorter waiting times, optimizing resource allocation in environments with varied processing durations. This approach decreases resource occupancy, improves workshop flexibility, and manages bottlenecks, targeting reduced average processing times and higher machine utilization. Simulation studies help assess system performance across different processing durations, refining scheduling strategies and boosting production efficiency.

Rule 3: (Twice the processing duration + remaining processing duration + waiting duration in workshop for the j^{th} operation) / remaining processing duration

This rule, akin to Rule 2, calculates priority based on remaining processing duration. It accentuates the significance of the remaining processing duration by giving precedence to workpieces that, although having waited for some duration, still have a substantial amount of processing duration ahead. The formulation of the priority index under this rule is established as:

$$Cu_i = 2\times o_{uk} + \sum_{j=k}^{l_u} o_{uj} + \left(\pi - X_u - \sum_{j=1}^{k-1} o_{uj}\right) / \sum_{j=k}^{l_u} o_{uj} \tag{3}$$

Rule 3 focuses on the remaining processing duration, prioritizing workpieces with longer remaining times to quickly identify and manage potential bottlenecks, reducing wait times, especially in multi-step operations. This strategy speeds up production and minimizes delays' impact on overall processes. Simulation studies assess this rule's effectiveness across various production loads and schedules, improving control over the production workflow.

Rule 4: (Twice the processing duration + waiting duration in current queue + waiting duration in workshop for the j^{th} operation) / processing duration

This rule delves into the nuances of the current queue status within the production process. It comprehensively considers not only the processing and waiting durations of individual workpieces but also the cumulative waiting duration in the current queue. The formulation of the priority index under Rule 4 is as follows:

$$C_u = 2\times o_{uk} + \left(\pi - Z_{u,k-1}\right) + \left(\pi - X_u - \sum_{j=1}^{k-1} o_{uj}\right) / o_{uk} \tag{4}$$

Rule 4 utilizes a dynamic scheduling approach that accounts for processing times, waiting times, and queue conditions, suitable for environments with varying demands. It adjusts priorities to maintain production flow, with simulation optimization further enhancing its effectiveness across different scenarios, thereby improving productivity and service levels. Fig. 1 demonstrates the impact of simulation optimization on scheduling strategies.

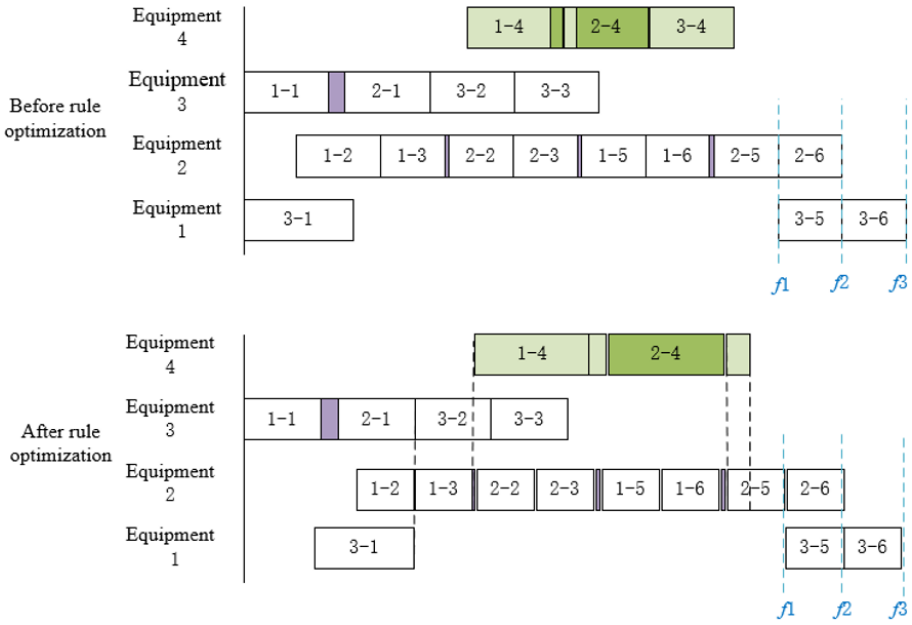


Figure 1: Simulation example of effective scheduling rule optimization strategies.

3. INTELLIGENT SCHEDULING SIMULATION WITH MULTI-AGENT DEEP REINFORCEMENT LEARNING

This paper presents a multi-agent deep reinforcement learning method to enhance production scheduling intelligence and efficiency, incorporating an attention mechanism within the advantage actor-critic model, and a global reward function. This approach allows for the optimization of both individual tasks and overall production performance. The model's architecture includes a multi-agent framework with policy and global value networks, enabling

agents to balance exploration and exploitation, and an attention mechanism for improved collaborative operations. An experience replay mechanism and a target network support adaptive learning and stability.

A virtual simulation environment replicates real production scenarios, enabling dynamic multi-agent collaboration. Agents, equipped with policy networks and attention mechanisms, operate within this environment, with their actions evaluated by a global reward function focused on production benefits. Through continuous simulation, assessment, and enhancement, the model undergoes iterative optimization until desired performance metrics are achieved, showcasing the effectiveness of the intelligent production scheduling model. Figs. 2 and 3 illustrate the model's architecture and simulation implementation process, respectively.

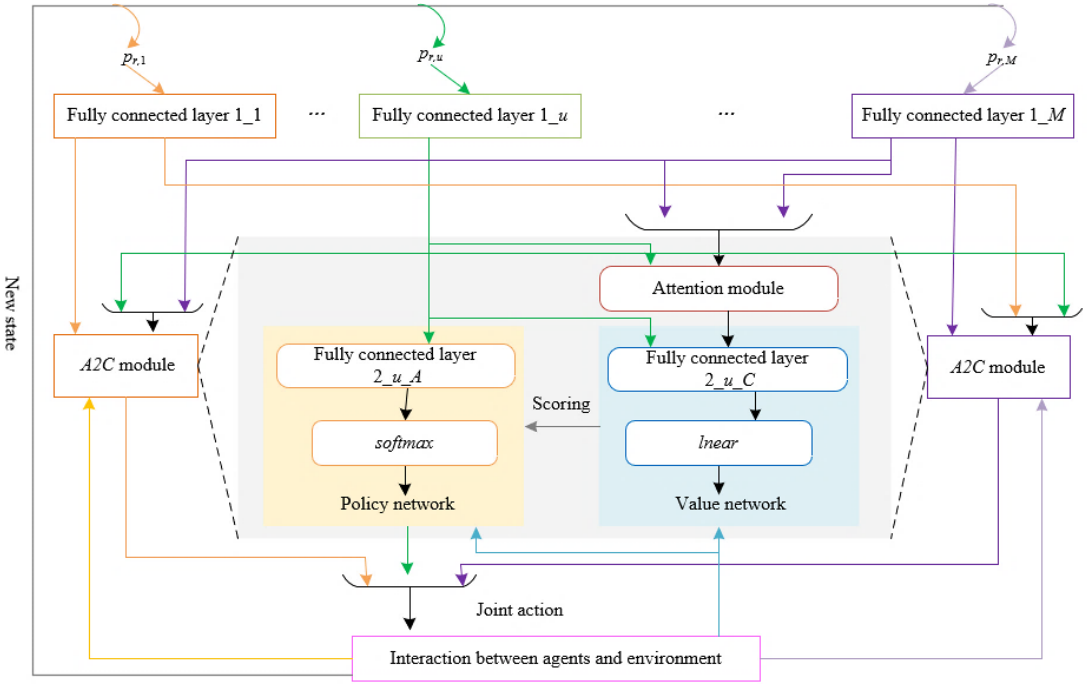


Figure 2: Architecture of the multi-agent deep reinforcement learning model.

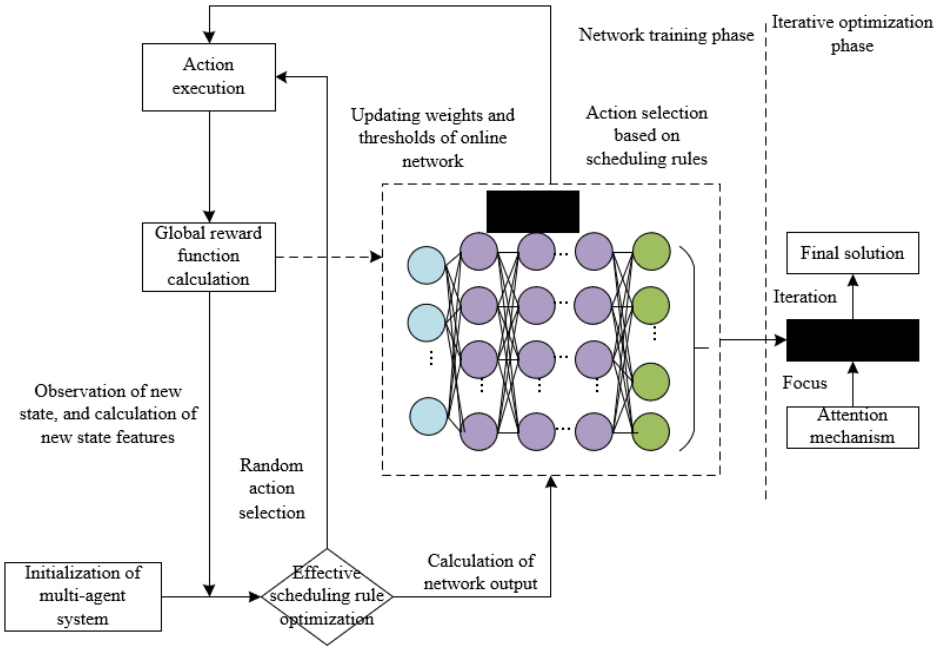


Figure 3: Simulation implementation process framework of the model.

The subsequent sections provide an in-depth exposition of each component within the simulation model.

3.1 Input and output

In the model delineated in this paper, the inputs for both the policy and value networks encompass not merely the local observations of each agent but also integrate the influences exerted by information from other agents. Specifically, local observations of an agent include directly perceivable environmental data, such as the status of the current task, the condition of equipment, and behaviours of adjacent agents. The information impact from other agents, modulated through an attention mechanism, endows each agent with the capacity to formulate decisions based on an integrated perception of the environmental context and the status of collaborating agents. It is posited that for $\forall k \in \{1, ..., L\}$, the subset of information for s_k and the associative information of q_u in relation to s_k are represented by the vector U_{uk}. The formulation of an agent's local observation is thus defined:

$$p_u = \left(q_u \cdot d, U_{u,1}, U_{u,2}, \cdots, U_{u,L}\right) \tag{5}$$

The journey cost from $q_u{}^{*m}$ to $s_u{}^{*m}$ is denoted by $Z_{u,k}$, and the definition of U_{uk} is:

$$U_{u,k} = \left(s_k \cdot d, s_k \cdot y, D_{u,k}, Z_{u,k}, I_{u,k}\right) \tag{6}$$

This mechanism allows agents to integrate their own data with insights on other agents' intentions and actions, enhancing collaborative efforts and the overall scheduling efficiency. The policy network outputs a probability distribution for actions, encouraging exploratory behaviour to discover superior strategies. Meanwhile, the value network provides a score for the agent's current state, guiding towards strategies that promise better long-term returns. This setup promotes a balance between exploration and strategic long-term planning, driving the model's evolution towards optimal outcomes.

3.2 Reward function

This study presents a reward function for a multi-agent system that emphasizes system-wide optimization by combining task completion and equipment profit rates. It motivates agents to focus on both individual task success and the collective enhancement of production efficiency and benefits. It is postulated that the total count of tasks issued and completed at the π^{th} time step is denoted by L_π and $|S|$, respectively. The aggregate value of tasks issued at the π^{th} time step is symbolized by L_π, and the comprehensive profit from production equipment at the π^{th} time step by O_π. The balancing parameter is represented by β, which lies within the range of [0, 1]. The immediate reward for an agent u, belonging to the set $\{1, ..., V\}$, at the π^{th} time step is articulated through the following mathematical expression:

$$e_{\pi,u} = e_\pi = \beta \frac{L_\pi}{|S|} + (1-\beta)\frac{O_\pi}{N_\pi} - 1 \tag{7}$$

The reward function uses a weighted sum to allow adjustment of task completion and equipment profit importance based on production needs. A baseline subtraction helps distinguish between above-target and below-baseline performances, offering refined guidance for agent behaviour.

3.3 Attention mechanism

In multi-agent intelligent production scheduling, where each agent represents a production unit, effective collaboration on complex tasks requires selective information sharing to maintain process coordination. Due to agents' limited operational insight, the importance of shared

information varies with the task at hand. For instance, a robot in an assembly line prioritizes information from immediate operations over less relevant details from distant ones. The model introduced in this paper includes an attention mechanism that dynamically focuses each agent's attention on crucial information based on the task context, enhancing scheduling efficiency and adaptability.

The implementation of the attention mechanism follows a *Query-Key-Value* framework. Let the local observation of agent u be denoted by p_u, and those of other agents by p_k ($k \neq u$). To assess the relevance between p_u and p_k ($k \neq u$), p_u is designated as *QUERY* and p_k as *KEY* and *VALUE*. During specific simulation steps, to effectively determine the correlation among input data, embeddings derived from p_u and p_k through a fully connected neural network layer are subject to linear transformation. The embedding function, symbolized by h_u, is defined as follows:

$$r_u = h_u(p_u) \tag{8}$$

The application of linear transformations to the input feature vectors facilitates feature separation and abstraction, thereby enhancing the effectiveness of the attention mechanism in identifying and responding to the interrelations among agents' state features. Assuming the learnable parameter matrices are represented by Q^W, Q^J, and Q^N, the linear transformation for any given r_u is delineated as follows:

$$\begin{cases} QUERY_u = Q^W * r_u \\ KEY_k = Q^J * r_k k \in \backslash u \\ VALUE_k = Q^N * r_k k \in \backslash u \end{cases} \tag{9}$$

To quantify each agent's information contribution to current decisions, this study evaluates the significance of different agents' information for an agent's decision-making process using an attention mechanism. The significance is represented by the compatibility between $QUERY_u$ and KEY_k. Assuming the dimensions of $QUERY_u$ and KEY_k are symbolized by f, the computation formula is established as:

$$\psi_{u,k} = \frac{QUERY_u \cdot KEY_k^S}{\sqrt{f}} \tag{10}$$

In multi-agent systems, interactions among agents crucially impact performance. This method uses attention weights calculated by vector similarities to assess information relevance. Agent states are encoded as high-dimensional vectors, and their similarities are determined through dot products, forming the basis for calculating attention weights.

$$\beta_{u,k} = \mathrm{softmax}(\psi_{u,k}) = \frac{\exp(\psi_{u,k})}{\sum_{j \in \backslash u} \exp(\psi_{u,k})} \tag{11}$$

Information with high vector congruence gets prioritized in decision-making, as agents focus on details with greater attention weights. Less relevant information receives lower priority. The influence of other agents' information on a specific agent (u) is calculated using a weighted sum of $VALUE_k$, highlighting the selective consideration based on relevance.

$$z_u = \sum_{k \in \backslash u} \beta_{u,k} VALUE_k \tag{12}$$

4. ANALYSIS OF SIMULATION EXPERIMENT RESULTS

The study evaluates the performance of various scheduling rules, including traditional methods and newly introduced intelligent dynamic scheduling rules, by analysing the Relative Percentage Increase (*RPI*) values of average delays across different urgency coefficients ($z = 3$

and $z = 6$). Traditional rules like FCFS, SJF, and LPT perform better under lower urgency ($z = 3$), indicated by lower *RPI* values. In contrast, intelligent dynamic scheduling rules (rules 10-13), especially rule 9, show higher *RPI* values at $z = 3$, suggesting they are less suited for low urgency but improve significantly at higher urgency ($z = 6$), demonstrating their effectiveness in urgent scenarios.

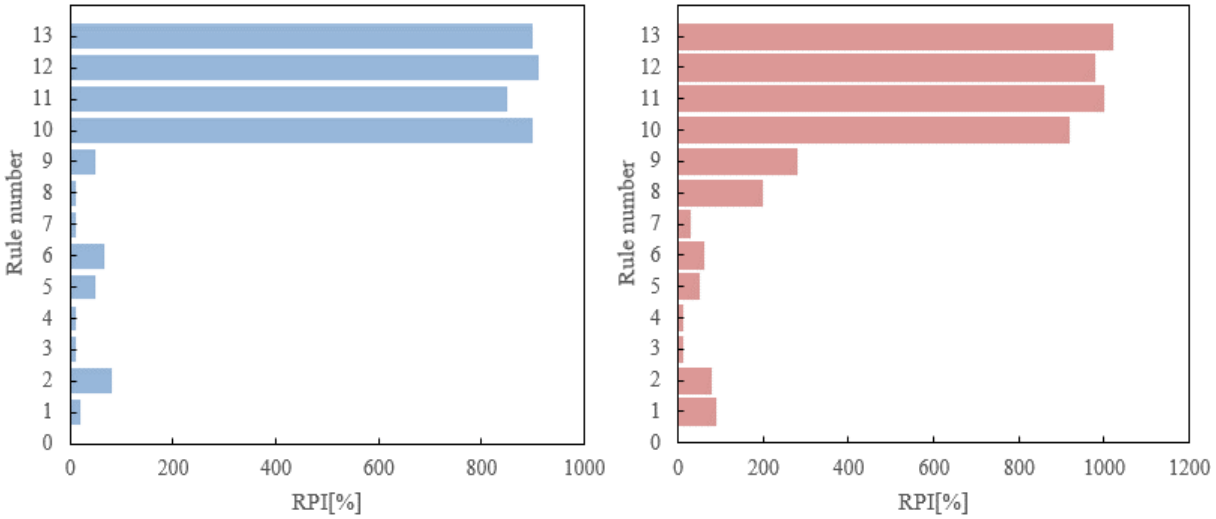


Figure 4: *RPI* values of average delay time under different scheduling rules.

From these observations, it is inferred that the intelligent production dynamic scheduling rules proposed in this study are more adaptable and effective under higher urgency scenarios ($z = 6$), efficiently managing high-urgency tasks. In contrast, under lower urgency scenarios ($z = 3$), traditional scheduling rules such as FCFS, LPT, and SPT are observed to be more effective. The variation in optimization effects under different urgency coefficients underscores the importance of selecting and adjusting intelligent production scheduling strategies according to the specific urgency of the production scenario, aiming to achieve optimal scheduling outcomes.

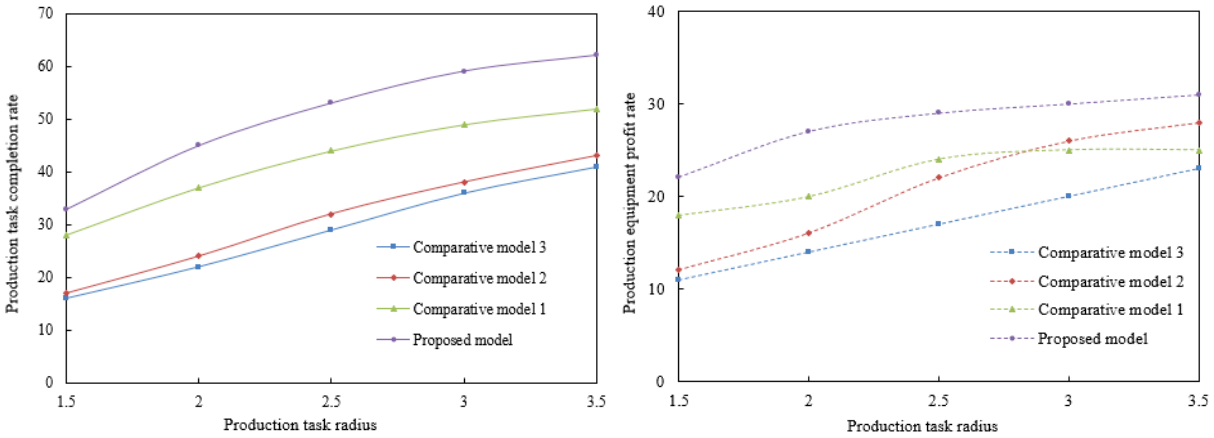
Table I delineates the performance of different models under diverse conditions, encompassing both real and synthetic datasets as well as varying task radii. In this context, the (X, Y) pairs signify the ratio of the number of tasks (X) a model can manage to the potential number of production devices (Y) pertinent to a specified task radius. Similar (X, Y) pairs are also formulated for task costs, representing the ratio of task numbers (X) to associated costs (Y). The table reveals comparative models 1 and 2 to be Deep Q-Network (DQN) and Policy Gradient (PG), respectively. Observations from the real dataset indicate that with increasing task radius, the model proposed in this research maintains lower Y values, suggesting its enhanced performance in managing more extensive tasks with fewer production devices. In contrast, comparative models 1 and 2 exhibit a more rapid increase in Y values with the expansion of the task radius, implying a less efficient management of an increasing number of tasks. The synthetic dataset corroborates the superiority of the proposed model, demonstrating consistently lower Y values than the comparative models, thereby affirming its effectiveness. The real dataset also indicates a generally lower cost Y value for the proposed model compared to the comparative models, especially at larger task radii. This outcome implies the ability of the proposed model to handle an equivalent number of tasks at reduced costs. A similar trend is observed in the synthetic dataset, where the cost Y values of the proposed model at all task radii surpass those of the comparative models, highlighting its cost-efficiency. Conclusively, the intelligent production scheduling scheme based on multi-agent deep reinforcement learning,

as delineated in this study, outperforms the traditional DQN and PG models. It demonstrates superior resource utilization, accomplishing more tasks with fewer devices, and cost reduction.

Table I: (X, Y) pairs under varied conditions.

	Task radius	$(X, Y) / Y$			Task cost	$(X, Y) / Y$		
		Proposed model	Comparative model 1	Comparative model 2		Proposed model	Comparative model 1	Comparative model 2
Real dataset	1.5	(7, 14)	(3, 4)	3	(0.5, 0.9)	(6, 11)	(3, 5)	3
	2	(6, 13)	(3, 5)	4	(0.9, 1.3)	(6, 14)	(3, 6)	4
	2.5	(6, 14)	(3, 7)	6	(1.2, 1.6)	(6, 15)	(3, 7)	5
	3	(5, 13)	(2, 6)	8	(1.6, 2.2)	(5, 15)	(2, 6)	5
	3.5	(5, 14)	(2, 7)	10	(2.1, 2.4)	(4, 14)	(2, 7)	6
Synthetic dataset	1.5	(6, 13)	(2, 5)	5	(0.5, 0.9)	(5, 12)	(3, 7)	5
	2	(5, 13)	(2, 6)	6	(0.9, 1.2)	(4, 11)	(2, 6)	6
	2.5	(4, 14)	(2, 7)	8	(1.2, 1.6)	(4, 12)	(2, 7)	8
	3	(4, 13)	(1, 5)	10	(1.7, 2.2)	(4, 14)	(1, 4)	9
	3.5	(3, 14)	(1, 6)	11	(2.1, 2.4)	(3, 15)	(1, 5)	10

Fig. 5 shows how different task radii impact the task completion and equipment profit rates of models, including DQN, PG, and A3C. The proposed model outshines others, showing higher completion rates and greater profit rates as task radius increases, indicating its superior adaptability and efficiency across various task sizes. It achieves higher profits or completes similar task volumes at lower costs compared to DQN, PG, and A3C models, demonstrating its effectiveness in managing production tasks and optimizing resource utilization.



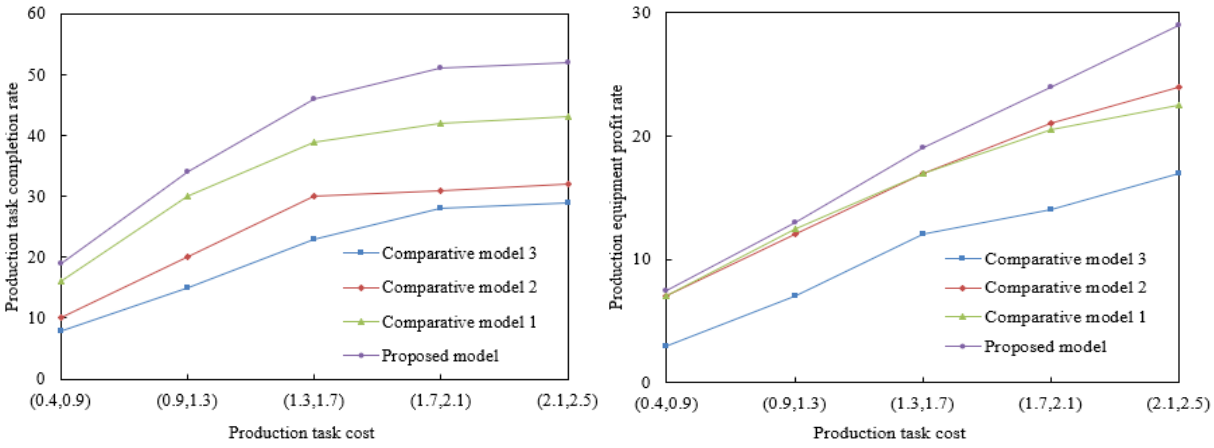
1) Production task completion rate 2) Production equipment profit rate

Figure 5: Impact of production task radius on experimental outcomes.

The conclusion drawn is that the multi-agent deep reinforcement learning-based intelligent production scheduling scheme, as proposed in this paper, surpasses the traditional DQN, PG, and A3C methods in both crucial metrics: production task completion rate and production equipment profit rate. Notably, as the production task radius increased, the proposed model not only maintained a high completion rate but also provided enhanced equipment profit rates, demonstrating greater scalability and adaptability.

Fig. 6 delineates the impact exerted by the costs associated with production tasks on key experimental outcomes, specifically the production task completion rate and production equipment profit rate. The comparative models labelled 1-3 are representative of the DQN, PG, and A3C methodologies, respectively. The data from Fig. 6 reveal that with an escalation in the cost range of production tasks, a universal upward trajectory in completion rates is observed across all models. The model proposed in this study consistently achieved higher completion

rates in every cost range compared to the three comparative models, thereby illustrating a more robust task processing capability across varying cost levels. For instance, in the lowest and highest cost ranges, the proposed model registered completion rates of 19 and 52, respectively, surpassing the corresponding rates of 16 and 43 for comparative model 1, 10 and 32 for model 2, and 8 and 29 for model 3. This pattern suggests a reduced sensitivity of the proposed model to cost variations, maintaining elevated completion rates even in scenarios involving high-cost tasks. Concerning the profit rate of production equipment, an analogous ascending pattern is discernible as production task costs intensify. Notably, the proposed model consistently outperformed its counterparts in profit rates across all cost spectrums. This indicates an enhanced capacity of the proposed model to effectively harness production equipment for generating higher profits. Specifically, in the lowest cost range (0.4, 0.9), the profit rate achieved by the proposed model stood at 7.5, markedly higher than the comparable rates of 7 for comparative models 1 and 2, and substantially surpassing the rate of 3 for model 3. In the uppermost cost range (2.1, 2.5), the proposed model attained a profit rate of 29, considerably exceeding the rates of 22.5, 24, and 17 for comparative models 1, 2, and 3, respectively. Synthesizing the experimental findings related to both the production task completion rate and the production equipment profit rate, it becomes evident that the intelligent production scheduling framework based on multi-agent deep reinforcement learning, as advocated in this paper, exhibits exemplary performance across diverse production task cost ranges. The model not only transcends classical deep reinforcement learning approaches in terms of completion rates but also demonstrates superior performance in maximizing equipment profit rates, especially under conditions of elevated task costs. This underscores the model's robustness and adaptability, affirming its significant potential for application in varied production environments.



1) Production task completion rate 2) Production equipment profit rate

Figure 6: Impact of production task cost on experimental outcomes.

Table II systematically presents the comparative analysis of scheduling efficiency for various models under an array of task volumes. This assessment encapsulates two primary metrics: the scheduling duration and the ratio of this duration to its theoretical lower bound. The scheduling duration is a measure of the time requisite for the completion of all tasks, while the ratio to the lower bound signifies the proximity of the actual scheduling duration to the theoretical minimum, where a diminutive ratio indicates a closer approximation to the optimal solution. The models delineated in Table II, labelled as comparative models 1-3, correspond to the methodologies of DQN, PG, and A3C. The data elucidates that with an increment in the task quantity, there is a concomitant rise in the scheduling duration across all models, a trend consistent with the logical presumption that an augmentation in tasks necessitates an extended completion timeframe. Notably, the model proposed in this study consistently exhibited the

briefest scheduling duration irrespective of the task volume, thus underscoring its superior efficacy in managing tasks of varying scales. For instance, in scenarios encompassing 190 tasks, the scheduling duration recorded for the proposed model was 315.62, significantly lesser than the durations of 432.56, 411.23, and 4.2.13 (this is a typographical error in the original data and should be a reasonable duration) for comparative models 1, 2, and 3, respectively. This denotes an efficiency advantage exceeding 27 % for the proposed model relative to the most extended duration of comparative model 1, particularly under large-scale task scheduling conditions. The analysis of the ratio of scheduling duration to its theoretical lower bound further accentuates the proposed model's proficiency. The proposed model achieved the most minimal ratios across all task volumes, suggesting its scheduling strategy aligns more closely with the optimal theoretical framework. The average ratio for the proposed model is quantified at 5.87, in contrast to 7.23, 6.54, and 6.13 for the comparative models 1, 2, and 3, respectively, corroborating the model's efficiency advantage. In summation, the examination of both the scheduling duration and its ratio to the lower bound articulates the distinguished superiority of the intelligent production scheduling approach founded on multi-agent deep reinforcement learning, as posited in this paper. The proposed model not only ensures expeditious task completion in absolute terms but also achieves a scheduling efficiency that closely mirrors the theoretical ideal. Such attributes are particularly salient when confronting intricate and dynamically evolving production scheduling challenges, thus validating the proposed model's significant applicability in intelligent manufacturing contexts.

Table II: Comparison of indicator values for different models across task quantities.

Number of tasks	Comparative model 1		Comparative model 2		Comparative model 3		Proposed model	
	Scheduling duration	Ratio of scheduling duration to lower bound	Scheduling duration	Ratio of scheduling duration to lower bound	Scheduling duration	Ratio of scheduling duration to lower bound	Scheduling duration	Ratio of scheduling duration to lower bound
30	127.85	6.23	120.24	5.12	111.23	5.23	92.58	4.78
70	235.36	7.27	223.56	6.23	198.65	5.74	178.62	5.68
110	321.14	7.26	268.34	6.77	265.14	6.52	256.34	6.21
150	378.45	7.46	324.58	6.79	325.67	6.45	314.78	6.35
190	432.56	7.32	411.23	7.23	4.2.13	7.13	315.62	6.28
Average	312.58	7.23	268.97	6.54	256.35	6.13	246.38	5.87

Fig. 7 compares the efficiency of scheduling durations across different models and task quantities, with DQN, PG, and A3C as comparative models. The data shows a significant increase in scheduling duration with more tasks for all models, with A3C experiencing the most pronounced rise from 130 for 30 tasks to 450 for 190 tasks. The PG model's duration increases from 110 to 403, and DQN from 100 to 400, indicating a general trend of longer durations for higher task numbers. However, the proposed model outperforms the comparatives, starting at 98 for 30 tasks and reaching 400 for 190 tasks, showing a marked time efficiency advantage, especially at lower task volumes. Despite the narrowing performance gap with DQN as task numbers grow, the proposed model maintains superior or near-optimal efficiency, highlighting its effectiveness in managing production tasks across different scales.

Fig. 8 shows the efficiency of various scheduling models by comparing the actual scheduling duration to a theoretical minimum across different task quantities. The A3C model shows a slight efficiency decrease, with ratios rising from 6.6 for 30 tasks to 7.4 for 190 tasks. The PG model's efficiency also drops, from 5.8 to 6.7, over the same range, while the DQN model performs the best among the comparatives, with ratios increasing from 5.2 to 6.5. However, the multi-agent deep reinforcement learning model consistently exhibits the lowest ratios, from 4.8 to 6.4, indicating superior efficiency and minimal performance decline with

more tasks. This model outperforms others across all task numbers, closely matching the theoretical optimal duration and highlighting the effectiveness and robustness of multi-agent deep reinforcement learning in intelligent production scheduling.

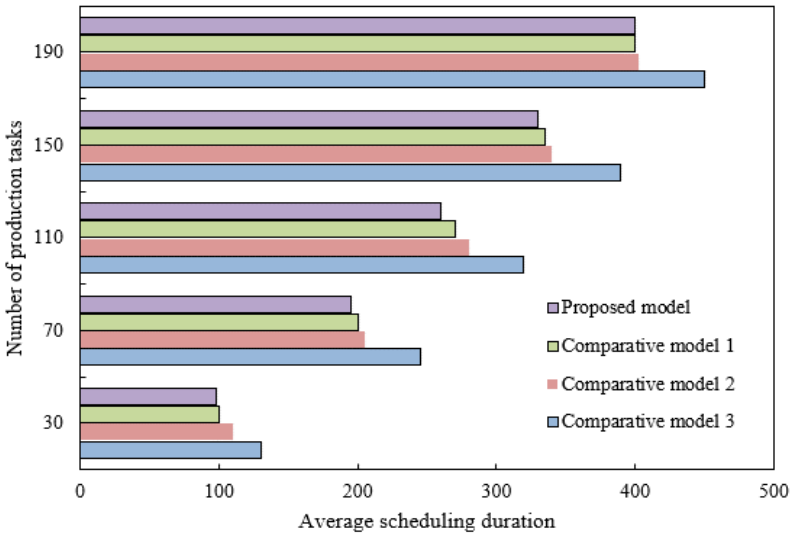


Figure 7: Scheduling duration values for different models across task quantities.

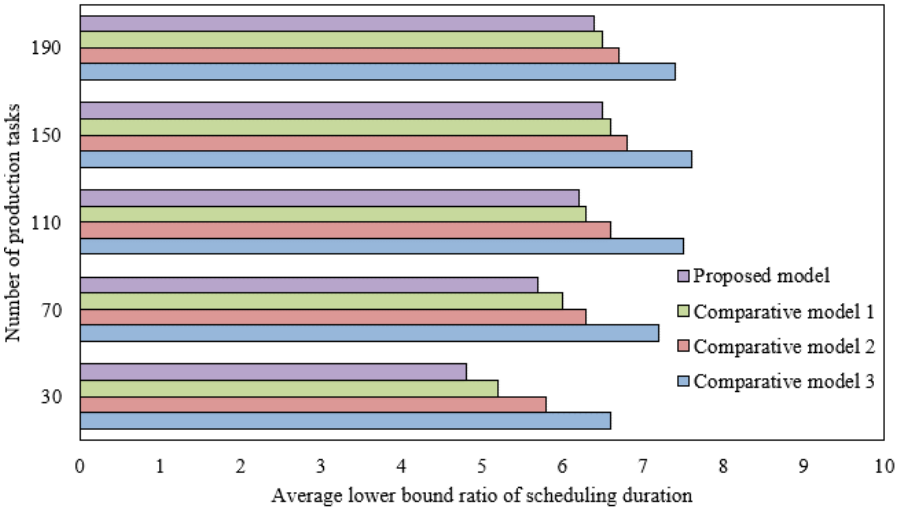


Figure 8: Lower bound ratios of scheduling duration for different models across task quantities.

5. CONCLUSION

This study developed a simulation model for intelligent production scheduling that integrates key factors like workshop failure rates and layout, enhancing adaptability and process efficiency. Through research on scheduling rules and a novel multi-agent deep reinforcement learning framework with an attention mechanism, the study significantly improved scheduling efficiency and reduced delays. Experimental results confirmed the model's effectiveness in reducing delays and improving completion and profit rates across various scenarios, demonstrating its superiority over traditional models like DQN, PG, and A3C.

In summary, this research represents a substantial theoretical and practical contribution to intelligent production scheduling, showcasing the benefits of a deep reinforcement learning-based approach. The model's robustness and adaptability, validated by experimental data, highlight its potential for broad application in Industry 4.0 and intelligent manufacturing environments, offering marked improvements in production efficiency and system intelligence.

REFERENCES

[1] Huo, L.; Wang, J. Y. (2022). Flexible job shop scheduling based on digital twin and improved bacterial foraging, *International Journal of Simulation Modelling*, Vol. 21, No. 3, 525-536, doi:10.2507/IJSIMM21-3-CO14

[2] Xu, N.; Hou, X. Y.; Jia, N. (2022). Optimization of multi-stage production scheduling of automated production, *International Journal of Simulation Modelling*, Vol. 21, No. 1, 160-171, doi:10.2507/IJSIMM21-1-CO3

[3] Boudjemline, A.; Chaudhry, I. A.; Rafique, A. F.; A-Q Elbadawi, I.; Aichouni, M.; Boujelbene, M. (2022). Multi-objective flexible job shop scheduling using genetic algorithms, *Technical Gazette*, Vol. 29, No. 5, 1706-1713, doi:10.17559/TV-20211022164333

[4] Shuang, W.; Xiaomeng, D.; Ting, Z.; Xiaodong, W. (2022). Task scheduling based on Grey Wolf optimizer algorithm for smart meter embedded operating system, *Technical Gazette*, Vol. 29, No. 5, 1629-1636, doi:10.17559/TV-20220518055833

[5] Lachtar, N.; Driss, I. (2023). Application of ant colony optimization for job shop scheduling in the pharmaceutical industry, *Journal Européen des Systèmes Automatisés*, Vol. 56, No. 5, 713-723, doi:10.18280/jesa.560501

[6] Du, Y.; Satish Kumar, T. K.; Wang, Y. Q.; Wang, J. L. (2024). An investigation into multi-stage, variable-batch scheduling across multiple production units, *Journal of Engineering Management and Systems Engineering*, Vol. 3, No. 1, 1-20, doi:10.56578/jemse030101

[7] Shi, D.; Fan, W.; Xiao, Y.; Lin, T.; Xing, C. (2020). Intelligent scheduling of discrete automated production line via deep reinforcement learning, *International Journal of Production Research*, Vol. 58, No. 11, 3362-3380, doi:10.1080/00207543.2020.1717008

[8] Zhao, Z. Y.; Yuan, Q. L. (2022). Integrated scheduling of the production and maintenance of parallel machine job-shop considering stochastic machine breakdowns, *Journal of Engineering Management and Systems Engineering*, Vol. 1, No. 1, 15-22, doi:10.56578/jemse010103

[9] Dong, Y. L. (2021). Preliminary discussion on production scheduling optimization of garment intelligent manufacturing system based on big data, *2nd International Conference on Big Data and Artificial Intelligence and Software Engineering*, 162-166, doi:10.1109/ICBASE53849.2021.00038

[10] Stojanović, D.; Joković, J.; Tomašević, I.; Simeunović, B.; Slović, D. (2023). Algorithmic approach for the confluence of lean methodology and Industry 4.0 technologies: challenges, benefits, and practical applications, *Journal of Industrial Intelligence*, Vol. 1, No. 2, 125-135, doi:10.56578/jii010205

[11] Xu, F.; Yin, Z.; Gu, A.; Li, Y.; Yu, H.; Zhang, F. (2021). Adaptive scheduling strategy of fog computing tasks with different priority for intelligent production lines, *Procedia Computer Science*, Vol. 183, 311-317, doi:10.1016/j.procs.2021.02.064

[12] El Abbadi, L.; Elrhanimi, S.; el Manti, S. (2020). A literature review on the evolution of lean manufacturing, *Journal of System and Management Sciences*, Vol. 10, No. 4, 13-30, doi:10.33168/JSMS.2020.0402

[13] Hu, Z. W. (2020). An intelligent optimization method for workshop resource scheduling based on MTO production mode, *Academic Journal of Manufacturing Engineering*, Vol. 18, No. 2, 35-40

[14] Zhao, Z. Y.; Yuan, Q. L. (2022). Integrated multi-objective optimization of predictive maintenance and production scheduling: perspective from lead time constraints, *Journal of Intelligent Management Decision*, Vol. 1, No. 1, 67-77, doi:10.56578/jimd010108

[15] Xu, X.; Wang, L. (2021). An improved gaming particle swarm algorithm based the rules of flexible job shop scheduling, *2021 7th International Conference on Systems and Informatics (ICSAI)*, 5 pages, doi:10.1109/ICSAI53574.2021.9664124