ICCEPM 2024 https://dx.doi.org/10.6106/ICCEPM.2024.0056
The 10th International Conference on Construction Engineering and Project Management
Jul. 29-Aug.1, 2024, Sapporo

Multi-objective production scheduling of precast concrete based on reinforcement learning

Leting ZU[1], Wenzhu LIAO[2]*

[1] *Department of Engineering Management, School of Management Science and Real Estate, Chongqing University, China,* E-mail address: 202203021035@stu.cqu.edu.cn
[2] *Department of Engineering Management, School of Management Science and Real Estate, Chongqing University, China,* E-mail address: liaowz@cqu.edu.cn

Abstract: To enhance energy efficiency and reduce emissions in prefabricated construction, optimizing the production scheduling of precast concrete is considered an effective approach. Due to the unique characteristics of precast concrete during production, traditional scheduling models are no longer applicable. This present study introduces practical considerations, such as a limited number of molds, buffers, uncertainty of order arrivals and vehicles. Furthermore, to meet the requirements of contemporary industrial development, a mulit-objective optimization scheduling model is formulated by integrating total processing time, on-time delivery rate and work station idle time. A solution based on reinforcement learning algorithm is devised. Results indicate that this method can effectively undergo training and achieve outstanding performance in addressing such issues. The model has the potential to significantly reduce decision-making time in precast production, thereby contributing to the sustainable development of prefabricated construction.
Keywords: reinforcement learning; production scheduling; precast concrete; prefabricated construction; optimization

1. INTRODUCTION

The construction method of prefabricated construction offers numerous advantages, including reduced labor costs, shorter construction durations, and enhanced construction quality compared to traditional construction approaches. As a result, it is regarded as an inevitable trend within the construction industry [1].However, cost control remains a significant challenge in the realm of prefabricated construction [2].

To reduce production costs for precast concrete, we can approach the challenge from both managerial and technological perspectives. On the technological front, this involves significant financial investments in cutting-edge machinery, real-time scheduling monitoring, and computer equipment. However, these investments may not be feasible for many smaller prefabrication facilities. In contrast, optimizing production scheduling at the management level offers a more cost-effective alternative that has gained widespread adoption. As a result, in recent years, there has been a growing emphasis on research related to production scheduling for precast concrect. In this research field, the current major innovations can be categorized into three aspects: modeling of precast concrete scheduling, optimization methods, and dynamic factors with re-scheduling. In terms of modeling, considerations include mold constraints [3], labor-related constraints [4], multiple production lines [5], adding new processes [6], and buffers [7]. In terms of optimization methods, currently, genetic algorithms remain mainstream. They are applied to multi-objective or multi-line precast concrete scheduling problems [8]. Some studies integrate genetic algorithms with other methods to enhance performance [6,9]. However, to address real-time scheduling in dynamic environments, reinforcement learning methods show promising prospects [10]. In terms of dynamic factors and re-scheduling, current research accounts for design changes, urgent orders, machine failures, etc. Solutions involve rescheduling across multiple production lines [11] and prioritizing product-based rescheduling [12]. Therefore, this study will utilize reinforcement learning as the optimization method, further exploring its applicability in the more complex environment of precast concrete scheduling.

2. A PRECAST CONCRETE PRODUCTION SCHEDULING MODEL

In the early stages of research, the precast concrete production process was primarily divided into six steps: mold assembling, rebars and embedded parts placement, concrete casting, curing, mold removal, and quality inspection with surface treatment. Wang and Hu [6], adopting a holistic perspective on the entire supply chain, expanded the production process of precast concretes to include three additional steps: mold production, storage, and transportation, making it a total of nine steps. They considered that, due to the insufficient standardization in the modular construction industry, many different types of components lack readily available molds, making the inclusion of mold production steps more aligned with practical needs. Additionally, recognizing the time required for storage and transportation post-production is essential, as neglecting these processes could lead to results inconsistent with reality, where the calculated completion time may be less than the actual delivery time. Considering the mold production stage will affect the training of reinforcement learning agents, as this environment significantly increases the scheduling uncertainty, the current use of reinforcement learning in the field of precast concrete scheduling is still not mature. RUAN (2022) configured the production process into eight steps, and after training and debugging, this study believes that RUAN's (2022) scheduling model is more conducive to the training of reinforcement learning[13]. Therefore, the precast concrete manufacturing process considered in this paper includes the following eight steps: (1) Mold assembling; (2) Rebars and embedded parts placement; (3) Concrete casting (concrete placement, concrete vibration, concrete scraping); (4) Concrete curing; (5) Mold removal; (6) Quality inspection and surface treatment; (7) Storing; (8) Transportation. A simplified process flowchart is illustrated in Fig 1. And the specific scheduling process of the eight production stages is illustrated in Fig 2.

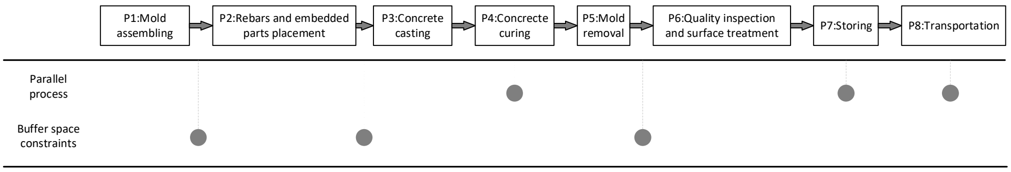

Fig 1. Eight processes of precast concrete

3. DEEP Q-NETWORK ALGORITHM

The DQN algorithm for the precast concrete production scheduling problem consists of three main components, the Fig 3. shows the interaction process between the Precast Concrete Production Simulator and the DQN algorithm.The first component is the Precast Concrete Production Simulator, designed to replicate the workshop's production environment. After the agent outputs an action, the simulator selects the next workpiece to be processed using a dispatching rule. It then simulates the processing environment for the chosen workpiece, calculating rewards and new states for each decision. The second component is the DQN algorithm, incorporating a Markov chain design based on actions, states, and rewards for training the agent. The third component is the Workpiece Generator, simulating the arrival of random orders and sequentially placing them in the processing queue. This allows the Precast Concrete Production Simulator to determine the next workpiece for processing based on actions and dispatching rules.

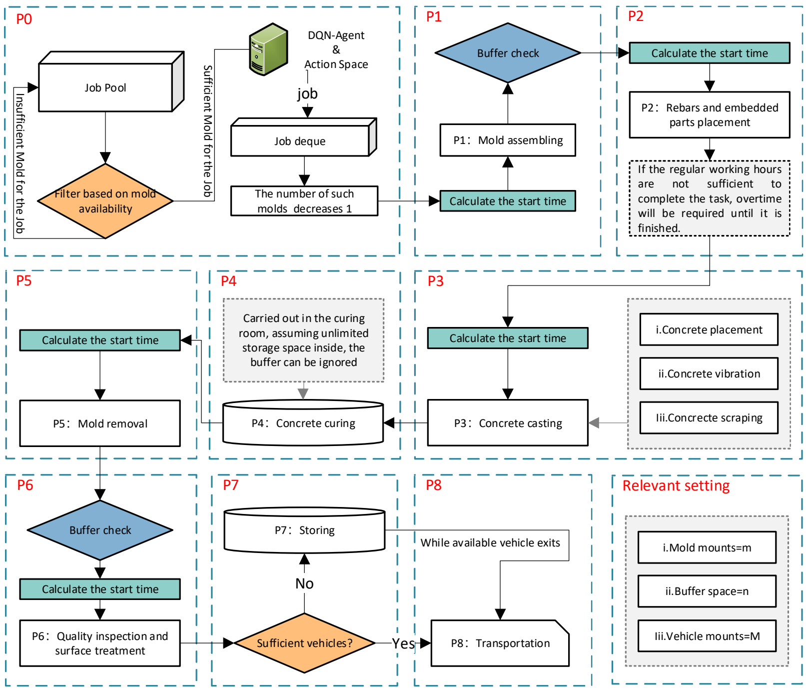

Fig 2. Production scheduling rules for precast concrete

In the design of the Markov chain, seven states are employed to describe the production status in the simulated workshop, namely:(1) s0 represents the current decision time point, which is also the start time of the next job in the first process; (2) s1 represents the current number of days elapsed; (3) s2 represents the remaining number of jobs; (4) s3 represents the average slack time, indicating the average duration each job waiting for processing deviates from its respective due time; (5) s4 represents the minimum slack time; (6) s5 represents the maximum slack time; (7) s6 represents the standard deviation of slack time for jobs waiting for processing. In terms of action selection, four commonly used dispatching rules in the production scheduling domain were chosen: (1) EDD (Earliest Due Date); (2) SPT (Shortest Processing Time); (3) CR (Critical Ratio); (4) FIFO (First In, First Out). In terms of the reward mechanism, three objectives are established: (1) total processing time（sparse reward）; (2) on-time delivery rate(dense reward); (3)work station idle time(spare reward).

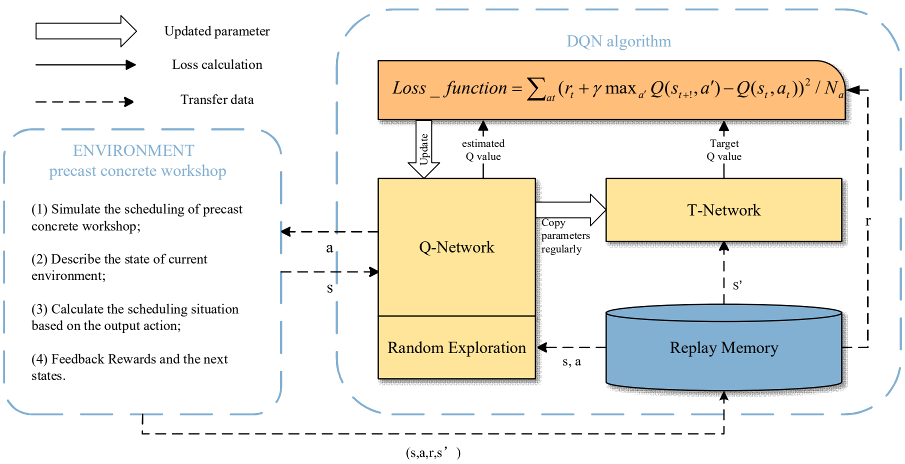

Fig 3. Principles of Algorithm Interaction

The agent module, as depicted in the Fig 4., comprises an input layer with seven states, an output layer with four actions corresponding to four scheduling rules, and four hidden layers. Each hidden layer contains 30 nodes, and hyperbolic tangent (Tanh) serve as activation functions between layers.

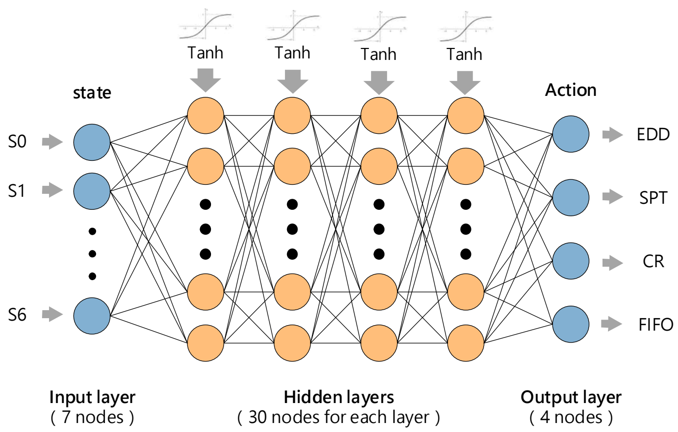

Fig 4. Schematic illustration of the neural network in PC-DQN

4. SIMULATION EXPERIMENTS

Simulation experiments were conducted using Python 3.9 on a computer equipped with an Intel i5-12490f processor, 16GB RAM, and an RTX 3060 graphics card. The simulation utilized background data commonly employed by researchers [6]. The processing times for each precast concrete product type are presented in Table 1. Notably, the processing time for process 7 is assumed to be 0 by default and will be calculated based on actual circumstances. Additionally, the processing time for process 8 is set to 2 hours in this study, representing the time required for transporting concrete products from the precast concrete plant to the construction site and back.

Table 1. Processing time for each PC product type.

Product type	Processing time(h)							
	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**
Type 1	1.5	2.0	0.5	8.0	1.0	0.5	0	2.0
Type 2	1.0	2.0	0.4	8.0	1.0	0.5	0	2.0
Type 3	1.0	1.5	0.5	8.0	0.5	0.5	0	2.0
Type 4	0.5	1.0	0.3	8.0	0.3	0.5	0	2.0
Type 5	1.0	0.8	1.0	8.0	1.5	0.5	0	2.0
Type 6	0.5	2.0	0.4	8.0	0.5	0.5	0	2.0
Type 7	1.5	2.0	0.5	8.0	1.0	0.4	0	2.0
Type 8	0.5	2.0	0.3	8.0	0.6	0.3	0	2.0
Type 9	1.5	1.8	1.2	8.0	1.5	1.5	0	2.0
Type 10	0.4	0.5	0.6	8.0	0.5	0.5	0	2.0

Table 2 displays the hyperparameters set in this study during the experiments. In the training phase, a total of 5000 episodes were trained with a replay buffer capacity of 10000. 64 sets of data are randomly selected from the buffer for each training iteration. The target model was updated every 10 steps to enhance training stability. Training utilized an ε-greedy strategy, starting with an exploration rate of 1, which decreased to zero at episode 3000, expediting model convergence and preventing potential issues such as gradient explosion.

Table 2. Hyperparameter for training DQN.

Hyperparameter	**Value**
Number of episodes (E)	3000
Minibatch size	64
update frequency i	10
Size of reply buffer (θ)	10000
Greedy exploration factor (ε)	Linearly increased from 0.00 to 1.00
Discount factor (γ)	0.95
Learning rate (α)	0.01

The Fig 5. and Fig 6. illustrates the changes in training loss and reward values during the training process of the agent over 3000 episodes. As training progresses, there is a noticeable decrease and convergence in the loss value. Additionally, the total reward value gradually increases, indicating that the agent has effectively learned to address the dual-objective PC scheduling problem.

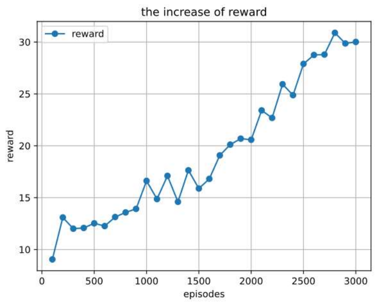

Fig 5. Training loss curve

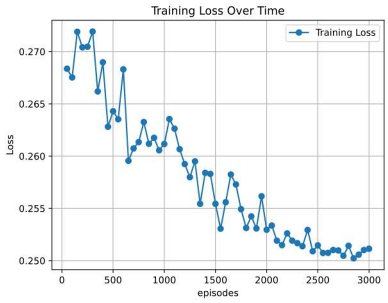

Fig 6. Reward curve

REFERENCE

[1] G. Tumminia, F. Guarino, S. Longo, et al. "Life cycle energy performances and environmental impacts of a prefabricated building module." Renewable and Sustainable Energy Reviews, vol. 92, pp. 272-283, 2018.

[2] J. Hong, G. Q. Shen, Z. Li, et al. "Barriers to promoting prefabricated construction in China: A cost–benefit analysis." Journal of Cleaner Production, vol. 172, pp. 649-660, 2018.

[3] V. Benjaoran, N. Dawood, B. Hobbs. "Flowshop scheduling model for bespoke precast concrete production planning." Construction Management and Economics, vol. 23, no. 1, pp. 93-105, 2005.

[4] J. Cao, P. Zhao, G. Liu. "Optimizing the production process of modular construction using an assembly line-integrated supermarket." Automation in Construction, vol. 142, p. 104495, 2022.

[5] Z. Yang, Z. Ma, S. Wu. "Optimized flowshop scheduling of multiple production lines for precast production." Automation in Construction, vol. 72, pp. 321-329, 2016.

[6] Z. Wang, H. Hu. "Improved Precast Production–Scheduling Model Considering the Whole Supply Chain." Journal of Computing in Civil Engineering, vol. 31, no. 4, p. 04017013, 2017.

[7] Y. R. Dan, G. W. Liu, Y. Fu. "Optimized flowshop scheduling for precast production considering process connection and blocking." Automation in Construction, vol. 125, p. 103575, 2021.

[8] W. Niu, J. Li. "A two-stage cooperative evolutionary algorithm for energy-efficient distributed group blocking flow shop with setup carryover in precast systems." Knowledge-Based Systems, vol. 257, p. 109890, 2022.

[9] B. Anvari, P. Angeloudis, W. Y. Ochieng. "A multi-objective GA-based optimisation for holistic Manufacturing, transportation and Assembly of precast construction." Automation in Construction, vol. 71, pp. 226–241, 2016.

[10] T. Kim, Y. W. Kim, D. Lee, et al. "Reinforcement learning approach to scheduling of precast concrete production." Journal of Cleaner Production, vol. 336, p. 130419, 2022.

[11] Z. L. Ma, Z. T. Yang, S. L. Liu, S. Wu. "Optimized rescheduling of multiple production lines for flowshop production of reinforced precast concrete components." Automation in Construction, vol. 95, pp. 86–97, 2018.
[12] Z. Wang, H. Hu, J. Gong. "Modeling worker competence to advance precast production scheduling optimization." Journal of Construction Engineering and Management, vol. 144, no. 11, p. 04018098, 2018.
[13] M. H. Ruan, F. Xu. "Improved eight-process model of precast component production scheduling considering resource constraints." Journal of Civil Engineering and Management, vol. 28, no. 3, pp. 208 – 222, 2022.