WARSAW UNIVERSITY OF TECHNOLOGY Index 351733
FACULTY OF CIVIL ENGINEERING
COMMITTEE FOR CIVIL AND WATER ENGINEERING
POLISH ACADEMY OF SCIENCES ISSN 1230-2945

DOI: 10.24425/ace.2024.148932
ARCHIVES OF CIVIL ENGINEERING
Vol. LXX ISSUE 1 2024

pp. 621 –634

Research paper

Design optimization of obstacle avoidance of intelligent building the steel bar by integrating reinforcement learning and BIM technology

Hong Chai[1], **Junchao Guo**[2]

Abstract: In promoting the construction of prefabricated residential buildings in Yunnan villages and towns, the use of precast concrete elements is unstoppable. Due to the dense arrangement of steel bars at the joints of precast concrete elements, collisions are prone to occur, which can affect the stress of the components and even pose certain safety hazards for the entire construction project. Because the commonly used the steel bar obstacle avoidance method based on building information modeling has low adaptation rate and cannot change the trajectory of the steel bar to avoid collision, a multi-agent reinforcement learning-based model integrating building information modeling is proposed to solve the steel bar collision in reinforced concrete frame. The experimental results show that the probability of obstacle avoidance of the proposed model in three typical beam-column joints is 98.45%, 98.62% and 98.39% respectively, which is 5.16%, 12.81% and 17.50% higher than that of the building information modeling. In the collision-free path design of the same object, the research on the path design of different types of precast concrete elements takes about 3–4 minutes, which is far less than the time spent by experienced structural engineers on collision-free path modeling. The experimental results indicate that the model constructed by the research institute has good performance and has certain reference significance.

Keywords: reinforcement learning, building information modeling, reinforcement steel bar, precast concrete elements, markov decision, BIM information

[1] MsC., Yellow River Conservancy Technical Institute, Department of Civil Engineering and Transportation Engineering, 475000 Kaifeng, China, e-mail: chaihong1977@126.com, ORCID: 0009-0008-1545-3500

[2] MsC., Yellow River Conservancy Technical Institute, Department of Civil Engineering and Transportation Engineering, 475000 Kaifeng, China, e-mail: m13476052810@163.com, ORCID: 0009-0009-9306-6909

1. Introduction

The steel bar can greatly affect the bearing capacity of prefabricated buildings, and the building structure collapse loss caused by the defect of the steel bar component is huge. Therefore, the planning and design of the steel bar route is an indispensable part in reinforced concrete structure and prefabricated component building placement. Due to the quantity and placement requirements of the steel bars in precast concrete elements (PCE), even if steel bars are arranged according to regulations, there may still be collisions and crowding problems between steel bars. However, most of the existing solutions to the steel bar conflict are based on building information modeling (BIM). However, algorithms based on BIM models can only match PCE with conventional shapes, and cannot adjust the position of steel bars to avoid obstacles [1]. To overcome the limitations of traditional design codes and construction, and realize the automatic arrangement and adjustment of the steel bar in a certain number of PCE in complex environments. The automatic steel bar obstacle avoidance (OA) model based on multi-agent reinforcement learning (MARL) and BIM is proposed, to complete the OA design of the steel bars. This research is divided into four parts. The first part is a description of relevant work in recent years. The second part is a collision-free framework based on MARL fusion of BIM. The third part is to test the model proposed by the research. The fourth part summarizes the results and draws conclusions and formulates future research directions.

2. Related words

Research on the design of OA using the MARL method has developed maturely. To solve the poor control rate and portability of traditional multi-agent formation OA algorithm, JI et al. proposed a method based on deep reinforcement learning (DRL), and proved the effectiveness of the proposed algorithm through simulation experiments [2]. Ang et al. proposed a game combination multi-agent depth determination strategy gradient method to optimize the trajectories of multiple unmanned aerial vehicles to solve the problem of uneven distribution of data size between ground users due to the heterogeneity of mobile tasks [3]. Through experiments, it was proven that the proposed algorithm can minimize unloading delay and improve the energy efficiency of unmanned aerial vehicles. Su et al. proposed a multi-agent modeling method to obtain an efficient cost strategy for serial production lines with multi-level design preventive maintenance actions, and verified its effectiveness through experiments [4]. Kaiwen et al. proposed a new distributed framework for multi-cell collaboration or competitive beamforming, designing limited information exchange schemes to improve global performance [5]. Yan et al. studied the problem of self-propelled fish swarm OA maneuver under intelligent control through numerical simulation, and proposed a hydrodynamic simulation method based on DRL and artificial intelligence control to help the potential application of bionic robot swarm in engineering [6]. Zhu et al. proposed a real-time robot anti-collision method to improve the overall quality and speed of human-machine cooperation engineering, learned direct control commands from the original depth image through the self-supervised reinforcement learning algorithm, and verified the effectiveness of its algorithm through experiments [7]. Cheng et al. proposed an algorithm of state feedback and output feedback control protocol to improve the robustness of the agent system to external interference, obtained inclusive conditions and conducted convergence analysis [8]. Nicholas et al. proposed an improved DRL controller model for decentralized anti-collision problem, and the effectiveness of the proposed model was proved through experiments in the 3D UAV environment [9]. Zhu et al. proposed a multi behavior critic reinforcement learning algorithm for autonomous underwater vehicle path-planning to solve the too slow rate of convergence of the model caused by the coupling of multiple behaviors in reinforcement learning, to overcome the problems of early training amplitude oscillation and low learning efficiency common in traditional actor critic algorithms [10]. Chen et al. proposed a robot dynamic OA path-planning algorithm based on DRL algorithm to avoid moving obstacles in the actual environment, and conducted real-time planning on the robot's formal path according to the comprehensive reward function of dynamic OA and target approaching, and verified the effectiveness of the algorithm in subsequent experiments [11]. Ejaz et al. proposed a collision-free autonomous navigation model based on DRL to achieve safe autonomous steering of tracked robots in various transformed environments, and verified the performance of the model in subsequent experiments [12]. Qingrui et al. proposed an algorithm model based on DRL to solve the intelligent tracking control of uncertain autonomous surface vehicles with anti-collision function, and verified the performance of the proposed model by comparing the constructed model with the traditional algorithm model [13].

To sum up, MARL method and BIM have been applied to path-planning problem more maturely, but there is less research on the steel bar collision problem. Based on this, a BIM-based on MARL is proposed to design the collision-free path of the steel bar in reinforcement components.

3. The steel bar OA path design based on MARL and BIM

This research mainly takes MARL as the framework, regards the steel bar as multi-agent, fuses BIM to construct barrier-free path and conduct three-dimensional display, and further solves the collision problem caused by the inability of the steel bar to adjust the track under traditional BIM.

3.1. The steel bar OA distribution design based on MARL structure

PCE, as the material foundation of building industrialization, are building components pre-made in factories using concrete as the basic material. General PCE includes beams, slabs, columns, and building decoration accessories. Compared to traditional on-site pouring construction methods, PCE can to some extent reduce pollution and protect the environment. PCE is generally connected by binding, welding, mechanical and cold pressing sleeves, etc. The structural diagram of PCE is shown in Fig. 1.

From Fig. 1, the components of PCE are divided into two types: steel bars that meet the bearing capacity and stirrups that meet the shear strength of the inclined section. According to

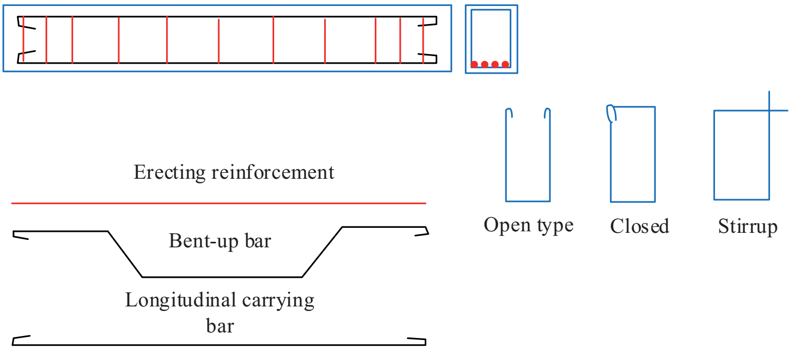


Fig. 1. Structural diagram of PCE

the practical application of steel bars and stirrups, they can be divided into categories such as longitudinal load-bearing steel bars, late-rising steel bars, erected steel bars, open stirrups, and closed stirrups. Research innovatively uses the steel bars as multi-agent systems, and designs corresponding OA routes for the steel bars through multi-agent path-planning. MARL can consider multiple goals at the same time to improve the decision, and consider the rejection of extreme deviations on some goals. Balanced decision-making makes it more reasonable. MARL systems are also often used in the design of path-planning problems. In the interaction between agents and the environment, the estimated values of future feedback are gradually derived based on the rewards or punishments of action feedback in the state [14]. The concrete modeling in the interaction between the environment and agents can be completed using Markov decision process (MDP), and the predicted values of expected rewards of MDP are represented by value functions, the value function calculates the total future cumulative reward for the behavior in the current state. Through the value function, the agent can focus more on maximizing the overall benefits in the long run. The calculation method is shown in Eq. (3.1).

$$Gt = \sum_{t=1} \varepsilon^k R_{t+1+k} \tag{3.1}$$

In Eq. (3.1), Gt represents cumulative rewards; ε means a discount factor, mainly used to control the impact of future rewards on the current agent's decision-making; R_t denotes the reward obtained by the agent at the FF moment after taking action a, and at the $t + 1$ moment. After exploratory learning, the agent learns the optimal behavior strategy and maximizes the cumulative reward. When the cumulative reward no longer changes, the reward function converges.

The MARL structure is divided into three levels: state, action, and reward. Let the input value of the state layer be $s = (s_1, s_2, \ldots s_n)$, the input value of the behavior layer be $a = (a_1, a_2, \ldots a_n)$, and the input value of the reward layer be r. It uses the Q function to guide the state and action of the agent and delimits the expected reward Q value corresponding to the state and action of the agent in the Q-table. After random assignment, it updates the $Q(s, a)$ with the help of the Bellman equation. The relevant expression is shown in Eq. (3.2).

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \eta[r_{t+1} + \gamma \max_{a_{t+1}} Q_t(s_{t+1}, a_{t+1})| - Q_t(s_{t+1}, a_{t+1})] \tag{3.2}$$

In Eq. (3.2), under the premise of selecting a action in a status, $Q_{t+1}(s_t, a_t)$ means the updated reward expectation; $Q_t(s_t, a_t)$ indicates the corresponding original reward expectation; $r + 1$ refers to the return reward;ηdenotes the Learning rate; γ stands for the decay rate; $\max_{a_{t+1}} Q_t(s_{t+1}, a_{t+1})$ indicates the max expected reward for all selectable a_{t+1} actions in the updated s_{t+1} state. In the automatic OA design of steel bars, MARL utilizes multiple intelligent agents to plan paths from one end of the beam or column to the other end, and navigate the starting and ending points of the driving path [15]. The relevant intelligent definition operation principle is shown in Fig. 2.

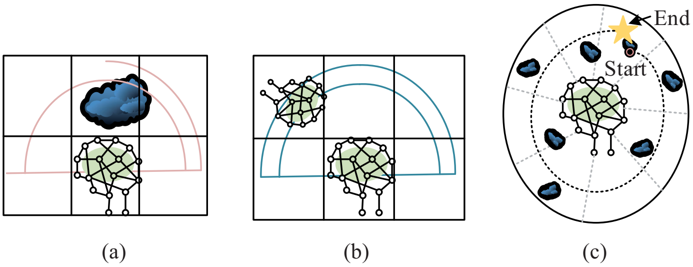


Fig. 2. Principle of intelligent OA design for MARL: Obstacle identification, b) Intelligent agent detection, c) Target detection

In Fig. 2, the five-star position is the endpoint, the circular position is the starting point, and the target recognition environment is three-dimensional space. In the case of obstacles surrounding, it is not possible to directly design a route from the starting point to the endpoint, so the virtual graph line is the optimal path-planning for the intelligent agent.

3.2. Rasterisation processing of the steel bar OA distribution design environment of BIM+MARL model

When using MALR to deal with the steel bar design path, it is necessary to convert the PCE from the BIM to the rasterisation model, and use the geometric information and boundary location to calculate the corresponding path [16]. Let the grid size in the rasterisation environment be the same, and the side length calculation of each small grid is Eq. (3.3).

$$D_i = \max(d_c + d_t) \tag{3.3}$$

In Eq. (3.3), D_i denotes the length of the grid edge; d_c refers to the diameter of the longitudinal compressive the steel bar in the beam or column; d_t indicates the diameter of the longitudinal tensile the steel bar in the beam or column. Using the size of PCE and the small grid edge length, the entire grid environment edge length is constructed [17]. The relevant function expression is shown in Eq. (3.4).

$$S_z = floor\,(D/D_i) \tag{3.4}$$

Eq. (3.4), S_z refers to the side length of grid environment, D represents the side length of reinforced concrete construction, and $floor(.)$ denotes the Floor and ceiling functions.

It constrains the edge length of the grid environment through the *floor*(.) function. When designing PCE, it is necessary to consider the influence of three variables: the cross-sectional area of longitudinal tensile steel bars, the cross-sectional area of longitudinal compressive steel bars, and the cross-sectional area of stirrups on it. The longitudinal tensile steel bar is represented by the symbol A_s, and the A_s calculation method is shown in Eq. (3.5).

$$A_s = \sum_{i=1}^{N_t} \frac{\pi \cdot d_{t,i}^2}{4} \tag{3.5}$$

In Eq (3.5), $d_{t,i}$ denotes the diameter of the i longitudinal tensile steel bar in the reinforced concrete component. N_t nindicates the total number of tensile steel bars, and the range of values for the number of tensile steel bars is shown in Eq. (3.6).

$$\begin{cases} N_{t,\min} = \dfrac{b-2c}{s_{ht,\max}} \\ N_{t,\max} = \dfrac{b-2c}{s_{ht,\min}} \end{cases} \tag{3.6}$$

In Eq. (3.6), b refers to the width of the reinforced concrete beam; c expresses the thickness of the concrete protective layer; $s_{ht,\max}$ denotes the maximum spacing between longitudinal tensile steel bars; $s_{ht,\min}$ indicates the minimum spacing between longitudinal tensile steel bars. Due to the consistent calculation of the cross-sectional area of longitudinal compressive steel bars and longitudinal tensile steel bars, the study will not repeat it. The character A_{sv} is used to represent the cross-sectional area of the stirrup, and its calculation method is shown in Eq. (3.7).

$$A_{sv} = n_s \cdot A_{sv1} = n_s \cdot \frac{\pi \cdot d_{s,i}^2}{4} \tag{3.7}$$

In Eq. (3.7), n_s means the amount of stirrups in a certain section of the reinforced concrete component, and $d_{s,i}$ denotes the diameter of the ith stirrup. The research focuses on designing collision-free steel OA routes, and thus sets up 15 structural components in the MARL framework. The 15 types of PCE are mainly generated by combining three variables: beam, column, and beam nodes of different shapes. Regarding the analysis of structural forces, the research mainly utilizes the structural calculation software PKPM for operational analysis. By calculating the structural forces, the corresponding steel bar bending moment, shear force, axial force, torque, and the number and type of longitudinal bars and hoops are obtained. By integrating the above description, a design framework for automatic OA of steel bars using MARL and BIM is obtained. The schematic diagram of the framework structure is shown in Fig. 3.

From Fig. 3, according to the calculated component coordinate data and geometric data information, MARL will generate the corresponding three-dimensional coordinate information of collision-free steel bar design, and then use the three-dimensional coordinates of BIM-built steel bar components to represent the rasterisation data information. Q-learning is used to select the selected action in the rasterisation environment and calculate the expectation obtained under

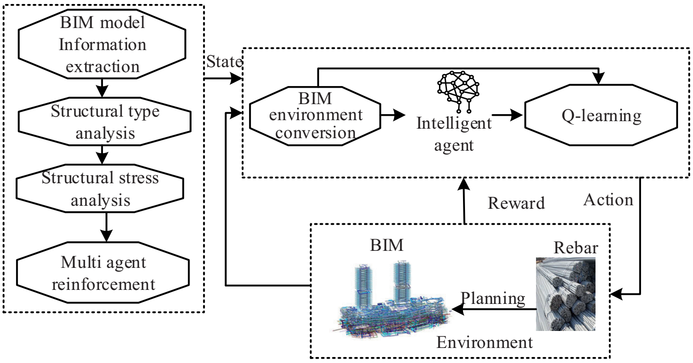

BIM model Information extraction
Structural type analysis
Structural stress analysis
Multi agent reinforcement
State
BIM environment conversion
Intelligent agent
Q-learning
Reward
Action
BIM
Rebar
Planning
Environment

Fig. 3. BIM+MARL Algorithm for intelligent the steel bar OA design model

the action, and select the action with maximum benefit according to the stored Q value [18]. The calculation method for action expectation is shown in Eq. (3.8).

$$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)] \tag{3.8}$$

In Eq. (3.8), $Q(s,a)$ indicates the estimated income of a action taken in s state; γ means the attenuation value of future rewards; α denotes the learning rate, and $Q(s',a')$ is the actual income of a action taken in s state.

4. The application of automatic OA design for steel bars based on MARL+BIM in practice

Prefabricated residential buildings in villages and towns have also developed into one of the commonly used forms of housing in rural areas. At present, prefabricated structures are mainly divided into three types: prefabricated steel, concrete and prefabricated wood structures. The research mainly focused on the automatic OA of steel bars under prefabricated concrete structure residential buildings in Dali Village, Yunnan Province. The design experiment verified the performance of MARL. In the experiment, it set the learning rate α of MARL to 0.05, the attenuation coefficient γ to 0.7, and the initial value of Q-Table to 0. The output of visual fitting results for the MARL model trained 3000 times is shown in Fig. 4.

In Fig. 4(a), from the starting point to the vicinity of the obstacle position, the output value of the MARL model was always 0, and the direction of the value 0 was specified as facing forward, which met the expected direction of the intelligent agent. From Fig. 4(b), the x-axis also represented the location environment of the intelligent agent, while the y-axis denoted the travel speed of the intelligent agent. From the output values of the MARL model, the intelligent agent has basically grasped the speed of travel: at the starting point, the speed of the intelligent agent would be faster, and near the obstacle, the speed of the intelligent agent would be slower. At this moment, the speed representation of the intelligent agent would not show a smooth

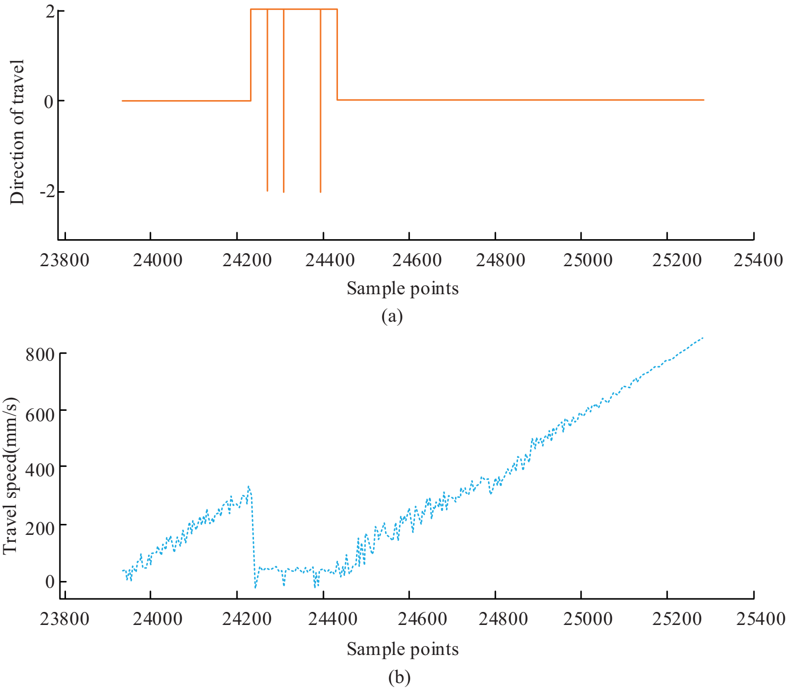


Fig. 4. Visualization fitting results of trained MARL model: a) Discrete actor, b) Continuous actor

curve, but would show a small range of fluctuations in the curve, where the amplitude of the speed curve would be inversely proportional to the distance of the obstacle. By visualizing the path and driving speed of the agent and the location of the agent, the MARL model's handling of obstacles could be intuitively observed, and the performance of the model could be judged. The trained MARL model was combined with BIM to conduct the steel bar crash test. The experimental results are shown in Fig. 5.

From Fig. 5, the BIM+MARL model was capable of reinforcing the y-axis stacked beam, raising it to a certain distance, and interleaving it with the X-axis stacked beam while ensuring the original direction. This ensured that the bottom steel bar with added concrete protection layer could avoid all obstacles after a single bend. After testing the OA performance of the model for beam-shaped steel bars, it designed experiments to verify the OA performance of the model under different types of steel bars. Three typical beam-column nodes, namely the cross-shaped node with bent bottom the steel bar node area, the T-shaped node, and the cross-shaped node with bent the steel bar end, were selected as the research objects to verify the performance of the three-dimensional beam-column node the steel bar OA route generated by the BIM+MACL model proposed by the research institute. The performance of BIM on the steel bar OA routes on three typical beam-column nodes was compared. The experimental results are shown in Table 1.

From Table 1, the average probability of non-reinforced collision in the BIM+MARL model constructed by the research institute was 98.45%. It suggested that the model constructed

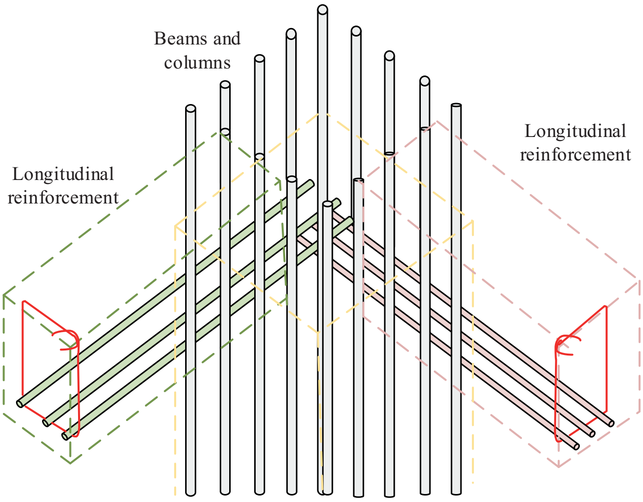


Fig. 5. Crash test results of the steel bar

Table 1. Comparison of two models on collision-free behavior of the steel bar in PCE

Non collision rate of steel bars (%)	Beam-column joints	MARL+BIM	BIM
Cross shaped node with bent bottom the steel bar node area	Type I Cross 1	98.32	82.54
	Type I Cross 2	98.26	83.63
	Type I Cross 3	97.54	81.29
	Type I Cross4	99.33	83.34
	Type I Cross5	98.78	85.62
Cross shaped nodes with bent ends of steel bars	Type II cross1	97.95	83.27
	Type II cross2	98.26	86.34
	Type II cross3	98.82	85.62
	Type II cross4	98.69	84.29
	Type II cross5	98.22	88.37
T-shaped cross	T-shaped cross1	98.77	79.68
	T-shaped cross2	97.68	82.34
	T-shaped cross3	99.36	81.15
	T-shaped cross4	98.26	80.34
	T-shaped cross5	99.02	82.09

by the research institute could increase the probability of the steel bar avoidance by 15.16% on the basis of the BIM model. In the cross-shaped beam-column node with bent steel bars at the end, the average collision probability without steel bars in the BIM+MARL model was 98.39%, and the minimum collision avoidance probability of steel bars in the cross-shaped beam-column node with bent steel bars at the end was 97.95%. However, the average collision avoidance rate of steel bars in the BIM model was 85.58%, which was 12.81% lower than the steel bar avoidance rate in the BIM+MARL model. Similarly, in the T-shaped cross-beam column node, the average probability of the steel bar collision avoidance for BIM+MARL was 98.62%, and the BIM model was 81.12%. Compared with the time spent on manual modeling by 40 experienced structural engineers, the experimental results are shown in Fig. 6.

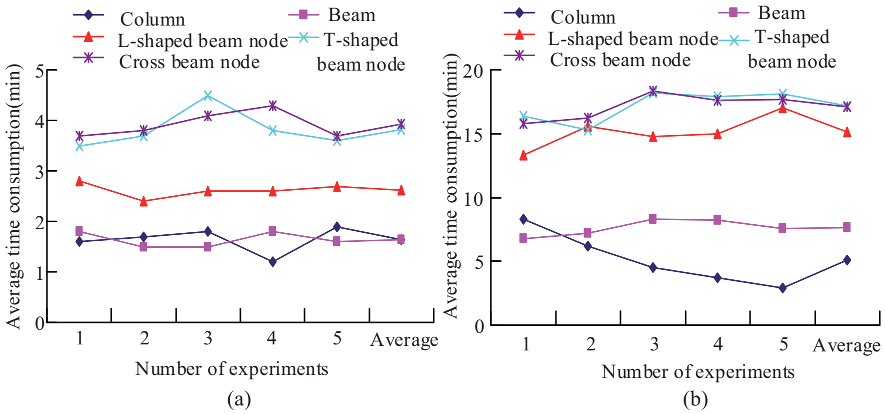
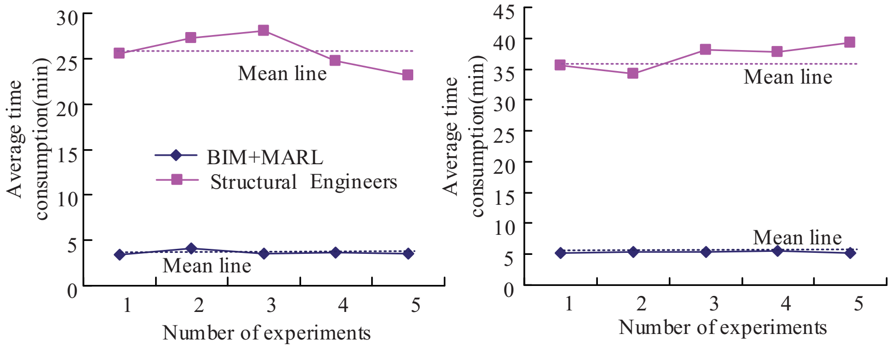


Fig. 6. Comparison of collision-free path generation time: a) The path time of BIM+MARL model on different types, b) Modeling time of 40 structural engineer

From Figs. 6(a) and 6(b), as the complexity of node structure types increased, the time required for both the model and manual processing of nodes would increase. Among them, the BIM+MARL model took an average time of 1.64 min to generate the barrier-free path of the column, which was 3.48 min less than the average time of structural engineers modeling the same column node. In terms of beam detection time, the average time for BIM+MARL model Generative model was 1.64 min, and the average time for manual modeling was 7.2 min. On the same index, the average modeling time of structural engineers was 12.52 min, 13.56 min, and 13.20 min more than that of BIM+MARL model. After checking the OA path design time of the model for relatively simple beam-column joints, further design experiments verified the time consumed by the BIM+MARL model for the OA path design of complex beam-column joints and compared the time consumed by structural engineers for collision-free modeling of the same complex nodes. The experimental results are shown in Fig. 7.

From Fig. 7(a), for the middle layer single beam and single column nodes, the overall time fluctuation of the BIM+MARL model constructed by the research institute was relatively small, and the average time for designing the steel bar OA path was 3.64 minutes. For structural engineers, the time required for designing collision-free the steel bar path modeling fluctuates greatly, among which the shortest time required for modeling was 23.2 min, the longest time



(a) (b)

Fig. 7. Comparison of generation time for collision-free paths of complex nodes: a) Time consuming comparison between model and structural engineer on single beam and single column joints in the middle floor, b) Time consuming comparison between model and structural engineer for *L*-shaped beam column joints in the middle floor

was 28.1 min, and the average time required for modeling was 25.8 min. The average modeling time of BIM+MARL model was 22.16 min lower than that of structural engineers. Fig. 7(b) shows the comparison of the design time consumption of OA path for *L*-shaped beam-column joints in the middle layer. The average time consumption of BIM+MARL model was 5.32 min, and the average time required for structural engineers to model was 37 min, which was 31.68 min higher than that of BIM+MARL model. The design experiment verified the time required for OA path-planning of the middle layer reverse beam-column joints, middle layer *T*-shaped beam-column joints and middle layer cross-shaped beam-column joints composed of 32, 44, and 44 longitudinal the steel bar in the model, and also compared the time required for structural engineers to model OA under the same experimental object. The experimental results are shown in Fig. 8.

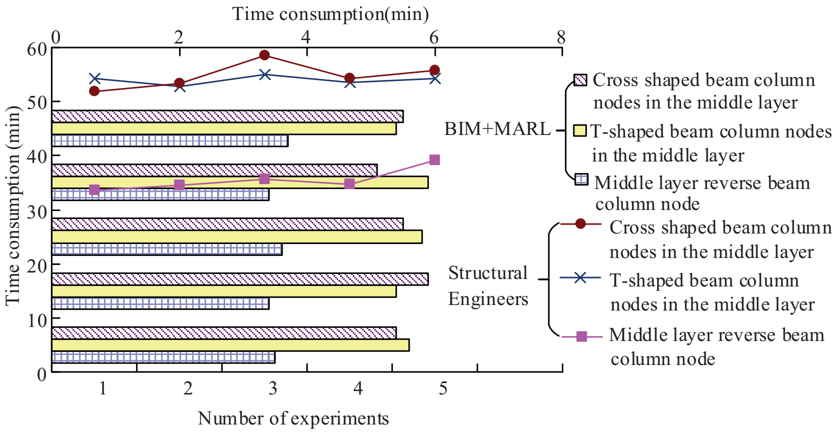


Fig. 8. Comparison of model and engineer time consumption

From Fig. 8, the modeling time of the model was significantly lower than that of engineers. For the OA path design of the reverse beam-column node in the middle layer, the BIM+RACL model took an average of 3.52 minutes, while the engineer modeling took an average of 35.5 minutes.In the design of OA paths for cross-shaped beam-column nodes in the middle layer, the BIM+RACL model took 39.38 minutes less than the engineer's modeling time, with an average OA path design time of 15.38 minutes. The experimental results indicated that the BIM+RACL model constructed by the research institute had good performance in designing OA paths for different types of reinforced component beam-column nodes.

5. Conclusions

When the traditional manual method or BIM method is used to deal with the problem of the steel bar collision in PCE, it can only adapt to specific objects and is limited to the actual construction situation. To achieve the requirement of adjusting the position of the steel bar to avoid obstacles, a MARL algorithm framework based on BIM was proposed. By treating the steel bar as an agent of reinforcement learning, the steel bar OA design was converted into a path-planning problem, to solve the problem of the steel bar collision in PCE. The experimental results showed that the MARL model fused with BIM could efficiently design OA routes for column shaped beam the steel bar. The OA rates for the design of typical three types of beam-column node the steel bar components were 98.45%, 98.62%, and 98.39%, respectively. In the comparison of the design time consumption of OA path for different types of the steel bar nodes, the average time consumption of the BIM+MARL model for column and column collision-free path design was 1.64 min, which was 3.48 min and 5.56 min lower than the modeling time of structural engineers, respectively. The time for the model to generate collision-free paths on L-shaped beam nodes, T-shaped beam nodes, and cross-shaped beam nodes was 2.62, 3.82, and 3.92 minutes, respectively. Compared with the design time of the collision-free modeling path of the structural engineer, the average time of the BIM+MARL model in the design of the OA path of the reverse beam-column node in the middle layer was 32 min lower than the average time of the engineer in modeling, 48.38 min lower than the design time of the OA path of the T-shaped beam-column node in the middle layer, and 39.38 min lower than the design time of the OA path of the cross-shaped beam-column node in the middle layer.

6. Future work

The results of this study are: 1. The problem of collision at steel joints in precast concrete members was solved. 2. The path designed by the constructed BIM + MARL model could effectively reduce the reinforcement collisions in PCE. 3. The BIM + MARL model constructed by the institute could greatly shorten the generation time of the OA path. 4. In the generation of accessible path of complex beam and column joints, the performance of BIM + MARL model was overall better than the traditional model.

Research shortcomings: This study only considered cast-in-place reinforced concrete components under regular geometry, and did not involve other cast-in-place components, so the comprehensiveness of the experiment needs to be improved.

Future expectations: (1) In the experimental test link, the feasibility of the experimental test is improved by adding the standard requirements of reinforcement. (2) In the experimental test, it needs to add the stress test and OA path generation test for different pouring components. (3) In the actual production and application, the research results should be appropriately combined to provide a scientific basis for the construction and production.

References

[1] J. Wang, "Optimized mathematical model for energy efficient construction management in smart cities using building information modeling", *Strategic Planning for Energy and the Environment*, vol. 41, no. 1, pp. 61–79, 2022, doi: 10.13052/spee1048-5236.4113.

[2] X.K. Ji, J.T. Hai, W.G. Luo, C.X. Lin, Y. Xiong, Z.K. Ou, and J.Y. Wen, "Obstacle avoidance in multi-agent formation process based on deep reinforcement learning", *Journal of Shanghai Jiao Tong University*, vol. 26, no. 5, pp. 680–685, 2021, doi: 10.1007/s12204-021-2357-6.

[3] A. Gao, Q. Wang, W. Liang, and Z.G. Ding, "Game combined multi-agent reinforcement learning approach for UAV assisted offloading", *IEEE Transactions on Vehicular Technology*, vol. 70, no. 12, pp. 12888–12901, 2021, doi: 10.1109/TVT.2021.3121281.

[4] J.Y. Su, J. Huang, S. Adams, Q. Chang, and P.A. Beling, "Deep multi-agent reinforcement learning for multi-level preventive maintenance in manufacturing systems", *Expert Systems with Application*, vol. 192, no. 4, pp. 62–74, 2022, doi: 10.1016/j.eswa.2021.116323.

[5] K.W. Yu, G. Wu, S.Q. Li, and G.Y. Li, "Local observations-based energy-efficient multi-cell beamforming via multi-agent reinforcement learning", *Journal of Communication and Information Networks*, vol. 7, no. 2, pp. 170–180, 2022, doi: 10.23919/JCIN.2022.9815200.

[6] L. Yan, X.H. Chang, N.H. Wang, R.Y. Tian, L.P. Zhang, and W. Liu, "Learning how to avoid obstacles: A numerical investigation for maneuvering of self-propelled fish based on deep reinforcement learning", *International Journal for Numerical Methods in Fluids*, vol. 93, no. 10, pp. 3073–3091, 2021, doi: 10.1002/fld.5025.

[7] X.J. Zhu, Y.H. Liang, H.X.X. Sun, X.Q. Wang, and B. Ren, "Robot obstacle avoidance system using deep reinforcement learning", *Industrial Robot*, vol. 49, no. 2, pp. 301–310, 2022, doi: 10.1108/IR-06-2021-0127.

[8] C.J. Xu, B.F. Li, Y. Yuan, "Disturbance observer-based robust formation-containment of discrete-time multiagent systems with exogenous disturbances", *Complexity*, vol. 2021, art. no. 5525067, 2021, doi: 10.1155/2021/5525067.

[9] N. Thumiger and M. Deghat, "A multi-agent deep reinforcement learning approach for practical decentralized UAV collision avoidance", *IEEE Control Systems Letters*, vol. 6, pp. 2174–2179, 2022, doi: 10.1109/LCSYS.2021.3138941.

[10] Z. Wang, S.W. Zhang, X.N. Feng, and Y.C. Sui, "Autonomous underwater vehicle path planning based on actor-multi-critic reinforcement learning", *Proceedings of the Institution of Mechanical Engineers, Part I. Journal of Systems and Control Engineering*, vol. 235, no. 10, pp. 1787–1796, 2021, doi: 10.1177/0959651820937085.

[11] P. Chen, J. Pei, W. Lu, and M. Li, "A deep reinforcement learning based method for real-time path planning and dynamic obstacle avoidance", *Neurocomputing*, vol. 497, no. 8, pp. 64–75, 2022, doi: 10.1016/j.neucom.2022.05.006.

[12] M.M. Ejaz, T.B. Tang, and C.K. Lu, "Vision-based autonomous navigation approach for a tracked robot using deep reinforcement learning", *IEEE Sensors Journal*, vol. 21, no. 2, pp. 2230–2240, 2021, doi: 10.1109/JSEN.2020.3016299.

[13] Q.R. Zhang, W. Pan, and V. Reppa, "Model-reference reinforcement learning for collision-free tracking control of autonomous surface vehicles", *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 8770–8781, 2022, doi: 10.1109/TITS.2021.3086033.

[14] J. Sobieraj, D. Metelski, and P. Nowak, "PMBoK vs. PRINCE2 in the context of Polish construction projects: Structural Equation Modelling approach", *Archives of Civil Engineering,* vol. 67, no. 2, pp. 551–579, 2021, doi: 10.24425/ace.2021.137185.

[15] Y.M. John, A. Sanusi, and I.R. Yusuf, "Reliability analysis of multi-hardware–software system with failure interaction", *Journal of Computational and Cognitive Engineering*, vol. 2, no. 1, pp. 38–46, 2023, doi: 10.47852/bonviewJCCE2202216.

[16] C. Hao, J.J. Xia, L. Wu, M. Xiao, "Reinforcement layout design of three-dimensional members under a state of complex stress", *Archives of Civil Engineering*, vol. 69, no. 1, pp. 421–436, 2023, doi: 10.24425/ace.2023.144181.

[17] P.A. Ejegwa and J.M. Agbetayo, "Similarity-distance decision-making technique and its applications via intuitionistic fuzzy pairs", *Journal of Computational and Cognitive Engineering*, vol. 2, no. 1, pp. 68–74, 2023, doi: 10.47852/bonviewJCCE512522514.

[18] P. Szeptyński and L. Mikulski, "Preliminary optimization technique in the design of steel girders according to Eurocode 3", *Archives of Civil Engineering*, vol. 69, no. 1, pp. 71–89, 2023, doi: 10.24425/ace.2023.144160.

Received: 2023-08-07, Revised: 2023-11-07