Automation in Construction 112 (2020) 103062

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Automated clash resolution for reinforcement steel design in concrete frames via Q-learning and Building Information Modeling

Jiepeng Liu[a,c], Pengkun Liu[a,2,*], Liang Feng[b,1,**], Wenbo Wu[b], Dongsheng Li[a], Y. Frank Chen[a]

[a] School of Civil Engineering, Chongqing University, Chongqing, China
[b] College of Computer Science, Chongqing University, Chongqing, China
[c] Chongqing University Industrial Technology Research Institute, Chongqing University, Chongqing, China

ARTICLE INFO

Keywords:
Building Information Modeling
Reinforcement learning
Multi-agent
Q-learning
Rebar design
Clash resolution
Reinforced concrete frame

ABSTRACT

The design of reinforcing steel bars (rebars) is critical to reinforced concrete (RC) structures. Generally, a good number of rebars are required by a design code, particularly at member connections. As such, rebar clashes (i.e., collisions and congestions) would be inevitable. It would be impractical, labor-intensive, and error-prone to avoid all possible clashes manually or even using standard design software. The building information modeling (BIM) technology has been utilized by the present architecture, engineering, and construction (ACE) industry for clash-free rebar designs. However, most existing BIM-based approaches offer the clash resolution strategy for moving components with an optimization algorithm, and are only applicable to the RC structures with regular shapes. In particular, the optimized path of rebars cannot be adjusted to avoid the obstacles, thus limiting the practical applications. Furthermore, most existing studies lack the learning from design code and constructibility constraints to realize automatic and intelligent arrangement and adjustment of rebars for avoiding the obstacles encountered in complex RC joints and frame structures. Considering these shortcomings, the authors have recently proposed an immediate reward-based multi-agent reinforcement learning (MARL) system with BIM, towards automatic clash-free rebar designs of RC joints without clashes. However, as the immediate reward is required in the MARL system for guiding the learning of a rebar design, it will not succeed in clash-free rebar designs of complex RC structures where immediate reward is often unavailable. In this study, this study further extends the previous work with Q-learning (a model-free reinforcement learning algorithm) for more realistic path planning considering both immediate and delayed rewards in clash-free rebar designs for real-world RC structures. In particular, the rebar design problem is treated as a path-planning problem of multi-agent system, where each rebar is deemed as an intelligence reinforcement learning agent. Next, by employing the Q-learning as the reinforcement learning engine, the particular form of state, action, and immediate and delayed rewards for the reinforcement MARL for automatic rebar designs considering more actual constructible constraints and design codes can be developed. Comprehensive experiments on three typical beam-column joints and a two-story RC building frame were conducted to evaluate the efficiency of the proposed method. The study results of paths of rebar designs, success rates, and average time confirm that the proposed framework with MARL and BIM is effective and efficient.

1. Introduction

Recently, building information modeling (BIM) has been widely used to reduce the errors in the building life cycles [1], to prevent construction wastes [2], and to improve the communication efficiency for all stakeholders [3] in the Architecture, Engineering, and Construction (AEC) industry [4]. Specifically, BIM has changed construction work routines by spawning new job positions and reorganizing work procedures. Furthermore, a Construction project consists of thousands of components that spatially depend on each other. One of the key activities during design is to coordinate the layout of these components. Among many complex activities, BIM has been

* Corresponding author.
** Corresponding author.
E-mail addresses: pengkunliu@cqu.edu.cn (P. Liu), liangf@cqu.edu.cn (L. Feng).
[1] Main Corresponding Author.
[2] Co-Corresponding Author.

https://doi.org/10.1016/j.autcon.2019.103062
Received 23 April 2019; Received in revised form 10 December 2019; Accepted 19 December 2019
Available online 31 January 2020

increasingly used for design coordination and one of its most widely used applications is clash detection and resolution.

Clash detection and resolution for reinforcing steel bars (rebars) design are critical to reinforced concrete (RC) structures. According to the Chinese design codes for RC members (GB50010-2010 [5], and GB 50011-2010[6]), the rebar design must meet the seismic and bearing capacity requirements of RC members. Besides, the rebar design has to be safe, constructible and cost-effective. Rebars are usually populated at member joints and must meet the restrictive code requirements. It would be impractical, labor-intensive and error-prone for a designer to avoid all collisions (hard clash) or congestions (soft clash) of rebars by manually using a trial-and-error approach, even with the aid of computer software [7, 8]. Ideally, any potential spatial clash of rebars can be identified and resolved before construction execution. However, current computer software including Autodesk Robot Structural Analysis Professional [9], CSI ETABS [10], PKPM [11], and YJK [12] can only calculate the rebar areas based on the rebar design. They all require a large amount of manpower to arrange the rebars to avoid spatial clashes as required by the design code GB50010-2010 [5]. Moreover, although some clash detection computer programs like Autodesk Navisworks Manage [13] and Solibri Model Checker [14] realize the detection and visualization of the clash members [15], they mainly focus on the clash identifications of structural members beyond the design stage. They cannot automatically avoid rebars clashes and offer an implementation resolution for the automatic arrangement of clash-free rebars.

With the popularization of BIM, the use of three-dimensional (3D) models to detect clashes is attracting wide attention due to the advantage of BIM such as high efficiency in spatial processing and visualization. BIM technology allows us to represent the rebar details digitally and transfer the detailed information to a structural analysis software [4]. However, existing BIM software packages do not provide automated resolution for rebar clashes. In the last decades, various researchers have tried to solve the issues of the clash detection and resolution with the aid of BIM technology, such as improving modeling accuracy or reorganizing model structure from modeling aspects [16, 17], improving clash detection algorithms [4, 18-20], and using historical data to improve clash detection from a knowledge management perspective [15, 21, 22]. The main drawbacks of existing approaches can be summarized as follows: (1) Due to the complex rebars as required by a design code, constructible constraints and design code requirements cannot be intelligently learned and stored; and (2) Most previous studies lack the automatic and intelligent arrangement and adjustment of rebars for avoiding the obstacles in real-world complex RC joints and frame structures.

In machine learning, Reinforcement Learning (RL) algorithms have achieved many important achievements in the field of complex adaptive systems such as mobile robot path planning [23]. Reinforcement learning (RL) creates an autonomous agent that learns and then adjusts its behavior through the action feedback (punishment and reward) from the environment, instead of explicit teaching. Following the framework of a Markov decision process (MDP), a RL agent performs learning through the cycle of sense, action, and learning. Furthermore, Q-Learning is a model-free reinforcement learning algorithm. The goal of Q-learning is to learn a policy, which tells an agent what action to take under different circumstances. “Q” denotes the function that returns the reward used to provide the reinforcement and stands for the “quality” of an action taken in a given state.

Recently, inspired by the similarity between the path planning of multi-agents and the arrangement of rebars, the authors have proposed a framework towards automatic clash-free rebar designs of RC joints using the multi-agent reinforcement learning (MARL) system and BIM. Each agent has a FALCON (fusion architecture for learning, cognition, and navigation) [24] RL architecture with immediate rewards. Furthermore, the clash detection and resolution problems for rebar designs can be treated as a path planning of multi-agents in order to achieve an automatic arrangement and adjustment of rebars to avoid obstacles. The authors previous work has progressed to achieve a clash-free rebar design in RC beam-column joints and one-story frames. However, it should be noted that the RL engine of FALCON [24] relies on the immediate reward obtained after performing each action. Since targets of agents may be blocked or invisible in the real world, it is unrealistic to obtain immediate reward after taking each action. Due to the huge search space of real complex RC members and the complicated rebar arrangement details, MARL may fail to find a feasible solution where the immediate reward is often not available for clash-free rebar designs.

With the above issues in mind, in this study, the authors further extend their work with Q-learning for path planning considering both immediate and delayed rewards for more realistic rebar designs according to more complex physical constraints and design codes of RC structures. The inadequacy of relying on immediate rewards in FALCON [24] may lead to the failure of finding feasible solutions for rebar design in real-world complex RC structures. By using Q-learning, the extended framework is able to overcome the inadequacy and achieve stable performance in a much faster pace than FALCON [24]. The specific objectives of this study are (1) to overcome the inadequacy of relying on immediate rewards and failing to find feasible solutions for real-world complex RC structures; (2) by employing the Q-learning method as the RL engine, to design the particular form of state, action, and rewards for the reinforcement MARL according to physical constraints; (3) to simulate and validate the clash-free rebar design results of the proposed MARL on a two-story RC building frame.

The following of this paper is organized as follows: Section 2 provides a brief review of related works in the literature. Section 3 presents the problem formulation of rebar design in RC frames and the motivation of the proposed RL for rebar design. The rebar spacing demand for RC members and the framework via the proposed MARL with BIM are also discussed in this section. Section 4 provides the details of solving the path planning problem for rebar designs via MARL. Section 5 reports the experiments and discuss the simulation results. Conclusions and future work are discussed in Section 6.

2. Preliminary

Concerning the clash problem of rebar designs of RC joints in 3-D space, Section 2.1 presents the research background about the clash identification and resolution problem. Section 2.2 describes the introduction of RL, and Section 2.3 introduces the robot navigation task in RL.

2.1. Research background

In order to expand the capacity of BIM technology for the clash detection and resolution of automated designs, Zhang and Hu [19] proposed a new approach for conflict and safety analysis during construction through the integration of construction simulation, four-dimensional construction management, and safety analysis. However, their approach does not offer implementation details for solving clashes. With the work breakdown structures, Gijezen [17] improved the efficiency of the clash detection process. Helm et al. [16] classified clash detections into four classes: (1) shapes comparison, (2) axis-aligned bounding boxes comparison, (3) ray-triangle intersection, and (4) industry foundation classes (IFC) structure method. However, all of these methods focused only on clash detection with no resolution offered. Park [20] developed a BIM-based simulator to determine the sequence of rebar placement and the clashes of rebars were identified using a previously-developed application programming interface previously. When the distance between the center lines of the rebars was smaller than the rebar diameter, the collision was automatically identified. Nevertheless, previous studies focus on the simulation of the rebar placement sequence, and slove the spatial clash problem manually. Moreover, Wang and Leite et al. [15] developed a knowledge

representation for conflicting spatial coordinates of MEP systems, which includes description, context, evaluation and management details. However, the developed representation pattern provides only documentation to store the clash-based information with no clash resolution strategy at all for identifying clashes. Radke et al. [18] proposed an automated identification and resolution for mechanical, electrical and plumbing (MEP) systems, where the offered resolution is to move one of the two clash entities to solve spatial conflicts. However, design constraints are not verified after moving one object. Besides, their resolution method is manual and is limited to certain types of clashes. Mangal and Cheng [4] proposed a framework based on the BIM and GA to realize rebar designs, which can avoid the clashes at RC beam-column joints. When the total number of rebars at a RC beam-column joint exceeds the maximum allowed number allowed in both directions (x and y) as specified in BS8110 [25], the clashes were identified. However, the proposed framework only offers the clash resolution strategy for moving components by GA and is applicable to regular-shaped RC structures only. In particular, the optimized path of rebars is not adjustable to avoid the obstacles, thus limiting its practical applications. The above-mentioned studies use the conventional methods of moving one of the two clash entities to solve spatial clashes by optimization methods with no implementation details for solving clashes offered. Moreover, to the best knowledge of the authors, no study has been done to investigate the full automation in the identification and resolution of rebar clashes of real complex RC joints and frame structures. Previous studies also lack the function of intelligent and gradual learning and the implementation of design codes and constructible constraints of rebars.

Therefore, the objective of the current research is to develop a framework based on Q-learning and BIM for achieving automatic rebar designs of actual complex RC frames. Furthermore, the clash detection and resolution problem of rebar designs can be treated as a path planning of multi-agents in order to achieve the automatic arrangements and adjustment of rebars to avoid obstacles.

2.2. Reinforcement learning

Reinforcement learning (RL) [26] is a natural learning paradigm to both single-agent and multiple-agent (Fig. 1). It creates an autonomous agent that learns and then adjusts its behavior through the action feedback (penalty and reward) from the environment, instead of explicit teaching. Following the framework of a Markov decision process (MDP), a RL agent performs learning through the cycle of sense, action, and learning [26]. In each cycle, the agent obtains the sensory input from its environment representing the current state (S) [24]. Depending on the current state, its knowledge, and goals, the system selects and performs the most appropriate action (A) [24]. Through receiving the feedback in terms of rewards (R) from the environment, the agent learns to adjust its behavior in the motivation of receiving positive rewards and avoid penalties in the future. Classical approaches of the RL involve learning one or both of the following functions, namely, *policy function*, which maps each state to a desired action and *value function*, which associates each pair of state and action to a utility

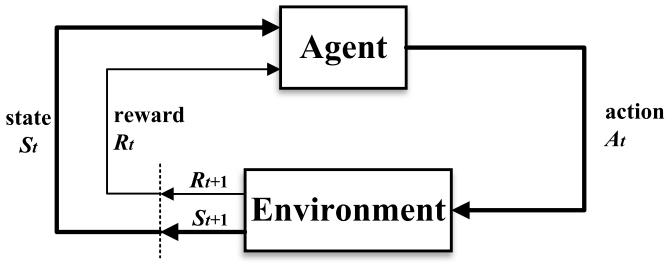

Fig. 1. Basic module of reinforcement learning.

value [27]. A popularly used method for learning value function is Q-learning [28, 29], which is a temporal-difference (TD) method to estimate the accumulative future rewards (or costs) of performing an action in a given state. It is important to note that how to turn a real-world environment into the digital environment with clear reward signals is a key point to carry out RL.

2.3. Robot navigation task with reinforcement learning

The robot navigation task [24, 27, 30] is a term used in robotics for the process of breaking down the desired movement task into discrete motions that satisfy movement constraints and possibly optimize some aspects of the movement. In the navigation task, several unmanned robots are tasked to navigate towards a randomly defined target safely, and across a field that is filled with randomly scattered obstacles. At present, in light of its strength, RL algorithms have achieved many important achievements in the field of mobile robot path planning [23].

The traditional approaches for robot navigation task, such as Voronoi diagram [31], A* algorithm [32], Dijkstra algorithm [33], Particle Swarm Optimization (PSO) [34], Ant Colony Optimization (ACO) [35], Simulated Annealing (SA) [36], and Genetic Algorithm (GA) [37] have several drawbacks including the high time complexity in high dimensions, making them practically inefficient [38]. To overcome the weakness, the first FALCON (Fusion Architecture for Learning, Cognition, and Navigation) system [30] was developed to learn a policy directly by creating category nodes, each associating a current state with a desirable action. As the FALCON relies on the availability of immediate feedback, it is not applicable to problems in which the reward of an action is only known several steps after the action is performed. To overcome this inadequacy, TD-FALCON [27] system using the TD learning method with Q-learning can be considered, which is effective in learning with both immediate and delayed rewards and achieve a stable performance in a much faster pace than FALCON.

3. Proposed method and framework

In order to solve the rebars clash problems of RC joints in a 3-D space, this section describes an approach of treating the code-specified rebar designs as a multi-agent path planning. The formulation of multi-agent path planning for rebar clashes at RC beam-column joints is also presented in Section 3.1. Section 3.2 presents the formulation of a grid environment for the rebar spacing requirements for RC members, (e.g., RC beams and RC columns) as specified in GB50010-2010 [5]. Section 3.3 presents the proposed framework via the MARL system with BIM for clash-free rebar designs.

3.1. Formulating rebar designs as path planning considering both immediate and delayed rewards of multi-agent for RC members

A MARL system can be used as an efficient and effective tool for solving the path planning problem [39]. To overcome the inadequacy of relying on immediate rewards and the failure of finding feasible solutions in real complex RC structures, the clash-free of rebar designs is modeled as a more realistic path planning problem of multi-agents considering both immediate and delayed rewards. In fact, it is more realistic to consider both immediate and delayed rewards since targets of agents may be blocked or invisible in the real world. It can be further modeled with a team of agents tasked to navigate from origins towards defined targets safely, while crossing a RC beam-column joint which is filled with the obstacles as shown in Fig. 2. In a 2-D environment, agents can choose three actions to avoid obstacles like forward move, right and left, while in the 3-D environment, there are two additional actions: up and down to ensure the traces of agents can be transformed into practical rebar designs. The RC beam-column joint with the rebar arrangement is gradually filled with obstacles which are the 3-D

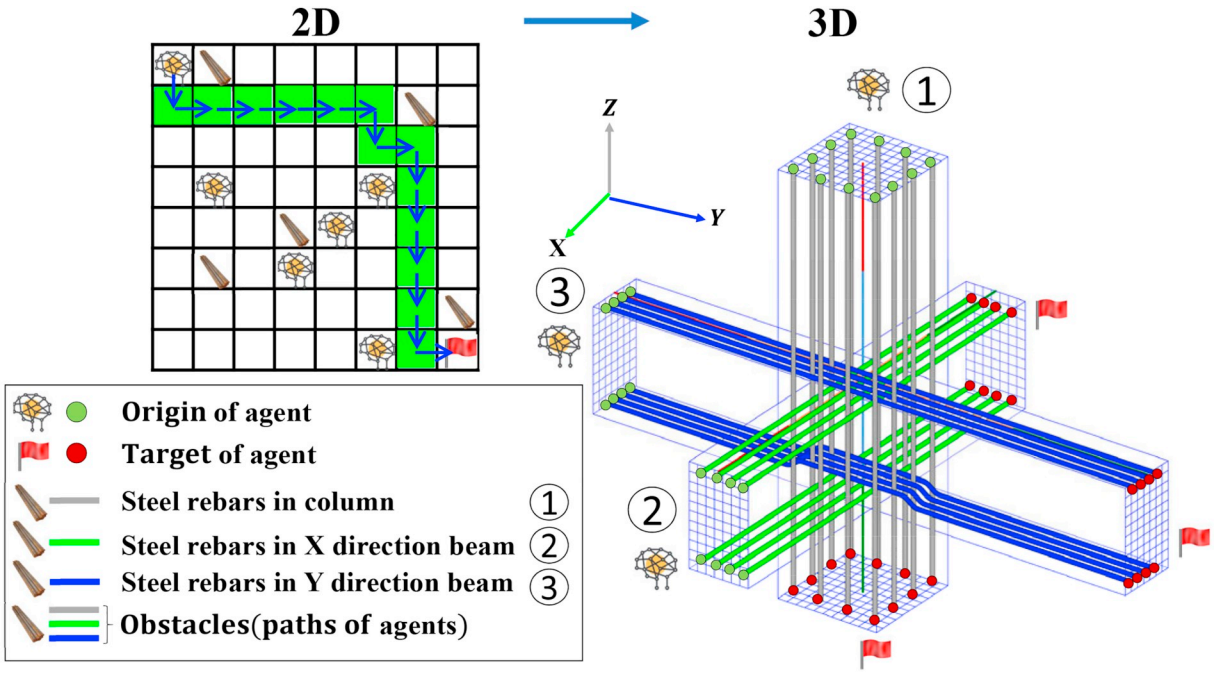

Fig. 2. Problem formulation for RC beam-column joints.

coordinates of the rebars generated in the previous steps. With the proposed MARL, the 3-D coordinates of the clash-free rebar designs are then obtained by collecting the traces of the agents.

Specifically, the rebar design process of a RC beam-column joint is divided into three phases as illustrated in Fig. 2: (1) In the first phase, the longitudinal rebars in a column (gray line) are regarded as a group of agents from the origins navigating to the targets across the column and beam-column joints in a 3-D environment and there are no other rebar obstacles at the joint area in this phase; (2) In the second phase, the x-direction longitudinal rebars (green line) in a beam are regarded as a group of agents, and the rebars in the column including longitudinal and shear rebars are regarded as obstacles; (3) In the third phase, the y-direction longitudinal rebars in a beam (blue line) are regarded as a group of agents and the rebars in the column and x-direction beam are regarded as obstacles.

3.2. Formulating grid environment for rebar spacing requirements in RC members

3.2.1. Grid environment transformation for RC joints

To carrying out the MARL system, RC members have to be transformed into a suitable digital environment. Moreover, the BIM models of RC members are transformed into grid environments approximating the geometry of the RC members with known boundary conditions, as shown in Fig. 3. The origins of agents in each mission are decided by $S_{ht,max}$ and $S_{ht,min}$ (S_{ht} being the horizontal spacing of longitudinal tension rebars). Meanwhile, the targets of agents in each mission are decided by $S_{hc,max}$ and $S_{hc,min}$ (S_{hc} being the horizontal spacing of longitudinal compression rebars), as described in Section 3.2.2.

D_i is the dimension of a single square grid in a mesh environment, determined by:

$$D_i = max(d_c \quad and \quad d_t) \tag{1}$$

where d_c is the diameter of longitudinal compression rebars in a column or beam, and d_t is the diameter of longitudinal tension rebars in a column or beam. In order to ensure the accuracy of the grid environment transformation for RC joints, make sure each rebar in the range of the grid and decide the dimension of a single grid in the mesh environment of the RC joints, the dimension of a single square grid must be larger than the maximum diameter of the rebar.

Therefore, the size of grid environment S_z depends on D_i and the dimension of RC members, namely

$$Sz = floor\,(D/Di) \tag{2}$$

where D is the dimension of RC members (length/width/height), D_i is the dimension of a single square grid in a mesh environment and *floor*() denotes the integer rounding down function for limiting the range of S_z.

3.2.2. Rebar spacing requirements for RC beams

According to Chinese design code GB5001-2010 [5], the three main considered variables (Fig. 4) during the rebar design of RC members are (1) the cross-sectional area of longitudinal tension rebars (A_s), (2) the cross-sectional area of longitudinal compression rebars ($A_{s'}$), and (3) the cross-sectional area of stirrups (A_{sv}), which depend on several parameters [4].

To pour concrete easily and ensure the compactness of concrete around rebars, the spacings between longitudinal rebars are determined as per the provisions of GB50010-2010 [5], as shown in Fig. 5, where $S_{hc} \geq 30$ and $\geq 1.5d_{c,max}$ and $S_{ht} \geq 25$ and $\geq d_{t,max}$. Furthermore, when the layers of longitudinal tension rebars are more than two, S_{vt} being the vertical rebars spacing should be as such: $S_{vt} \geq 25$ and $\geq d_{t,max}$ [5].

A_s is the total cross-sectional area of tension rebars provided at a section, as given by Eq. (3).

$$A_s = \sum_{i=1}^{N_t} \frac{\Pi \cdot d_{t,i}^2}{4} \tag{3}$$

where $d_{t,i}$ is the diameter of tension rebars i in an RC member; and N_t is the total number of tension rebars ($n_{t,min} \leq N_t \leq n_{t,max}$).

$$n_{t,min} = \frac{b - 2 \cdot c}{S_{ht,max}} \tag{4}$$

$$n_{t,max} = \frac{b - 2 \cdot c}{S_{ht,min}} \tag{5}$$

where b is the width of RC beam, c is the concrete cover and $S_{ht,max}$ and $S_{ht,min}$ are respectively the maximum and minimum spacings between

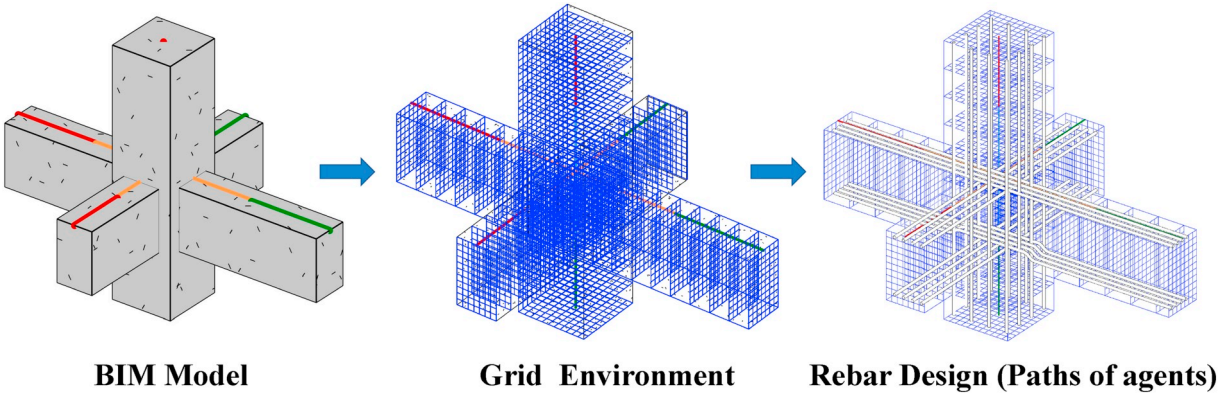

Fig. 3. Grid environment transformation for RC joints.

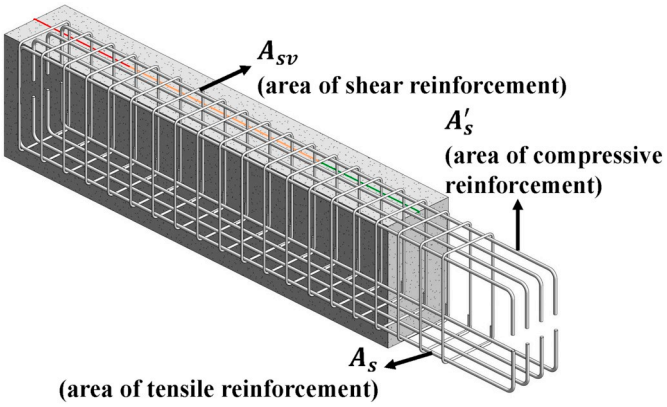

Fig. 4. Three main considered variables during rebar designs in an RC beam.

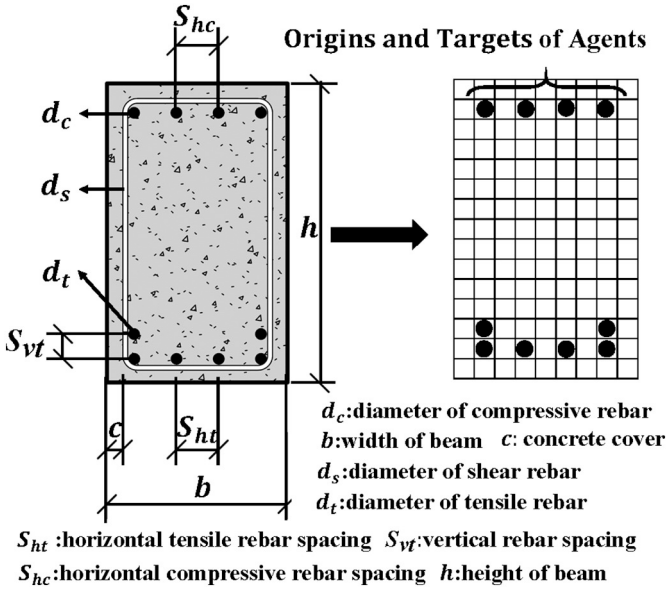

Fig. 5. Illustration of rebars spacing, origins and targets of agents and transformation to grid environment for an RC beam.

tension rebars. Moreover, the $S_{ht,max}$ and $S_{ht,min}$ values are used in MARL to decide the origins and targets of agents in each mission.

Similar to A_s, $A_{s'}$ represents the total cross-sectional area of compressive rebars provided at a section, as given by Eq. (6).

$$A_{s'} = \sum_{i=1}^{N_c} \frac{\Pi \cdot d_{c,i}^2}{4} \tag{6}$$

where N_c is the total number of compressive rebars ($n_{c,min} \le N_c \le n_{c,max}$) and $d_{c,i}$ is the diameter of compressive rebars i in an RC member.

$$n_{c,min} = \frac{b - 2 \cdot c}{S_{hc,max}} \tag{7}$$

$$n_{c,max} = \frac{b - 2 \cdot c}{S_{hc,min}} \tag{8}$$

where $S_{hc,max}$ and $S_{hc,min}$ are the maximum and minimum spacings between compressive rebars, respectively. Furthermore, the $S_{hc,max}$ and $S_{hc,min}$ are used in MARL to decide origins and targets of agents in each mission.

3.2.3. Rebar spacing requirements for RC columns

Due to length limitations and the determination of the three main considered variables for RC column designs similar to that of RC beam designs, the rebar spacing requirements are briefly described in Fig. 6. The spacing between longitudinal rebars S_h is determined according to GB50010-2010 [5], as shown in Fig. 6, where $50mm \le S_h \le 300mm$. However, when the width of the column is more than 400 mm, the spacing of longitudinal rebars (S_h) has to be kept under 200 mm [5].

3.3. Framework for path planning problem of clash-free rebar designs

The framework via the MARL system with BIM for clash-free rebar designs is based on a framework presented by Mangal and Cheng [4], as it was proposed for automated rebar optimization for RC frames. The framework consists of 4 modules (Fig. 7) : (1) BIM Model Information Extraction, (2) Structural Type Analysis, (3) Structural Analysis, and (4) Multi-Agent Reinforcement Learning (MARL) System. BIM Model

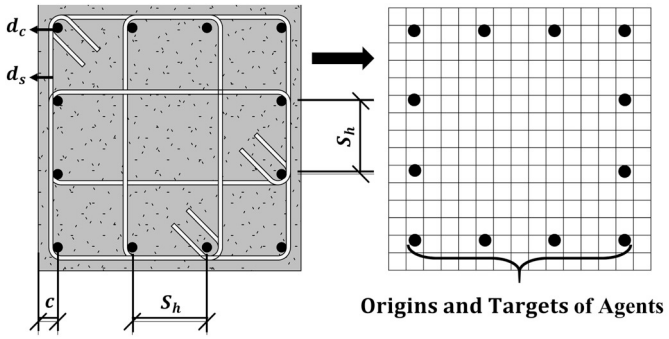

Fig. 6. Illustration of rebars spacing, origins and targets of agents and transformation to grid environment for an RC column.

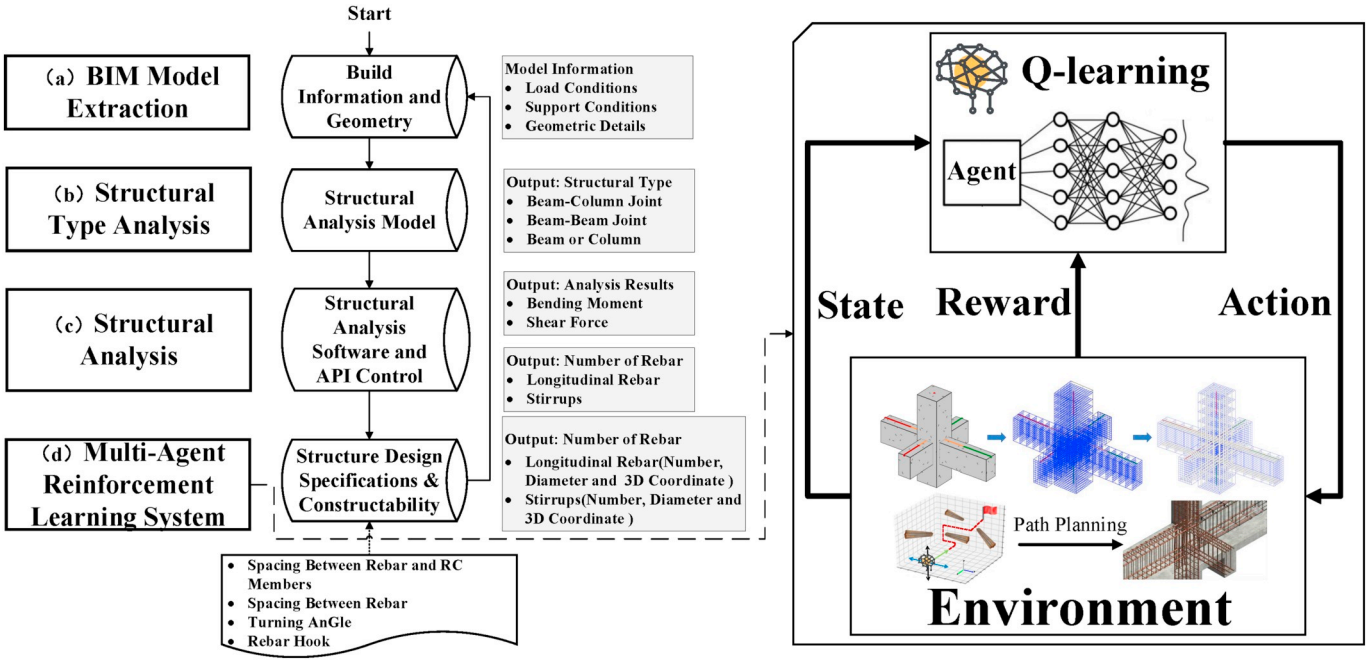

Fig. 7. Proposed BIM-based framework with multi-agent reinforcement learning (MARL) for clash-free rebar designs.

Extraction and Structural Analysis modules are clearly stated in [4]. The modules of Structural Type Analysis and MARL System, are explained in Sections 3.3 and 5, respectively. Each module is briefly described as follows.

(1) BIM model information extraction: The detailed information including behavior, physical and material characteristics [40] in BIM is extracted for the next two modules consisting of structural stress analysis and structural type analysis. Behavior characteristics include the loading information of RC members, physical characteristics include the support conditions and geometry of RC members, and material characteristics include the concrete grade and rebar strength.

The information extracted from the BIM model can be automatically achieved by the application programming interface (API) [41] or Dynamo [42] in Autodesk Revit [41]. In the suggested process, the beam or column model can be selected in quantities by the function node named All Elements of Category in Dynamo [42] is shown in Fig. 8. And by using a function node named Element.GetLocation and Element.Parameters [42], the corresponding model information like cross-sectional area, length, concrete cover, the start and end points of an RC member can be transformed into common formats like extensible markup language (XML), Microsoft Excel or CSV by Dynamo for further analysis. As for the single beam-column joint, the program takes within one second to transform the information into CSV file. And the information extraction of two-story RC frames in the study takes a few seconds like 5–6 s to complete the transformation.

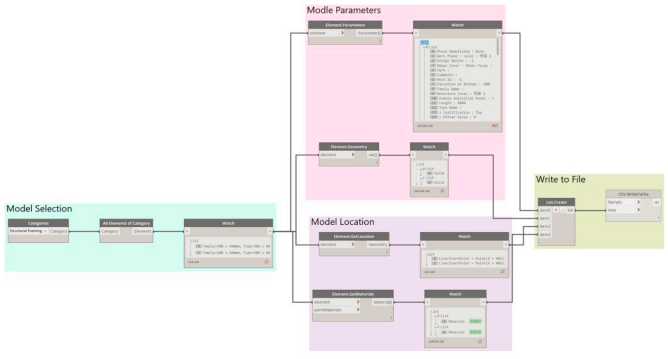

Fig. 8. BIM model information extraction by Dynamo.

(2) Structural type analysis: Different types of beams, columns, and member connections at each level of RC frame are classified into typical structural types, based on the orientation and interaction of RC beams and columns. The classified structural types will further be transformed to structural analysis.

The structural types are determined by the connection relationships between the beams and columns calculated by the location points of RC members. In particular, in the BIM Model Information Extraction module, the location point of an RC member can be extracted from the BIM model as shown in Fig. 8. Based on the location points, the information about the connection relationships between beams and columns like the numbers of beams and columns connected to the points can be calculated. Furthermore, according to the level of RC frame and the connection relationships, the RC members are classified into typical structural based on rules. For example, one column and three beam connect on the same location point, then the joint on this point is T beam-column joint. Using a machine learning classification algorithm to realize the structural type analysis will much be beneficial for more complex RC frames. Therefore, extending the framework including classification algorithm will be considered in the future work.

Various kinds of structural types of beams, columns, beam-column joints and beam-girder joints in a RC frame are shown in Fig. 9. Structural type analysis was carried out to classify the type of RC member in a given RC frame. A total of 15 different structural types of RC members were considered in the proposed BIM-based framework for clash-free rebar designs. The considered RC frame was divided into separate floors. In each separated floor, RC beams, columns and their connections were taken into consideration in the proposed framework.

(3) Structural analysis: The extracted information is input into the structural analysis software PKPM [11] to calculate bending moments, shearing forces, axial forces and torsional moments of the RC members according to the design specifications, such as GB 50010-2010 [5] and GB 50011-2010 [6]. The number of longitudinal rebars and stirrups that meet the code-specified requirements will then be calculated.

(4) MARL system: With the number of longitudinal rebars and stirrups, structural type, coordinates of RC members and geometric details, the MARL will then generate the 3-D coordinate information of the clash-free rebar designs. The design meets the rebar requirements

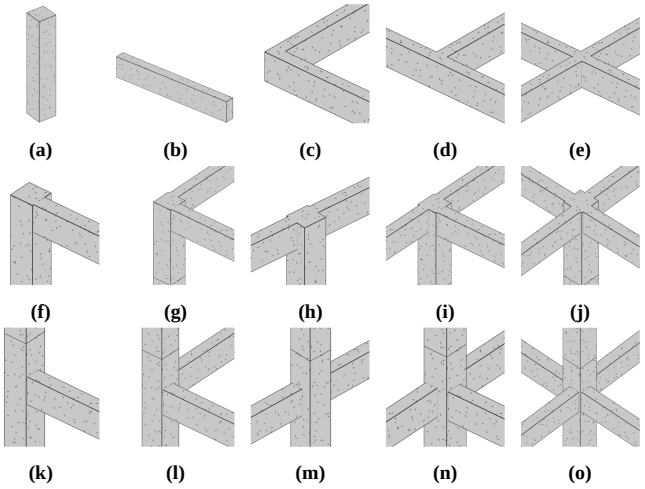

Fig. 9. Various structural types in a RC frame: (a) single column; (b) single beam; (c) L beam joint; (d) T beam joint; (e) + beam joint; (f) top floor single beam joint; (g) top floor L joint; (h) top floor opposite joint; (i) top floor T joint; (j) top floor + joint; (k) middle floor single beam joint; (l) middle floor L joint; (m) middle floor opposite joint; (n) middle floor T joint; (o) middle floor + joint.

specified in the design codes [5, 6], such as the spacing, bent angle, and hook demands. Specifically, the BIM model is set as the environment, and grid environment transformation is the process of extracting the information of physical characteristics (e.g., support conditions and geometry of RC members) from the BIM model, as described in Section 3.2. According to Q-learning [28, 29], the agent selects the suitable action as described in Sections 4.1 to 4.3. Furthermore, as described in Section 5, the paths of agents are converted to BIM generating clash-free rebar deign.

4. Proposed multi-agent reinforcement learning system for solving rebar designs as path planning

By employing the Q-learning method as the RL engine, the particular forms of state, action and rewards are designed for the reinforcement MARL. Furthermore, the design of state, action and reward must meet the physical constraints and design codes of rebar in order to achieve more stable and efficient performance for rebar designs in complex RC members.

4.1. Neural network architecture of each agent

The architecture of each agent takes the form of TD-FALCON [27-29], which has the three-channel neural network architecture, consisting of three modules: (1) State, (2) Action, and (3) Reward (Fig. 10). State module is a sensory field F_1^{c1} for saving and representing current agent states; Action module is a motor field F_1^{c2} for representing available actions and Reward module is a feedback (reward) field F_1^{c3} for representing the internal states of an agent as well as the external feedback from the environment. The architecture has a cognitive field F^2 where agents calculate the maximum expected future rewards for action at each state, which encodes a relation among the patterns in the three input channels.

4.1.1. State module

MARL involves multiple agents that start from one side of beams or columns. All of these agents are equipped with a set of sonar sensors that has a 180° forward view. Meanwhile, input attributes of sensory (state) vector consist of obstacles detection, other agent positions detection and the bearing of the target from the current position in this mission, as shown in Fig. 11. In a discrete time step, the example of

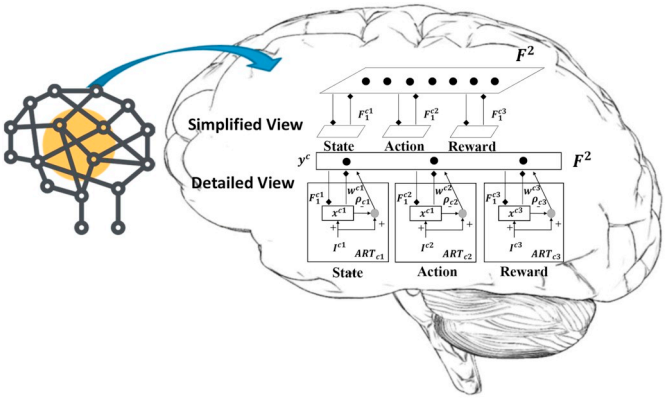

Fig. 10. Neural network architecture of each agent.

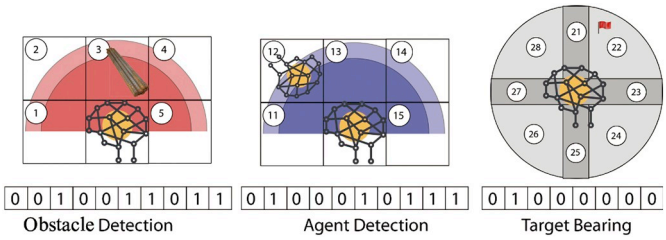

Fig. 11. Example of states (input attributes) in 2D.

input attributes for the agent are shown in Fig. 11. The states (input attributes) shown denote that an agent detects an obstacle (path of other agent) above itself, another agent diagonally to the left, and a target diagonally to the right position. For each direction i of sonar sensor, the sonar signal is calculated by $S_i = 1/d_i$, where d_i is the distance in the i direction between the agent and an obstacle,which can be paths of other agents, another agent, or the boundary of a RC member. If S_i is smaller than 1, S_i will be set as 0. Therefore, without a priori knowledge of the 3-D coordinate information of the obstacles and targets, each agent is equipped with a localized view of its environment.

4.1.2. Action module

In the MARL system, the agent can choose one of the five possible actions (left, forward move, right, up and down) at each discrete time step (Fig. 12).

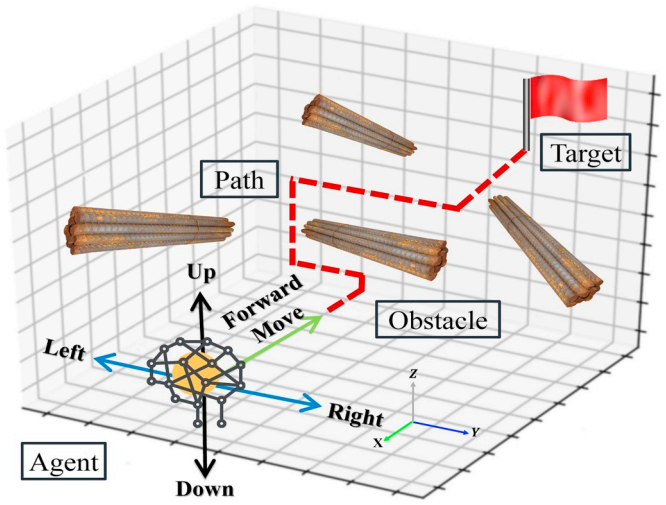

Fig. 12. Illustration of five possible actions in 3-D space.

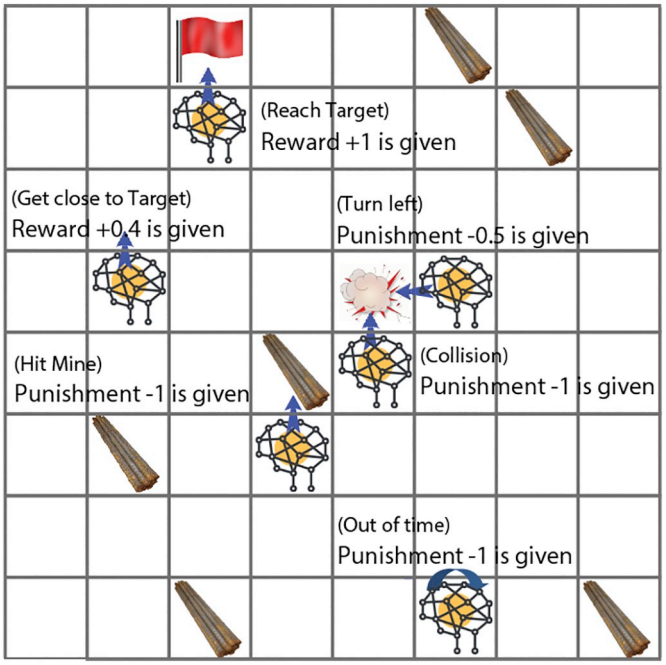

Fig. 13. Illustration of multi-agent path planning including reward, penalty and mission endings for rebar designs.

4.1.3. Reward module

In MARL, the design of reward, penalty and some specific strategies are presented for build-ability constraints (Fig. 13). In the experiments, it is not necessary that the agent receives immediate evaluation feedback from the environment for each action taken. Since in the realistic world, it is difficult to reach the targets as they may be invisible or blocked. The reward and penalty strategies are described in Table 1:

A reward of +1 is given when the agent reaches the target without hitting obstacles or running out of time.

A reward of +0.4 is given when the agent takes action that can get close to the target to encourage agents searching for defined targets.

A penalty of −1 is given when the agent hits an obstacle (paths of other agents), collides with another agent or runs out of maximum time in order to avoid rebar clashes.

A penalty of −1 is given when the agent moves into the specified range (1.5 × rebar diameter) of paths or positions of other agents, thus satisfying the rebar spacing demand.

A penalty of −0.5 is given when the agent takes actions including left, right, up, and down in order to ensure that the agent moves as straight as possible, therefore the layout of rebars is most likely to be a straight line unless obstacles are encountered.

A reward of 0 is also assigned when the agent moves forward and does not find the target in the maximum allowable time.

Table 1
Reward and penalty strategies for agents.

Reward and penalty strategies	
Reach targets without hitting obstacles	+1.0
The distance between agents and targets decreases	+0.4
Hit obstacles (paths of other agents)	−1.0
Hit other agents	−1.0
Within the range of other agents' paths	−1.0
Run out of time	−1.0
Take actions (left, right, up and down)	−0.5
Take action (forward move)	0

4.2. Q-learning algorithm in MARL

Algorithm 1. The Q learning algorithm

1: **Initialize** Q-values $Q(s,a)$ arbitrarily for all state-action pairs.;
2: **For** life or until learning is stopped $\cdots$;
3: **Choose** an action (a) in the current world state s based on current Q-values estimates $(Q(s,\cdot))$.;
4: **Take** the action (a) and observe the outcome state (s') and (r').;
5: **Update** $Q(s,a)$: $Q(s,a) = Q(s,a)+\alpha[r+\gamma max_{a'} Q(s',r')-Q(s,a)]$;

The thought of Q-learning [28, 29] with both immediate and delayed evaluative feedback is that agents evolve from learning by a sequence of trials and thereby adjust their behavior (Fig. 15). The Q learning algorithm pseudo-code is summarized in Algorithm 1. Different aspects of Q-learning are described as follows:

Q-Table ("Q" for action-utility function) is a lookup table where agents calculate the maximum expected future rewards for action at each state as presented in Fig. 14. Basically, this table will guide agents to the best action at each state. In terms of computation, this environment can be transformed into a table.

In the Q-Table, the columns will be the available actions and the rows will be the states. Each Q-table score will be the maximum expected a future reward that the agent will get if it takes that action at that state. In order to learn and improve each value of the Q-table in each iteration during the iterative process, the Q-learning algorithm [28, 29] is executed.

4.2.1. Q-function

The Q-function [28, 29] requires two inputs: State (**s**) and Action (**a**). It returns the expected future reward of that action at that state as:

$$Q^{\pi}(s_t, a_t) = E(R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots | s_t, a_t) \quad (9)$$

where $Q^{\pi}(s_t,a_t)$ is the Q-values for the state s_t given a particle action a_t, $E(R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots)$ is the expected discounted cumulative reward, and $s_t and a_t$ are respectively the given the state and action.

Q-function is a reader that scrolls through the Q-table to find the line associated with the agent's state, and the column associated with agent's action. It returns the Q-value from the matching cell, named as "expected future reward". Before agents explore the environment, the Q-table gives the same arbitrarily fixed value (0 for most of the time). As agents explore the environment, the Q-table will give agents a better and better approximation by iteratively updating $Q(s,a)$ using the Bellman Equation shown in Eq. (10).

4.2.2. Initialize Q-values

A Q-table is first built with m columns (m = number of actions) and n rows (n = number of states), and then initializes the values at 0. In Fig. 14, agents have four actions (a = 4) and sixty four states (s = 64). So a table is built with 4 columns and 64 rows (Fig. 14).

4.2.3. Choose and perform an action

Agents will choose an action (**a**) in the state (**s**) based on the Q-Table. In Fig. 16, there are two actions for agents to choose from: down or right. However, as mentioned earlier in Section 4.2.2, when the episode initially starts, every Q-value is initialized as 0.

In the training process, there are two strategies namely exploration and exploitation. Exploration is finding more information about the environment and exploitation is exploiting known information to maximize the reward. The goal of the agent is to maximize the expected cumulative reward. However, agents may trap in local optimum and fail to find a feasible solution. Therefore, in Q-learning, it uses the epsilon greedy strategy as shown in Fig. 17. An exploration rate ϵ is specified as the rate of steps that agents take actions randomly. At the

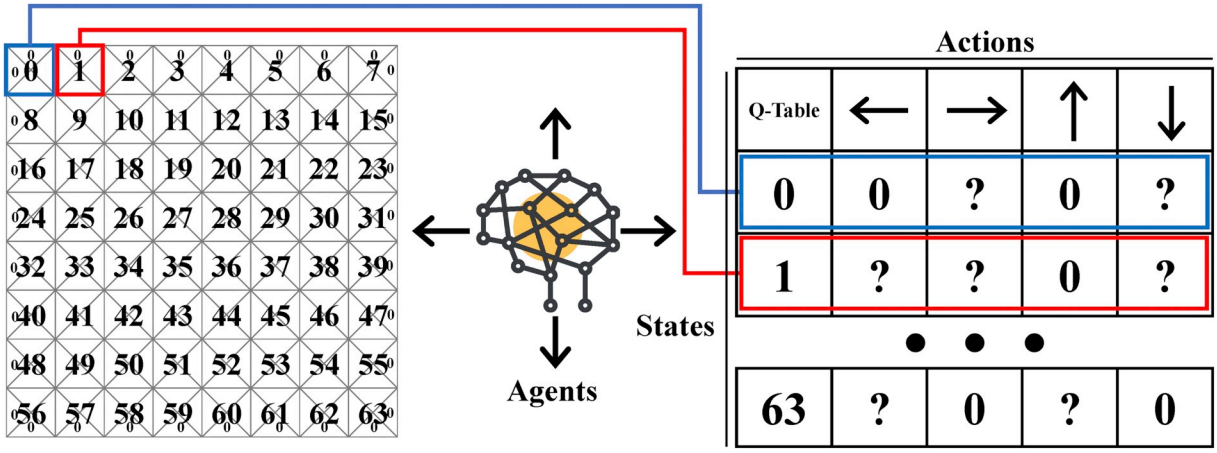

Fig. 14. Illustration of the Q-Table.

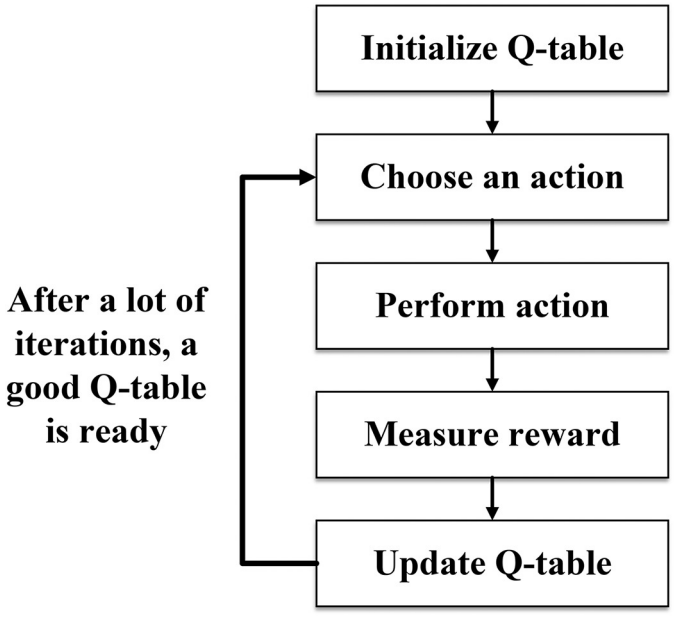

Fig. 15. The Q-learning algorithm process.

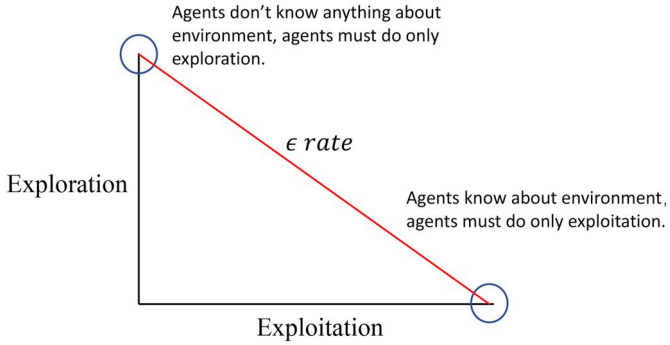

Fig. 17. Illustration of exploration/exploitation trade-off.

beginning of the mission, ϵ rate must be at its highest value, as agents do not know anything about the values in Q-table, implying that agents need to do a lot of exploration by randomly choosing their actions. Therefore, a random number N is generated. If $N > \epsilon$, agents will then do the exploitation, meaning that agents use what is already known to select the best action in each step. Else, agents will do the explorations. The idea is that agents must have a big ϵ rate at the beginning of the training of the Q-function. As the agents explore the environment, the ϵ rate decreases and the agents start to exploit the environment. Then, reduce ϵ rate progressively as the agent becomes more confident at estimated Q-values.

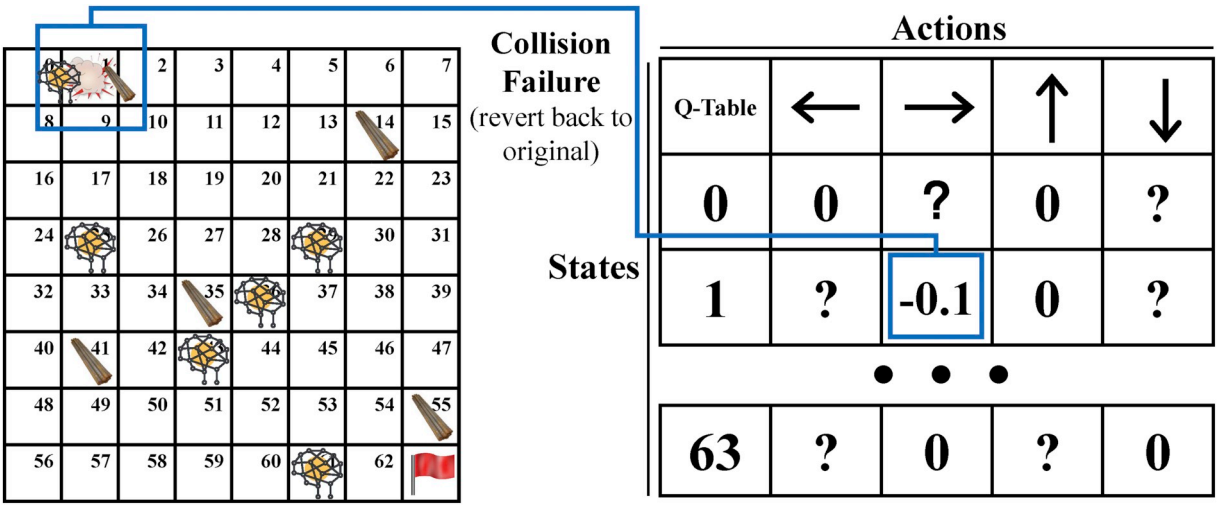

Fig. 16. Illustration of updating Q-table.

In Fig. 17, at the beginning of training, an agent has a higher ϵ rate but knows nothing about the environment, the agent chooses a random action from the right. The agent hits an obstacle and gets penalty -1 according to the reward and penalty system as described in Section 4.2. According to the Bellman equation (Eq. (11)), the Q-values are updated for being at the start and moving to the right getting collision as described in Section 4.2.4.

4.2.4. Evaluate

Agents take the action (**a**) and then perform corresponding observation, the outcome state (**s**), and reward **r**. The function $Q(s,a)$ is updated by the Bellman equation:

$$NewQ(s,a) = Q(s,a) + \alpha[R(s,a) + \gamma maxQ'(s',a') - Q(s,a)] \quad (10)$$

where $NewQ(s,a)$ is the new Q-value for state s and action a, $Q(s,a)$ is the current Q-value, α is a learning rate, $R(s,a)$ is the reward by taking action a at state s, γ is a discount rate, and $maxQ'(s',a')$ is the maximum expected future reward for given the new state s' and all possible actions a'.

4.3. MARL system based on Q-learning

Algorithm 2. Pseudo Code of MARL System

```
Initialize: Generate the initial m agents
While(a mission ending conditions are not satisfied)
For each agent
    If (the agent does not fail or not arrive the target)
        Perform Q-learning algorithm
    Else
        Stop training
    End If
End For
End While
```

The basic steps of the MARL system are outlined in Algorithm 2. In the first step, a population of m agents is initialized. While mission ending conditions are not satisfied, each agent will perform Q-Learning algorithm until agents arrive the targets or agents fail in a mission. An agent fails when hitting obstacles, exceeding 30 sense-act-learn cycles (running out of time). An agent having reached the target without hitting obstacles or running out of time is defined as a success. A mission ends when all agents fail or arrive at the target successfully. A mission will also be deemed to have failed if an agent collides with another, as depicted in Fig. 18.

The positions of origins and targets are automatically chosen by MARL on the one side of columns or beams as described in Section 3.2. The positions of obstacles are generated according to paths of other agents in the previous steps. In a mission i, the positions of the targets remain stationary, while the obstacles are gradually generated by MARL, therefore the positions of obstacles are dynamically increasing. As the configuration of the obstacle is changing within missions, agents need to learn strategies that can be carried out onto future unseen missions.

5. Empirical study

The effectiveness and efficiency of the proposed framework via the MARL system with BIM have been examined empirically in this paper. The authors also proposed two evaluation indexes like success rate *Sr* and the spent time investigating the performance of MARL. Section 5.1 presents the detailed experimental configurations including setting

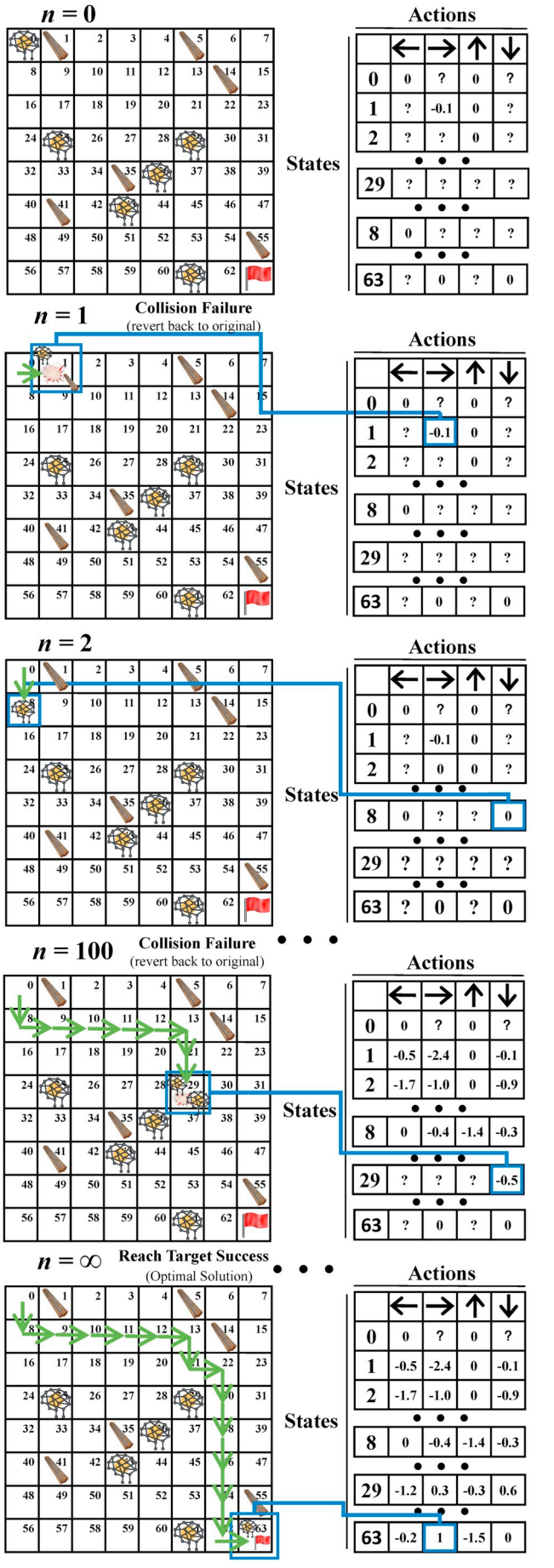

Fig. 18. Illustration of learning process of agents.

parameters and evaluation indicators such as success rate for MARL. Two illustrative examples of three beam-column joints (Section 5.2) and two-story RC frames (Section 5.3) were used to test the proposed framework.

In these experiments, the BIM models of RC members were first built in Autodesk Revit [41], and PKPM [11] to serve as structural analysis applications. Then, programming based on python and Dynamo [42] was developed to automatically transform the support conditions, load conditions, and geometry details to PKPM [11]. The structural analysis of PKPM [11] was then extracted to the MARL method to generate the clash-free rebar designs. At last, the calculated information about number, diameters, and 3-D of rebar by the MARL system was transformed to Autodesk Revit [41] to generate 3-D rebars BIM models using application programming interface (API) in Autodesk Revit [41] and Autodesk plug-in.

5.1. Experimental configurations

In the experiments of illustrative examples, 40 independent simulations for each experiment were carried out, where each simulation involves 1000 missions (i.e., $N_t = 1000$). The setting of parameters of MARL is summarized in Table 2. To ensure the independence in each simulation, the initial Q-value is to be reset as 0 in each simulation. Meanwhile, the initial positions of multiple agents' origins and targets, obstacles are also reset at the start of each mission. An agent having reached the target without hitting obstacles or running out of time is defined as a success. The success rate S_r for a group of agents is measured by the percentage of agents that arrives the defined target positions without hitting obstacles or running out of time within the total missions, which can be calculated by:

$$S_r = \frac{1}{N_m} \times \sum_{i=1}^{N_m} \frac{N_s^i}{N_s} \times 100\% \tag{11}$$

where N_m is the number of missions performed, N_s is the total number of agents that participated in all missions, and N_s^i is the number of agents arriving the targets successfully in mission i without hitting obstacles. S_r was used to evaluate the performance of the proposed framework with MARL.

5.2. Illustrative example of 1 - RC beam-column joints

In this section, three common RC beam-column joints are considered as the case study. Obviously, these joints are prone to rebar clashes, since the rebars are densely arranged three-dimensionally.

Design drawings for the beam-column joints are shown in Fig. 20. Each column is 3500 mm in height with a cross-section of 500 mm × 500 mm. The yield strength(f_y) of longitudinal rebars is 360 N /mm^2. The longitudinal rebars in each column consist of 4 20 mm-diameter bars and 16 18 mm-diameter bars. The beams have a typical cross-section of 500 mm × 300 mm. There are 12 18 mm-diameter longitudinal bars at the top and bottom of each beam.

5.2.1. The example of training process in RC beam-column joint by MARL

The training processes for agents in + beam-column joints by MARL are given in Fig. 19. The gray vertical line indicates the longitudinal rebars in the column which is the obstacle for agents. The origins and targets are represented by red dots and red triangles, respectively.

In this mission, a group of agents are tasked to navigate from the origins (red dots) towards defined targets (red triangles) safely, while crossing a RC beam-column joint. The RC beam-column joint is gradually filled with obstacles which are the 3-D coordinates of the rebars generated in the previous steps.

In the initial stages of mission as shown in Fig. 19 (a) –(c), agents having a higher value of ϵ rate are encouraged to explore new possibilities and try to reach defined targets without hitting obstacles or running out of time. Therefore the paths of the agents look messy, cluttered, or indirect. In the late stage of mission as shown in Fig. 19 (d), agents having a lower value of ϵ rate attempt to converge gradually to the global optimum and to find the optimal paths for the rebar designs. Moreover, along with the experimental training, the paths of agents have also gone from chaos to the gradual and orderly process of development and finally, the global optimum of the path will be selected to generate a suitable rebar design.

Table 2
Summary of the parameters setting for the MARL system.

MARL parameters	
Temporal difference learning parameters	
TD learning rate α	0.05
Discount factor γ	0.7
Initial Q-value	0
ϵ-Greedy action policy parameters	
Initial value of ϵ	0.6
ϵ Decay rate	0.0004

5.2.2. Results and discussions

The centerline coordinates of clash-free rebar designs are generated by the proposed system, with the simulation results shown in Fig. 21. The automated 3-D BIM outputs of the rebars in RC beam-column joints are also given in Fig. 21. Rebar designs for RC beam-column joints are based on the result of the proposed system and design code (GB50010-2010 [5]. As observed, there is no rebar clash in RC beam-column joints using the clash detection of the 3-D BIM output.

The averaged success rates (S_r) of 40 simulations on the proposed MARL were analyzed. The averaged S_r values are plotted in Fig. 22, which indicates the percentage that the proposed MARL has successfully solved the path planning problem of clash-free rebar designs in 40 simulations.

5.3. Illustrative example of 2 - RC frames

A two-story RC frame shown in Fig. 23 with fixed end RC beams and columns was considered as the case study. RC frames have been commonly used in practice. Therefore, they are selected to illustrate and test the proposed framework with MARL.

5.3.1. Experimental environment

In this tested frame, there are 63 RC beams and 23 RC columns in the first floor. The numbers of RC beams and RC columns in the second floor are 47 and 23, respectively. Concrete compressive strength is 40 N /mm^2 (C40 grade) for the RC beams and 60 N /mm^2 (C60 grade) for the RC columns. The concrete cover is 20 mm typically. There are three different rectangular cross-sections for the beams: 150 mm × 350 mm,175 mm × 350 mm, and 200 mm × 350 mm. The RC beams are fixed at the ends with end-support moments in the RC frame. The cross-section of RC columns is 300 mm × 300 mm typically.

The yield strength (f_y) of the longitudinal rebars in RC beams and columns is 360 N /mm^2 and the counterpart (f_{yv}) for the stirrups is 270 N /mm^2. The proposed framework with MARL for rebar designs was then tested on this RC frame. The RC frame BIM model was built by Autodesk Revit [41] and PKPM [11], which serves as the structural analysis application.

5.3.2. Results and discussions

The simulation rebar results and automated 3-D BIM detailing of the rebar details by MARL for the studied RC frame are given in Fig. 25. Therefore, the developed framework via the MARL with BIM helps achieve the rebar 3-D visualization and generate construction drawing details.

Table 3 shows the comparison between the framework with MARL (including the time spent on the BIM model information extraction) and

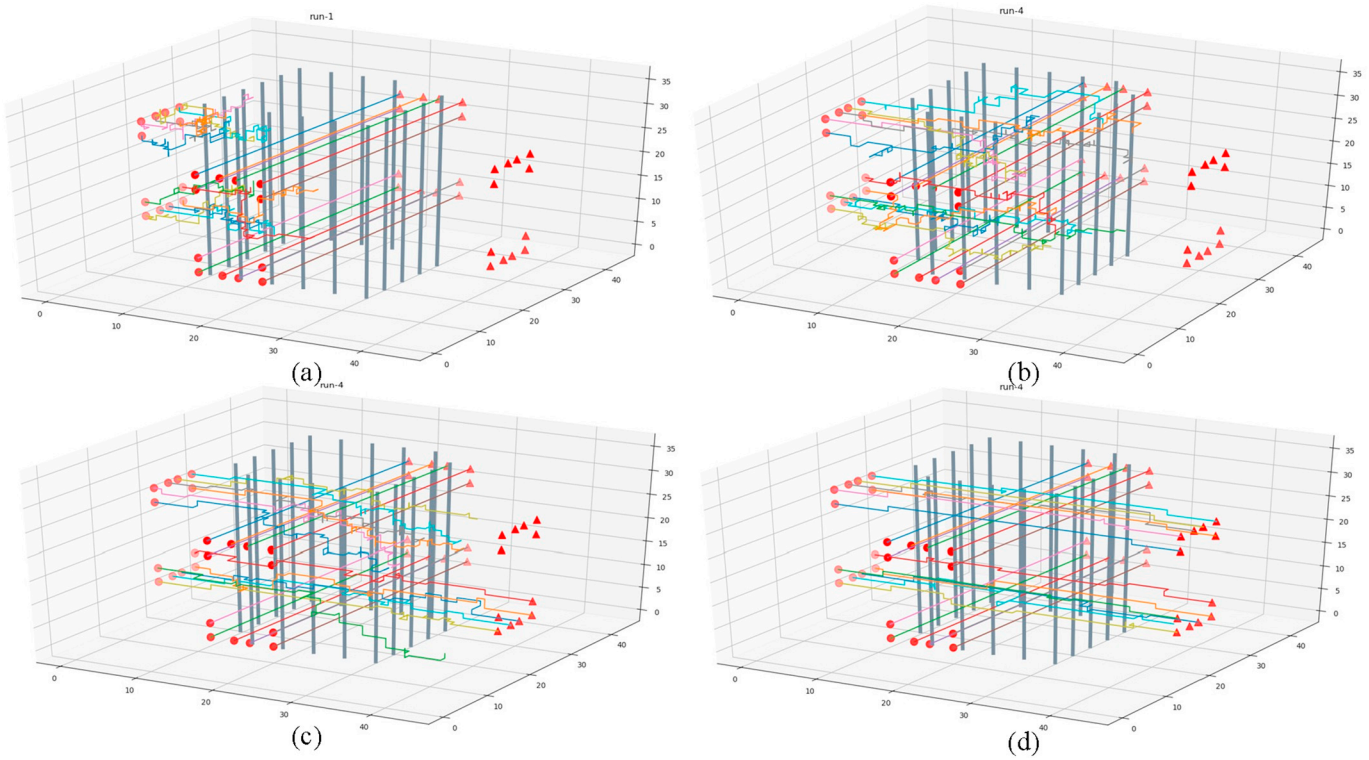

Fig. 19. The training process in + beam-column joint by MARL.

manual arrangement in terms of the average time for rebar designs of RC members. A total of 40 construction designers or engineering consultants were invited to participate in the experiments having at least 2 years construction experience from famous construction design institute and engineering consulting corporation in Chongqing, China. In particular, the participants were invited to design and draw the rebar models of the specific joint types in Table 3, like L, opposite, T, + beam-column joints by Autodesk Revit and record the time used. As for the two-story RC frame, due to expensive manpower, the participators estimated the time based on the time cost spent on the rebar design of the previous joints.

As observed, the average time (min) taken by the framework with the MARL is more optimal than the manual arrangement (Fig. 24). For example, for the considered L, opposite, T, + beam-column joints with rectangular beam cross-sections of 500mm × 300 mm and rectangular column cross-sections of 500 mm × 500 mm, the average time (min) taken by the framework with MARL is 5–6 min, whereas the manual arrangement would take about 50–60 min for each beam-column joint. Moreover, another advantage of the developed framework is the ability to provide fast and consistent clash-free rebar designs every time. In addition, for the considered RC frame with 110 RC beams, 46 RC columns, and 2240 longitudinal rebars, the proposed framework consumes about 180–200 min to generate a clash-free rebar design, whereas a designer must spend 2900–3400 min or more to arrange rebars, while still cannot completely avoid rebar clashes in RC members. The two-story RC building frame case indicates that the proposed framework can reduce engineering time for rebar designs by 90%. As a result, the proposed framework with MARL can provide much more effective and efficient rebar designs for RC members than the manual arrangement.

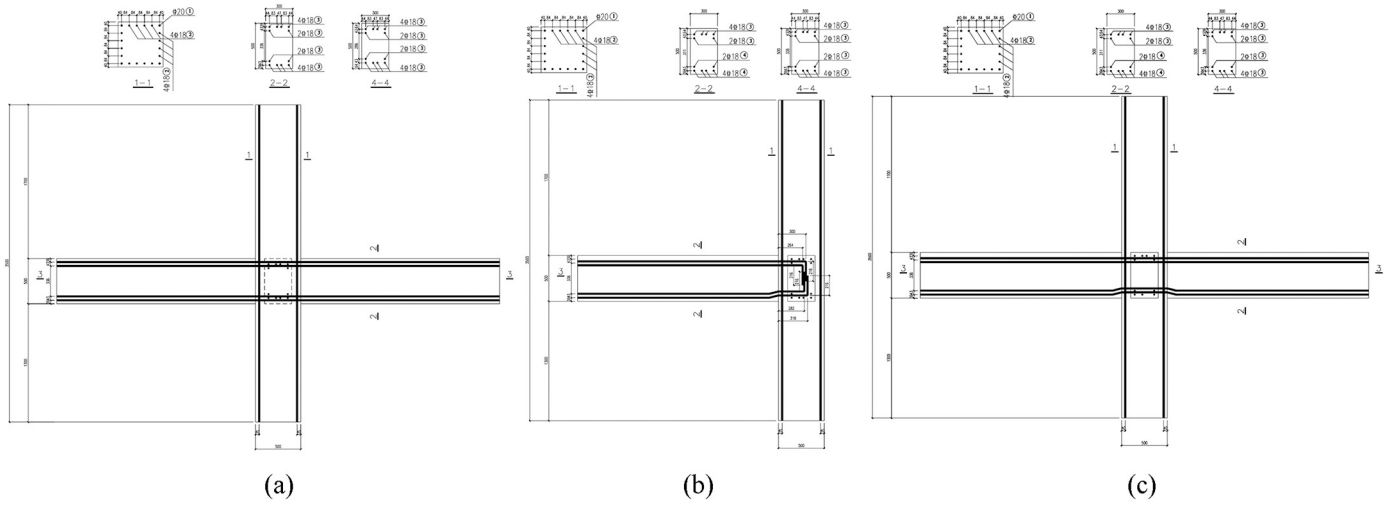

Fig. 20. Design drawing of RC beam-column joints: (a) + Joint (b)T Joint (c) + Joint(with another form of rebar).

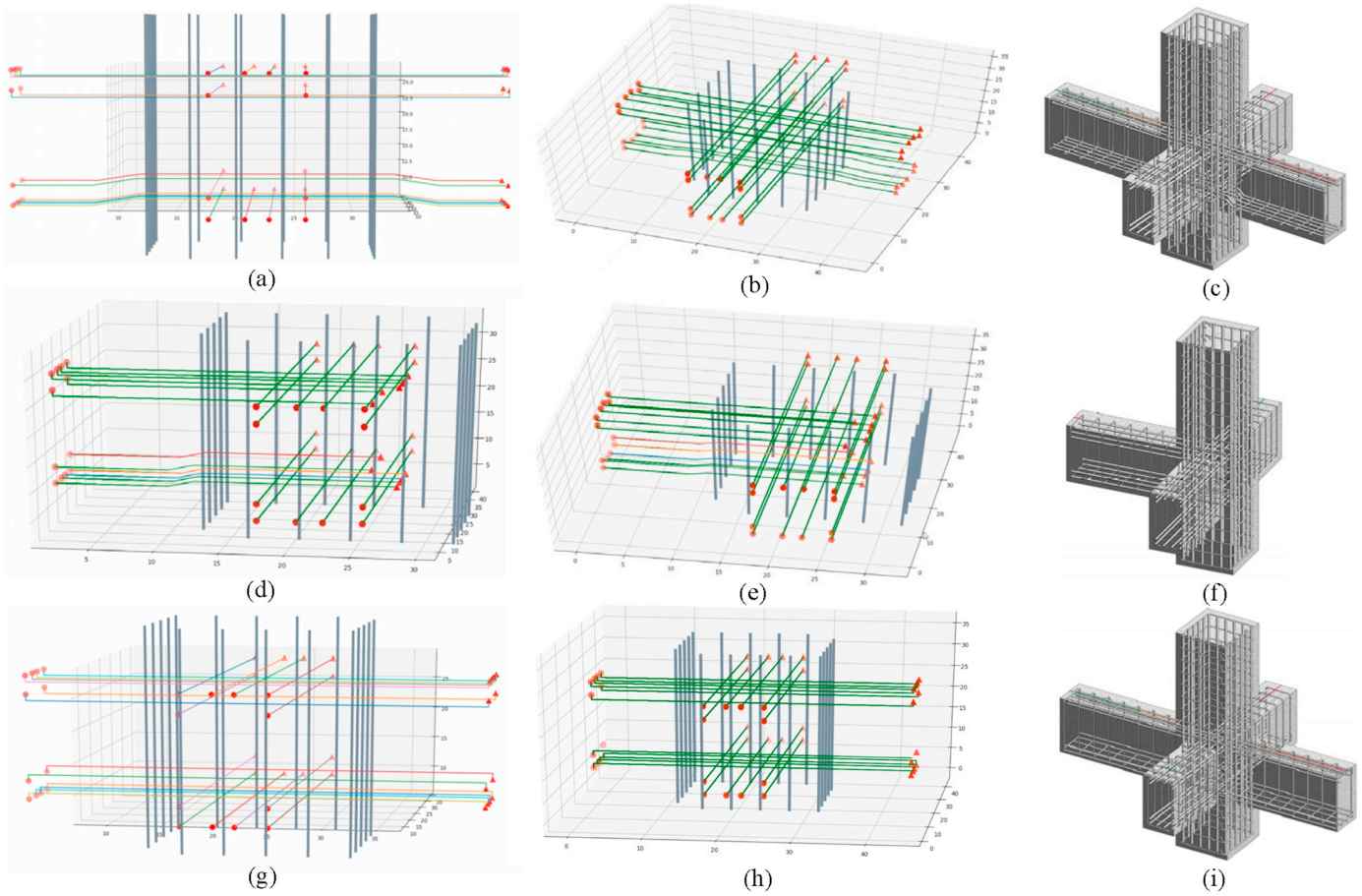

Fig. 21. The simulation result and 3-D BIM output of the beam-column joint based on MARL method: (a),(b) &(c): + joint; (d),(e) &(f):T Joint; (g),(h) &(i): + joint (with another form of rebars).

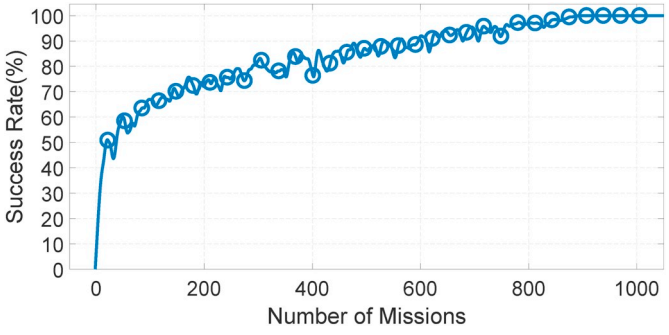

Fig. 22. The success rates of the MARL system.

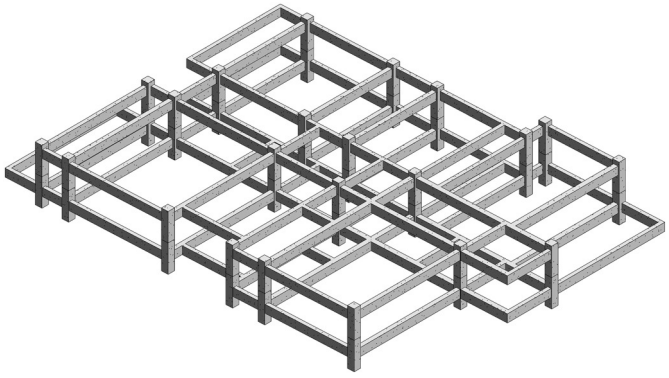

Fig. 23. A two-story RC frame BIM model.

6. Conclusions and future research

The traditional approaches for rebar designs require manual calculations for each RC element. Due to the large number of rebars and the restrictive design code requirements for rebar arrangement, it is impractical, labor-intensive and error-prone for designers to avoid all rebar clashes, even with the use of computer software. For clash detection and resolution, several research efforts have been also made to solve this issue.

Furthermore, most of the existing studies lack the automatic and intelligent arrangement and adjustment of rebars to avoid obstacles for real complex RC joints and frame structures.

In previous work, clash detection and resolution problem for rebar designs is first modeled as a path-planning problem of multi-agent in order to achieve the automatic arrangement and adjustment of rebars according to obstacles. However, the reinforcement learning engine FALCON relies on the availability of immediate rewards and is not applicable to clash-free designs in real complex RC structures where the immediate reward is often not available. To overcome this inadequacy, the authors further extend previous work with Q-learning by considering both immediate and delayed rewards and achieve more stable and efficient performance in rebar designs of real complex RC frames.

The contributions of the present study are summarized as follows:

(1) To overcome the inadequacy of relying on immediate rewards and failing to find feasible solutions for real-world complex RC structures, this paper presents the use of Q-learning based MARL for more realistic path planning problem considering both immediate and delayed rewards for the clash-free rebar designs in real-world

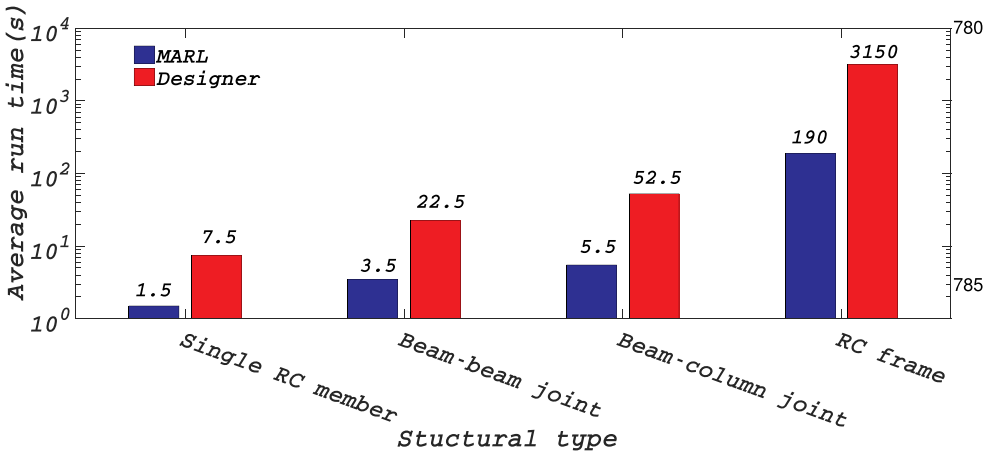

Fig. 24. Average time (min) for the rebar designs of RC members between the framework with MARL and manual arrangement.

RC structures.

(2) To achieve more stable and efficient performance for rebar designs in complex RC members, the particular form of state, action, and immediate as well as delayed rewards for the reinforcement MARL are designed by employing Q-learning as the RL engine.

(3) Comprehensive experiments on a two-story RC building frame have been conducted to verify the effectiveness and efficiency of the proposed framework.

To validate the effectiveness and efficiency of the proposed framework with MARL system, comprehensive simulations were conducted on three typical RC beam-column joints and two-story RC frames. The simulation results of success rate and the average time for rebar designs in RC members indicate that the proposed framework can significantly reduce the engineering time (up to 90% in this study) and avoid spatial rebars clashes. The study results also reveal that: (1) The average time taken by the framework is more optimal than the manual arrangement; (2) The developed framework has the ability to provide fast and consistent clash-free rebar designs every time; and (3) The proposed framework with MARL can provide much more effective and efficient rebars design for RC members compared with the manual arrangement.

However, the proposed framework with MARL system still has some

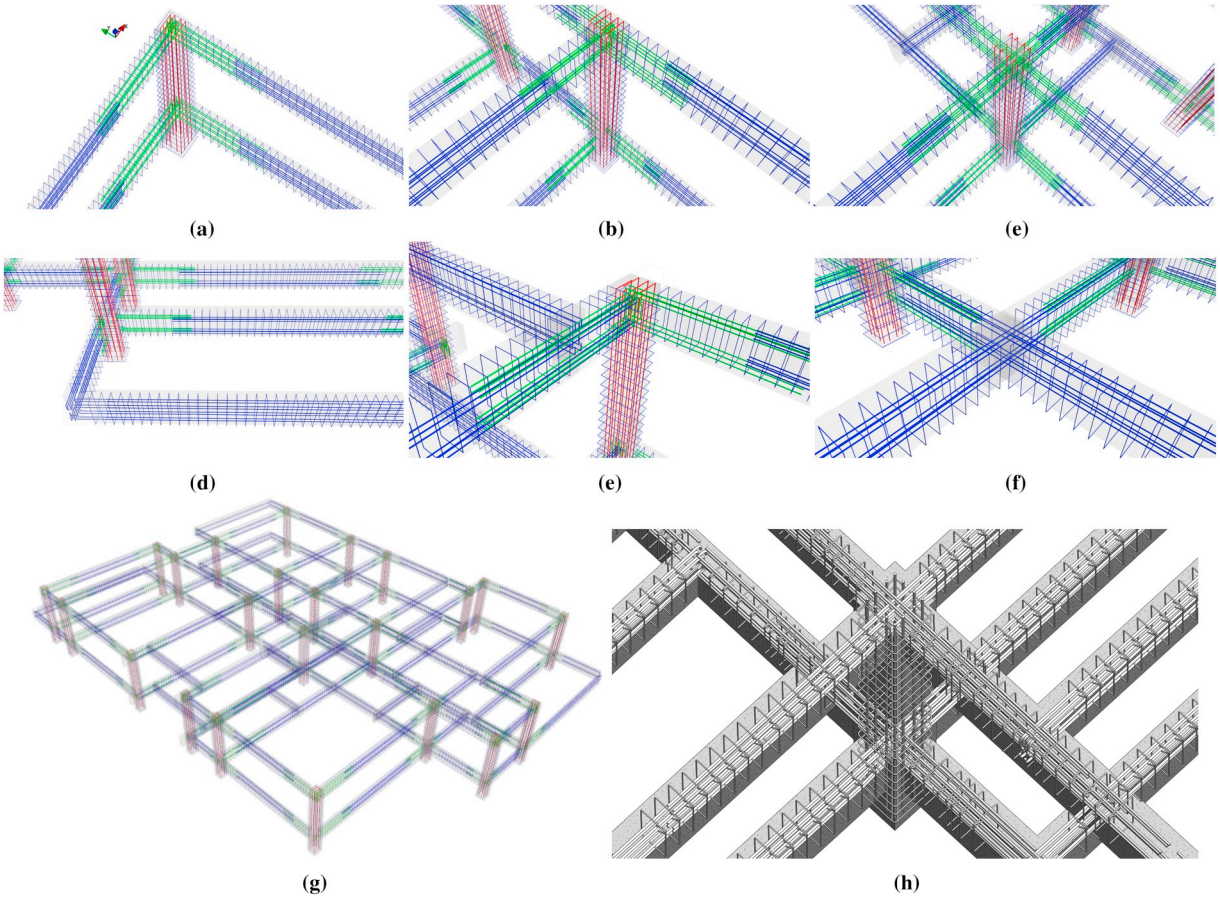

Fig. 25. 3D BIM output of the RC beam-column joints based on the MARL system: (a) L beam-column joint; (b)T beam-column joint; (c)+ beam-column joint; (d)L beam-beam joint; (e) T beam-beam joint and (f) + beam-beam joint.

Table 3
Average time (min) for the rebar design of RC members between the framework with MARL and manual arrangement.

Structural type	Description	Average run time(min)	
		MARL	Designer
Single RC column	500 × 500 mm, 20 rebars	1–2	5–10
Single RC beam	500 × 300 mm, 12 rebars	1–2	5–10
L beam-beam joint	500 × 300 mm, 24 rebars	3–4	10–15
T beam-beam joint	500 × 300 mm, 24 rebars	3–4	20–30
+ beam-beam joint	500 × 300 mm, 24 rebars	3–4	20–30
Top floor single beam-column joint	500 × 300 mm and 500 × 500 mm, 32 longitudinal rebars	3–4	20–30
Top floor L beam-column joint	500 × 300 mm and 500 × 500 mm, 44 longitudinal rebars	5–6	30–40
Top floor opposite beam-column joint	500 × 300 mm and 500 × 500 mm, 32 longitudinal rebars	5–6	30–40
Top floor T beam-column joint	500 × 300 mm and 500 × 500 mm, 44 longitudinal rebars	5–6	50–60
Top floor + beam-column joint	500 × 300 mm and 500 × 500 mm, 44 longitudinal rebars	5–6	50–60
Middle floor single beam-column joint	500 × 300 mm and 500 × 500 mm, 32 longitudinal rebars	3–4	20–30
Middle floor L beam-column joint	500 × 300 mm and 500 × 500 mm, 44 longitudinal rebars	5–6	30–40
Middle floor opposite beam-column joint	500 × 300 mm and 500 × 500 mm, 32 longitudinal rebars	3–4	30–40
Middle floor T beam-column joint	500 × 300 mm and 500 × 500 mm, 44 longitudinal rebars	5–6	50–60
Middle floor + beam-column joint	500 × 300 mm and 500 × 500 mm, 44 longitudinal rebars	5–6	50–60
Two-story RC frames	110 RC beams, 46 RC columns, 2240 longitudinal rebars	180–200	2900–3400

limitations: (1) This work only applied the proposed method for regular-shaped cast-in-place RC members (beams and columns) and their connections. For other structures like prefabricated structure in the Chinese construction sector, additional rewards and punishments schemes are required in the proposed MARL. Based on this framework, a plug-in application on Revit has been developed for regular-shaped cast-in-place RC frames, however, it is currently in the test phase and it cannot be widely used in the Chinese construction sector now. Because as for other structure types like prefabricated structure, additional rewards and punishments should be designed for the specific structure type to meet the design codes; (2) As for the other concrete technologies, the density of the rebars will affect the origins and targets of the rebar agents and increase the collision probability as well as computational complexity. For fiber reinforced concrete, it depends on the size of the fiber. The size of steel fiber or other fibers are generally too small to affect the spacing of the rebars, therefore they are not taken into consideration in the rebar clash problem. Furthermore, there are similar regulations in the current design code. If the size of the fiber is large enough to disturb the rebar arrangement, the rebar spacing should be larger than a certain range to maintain the workability of concrete. In this case, the corresponding rules suitable for the rebar design of fiber reinforced concrete, e.g., if agents (rebar) within the range of fiber, a plenty of -1 is given, would be included in the rewards and penalties design in MARL; (3) Only general concrete structure design code in GB50010-2010 [5] is considered in this study. For tackling other complex and specific construction requirements and standards in the design code, expert knowledge system for spatial conflict coordination will be very helpful in the reinforcement learning design. In the future, further extensions of the proposed MARL will be considered to address the limitations as discussed above.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (grant nos. 51622802, and 61876025), Scientific and Technological Innovation Foundation of Chongqing (cstc2019yszx-jscxX0001), to which the authors are very grateful.

References

[1] D. Bryde, M. Broquetas, J.M. Volm, The project benefits of building information modelling (BIM), Int. J. Proj. Manag. 31 (7) (2013) 971–980, https://doi.org/10.1016/j.ijproman.2012.12.001.
[2] J. Won, J.C. Cheng, G. Lee, Quantification of construction waste prevented by BIM-based design validation: case studies in South Korea, Waste Manag. 49 (2016) 170–180, https://doi.org/10.1016/j.wasman.2015.12.026.
[3] B. Succar, Building information modelling framework: a research and delivery foundation for industry stakeholders, Autom. Constr. 18 (3) (2009) 357–375, https://doi.org/10.1016/j.autcon.2008.10.003.
[4] M. Mangal, J.C. Cheng, Automated optimization of steel reinforcement in RC building frames using building information modeling and hybrid genetic algorithm, Autom. Constr. 90 (2018) 39–57, https://doi.org/10.1016/j.autcon.2018.01.013.
[5] Chinese Standard, GB50010-2010, Code for design of concrete structures, China Architecture & Building Press, Beijing, China, 2010 ISBN: 9787111356301 Available from: https://www.chinesestandard.net/PDF/BOOK.aspx/GB50011-2010 Access date: 8 December 2019.
[6] Chinese Standard, GB50011-2010, Code for seismic design of buildings, China Architecture & Building Press, Beijing, China, 2010 ISBN: 9787112203642, Available from:https://iisee.kenken.go.jp/worldlist/10_China/China%20GB50011-2010(China).pdf Access date: 8 December 2019.
[7] S. Staub-French, A. Khanzode, 3D and 4D modeling for design and construction coordination: issues and lessons learned, Electron. J. Inf. Technol. Constr. (ITcon) 12 (26) (2007) 381–407. Available from: http://www.itcon.org/2007/26/ . Access date: 8 December 2019.
[8] A.R. Tabesh, S. Staub-French, Case study of constructability reasoning in MEP coordination, Construction Research Congress 2005: Broadening Perspectives, 2005, pp. 1–10, , https://doi.org/10.1061/40754(183)119.
[9] Autodesk Inc., Robot Structural Analysis Professional, California, Available from: http://www.autodesk.in/products/robot-structural-analysis/overview , Accessed date: 12 August 2019.
[10] Computers Structures Inc., ETABS, California, Available from: https://www.csiamerica.com/products/etabs , Accessed date: 12 August 2019.
[11] PKPM Inc., PKPM V4.3, China, Available from: https://www.pkpm.cn/ , Accessed date: 12 August 2019.
[12] YJK Inc., YJK, China, Available from: http://www.yjk.cn/ , Accessed date: 12 August 2019.
[13] Autodesk Inc., Navisworks, California, Available from: https://www.autodesk.com/products/navisworks/overview , Accessed date: 12 August 2019.
[14] Solibri Inc., Solibri Model Checker, Helsinki, Finland, Available from: https://www.solibri.com/ , Accessed date: 12 August 2019.
[15] L. Wang, F. Leite, Formalized knowledge representation for spatial conflict coordination of mechanical, electrical and plumbing (MEP) systems in new building projects, Autom. Constr. 64 (2016) 20–26, https://doi.org/10.1016/j.autcon.2015.12.020.
[16] P. van den Helm, M. Böhms, L. van Berlo, IFC-based clash detection for the open-source BIMserver, Proceedings of the 2010 International Conference on Computing in Civil and Building Engineering, 181 Nottingham University Press, Nottingham, UK, 2010ISBN: 978-1-907284-60-1, Available from: https://www.academia.edu/1905724/IFC-based_clash_detection_for_the_open-source_BIMserver . Access date: 8 December 2019.
[17] S. Gijezen, Organizing 3D building information models with the help of work breakdown structures to improve the clash detection process, Master's thesis University of Twente, 2010 Available from: http://essay.utwente.nl/59401/ . Access date: 8 December 2019.
[18] A.M. Radke, T. Wallmark, M.M. Tseng, An automated approach for identification and resolution of spatial clashes in building design, 2009 IEEE International Conference on Industrial Engineering and Engineering Management, IEEE, 2009, pp. 2084–2088, , https://doi.org/10.1109/IEEM.2009.5373167.
[19] J. Zhang, Z. Hu, BIM-and 4D-based integrated solution of analysis and management for conflicts and structural safety problems during construction: 1. Principles and methodologies, Autom. Constr. 20 (2) (2011) 155–166, https://doi.org/10.1016/j.autcon.2010.09.013.
[20] U.Y. Park, BIM-Based Simulator for Rebar Placement, J. Korea Inst. Build. Constr. (JKIBC) 12 (1) (2012) 98–107, https://doi.org/10.5345/JKIBC.2012.12.1.098.
[21] L. Wang, F. Leite, Knowledge discovery of spatial conflict resolution philosophies in BIM-enabled MEP design coordination using data mining techniques: a proof-of-concept, Computing in Civil Engineering (2013), 2013, pp. 419–426, , https://doi.

org/10.1061/9780784413029.053.
[22] Y. Hu, D. Castro-Lacouture, Clash relevance prediction based on machine learning, J. Comput. Civ. Eng. 33 (2) (2018) 04018060, https://doi.org/10.1061/(ASCE)CP.1943-5487.0000810.
[23] S. Zhou, X. Liu, Y. Xu, J. Guo, A Deep Q-network (DQN) based path planning method for mobile robots, 2018 IEEE International Conference on Information and Automation (ICIA), 2018, pp. 366–371, , https://doi.org/10.1109/ICInfA.2018.8812452.
[24] A. Tan, N. Lu, D. Xiao, Integrating temporal difference methods and self-organizing neural networks for reinforcement learning with delayed evaluative feedback, IEEE Trans. Neural Netw. 19 (2) (2008) 230–244, https://doi.org/10.1109/TNN.2007.905839.
[25] British Standards Institution (BSI), BS 8110-1:2005, Structural Use of Concrete: Part 1: Code of Practice for Design and Construction, British Standards Institution, London, 1997 ISBN: 978 0 580 59893 7, Available from: https://www.itcon.org/papers/2008_22.content.04920.pdf . Access date: 8 December 2019.
[26] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT press, 2018 ISBN: 0-262-19398-1, Available from: https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf . Access date: 8 December 2019.
[27] D. Xiao, A. Tan, Self-organizing neural architectures and cooperative learning in a multiagent environment, IEEE Trans. Syst. Man Cybern. B Cybern. 37 (6) (2007) 1567–1580, https://doi.org/10.1109/TSMCB.2007.907040.
[28] C.J.C.H. Watkins, Learning from delayed rewards (Ph.D. thesis), King's College, Cambridge, 1989https://ci.nii.ac.jp/naid/10000072699/en/ , Accessed date: 8 December 2019.
[29] C.J.C.H. Watkins, P. Dayan, Q-learning, Mach. Learn. 8 (3) (1992) 279–292, https://doi.org/10.1007/BF00992698.
[30] Ah-Hwee Tan, FALCON: a fusion architecture for learning, cognition, and navigation, 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541), 4 IEEE, 2004, pp. 3297–3302, , https://doi.org/10.1109/IJCNN.2004.1381208.
[31] F. Aurenhammer, Voronoi diagramsa survey of a fundamental geometric data structure, ACM Comput. Surv. (CSUR) 23 (3) (1991) 345–405, https://doi.org/10.1145/116873.116880.
[32] R.J. Szczerba, P. Galkowski, I.S. Glicktein, N. Ternullo, Robust algorithm for real-time route planning, IEEE Trans. Aerosp. Electron. Syst. 36 (3) (2000) 869–878, https://doi.org/10.1109/7.869506.
[33] E.W. Dijkstra, A note on two problems in connexion with graphs, Numer. Math. 1 (1) (1959) 269–271, https://doi.org/10.1007/BF01386390.
[34] J. Kennedy, Particle Swarm Optimization, Springer US, Boston, MA, 2010, pp. 760–766, https://doi.org/10.1007/978-0-387-30164-8_630.
[35] X. Fan, X. Luo, S. Yi, S. Yang, H. Zhang, Optimal path planning for mobile robots based on intensified ant colony optimization algorithm, IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, 2003. Proceedings. 2003, 1 IEEE, 2003, pp. 131–136, , https://doi.org/10.1109/RISSP.2003.1285562.
[36] H. Miao, Y. Tian, Robot path planning in dynamic environments using a simulated annealing based approach, 2008 10th International Conference on Control, Automation, Robotics and Vision, IEEE, 2008, pp. 1253–1258, , https://doi.org/10.1109/ICARCV.2008.4795701.
[37] J. Tu, S.X. Yang, Genetic algorithm based path planning for a mobile robot, 2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422), 1 IEEE, 2003, pp. 1221–1226, , https://doi.org/10.1109/ROBOT.2003.1241759.
[38] A. Konar, I. Goswami Chakraborty, S.J. Singh, L.C. Jain, A.K. Nagar, A deterministic improved Q-learning for path planning of a mobile robot, IEEE Trans. Syst. Man Cybern. Syst. Hum. 43 (5) (2013) 1141–1153, https://doi.org/10.1109/TSMCA.2012.2227719.
[39] L. Feng, Y.-S. Ong, A.-H. Tan, X.-S. Chen, Towards human-like social multi-agents with memetic automaton, 2011 IEEE Congress of Evolutionary Computation (CEC), IEEE, 2011, pp. 1092–1099, , https://doi.org/10.1109/CEC.2011.5949739.
[40] Q. Lu, J. Won, J.C. Cheng, A financial decision making framework for construction projects based on 5D Building Information Modeling (BIM), Int. J. Proj. Manag. 34 (1) (2016) 3–21, https://doi.org/10.1016/j.ijproman.2015.09.004.
[41] Autodesk Inc., Revit, California, Available from: http://www.autodesk.in/products/revit-family/overview , Accessed date: 12 August 2019.
[42] Autodesk Inc., Dynamo, California, Available from: http://dynamobim.org , Accessed date: 12 August 2019.