Automated Clash Resolution of Rebar Design in RC Joints using Multi-Agent Reinforcement Learning and BIM

J. Liu [ac], P. Liu [ac], L. Feng [bc], W. Wu[bc], and H. Lan[c]

[a]School of Civil Engineering, University of Chongqing, China
[b]School of Computer Science, University of Chongqing, China
[c]Chongqing University Industrial Technology Research Institute, University of Chongqing, China
E-mail: liujp@cqu.edu.cn, pengkunliu@cqu.edu.cn, liangf@cqu.edu.cn, wuwenbo@cqu.edu.cn

Abstract –

The design of rebar in reinforced concrete (RC) structures is a mandatory stage in building construction projects. Due to the large number and complicated arrangement rules of rebar in each design code, it is impractical, labor-intensive and error-prone for designers to avoid all clashes (i.e., collisions and congestion) manually or partial automation by using computer software. Therefore, the building information modeling (BIM) technology has been employed in the present architecture, engineering, and construction (ACE) industry for clash free rebar design. However, it is worth noting that most of existing BIM based approaches are using optimization algorithms for moving components, which can only be applied for regular shaped RC structures. In particular, shapes of rebar are fixed which means the optimized path of rebar cannot bend to avoid the obstacle in current studies. Furthermore, most of the existing studies cannot meet design constraints after avoiding clash, lack automatic and intelligent identification and resolution of rebar clash for complex RC joints and frame structures.

Therefore, we present a framework towards automatic rebar design in RC frame without clashes via multi-agent reinforcement learning (MARL) system with BIM. In particular, by treating each rebar as an intelligence reinforcement learning (RL) agent, we propose to model the rebar design problem as a path-planning problem of multi-agent system. Next, by employing FALCON (A fusion architecture for learning, cognition, and navigation) with immediate evaluative feedback as the reinforcement learning engine, we design particular form of state, action, and rewards for reinforcement MARL for automatic rebar design. In addition, the design of rewards and some strategies in MARL are presented for build-ability constraints. Comprehensive experiments on one-story RC building frame have been conducted to evaluate the efficacy of the proposed framework. The obtained results confirmed that the proposed framework with MARL is effective and efficient.

Keywords –

Building Information Modeling; Reinforcement Learning; Multi-agent; Rebar Design; Clash Resolution; RC Joints

1 Introduction

Rebar design is a mandatory and important stage in reinforced concrete (RC) structures construction projects. According to Chinese design codes for RC members design (GB50010-2010), the design of rebar has to meet the seismic and the bearing capacity requirements of RC members. Besides, rebar design needs to be constructed easily, safe and cost effective. Since the rebar are densely located and the arrangement rules for rebar are extremely complex as per design codes, it is impractical, labor-intensive and error-prone for the designers to manually avoid all collisions (hard clash) or congestions (soft clash) in the RC structures even using computer software [1]. In addition, current clash detection software like Autodesk Navisworks Manage and Solibri Model Checker, have realized detection and visualization of the clash members [2]. However, the current software mainly focuses on the clash identifications of construction members after the design stage. It cannot automatically avoid the clash of the rebar or offer implementation resolution for solving clashes, which thus are lack of automatic arrangement in rebar design.

Recently, building information modeling (BIM) has been widely in the current Architecture, Engineering and Construction (AEC) industry. BIM technology allows us to represent the detailing of rebar digitally and transfer the detailing information to structural analysis software [5]. However, automated resolution of rebar clashes is lacking in the existing BIM software packages. Therefore, developing a framework for solving the problem of clash detection and resolution for automated design of rebar connects with the exiting BIM technology will be significant value in the AEC industry.

Various researchers in the past have tried to solve the problem of the clash detection and resolution for automated design of rebar with the aid of BIM technology, in the literature, Park [3] developed a BIM-based simulator to determine the sequence of rebar placement, and the clashes of rebar were identified by a developed application programming interface. Nevertheless, it focused on the simulation of the placement sequence and the spatial clash has to be solved manually. Next, Radke *et al.* [4] proposed an identification and resolution for mechanical, electrical and plumbing (MEP) systems. The offered resolution was moving one of the two clash entities to solve spatial conflicts. In fact, design constraints were not verified after moving one object. Besides, it provided manual resolution for resolving limited types of clashes. Moreover, Wang *et al.* [2] carried out a knowledge representation for spatial conflict coordination of MEP systems. The clash knowledge representation included description, context, evaluation and management details. However, the developed presentation pattern only provided a documentation to store clash information without any clash resolution strategy for identified clashes. Mangal and Cheng [5] proposed a framework based on BIM and genetic algorithm (GA) to realize rebar design and avoid clash at RC beam-column joints. However, the proposed framework only offered clash resolution strategy for moving components by using GA and can only applied for regular shaped RC structures. In particular, the optimized path of rebar cannot bend to avoid the obstacles, which thus limits its practicability in real-world complex RC joints. The main drawbacks of existing approaches can be summarized as follows: (1) Due to the complex design codes of rebar, most of the above studies cannot meet design constraints after avoiding clash by moving one object. (2) Most of the above studies lack automatic and intelligent identification and resolution of rebar clash for real-world complex RC joints and frame structures.

In machine learning, in light of its strength, RL algorithms have achieved many important achievements in the field of complex adaptive systems such as mobile robot path planning. What's more, a MARL system can lead to greater level of adaptivity and effective problem-solving [6]. Furthermore, the clash detection and resolution problem for the rebar design can be treated as a path planning of multi-agents in order to achieve automatic arrangements and bending of rebar to avoid obstacles. The similarity between the path planning of multi-agent and the arrangement of rebar, enlightens our work in this paper. Therefore, we propose a framework via a MARL system with BIM for automatically and intelligently provide clash resolution of rebar design in RC frames. To the best of our knowledge, this is the first modeling clash detection and resolution problem for the rebar design as a path-planning of multi-agent in the literature.

In particular, the three-dimensional coordinate information of the clash free rebar is then obtained by collecting the traces of the agents, considering longitudinal tensile, longitudinal compressive and shear rebar. To evaluate the efficiency and effectiveness of the proposed MARL, comprehensive experiments about one-story RC building frame having RC beams, columns, beam-column joints and beam-beam joints, including 63 RC beams and 23 RC columns with 1120 longitudinal rebars. Lastly, the obtained results including the success rates confirm that the proposed system is effective and efficient.

The contributions of the present study can then be summarized as follows:(1) To the best of our knowledge, this is the first modeling clash detection and resolution problem for rebar design as a path-planning problem of multi-agent in the literature. (2) To achieve automatic rebar design in complex RC joints, by employing FALCON as the reinforcement learning engine, we design the particular form of state, action, and rewards for the reinforcement MARL. (3) Comprehensive experiments on one-story RC building frame are performed to verify the effectiveness of the proposed framework.

2 Preliminary

Section 2.1 introduces the basic module of reinforcement learning. In Section 2.2, we describe the formulation of multi-agent path planning for rebar clash problem at RC beam-column joints. Section 2.3 presents rebar spacing requirements for RC members.

2.1 Introduction to Reinforcement Learning

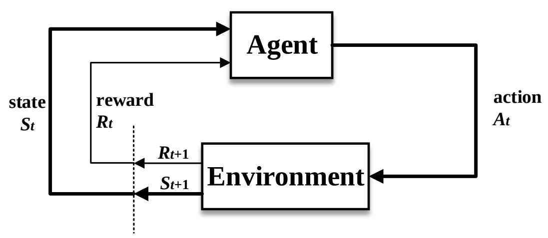


Figure 1. Basic Module of Reinforcement Learning

A multi-agent reinforcement learning (MARL) system [6] can be developed as effective tools for path planning problem-solving. RL is a natural learning paradigm to both single-agent and multiagent-agents. It 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 [6]. In each cycle, the agent obtains sensory input from its environment representing the current state (**S**), performs the most appropriate action (**A**) and then receives feedback in terms of rewards (**R**) from the environment. It is important to note that how to turn a real-world environment into digital environment with clear reward signals is a key point to carry out RL.

2.2 Formulating Rebar Design as Path Planning of Multi-agent System

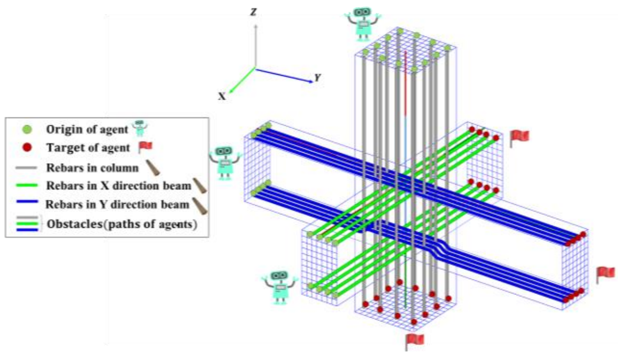


Figure 2. Problem formulation for RC beam-column joint

In particular, by treating each rebar as an intelligence reinforcement learning agent, we propose to model the rebar design problem as a path-planning problem of multi-agent system. It can be further modeled with a team of agents tasked to navigate towards defined targets safely across a RC beam-column joint that is gradually filled with obstacles which are rebars generated in the previous steps. available. In this task, the unmanned vehicle can choose one of the five possible actions, namely, up, down, forward move, left, and right at each discrete time step. The task or objective of the agent is to navigate successful through the joint towards assigned targets within the stipulated time, without hitting any obstacle. With the proposed MARL, the three-dimensional coordinates of the clash free rebar design are then obtained by collecting the traces of the agents.

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

2.3 Formulating Steel Rebar Spacing Requirements for RC Members and Origins and Targets for Agents

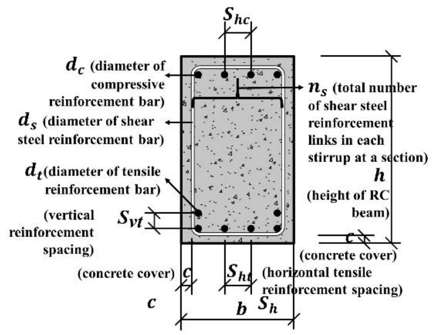


Figure 3. Spacing between rebar in an RC beam

In order to pour concrete easily and ensure the compactness of concrete around rebar, the spacings between longitudinal rebar are determined as per the provisions of GB50010-2010 (Figure 2). $S_{hc}$is horizontal compressive spacing for longitudinal compressive rebar, which is specified as $S_{hc} \geq 30$ and $\geq 1.5 d_{c,max}$. In addition, S_{ht} is horizontal tensile spacing for longitudinal tensile rebar, which is specified as $S_{ht} \geq 25$ and $\geq d_{t,max}$.

N_t denotes the total number of tensile rebars, and $n_{t,min} \leq N_t \leq n_{t,max}$.

$$n_{t,min} = (b - 2 \times c) / S_{ht,max} \quad (1)$$
$$n_{t,max} = (b - 2 \times c) / S_{ht,min} \quad (2)$$

Where b denotes the width of RC beam, and c denotes the concrete cover. $S_{ht,max}$ and $S_{ht,min}$ are the maximum and minimum spacing between tensile rebars, respectively. Further, $S_{ht,max}$ and $S_{ht,min}$ are used in MARL to decide the origins of agents in each mission.

Similar to the longitudinal tensile rebar, spacing demands for compressive rebar are straight forward and require no explanation.

The calculations of spacing demands in RC column design are similar to those in RC beam design and require no explanation. The spacings between longitudinal steel reinforcement bars S_h are decided as per the provisions of GB50010-2010(Figure 3), and 50mm$\leqslant S_h \leqslant$300mm. Specifically, when the width of column is more than 400mm, the spacings between longitudinal rebars S_h must to be less than 200mm [1].

The origins of agents in each mission are decided by the $S_{ht,max}$ and $S_{ht,min}$, meanwhile the targets of agents in each mission are also decided by the $S_{hc,max}$ and $S_{hc,min}$ as per the provisions of GB50010-2010, as illustrated in subsection 2.3.

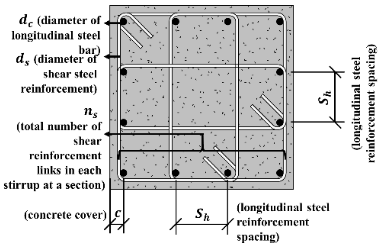


Figure 4.Spacing between rebar in an RC column

3 Proposed MARL system with BIM Solving the Path-Planning Problem for Clash Free Rebar Design

The presented framework for clash free rebar design is based on Mangal and Cheng [5] framework. The framework consists of 4 modules named (a) BIM Model Extraction, (b) Structural Analysis, (c) Structural Type Analysis, (d) Multi-Agent Reinforcement Learning System shown in Figure 5. The first three modules are clearly clarified in Mangal and Cheng [5] framework. The last module is explained in the following subsection.

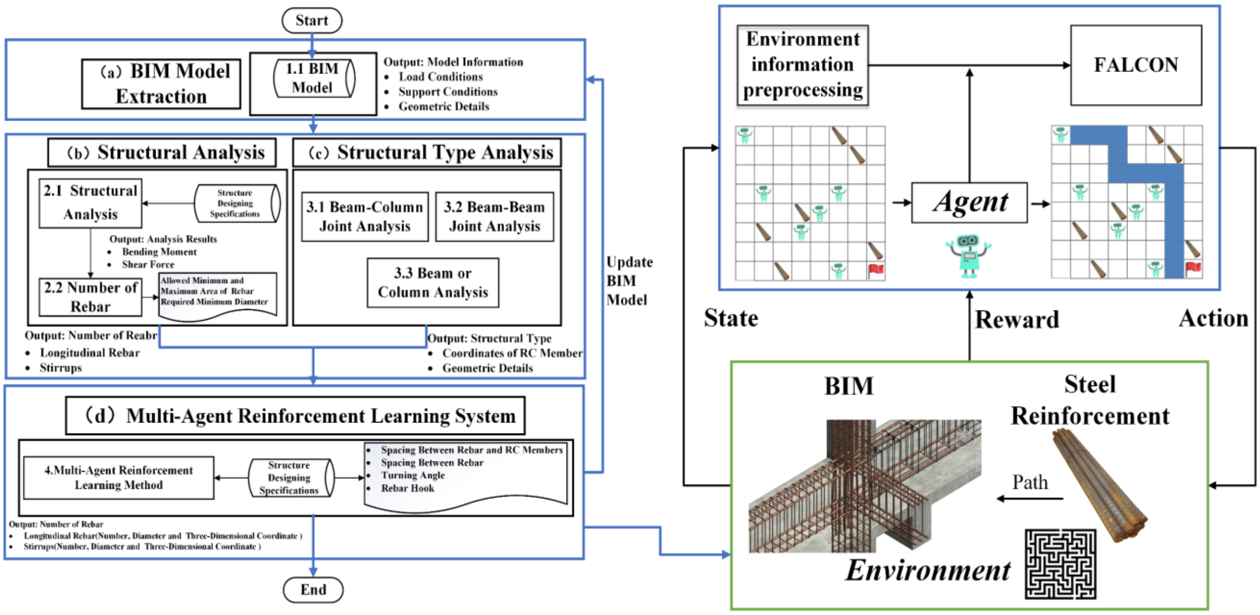


Figure 5. The presented framework for clash free rebar design via MARL with BIM.

3.1 Environment Information Pre-processing of RC Members

In addition, the RC members have to be transformed into a digital environment that is suitable for MARL system. Furthermore, we transform the BIM model of RC member into tessellated mesh environments approximating the geometry of the RC members with known boundary conditions, as illustrated in Figure 6. Then in tessellated mesh environment, a team of agents tasked to navigate towards defined targets safely.

Each tessellated mesh dimension Di of environment is the dimension of a single square mesh, which can be calculated as:

$$Di = \min(d_c \text{ } and \text{ } d_t) \quad (3)$$

Where d_c denotes the diameter of longitudinal compressive rebar, and d_t denote the diameter of longitudinal tensile rebar.

Therefore, the size of tessellated mesh environment Sz depends on Di and the dimension of RC members,

$$Sz = floor(D/Di) \quad (4)$$

Where D denotes the dimension of RC members, which can be length, width and height of RC members, and $floor()$ denotes the integer rounding down function in order to limit the range of Sz.

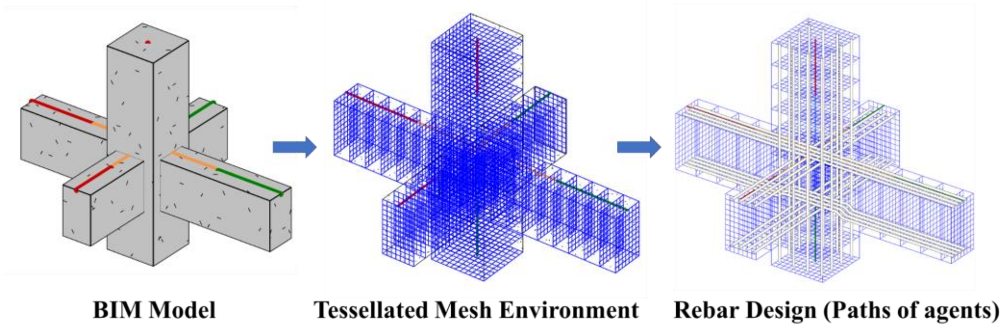


Figure 6. Environment information pre-processing of RC members

3.2 Neural Network Architecture of Each Agent

The architecture of each agent takes the form of FALCON [6], which has a three-channel neural network architecture (Figure 7), consisting of three modules: (1) State: a sensory field F_1^{c1} for saving and representing current agent states, (2) Action: a motor field F_1^{c2} for representing available actions, and (3) Reward: a feedback (reward) field F_1^{c3} for representing the internal states of an agent, as well as external feedbacks from the environment. It 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.

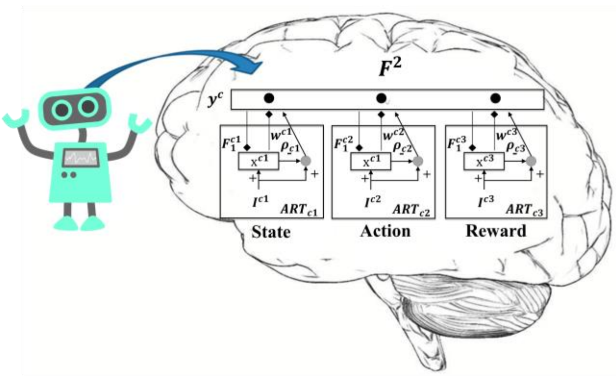


Figure 7. Neural network architecture of each agent

3.2.1 State Module

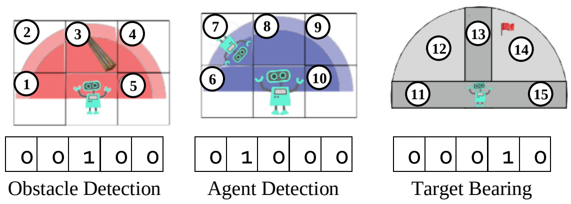


Figure 8. Illustration of states in 2D

MARL system involves numbers of agent equipped with a set of sonar sensors that has a 180° forward view. Meanwhile, input attributes of sensory (state) vector consist of obstacle (path of other agent) detection, other agent position detection and the bearing of the target from the current position. Therefore, without a priori knowledge of the three-dimensional coordinate information of the obstacle and targets, each agent is equipped with a localized view of its environment.

3.2.2 Action Module

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

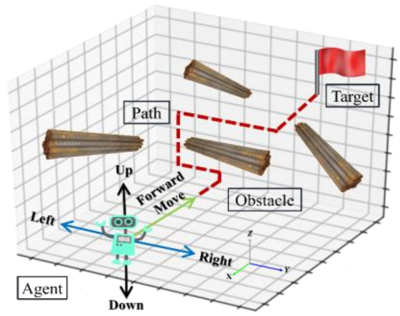


Figure 9. Illustration of five possible actions

3.2.3 Reward Module

In MARL, the design of reward, punishment and some specific strategies are presented for build-ability constraints. In particular, the reward and punishment strategies are described in Table.1:

Table 1. Reward and punishment strategies for agents.

Reward and Punishment 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

A reward of +1 is given when the agent reaches the target without hitting obstacles and 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 search for defined targets. A punishment 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 clash of rebar. A punishment of -1 is given when the agent moves into the specified range (1.5 × diameter of rebar) of paths or positions of other agents, therefore the spacing demand of rebar is satisfied. A punishment of -0.5 is given when the agent takes actions including left, right, up and down in order to make sure agent to move as straight as possible, therefore the layout of rebar 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.

Figure 10. Illustration of multi-agent path planning including reward, punishment and mission endings in 2D

3.3 FALCON in MARL

The architecture of FALCON based on 3-channel Adaptive Resonance Associative Map (multichannel ARAM) [6], an extension of predictive Adaptive Resonance Theory (ART) networks (Figure 7).

Input vectors: Let {**S**,**A**,**R**} denote the input vector, where **S** = $(s_1, s_2, \dots, s_n)$ denotes the state input, and s_i indicates the value of sensory input i ; **A** = $(a_1, a_2, \dots, a_n)$ denotes the action vector, and a_i indicates a possible action i; **R** = (r) denotes the reward vector, and $r \in [-1,1]$is the reward signal value.

Activity vectors: Let x^{ck} denote the F_1^{ck}activity vector. Let y^c denote the F_2^c activity vector.

Weight vectors: Let w_j^{ck} denote the weight vector associated with the jth node in $F_2^c$for learning the input representation in F_1^{ck} . Initially, all F_2^c nodes are uncommitted, and the weight vectors contain all 1's.

Parameters: The FALCON's dynamics is determined by choice parameters $\alpha^{ck} > 0$ for $k = 1, \dots, K$; learning rate parameters $\beta^{ck} \in [0;\ 1]$ for $= 1, \dots, K$; contribution parameters $\gamma^{ck} \in [0;\ 1]$ for $= 1, \dots, K$; and vigilance parameters $\rho^{ck} \in [0;\ 1]$ for $= 1, \dots, K$.

3.3.1 From Sensory to Action

Given the state vector **S**, the system performs code competition and selects an action based on the output activities of action vector **A**. The detailed algorithm is presented below.

Code activation: Given activity vectors $x^{c1}, x^{c2}, \dots, x^{cK}$ for each F_2^c node j, the choice function T_j^c is computed as follows:

$$T_j^c = \sum_{k=1}^{K} \gamma^{ck} \frac{|x^{ck} \wedge w_j^{ck}|}{\alpha^{ck} + |w_j^{ck}|} \tag{5}$$

Code competition: All F_2^c nodes undergo a code competition process. The winner is indexed at $J$where

$$T_J^c = \max\{T_j^c : \text{for all } F_2^c \text{ node } j\} \tag{6}$$

Action selection: The chosen F_2^c node J performs a readout of its weight vector to the action field F_1^{c2}such as

$$x^{c2} = w_J^{c2} \tag{7}$$

The chosen action a_I is then determined by

$$x_I^{c2} = \max\{x_i^{c2} : \text{for all } F_1^{c2} \text{ node } i\} \tag{8}$$

3.3.2 From Feedback to Learning

Upon receiving a feedback from its environment after performing the action a_I, the system adjusts its internal representation based on the following principles. Given a reward (positive feedback), the agent learns that an action executed in a state will result in a favorable outcome. Therefore, the system learns to associate the state vector **S**, the action vector **A**, and the reward vector **R**.

Template matching: Before code J can be used for learning, a template matching process checks that the weight templates of code J are sufficiently close to their respective input patterns. Specifically, resonance occurs if for each channel k, the *match function* m_J^{ck} of the chosen code J meets its vigilance criterion:

$$m_J^{ck} = |x^{ck} \wedge w_J^{ck}| / |x^{ck}| \geq \rho^{ck} \tag{9}$$

Learning then ensues, as defined below. If any of the vigilance constraints is violated, mismatch reset occurs in which the value of the choice function T_J^c is set to 0 for the duration of the input presentation. The search process repeats to select another F_2^c node J until resonance is achieved.

Template learning: Once a node J is selected for firing, for each channel k, the weight vector $w_J^{ck}$is modified by the following learning rule:

$$w_J^{ck(new)} = (1-\beta^{ck})w_J^{ck(old)} + \beta^{ck}|x^{ck} \wedge w_J^{ck(new)}| \tag{10}$$

3.4 Proposed MARL System

Algorithm 2 Pseudo Code of MARL System

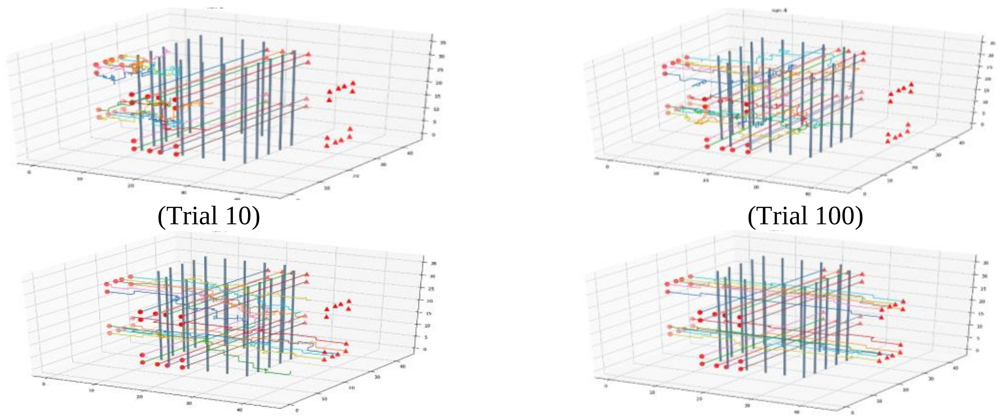
Initialization: Generate the initial m agents
While (a mission ending conditions are not satisfied)
For each agent
    If (agent dose not fail or not arrive the target)
        Perform FALCON algorithm
    Else
        Stop training
    End If
End For
End While

The basic steps of the proposed MARL system are outlined in Algorithm 2. In the first step, a population of m agents is initialized. An agent fails when hitting obstacles, exceeding 30 sense-act-learn cycles (running out of time). A mission ends when all agents fail or arrives at the target successfully. A mission will also be deemed to have failed if an agent collides with another, as depicted in Figure 10.

4 Empirical Study

4.1 The Example of Training Process in RC Beam-Column Joint by MARL

In the initial stage of mission as shown in Figure 11, agents 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 looks messy, cluttered or indirect in trial 10 and 100. In the late stage of mission such as trial 500 and 1000, agents converge gradually to the global optimum and find the optimum paths for the clash free rebar design. Furthermore, along with the experimental training, the paths of agents have also gone from chaos to the gradual and orderly process of development. Finally, the global optimum of the agents' path will be selected to generate the clash free rebar design.



(Trial 10) (Trial 100)

(Trial 500) (Trial 1000)

Figure 11. The training process in beam-column joint by MARL

4.2 The Example of RC Frame

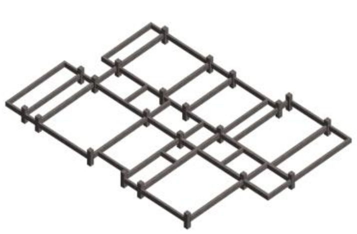


Figure 12. one-story frame

In this section, the empirical study is established to study the effectiveness of the proposed model. One illustrative example about one-story frame as shown in Figure 12 will be used to test the proposed model. In this tested frame, there are 63 RC beams and 23 RC columns with 1120 longitudinal rebars.

An agent having reached the target without hitting obstacles or running out of time is defined as a success. The success rate S_r can be calculated by Eq. 1:

$$S_r == \frac{1}{Nm} \times \sum_{i=1}^{N_m} \frac{N_t^i}{N_t} \times 100\% \tag{7}$$

where N_m denotes the number of total missions, N_t denotes the total number of participating agents and N_t^i stands for the number of agents that reach the targets successfully in mission i without hitting obstacles.

We analyzed the averaged success rate S_r on 40 simulations on the proposed MARL. The averaged success rates S_r presented in Figure 13 indicate that the proposed MARL successfully solving the path planning problem of rebar design after training. The design of reward, punishment and some specific strategies can satisfy build-ability constants.

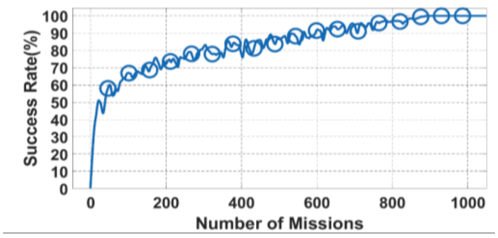


Figure 13. The success rates of MARL system

The automated 3D BIM outputs of the rebar in RC frame are given in Figure 14 and 15. Design of rebars of the RC frame are based on the result of the proposed system. It can be observed that there is no rebar clash in RC beam-column joints and frame by clash detection of the 3D BIM output.

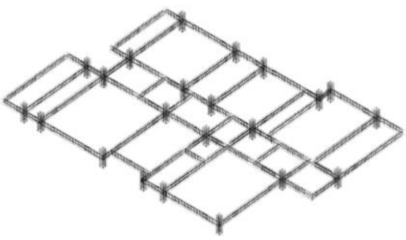


Figure 14. The simulation result of considered one-story RC frame

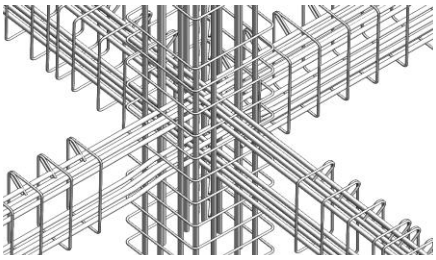


Figure 15. The simulation result of considered one-story RC frame

5 Conclusions and Future Research

In this paper, we model the clash detection and resolution problem for clash free rebar design as a path planning problem of multi-agents in order to achieve automatic arrangements and bending of rebar to avoid obstacles. Therefore, a framework via MARL system with BIM has been proposed to identify and avoid rebar spatial clash in complex RC frames. The three-dimensional coordinate information of the clash free rebar including bending of rebar is then obtained by collecting the traces of the agents. The design of reward, punishment and some specific strategies for build-ability constants are also put forward in MARL. Next, according to FALCON, the agent selects the suitable action and reaches the defined targets without hitting obstacles or running out of time. Subsequently, agents converge gradually to the global optimum along with the experimental training. Finally, the paths of agents are extracted to BIM model generating the rebar design. The simulation study in terms of the success rate have shown the effectivity and efficiency of the proposed system on the design of rebar in one-story RC frame.

However, the proposed framework via MARL system with BIM still has a few limitations. (1) It only applied for regular RC beams, columns, beam-column joints and beam-beam joints. (2) Furthermore, the system only considered the design codes GB50010-2010. Therefore, extending the system for more complex RC members or fabricated RC members and other design codes will be considered in the future work.

References

[1] Tabesh, A. Reza, and Sheryl Staub-French. Case study of constructability reasoning in MEP coordination. *Construction Research Congress 2005: Broadening Perspectives*. 2005.

[2] Wang L. and Leite F. Formalized knowledge representation for spatial conflict coordination of mechanical, electrical and plumbing (MEP) systems in new building projects. *Automation in Construction*, 64 (1): 20-26, 2016.

[3] Park U. BIM-Based Simulator for Rebar Placement. *Journal of the Korea Institute of Building Construction*, 12 (1): 98-107, 2012.

[4] Radke A., Wallmark T. and Tseng M.M. An automated approach for identification and resolution of spatial clashes in building design. In *IEEM 2009-IEEE International Conference on Industrial Engineering and Engineering Management*, pages 2084-2088, Hong Kong, 2009.

[5] Mangal, M., and Cheng, J. C. Automated optimization of steel reinforcement in RC building frames using building information modeling and hybrid genetic algorithm. *Automation in Construction*, 90, 39-57,2018.

[6] Feng, L., Ong, Y. S., Tan, A. H., and Chen, X. S. (2011, June). Towards human-like social multi-agents with memetic automaton. In *2011 IEEE Congress of Evolutionary Computation (CEC)*, pages 1092-1099, 2011.