Fire Evacuation Path Planning Based on Improved MADDPG (Multi-Agent Deep Deterministic Policy Gradient) Algorithm

Qiong Huang[1], Ying Si[2], Haoyu Wang[3]*
Department of Basic, China Fire and Rescue Institute, Beijing, 102202, China[1, 2]
Department of Fire Engineering, China Fire and Rescue Institute, Beijing, 102202, China[3]

***Abstract*—The lack of a scientific and reasonable optimal evacuation path planning scheme is one of the main causes of casualties in fire accidents. In addition to the high temperature and harmful smoke in the fire environment, the crowding problem caused by the change of the position of the crowd in the evacuation process will also affect the evacuation effect. Therefore, by improving the multi-agent depth deterministic strategy gradient algorithm, an AMADDPG (Adjacency Multi-agent Deep Deterministic Policy Gradient) model suitable for fire evacuation is proposed. First, the dangerous grid area is defined, and the influence of congestion degree and nearest exit is considered at the same time. The learning framework of "distributed execution and centralized local learning" is adopted to realize experience sharing among neighboring agents. Improve the learning efficiency and evacuation effect of the model. The experimental results show that the model can basically adapt to the complex and dynamic fire environment well, achieve the optimal path planning within 30, and ensure that the degree of congestion on the evacuation path is maintained within 0.5, which can achieve the safe evacuation goal. Meanwhile, compared with the MADDPG algorithm, the model has obvious advantages in terms of training efficiency and stability. It has good application value.**

Keywords*—Fire evacuation path; congestion degree; dangerous grid; multi-agent; Multi-Agent Deep Deterministic Policy Gradient

I. INTRODUCTION

In recent years, fire accidents occur frequently and become one of the major disasters threatening public safety, which not only brings huge property losses, but also often causes serious casualties. According to statistics, 825,000 fires were reported in China in 2022, with 4,175 casualties and direct property losses of 7.16 billion. The main causes of casualties are the sudden occurrence of fire, the blindness of crowd evacuation and the lack of scientific measures to guide crowd evacuation [1]. It is an urgent problem to provide reasonable and feasible optimal evacuation path planning scheme for trapped personnel, improve the safety evacuation efficiency of the crowd, and maximize the reduction of casualties in the fire accident.

At present, many scholars have carried out a lot of research on fire evacuation path planning. Zhong and Yu [1] proposed the idea of building a real-time fire evacuation system for smart cities based on the Internet of Things, using Floyd algorithm and building topology to plan the optimal evacuation path for personnel; Ye and Pan [2] proposed a path planning intelligent model based on BIM (Building Information Modeling) and cellular automata, and added dynamic obstacle model and random catastrophic fire model to the fire field, which can scientifically and efficiently avoid static and dynamic obstacles; Choi and Chi [3] used the smoke propagation prediction data provided by the fire dynamics simulator to improve the A* algorithm [4-5] on the basis of considering the safety status of subsequent nodes in the path, so as to find the optimal evacuation path and improve the algorithm; Liang and Wang [6] targeted the comprehensive building fire, considering the effects of fire products and crowd density on personnel escape speed, a personnel evacuation path planning model based on improved ant colony algorithm [5,7-8] was constructed; Dong et al. [9] applied the combination empowerment method to assign reasonable evacuation priorities to different crowd gathering points in view of indoor fires in commercial buildings, and optimized Dijkstra algorithm [10-12] to solve the congestion problem on the evacuation path.

The traditional path planning algorithms above are mostly based on static scenes and require complete evacuation environment information, which is not consistent with the actual evacuation situation. DRL (Deep Reinforcement Learning) algorithm [13] is one of the hotspots in the field of artificial intelligence research and is suitable for solving complex decision problems in unknown environments [14]. It has been applied to many fields such as robot control [15], military deduction, path planning [16-17], etc. Ni et al. [18] proposed a collaborative double-depth Q network algorithm to obtain good path planning results through the interactive learning experience between agents and dynamic environments in multi-exit fire scenarios. Zhang et al. [19] combined Deep Reinforcement Learning with multiple agents to improve the global guidance strategy and neural network structure, so as to be suitable for personnel evacuation in complex dynamic and multi-exit environments. Although the above crowd evacuation planning method has effectively solved the evacuation path optimization problem of the dynamic change of the fire danger area over time in the multi-exit fire scenario in practical application, there are still some problems, such as not considering the congestion caused by the change of the crowd position status in the evacuation process, and the punishment for deviating from the nearest exit in the multi-exit fire scenario. Therefore, this paper proposes an AMADDPG (Adjacency Multi-agent Deep Deterministic Policy Gradient) model suitable for fire evacuation. Through mathematical modeling of fire path planning, The MADDPG (Multi-agent Deep Deterministic Policy Gradient) algorithm is improved to realize evacuation under complex dynamic fire environment.

This study makes two primary contributions. First, the danger grid, congestion degree, and distance from the agent to the exit in fire evacuation path planning are mathematically defined, and the fire evacuation is modeled as a reinforcement learning problem in multi-agent environment. Through the definition of multi-agent state space and action space as well as reward function, the optimal fire evacuation path planning is realized to maximize the reduction of personnel crowding, avoidance of dangerous areas, and multi-exit fire scenarios. Second, the learning framework of "distributed execution and centralized local learning" is adopted to reduce the complexity of network training and show obvious advantages in training speed and system stability.

The rest of this article is organized as follows. Section I is given to model the fire environment and establish the mathematical model of fire path planning. Section II introduces the basic principle of MADDPG algorithm and the specific implementation of its improved algorithm AMADDPG. Section III introduces the construction of the fire environment of the fire evacuation experiment. In Section IV, the results of the fire evacuation experiment are analyzed. Discussion is given in Section V. Finally, in Section VI, the main research results of this study and the next research plan are summarized.

II. Material and Research Method

A. Problem Description and Modeling

In a multi-exit fire environment, personnel in each room of the building must avoid dangerous roads such as high temperature and heavy smoke, and quickly arrive at the nearest exit under the guidance of reasonable evacuation path planning to minimize casualties and property losses. To facilitate the research, the following assumptions are made:

1) The building is simplified into a two-dimensional finite plane space where the location of obstacles and safety exits is known.

2) Fire site information can be obtained in real time through sensor devices, such as temperature, smoke and toxic gas detection.

3) All evacuees in each room are regarded as one agent, and each agent is numbered as $\{Agent_i, i=1,2,3\ldots,n\}$, and the initial location of each agent is known, ignoring the impact of individual differences of evacuees on the speed of personnel movement.

a) Environmental modeling: In this paper, the grid method [20] is adopted to model the building plan, which is divided into several grids of equal size and non-overlapping. Each grid represents a feasible area or obstacle area with a length of 1m. In the feasible area, personnel can move freely, which is represented by white grid. While in the obstacle area, personnel cannot pass through, which is represented by black grid, usually a wall, column of a building. Grid coordinates increase from left to right, from bottom to top, and are represented by their center point coordinates. Fig. 1 shows the building plan created using the grid method. The grid coordinates in the lower left corner are (0, 0), and, the grid coordinates in the upper right corner are (m-1, n-1), and m, n are the number of grids in the horizontal and vertical directions, respectively. When a fire occurs, if the temperature, smoke visibility and toxic gas volume concentration in the feasible area grid exceed the preset critical value, it becomes an impassable dangerous grid.

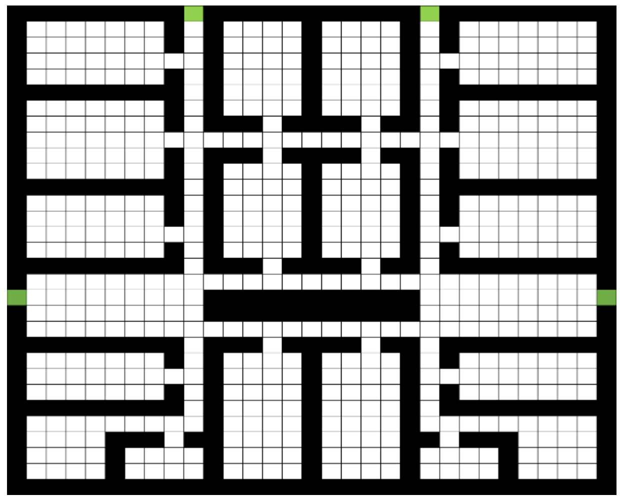


Fig. 1. Building modeling with grid method.

b) Mathematical model of fire path planning: (1) Definition of dangerous grid: In a fire environment, fire products such as the volume fraction of toxic gas (CO), smoke visibility, and temperature will affect the life safety of evacuees. If the evacuation path planning guides personnel to enter dangerous areas, casualties may be caused. Therefore, according to the effect of CO volume fraction, smoke visibility and temperature on human body in fire [21-22], the dangerous grid is defined,

$$G_{xy} = \{(x,y) \mid 0 < VIS < 3 \cup \varphi \ \ (CO) \geq 0.5 \cup T_s \geq 70\} \quad (1)$$

where, G_{xy} is the actual state of the grid at coordinates (x, y), *VIS*, $\varphi(CO)$ and T_s are smoke visibility, CO volume fraction and ambient temperature in the grid respectively.

1) Congestion degree

The congestion degree of evacuation directly affects the speed of movement and evacuation time of personnel, and will greatly reduce the efficiency of evacuation in serious cases. In order to represent the congestion degree of evacuees in a certain area during evacuation, the concept of congestion degree [23] is introduced to reflect the congestion of evacuation channels with time and space dimensions.

$N(t)$ is defined as the number of evacuees in the evacuation channel at evacuation time *t*, $C(t)$ is the passage capacity of the channel,

$$C(t) = \frac{SA_q}{\pi r^2} \tag{2}$$

where SA_q is the area of the evacuation channel, and the evacuees are regarded as circular particles, r is 1/2 of the normal shoulder width of people, usually takes a value of 0.25 m. $c = N(t)/C(t)$ represents the congestion degree of the evacuation channel at time *t*. The greater the value, the more serious the congestion degree in the corresponding channel. When $c \leq 0.5$, the interference between evacuees is small and has no impact on the evacuation process and efficiency. But when $c > 0.5$, the evacuation pedestrians began to be crowded, and the degree of congestion increased exponentially with the increase of saturation.

Calculate the congestion degree c_t^i of the $Agent_i$ located at coordinate (x^i, y^i) at time *t* in evacuation channel *P*. For the convenience of the research, assume that the evacuees in each agent are evacuated in a one-line formation, $(x^i, y^i) \in P$. The area of evacuation channel is denoted as SA_{real}, and N^i is the number of evacuees of the $Agent_i$. Consider it in the following two cases:

i) When there is no other agent in the evacuation channel, $N(t) = N^i, SA_q = SA_{real}$.

ii) Otherwise, $N(t) = N^i + \sum_{j=1}^{k} N^j$, $SA_q = SA_{real}$, where N^j is the number of evacuees of the $Agent_j$ located at coordinates (x^j, y^j), $(x^j, y^j) \in P$.

2) The distance between the agent and the exit

For the need of fire safety, the Code for Fire Protection in Building Design expressly stipulates the number of safety exits for public buildings: each fire protection zone or each floor of a fire protection zone in a public building shall have no less than 2 safety exits. In the multi-exit fire environment, when the number of evacuees and the initial location are determined, the choice of the nearest exit must be considered to achieve the optimal path planning. In this paper, Manhattan distance is used to define the distance between evacuees and each exit. Assuming that the $Agent_i$ is located at the evacuation grid at coordinates (x^i, y^i), and define the distance between the j exit at coordinates (x^j, y^j) is defined, $d_j^i = \left|x^i - x^j\right| + \left|y^i - y^j\right|$. And for the exit set $\{e_j, j = 1,2,3,\ldots,m\}$, the exit number closest to the evacuee $Agent_i$ is $q = \arg\min_j \{d_j^i, j = 1,2,3,\ldots,m\}$, the coordinate is (x^q, y^q).

B. Principle of Algorithm

1) MADDPG algorithm: Multi – agent Deep Deterministic Policy Gradient algorithm (MADDPG) applicable to traditional reinforcement learning method to handle the multi-agent cooperation task [24], by empirical playback mechanism and "centralized training, distributed execution" framework to learn. As shown in Fig. 2, each agent has an Actor network and a Critic network. During the training process, each agent interacts with the environment through its own Actor network according to the local information of its own state, to obtain action strategies, and evaluates the action of the Actor network according to the global information of the action state of all agents through the Critic network. This network structure effectively improves the policy stability and robustness of multi-agent systems.

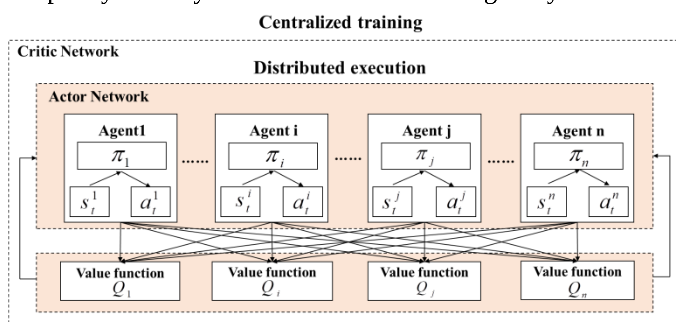


Fig. 2. Multi-agent depth deterministic strategy gradient algorithm MADDPG.

a) State space: In the process of fire evacuation, the location of evacuees and the congestion degree of adjacent areas have an impact on the choice of evacuation path, therefore, the state s_t^j of $Agent_i$, $i=1,2,3\ldots,n$ at time t as $(x_t^i, y_t^i, c_t^{i1}, c_t^{i2}, c_t^{i3}, \ldots, c_t^{i8}, d_t^{i1}, d_t^{i2}, d_t^{i3}, \ldots, d_t^{im})$, where x_t^i, y_t^i represent the horizontal and vertical coordinates of the $Agent_i$ position; $c_t^{i1}, c_t^{i2}, c_t^{i3}, \ldots, c_t^{i8}$ represents the congestion degree of the 8 areas closest to the $Agent_i$. If one area is an obstacle, the congestion degree of the corresponding area is set to infinity. $d_t^{i1}, d_t^{i2}, d_t^{i3}, \ldots, d_t^{im}$ represents the distance between the $Agent_i$ and m exits.

b) Action space: During fire evacuation, on the basis of environmental rasterization, the agent can select actions according to the observed environmental state information in 8 directions around it. Therefore, this paper defines that the $Agent_i$ can select actions in eight directions (up, down, left, right, upper left, upper right, lower left and lower right) in any state at time t, denoted as a_t^j.

c) Reward function: The reward function is an important reference for the agent to judge its own strategy, and it affects the learning effect and convergence speed of the algorithm to some extent. According to the goal of path planning, which is to find the shortest distance between all agents and the nearest exit on the basis of minimizing overcrowding and avoiding dangerous areas. In this paper, for each action performed by the $Agent_i$, if the agent is in a non-free active grid, a negative reward $R_a^i = -20$ is given; otherwise, whether the agent reaches the exit is judged. If yes, a larger positive reward $R_a^i = 100$ is given; otherwise, a negative reward $R_a^i = -1$ is given. In order to ensure the maximum reduction of personnel congestion during the evacuation process, the congestion degree C^i of the evacuation area of the $Agent_i$ is specified. When $C^i > 0.5$, negative reward $R_c^i = -\exp\left(c^i\right)$ is given; At the same time, in order to make the agent move towards the direction of the nearest exit and avoid entering the dangerous grid G_{xy}, suppose that the angle between the vector of $Agent_i$ from the position $\left(x_t^i, y_t^i\right)$ at time t to the position $\left(x_{t+1}^i, y_{t+1}^i\right)$ at time $t+1$ and the vector from its position $\left(x_t^i, y_t^i\right)$ at time t to the nearest exit $\left(x^q, y^q\right)$ is denoted as α, and the reward value R_e^i is divided into the following four cases, as shown in Table I. In conclusion, this paper defines the reward function as: $R^i = R_a^i + R_c^i + R_e^i$.

TABLE I. REWARD FUNCTION OF FIRE EVACUATION PATH PLANNING

	$(x_{t+1}^i, y_{t+1}^i) \notin G_{xy}$	$(x_{t+1}^i, y_{t+1}^i) \in G_{xy}$
$\alpha < 90^\circ$	$R_e^i = 0.5$	$R_e^i = -10$
$\alpha \geq 90^\circ$	$R_e^i = -1$	$R_e^i = -20$

2) Improved MADDPG algorithm: Lowe et al. [25] pointed out that the "distributed execution, centralized training" learning framework of MADDPG is suitable for multi-agent interaction scenarios. However, with the increase in the number of agents, the input dimension and training parameter scale of the centrally trained Critic network increase rapidly, which will greatly increase the training difficulty of the network. This makes it impossible to deal with large-scale multi-agent learning problems. In fact, in the process of fire evacuation, the action strategy of the agent is only affected by its surrounding environment and the agent close to it. Therefore, this paper proposes AMADDPG to improve it and adopts the learning framework of "distributed execution and centralized local learning", that is, only the status and action data of top-k other Agent Actors closest to the current Agent are considered as the input of the current Agent Critic network. The block diagram of AMADDPG algorithm is shown in the Fig. 3.

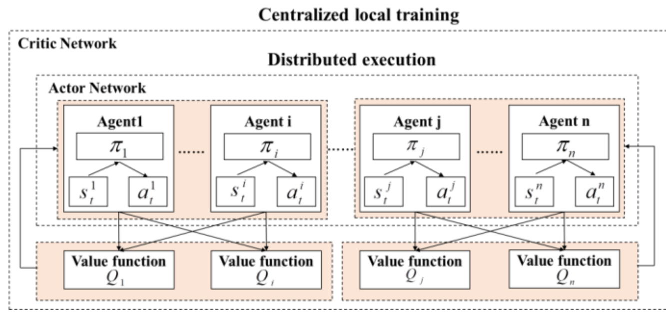


Fig. 3. Block diagram of improved AMADDPG algorithm.

The Actor of each $Agent_i$ independently uses local information to complete the interaction with the surrounding environment. During model training, it maximizes the cumulative expected return $J\left(\theta_i\right) = Q\left(s_t^i, s_t^{kNN(i)}, a_t^i, a_t^{kNN(i)}\right)\Big|_{a_t^j = \mu^j\left(s_t^j\right)}$ and minimizes the loss function of the locally centralized action value function in their respective Critic networks as follows:

$$L\left(\theta_i\right) = E\left[\left(Q_i^\mu\left(s_t^i, s_t^{kNN(i)}, a_t^i, a_t^{kNN(i)}\right) - y\right)^2\right] \quad (3)$$

$$y = Ri + \gamma Q_i^\mu\left(s_{t+1}^i, s_{t+1}^{kNN(i)}, a_{t+1}^{i'}, a_{t+1}^{kNN(i)'}\right)\Big|_{a_{t+1}^{j'} = \mu^{j'}\left(s_t^j\right)} \quad (4)$$

where $s_t^{kNN(i)}$、$a_t^{kNN(i)}$ indicate the status and actions of the top-k agents closest to the $Agent_i$.

The pseudo-code of the AMADDPG algorithm is as follows:

Initialize environment parameters, parameter variables;

for episode=1 to M do

Initialize random noise N;

Initialize the initial state of fire evacuation S0;

for t=1 to max- episode-length do

for agent i = 1 to n do

According to the state s_t^i, the random policy is used to perform an action a_t^i, get the immediate reward R_t^i, and reach the new state s_{t+1}^i;

Find the top-k other Agent sets $kNN(i)$ that are closest to.

Store $< s_t^i, s_t^{kNN(i)}, a_t^i, a_t^{kNN(i)}, R_t^i, a_{t+1}^i, a_{t+1}^{kNN(i)} >$ in the experience pool

end for

$S_t \leftarrow S_{t+1}, S_t = \{s_t^i, i = 1,2,3..,n\}$, $S_{t+1} = \{s_{t+1}^i, i = 1,2,3..,n\}$

for agent j = 1 to n do

Random sampling $< s_t^i, s_t^{kNN(i)}, a_t^i, a_t^{kNN(i)}, R_t^i, a_{t+1}^i, a_{t+1}^{kNN(i)} >$ from empirical pool.

Set the target Critic network function value

$$y = R^i + \gamma Q_i^{\mu'}(s_{t+1}^i, s_{t+1}^{kNN(i)}, a_{t+1}^{i'}, a_{t+1}^{kNN(i)'})|_{a_{i+1}^{j'} = \mu^{j'}(s_t^j)}$$

Minimize the loss function $L(\theta_i)$ update Critic.

Policy gradient $\nabla J(\theta_i)$ update Actor for calculating expected return.

end for.

Update target network parameters: $\theta_t' \leftarrow \alpha\theta_t + (1-\alpha)\theta_t'$

end for

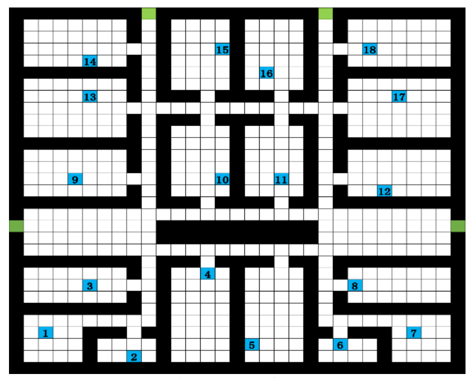
end for

III. Fire Environment Construction for Fire Evacuation Experiment

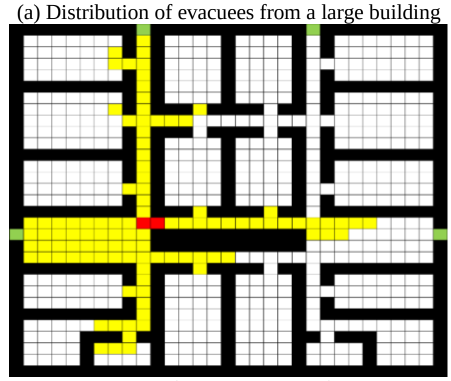
In order to verify the effectiveness of the AMADDPG algorithm on fire evacuation path planning, python 3.7 was used to simulate the algorithm. The hardware configuration of the experiment environment is as follows: the CPU is Intel Xeon (R) Bronze 3104, the operating system is Windows Sever2012 R2, and the deep learning framework is Pytorch 1.4.

In this experiment, the evacuation situation of a large building fire scene is simulated. As shown in Fig. 4(a). The building has an area of approximately 900 m^2 and constructs grid maps with dimensions of 30 × 30, each grid side length of 1m. There are four safety exits in the building, as shown by the green grid in the figure. The blue grid in the figure is the initial position of the agents, which are numbered in turn. The number of evacuees represented by each agent is randomly generated at the beginning of each training round, and the total number of evacuees is about 200. When all agents reach the safety exit in the shortest time, it is considered as a successful evacuation.

The main materials of combustibles in buildings are wooden furniture and fabrics, and the fire area is an indoor environment in the building, as shown by the red grid in the Fig. 4(b). In order to make the simulation test more close to the real fire, the heat release rate changes according to the fast t square fire, and the maximum heat release rate is set to 4 MW/m2. It is assumed that the fire continues to maintain the burning state after reaching the maximum heat release rate. The continuous change of fire environment information, such as smoke visibility, CO volume fraction and ambient temperature, is obtained through the numerical simulation results of FDS fire simulator. According to the definition of danger grid, the danger area in the process of fire spread is represented in yellow as shown in Fig. 4(b).



(a) Distribution of evacuees from a large building



(b) The spread of a large building fire at t=30s

Fig. 4. Fire scene of a large building.

IV. Experimental Results and Analysis

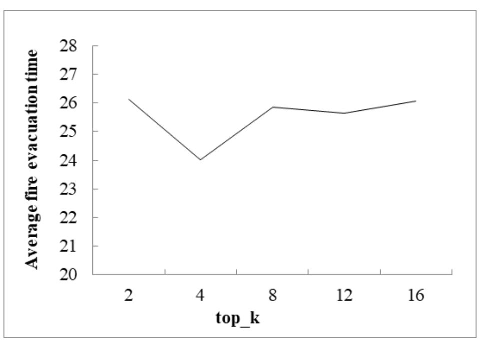
In order to evaluate the evacuation path planning ability of AMADDPG algorithm, the total number of training rounds was set to 1000 in the experiment, and the maximum evacuation simulation steps in each training round was set to 500. The evacuation time $T_e = D_e / v$ is defined, where D_e is the evacuation distance and v is the evacuation speed. Since the evacuation speed will be affected by the congestion degree, the greater the congestion degree, the more serious the degree of congestion, and the slower the moving speed of the evacuees. Therefore, according to literature, an exponential function is used to describe the evacuation speed, and $V_{\max}$ is the walking speed of normal people, which takes a value of 1.5m/s.

$$v = \begin{cases} V_{max}, c < 0.5 \\ V_{max} * e^{-0.5c}, other \end{cases} \tag{5}$$

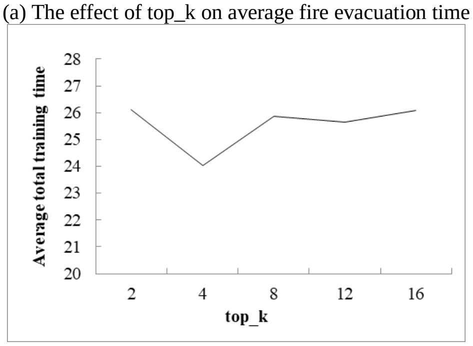
A. Sensitivity Analysis

In AMADDPG algorithm, the value of top_k will affect the evacuation ability and training time complexity of the model to some extent. Specifically, the smaller the value of top-k, the less surrounding information the agent obtains, which affects the value evaluation of the action of the Actor network. Otherwise, the model parameters will increase rapidly, which increases the difficulty of training and makes it difficult to converge. Therefore, it is necessary to discuss top_k. By using the average of the fire evacuation time and the total model training time of the three experiments, this study compared and analyzed the influence of top-k of 2, 4, 8, 12 and 16 on the evacuation ability and model training. As shown in Fig. 5.

As shown in Fig. 5, when top_k is 2, 4, 8, 12, 16, the fire evacuation time and model training time reach the lowest value when top-k is 4. After analysis and research, it is believed that when top_k is small, although the model parameters are small, the ability of the agent to perceive the environment becomes weak, the stability is poor, and it is not conducive to convergence during training. Therefore, it is more appropriate to set top_k to 4, and take this value as the fixed parameter value for subsequent experiments.



(a) The effect of top_k on average fire evacuation time



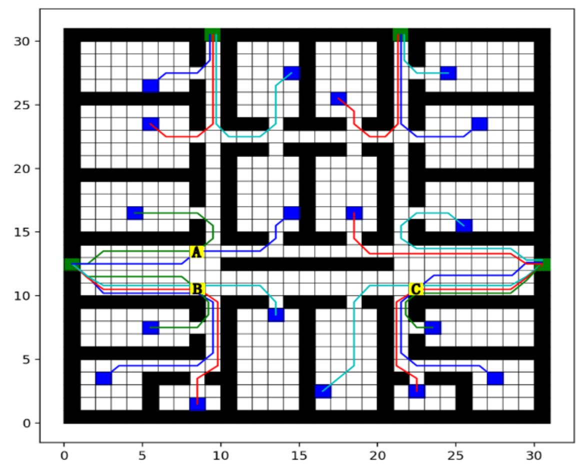
(b) The effect of top_k on average total training time

Fig. 5. Influence of top_k on the model.

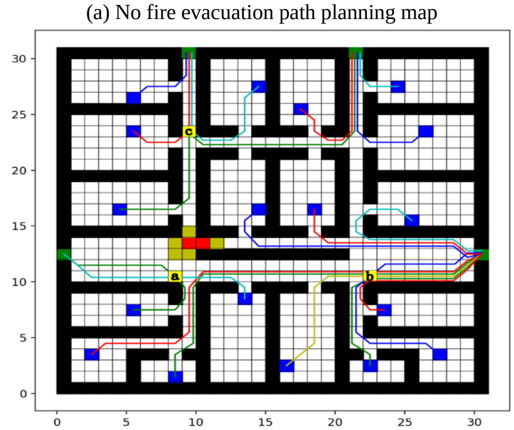
B. Evacuation Effect Analysis

According to the discussion of top_k in 4.2.1, on this basis, the personnel evacuation situation before and during fire is analyzed and compared, and the experimental results are shown in Fig. 5 and Table II.

From the comparison of Fig. 6 (a) and Fig. 6 (b), it can be seen that all agents can complete the total evacuation of personnel within 35 steps regardless of whether the fire occurs or not. According to the definition of evacuation time, the evacuation time can be controlled within 30s. In addition, when a fire occurs, due to the spread of the fire and the generation of combustion products, the evacuation path of some people is changed, so that they can bypass the dangerous area to reach the appropriate safety exit, as shown in the figure of agents 1, 2, 9 and 10. It shows that AMADDPG algorithm can adapt to the influence of dynamic environment change on path planning, and can plan the optimal path for multi-agent system.



(a) No fire evacuation path planning map



(b) Fire evacuation path planning maps are available

Fig. 6. Evacuation path planning diagram of a large building.

In the process of evacuation, the congestion phenomenon caused by the dynamic change of crowd location will have a certain impact on the evacuation effect. Combined with Fig. 6, the congestion degree of agent paths in A, B, C and a, b respectively without fire and when fire occurs is shown in the Table II.

TABLE II. CROWDING DEGREE OF AGENT EVACUATION TRAJECTORY

No-fire condition		Fire condition	
Tracking point	Congestion degree	Tracking point	Congestion degree
A	0.613	a	0.617
B	0.662	b	0.668
C	0.539	c	0.643

It can be seen from Fig. 6 and Table II that in the absence of fire, the algorithm adjusts the evacuation paths of agents 10, 3 and 7 when the congestion exceeds the threshold value 0.5 at points A, B and C respectively. Similarly, in the case of fire, the evacuation paths of agents 3, 7 and 9 are dynamically planned according to the congestion degree at points a, b and c respectively, which indicates that AMADDPG algorithm can effectively solve the congestion problem caused by the dynamic change of crowd location in the evacuation process, so as to ensure that all agents can quickly complete the evacuation within the safe evacuation time.

C. Comparative Analysis of AMADDPG Algorithm and MADDPG Algorithm

In order to reduce the training difficulty of the network and improve the computational efficiency of the algorithm, AMADDPG algorithm is an improvement of the MADDPG algorithm's centralized global learning, which only considers the state and action of the agent near the current agent. Through three experiments, this study analyzed and compared the training and evacuation conditions of AMADDPG algorithm and MADDPG algorithm in fire scenarios, and evaluated the efficiency of AMADDPG algorithm.

1) Comparison of evacuation time: Evacuation time refers to the time between the start of evacuation movement and the evacuation of all personnel to indoor or outdoor safe areas, and its definition is the same as 4.2.1. In order to ensure the safe evacuation of personnel in the building, the evacuation time of the fire site should be controlled within 90 seconds according to the requirements of the Code for Fire Protection in Building Design. The evacuation time in a fire scenario is shown in Table III.

TABLE III. COMPARISON OF EVACUATION RESULTS BETWEEN AMADDPG ALGORITHM AND MADDPG ALGORITHM IN FIRE SCENARIO

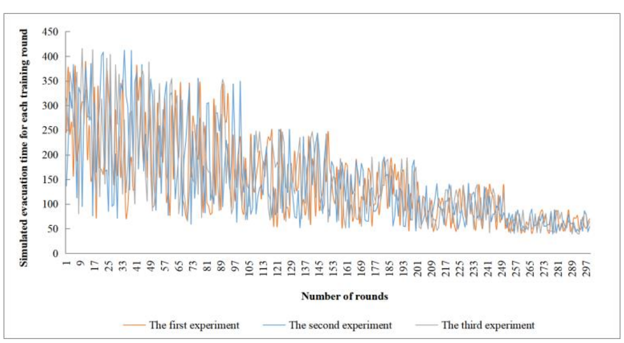
Algorithm evacuation result	AMADDPG algorithm	MADDPG algorithm
The first experiment	23.49	26.14
The second experiment	23.46	25.85
The third experiment	23.50	28.37
Mean evacuation time	23.48	26.79

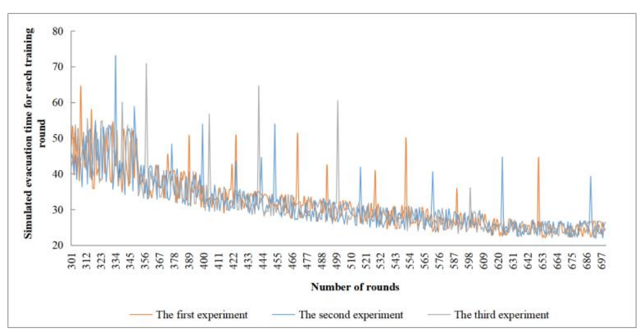
As can be seen from Table III, in a fire environment, the average evacuation time of the AMADDPG algorithm and the MADDPG algorithm three times is less than 30 s, and there is no significant change in the results of the three repeated experiments of the AMADDPG algorithm, while the evacuation time of the MADDPG algorithm is different. This shows that the two algorithms are acceptable in terms of path planning ability in fire scenes, and the evacuation effect is good, and the AMADDPG algorithm is better than the MADDPG algorithm in terms of algorithm stability. To analyze the reasons, the MADDPG algorithm needs to evaluate the status and actions of all agents in a complex and changeable fire environment. As a result, the input dimension of Critic network is too large, the complexity is too high, the convergence is difficult, and the stability of the algorithm is also affected. Therefore, compared with MADDPG algorithm, AMADDPG algorithm can obtain more stable optimal path planning results under complex dynamic environment and achieve the goal of safe evacuation.

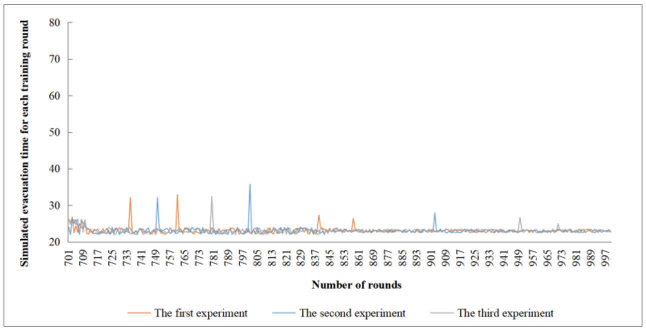
2) Convergent rounds: The training running time and convergence of AMADDPG algorithm and MADDPG algorithm are shown in Table IV and Fig. 7.

TABLE IV. TRAINING OF AMADDPG ALGORITHM AND MADDPG ALGORITHM IN FIRE SCENARIO

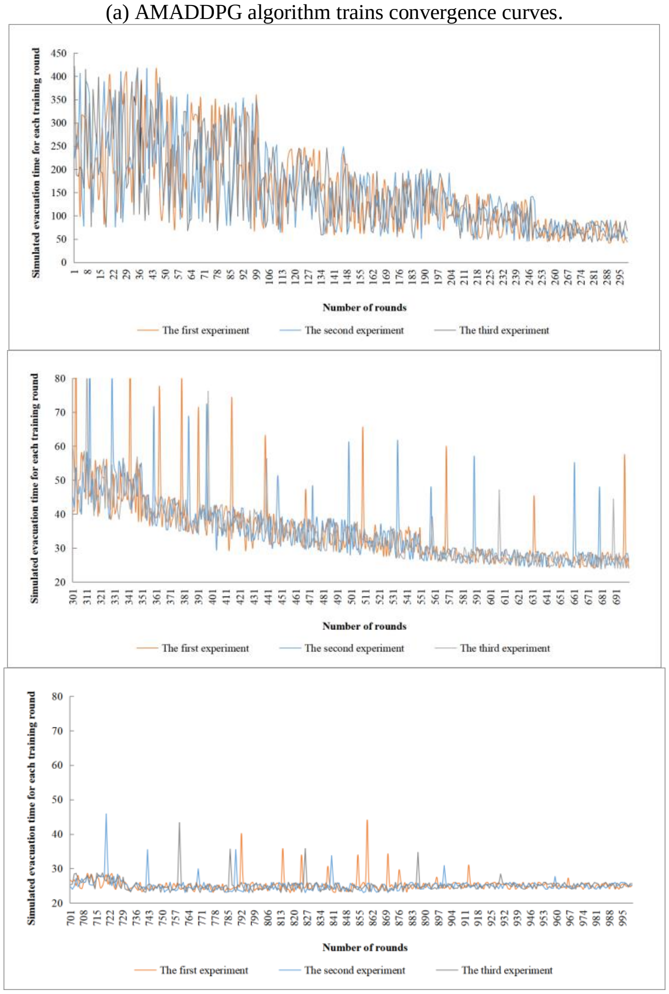
Algorithm	1000 rounds total training time			Average training time per round	Average convergent iteration rounds
	The first experiment	The second experiment	The third experiment		
AMADDPG algorithm	62467	64003	63861	63.44	708
MADDPG algorithm	68242	67508	67022	67.59	734







(a) AMADDPG algorithm trains convergence curves.



(b) MADDPG algorithm trains convergence curve.

Fig. 7. Convergence curves of AMADDPG and MADDPG algorithms in a fire scenario.

As can be seen from Table IV, the average running time of each training round of the AMADDPG algorithm is 4.15s shorter than that of the MADDPG algorithm, the average number of convergent iteration rounds is also reduced by 26 rounds, and the training efficiency is improved by 6.14% compared with the MADDPG algorithm. It shows that AMADDPG algorithm has higher training and learning effect, can effectively improve the convergence speed of the algorithm, and obtain the optimal evacuation path with less training time.

As can be seen from Fig. 7, in the first 400 rounds of the two algorithms, the evacuation time of each training round is generally above 30s, which is basically exploration oriented. With the accumulation and utilization of learning experience, the evacuation time of each training round converges to 23s until the 700 rounds, and the curve fluctuation of the AMADDPG algorithm is significantly less than that of the MADDPG algorithm, and is relatively stable. Therefore, AMADDPG algorithm can make the whole fire evacuation model convergence better and more stable.

V. Discussion

In order to verify the effectiveness of the algorithm, the evacuation process was simulated with or without fire, and the AMADDPG algorithm proposed in this study was compared with the MADDPG algorithm, as follows:

1) Through the simulation experiment with or without fire, it is verified that AMADDPG algorithm can basically adapt to the influence of dynamic environment changes on path planning, and can plan the optimal path for multi-agent system within 30s regardless of whether a fire occurs. Moreover, it can effectively solve the crowding caused by the dynamic change of the crowd position during the evacuation process, and the congestion degree on the multi-agent evacuation path is basically maintained within 0.5, so that the trapped people can move orderly at normal walking speed in the evacuation channel, and ensure a better evacuation effect.

2) By comparing the evacuation effect of AMADDPG algorithm and MADDPG algorithm in fire scenarios through three experiments, it can be concluded that AMADDPG algorithm can get the optimal path solution after 700 iterations. Compared with MADDPG algorithm, the average convergence iteration rounds of AMADDPG algorithm are reduced by 26 rounds, and the curve fluctuation is significantly less than MADDPG algorithm. Therefore, AMADDPG algorithm can achieve more stable and efficient optimal path planning in complex dynamic environment, and achieve safe evacuation goal.

VI. Conclusion

In this paper, an AMADDPG model suitable for fire evacuation is proposed by improving MADDPG algorithm. The experimental results show that the AMADDPG model can adapt to complex and dynamic fire environment, maximize the reduction of personnel congestion and avoid dangerous areas, and efficiently realize the optimal path planning for multi-exit fire scenarios.

From the technical point of view, the model has certain application potential and can be used in fire evacuation path planning. However, since the current research is still based on simulation, the subsequent research needs to be applied to the actual fire environment, and the algorithm should be deployed on multiple large public building [26] fire evacuation systems to further optimize the model and improve the generalization ability and robustness of the model [27] under different complex evacuation scenarios.

REFERENCES

[1] W. Zhong, J.Y. Yu, "Real-time fire evacuation system under smart city background," China Safety Science Journal, vol. 33, no. 2, pp. 179-184, 2023. Doi:10.16265/j.cnki.issn1003-3033.2023.02.2798.

[2] J.H. Ye, J.S. Pan, "Real-time path planning for safe rescue in building fire based on BIM and cellular automata," China Civil Engineering Journal, vol. 53, no. 08, pp. 1-8, 2020. Doi: 10.15951/j.tmgcxb.20200420.001.

[3] M. Choi,& S. Chi, "Optimal route selection model for fire evacuations based on hazard prediction data," Simulation Modelling Practice & Theory, vol. 94, pp. 321-333, 2019. Doi: 10.1016/j.simpat.2019.04.002.

[4] H. Min, X. Xiong, P. Wang, & Y. Yu, "Autonomous driving path planning algorithm based on improved A* algorithm in unstructured environment," Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, vol. 235, no. 2-3, pp. 513-526, 2021. Doi:10.1177/0954407020959741.

[5] L. Xu, K. Huang, J. Liu, D. Li, & Y.F. Chen, "Intelligent planning of fire evacuation routes using an improved ant colony optimization algorithm," Journal of Building Engineering, vol. 61, pp. 105208, 2022. Doi: 10.1016/j.jobe.2022.105208.

[6] J. Liang, & H. Wang, "Study on Building Fire Evacuation Path Planning Based on Improved Ant Colony Algorithm," Journal of System Simulation, vol. 34, no. 05, pp. 1044-1053, 2022. Doi:10.16182/j.issn1004731x.joss.20-0978.

[7] M. Sumathi, N. Vijayaraj, S.P. Raja, & M. Rajkamal, "HHO-ACO hybridized load balancing technique in cloud computing," International Journal of Information Technology, vol. 15, pp. 1357-1365, 2023. Doi: 10.1007/s41870-023-01159-0.

[8] E.A. Alhenawi, R.A. Khurma, A.A. Sharieh, O. Al-Adwan, A. Al Shorman, & F. Shannaq, "Parallel ant colony optimization algorithm for finding the shortest path for mountain climbing," IEEE Access, vol. 11, pp. 6185-6196, 2023. Doi: 10.1109/ACCESS.2022.3233786.

[9] H.N. Dong, X.T. Ye, C.P. Hao, "Emergency Evacuation Path Planning Algorithm for Indoor Fire in Commercial Buildings," Journal of Geomatics, vol. 46, no. S1, pp. 40-43, 2021. Doi:10.14188/j.2095-6045.2019351.

[10] L.W. Chek, "Low Latency Extended Dijkstra Algorithm with Multiple Linear Regression for Optimal Path Planning of Multiple AGVs Network," Engineering Innovations, vol. 6, pp. 31–36, 2023. Doi: 10.4028/p-t122xr.

[11] N. Khakzad, "A methodology based on Dijkstra's algorithm and mathematical programming for optimal evacuation in process plants in the event of major tank fires," Reliability Engineering and System Safety, vol. 236, pp. 109291, 2023. Doi: 10.1016/j.ress.2023.109291.

[12] P.M. de las Casas, A. Sedeno-Noda, & R. Borndörfer, "An improved Multi-objective Shortest Path algorithm," Computers & Operations Research, vol. 135, pp. 105424, 2021. Doi: 10.1016/j.cor.2021.105424.

[13] G. Li, S. Lin, S. Li, & X. Qu, "Learning Automated Driving in Complex Intersection Scenarios Based on Camera Sensors: A Deep Reinforcement Learning Approach," IEEE Sensors Journal, vol. 22, no. 5, pp. 4687-4696, 2022. Doi: 10.1109/JSEN.2022.3146307.

[14] R. Xie, Z. Meng, L. Wang, H. Li, K. Wang, & Z. Wu, "Unmanned aerial vehicle path planning algorithm based on deep reinforcement learning in large-scale and dynamic environments," IEEE Access, vol. 9, pp. 24884-24900, 2021. Doi: 10.1109/ACCESS.2021.3057485.

[15] H. Guan, Y. Gao, M. Zhao, Y. Yang, F. Deng, & T.L. Lam, "AB-Mapper: Attention and BicNet based Multi-agent Path Planning for Dynamic Environment," 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, IEEE, pp. 13799-13806, 2022. Doi:10.1109/IROS47612.2022.9981513.

[16] M. Pei, H. An, B. Liu, & C. Wang, "An improved dyna-Q algorithm for mobile robot path planning in unknown dynamic environment," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 52, no. 7, pp. 4415-4425, 2022. Doi: 10.1109/TSMC.2021.3096935.

[17] X. Tang, Y. Yang, T. Liu, X. Lin, K. Yang, & S. Li, "Path Planning and Tracking Control for Parking via Soft Actor-Critic under Non-Ideal Scenarios," IEEE/CAA Journal of Automatica Sinica, vol. 11, no. 1, pp. 181-195, 2024. Doi: 10.1109/JAS.2023.123975.

[18] L. Ni, X. Huang, H. Li, & Z. Zhang, "Research on Fire Emergency Evacuation Simulation Based on Cooperative Deep Reinforcement Learning," Journal of System Simulation, vol. 34, no. 06, pp. 1353-1366, 2022. Doi: 10.16182/j.issn1004731x.joss.21-0108.

[19] Z.B. Zhang, X.X. Huanng, H.G. Li, L.J. Ni, X.M. Xue, "Research on multiagent fire evacuation based on global guidance strategy," Modern Electronics Technique, vol. 45, no. 14, pp. 153-158, 2022. Doi:10.16652/j.issn.1004-373x.2022.14.027.

[20] G.L. Han, "Automatic Parking Path Planning Based on Ant Colony Optimization and the Grid Method," Journal of Sensors, vol. 2021, pp. 8592558, 2021. Doi: 10.1155/2021/8592558.

[21] C. Caliendo, P. Ciambelli, R. Del Regno, M.G. Meo, & P. Russo, "Modelling and numerical simulation of pedestrian flow evacuation from a multi-storey historical building in the event of fire applying safety engineering tools," Journal of Cultural Heritage, vol. 41, pp. 188-199, 2020. Doi: 10.1016/j.culher.2019.06.010.

[22] H.J. Zhang, Y. Liu, Z.Y. Zhu, B.Y. Chen, M. Yan, X. Luo, "Optimal Evacuation Path Planning Under Dynamic Spreading of Cruise Ship Fire," China Safety Science Journal, vol. 33, no. 01, pp. 183-190, 2023. Doi:10.16265/j.cnki.issn1003-3033.2023.01.0246.

[23] M. Zhang, M. Yang, Y. Li, J. Chen, & D. Lei, "Optimal Electric Bus Scheduling with Multiple Vehicle Types Considering Bus Crowding Degree," Journal of Transportation Engineering, Part A: Systems, vol. 149, no. 2, pp. 518-528, 2023. Doi: 10.1061/JTEPBS.TEENG-7518.

[24] J. Xue, J. Zhu, J. Du, W. Kang, & J. Xiao, "Dynamic Path Planning for Multiple UAVs with Incomplete Information," Electronics, vol. 12, no. 4, pp. 980, 2023. Doi:10.3390/electronics12040980.

[25] R. Lowe, Y.I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, & I. Mordatch, "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments," NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach California USA, pp. 6382-6393, 2017.

[26] Z. Feng, C. Wang, J. An, X. Zhang, X. Liu, X. Ji, W. Quan, "Emergency fire escape path planning model based on improved DDPG algorithm," Journal of Building Engineering, vol. 95, pp. 110090, 2024. Doi: 10.1016/j.jobe.2024.110090.

[27] Y. Q. Zhang, J. H. Wang, Y. Wang, Z. C. Jia, Q. Sun, Q. Y. Pei, D. Wu, "Intelligent planning of fire evacuation routes in buildings based on improved adaptive ant colony algorithm," Computers & Industrial Engineering, vol. 194, pp. 110335, 2024. Doi: 10.1016/j.cie.2024.110335.