Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 6, No. 3, 2023

Based on Hierarchical Reinforcement Learning for Large-scale Pedestrian Trajectory Prediction

Guokang Gao *, Yue Li

Software College, Chengdu University of Information Technology, Chengdu 610225, China

* **Corresponding author**: Guokang Gao (Email: 18782128216@qq.com)

Abstract: This paper describes the construction of an airport terminal simulation model using AnyLogic simulation software. It considers the advantages of hierarchical reinforcement learning and divides the complete process of pedestrian trajectories at the airport into layers. Pedestrians are treated as intelligent agents for hierarchical reinforcement learning. A large-scale pedestrian trajectory planning algorithm based on hierarchical reinforcement learning is designed to match the hotspots in the airport region simulated by pedestrian trajectories with congested areas in the terminal scene. A comparison is made with traditional multi-agent Q-learning algorithms and single-table hierarchical reinforcement learning. The results show that our algorithm can accurately identify the pedestrian flow hotspots in the actual terminal, with improved matching accuracy compared to traditional multi-agent Q-learning algorithms and single-table hierarchical reinforcement learning. The algorithm also exhibits faster convergence speed.

Keywords: Hierarchical Reinforcement Learning; Reinforcement Learning; Pedestrian Trajectory; Anylogic Simulation Software.

1. Introduction

Air transport is an indispensable part of society, and the operational efficiency of airport terminals, as an important part of air transport, is crucial to the development of the entire industry. However, the traditional manual scheduling and management methods can no longer meet the operational needs of modern terminals, and the use of artificial intelligence technology for terminal operation management can improve terminal efficiency and safety, so that passengers can have a higher experience, and reduce the operating costs of the airport. In terms of terminal management and operation, it is key for airport managers to predict the congestion degree of key process nodes in advance, and congestion at key nodes is a major problem that affects passenger travel satisfaction and troubles managers, so effectively solving the congestion of key nodes can often improve passenger satisfaction.

In this paper, a large-scale pedestrian trajectory prediction algorithm based on hierarchical reinforcement is proposed to train the pedestrian boarding model by using the simulation software Anylogic, and using hierarchical reinforcement and multi-agent collaboration technology, so as to find out the key nodes and causes of airport terminal congestion. The following use case 1 illustrates the basic process of large-scale pedestrian trajectory prediction based on hierarchical reinforcement.

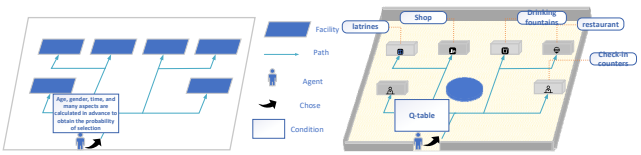

Figure 1. Simulations require a large amount of data to guide action selection

Example 1: We use the simple airport model shown in Figure 1 to explain our approach. At airports, pedestrians choose at the gate to go to different locations. In the airport model, each pedestrian arrives randomly in the span before the departure time of the pedestrian to take the flight, and the pedestrian will be generated at the airport gate, and will go through necessary and non-essential processes inside the airport. When a pedestrian arrives at the gate where they are boarded for their flight, the individual pedestrian flow ends. The pedestrian shown in the figure has six original actions in the check-in hall, (a) two can perform check-in actions for the service of the flight the pedestrian is taking, and (b) four different types of service actions. There are different rewards for performing different actions in different states. However, if you arrive at your destination after multiple choices or if you are unable to complete all the pre-boarding operations within the specified time, the individual pedestrian process ends.

Many scholars at home and abroad have adopted hierarchical reinforcement to improve the efficiency of algorithms, but they have not considered using multi-agent hierarchical reinforcement technology to solve related problems. In this paper, the algorithm proposes to use the combination of multi-agent strong chemical technology and hierarchical strong chemical to predict the pedestrian trajectory of the airport terminal, and then predict the potential congestion nodes of the terminal.

(1) The multi-agent reinforcement algorithm can be regarded as an extension of single-agent reinforcement. However, empirical evaluations show that directly implementing single-agent reinforcement to multi-agent reinforcement does not converge to the optimal solution because the environment is no longer static from each agent's perspective. In fact, actions performed by one agent can yield different rewards depending on the actions taken by other agents. This challenge, known as environmental nonstationarity, is the main problem to be solved in the development of efficient multi-agent reinforcement algorithms.

(2) In the airport terminal environment, the learning goals of the agent at different stages are not consistent, and it is

necessary to ensure that the learning goals of the agent are completely correct in the current environment. If the agent the target offset, the experimental results will not accurately express the solution of the agent problem at the current stage, so that the final result will deviate from the truth.

2. Formulation of the Problem

In order to solve the above problems, this paper proposes a large-scale passenger trajectory prediction model based on hierarchical reinforcement. The overall process of passengers from entering the waiting hall to boarding is divided into different sub-processes, and the sub-processes are divided into different sub-tasks through the task diagram, which realizes the hierarchical and reduces the dimension of sub-tasks, so as to solve the problem of accurate solution of sub-task objectives at different stages in the whole boarding process of the agent.

Definition 1: The problem in intensive chemistry is generally described as a Markov process, but the action in the Markov process decision process is instantaneous and does not represent the situation that the decision is made at a changing time interval, and the semi-Markov decision process (SMDP) can be regarded as an extension of the Markov process can be defined as:

$$< S, A, P, R > \tag{1}$$

$s \in S$ is denoted as the true state of the system; A is the action set; P is denoted as the probability function of multi-step transfer, and $P(s', N|s, a)$ is denoted as the probability of performing action a in state s to be transferred to s' through N steps; $r = R(s, a)$ is denoted as the expected reward obtained by the agent during the selection of action a in state s; γ is the discount rate∈ $(0,1]$; The goal of agent optimization is to ma mize the reward for the duration of the action and obtain the optimal strategy. The optimal Bellman equation and the optimal state-action pair-value function Bellman equation based on SMAP are as follows:

$$V^*(s) = \max_{a \in A} \left[R(s,a) + \sum_{s',N} \gamma^N P(s', N \mid s, a) V^*(s') \right] \tag{2}$$

$$Q^*(s,a) = R(s,a) + \sum \gamma^N P(s', N \mid s, a) \max Q^*(s', a') \tag{3}$$

In k iterations, the Q value is updated with the following formula:

$$\begin{aligned} &Q_{k+1}(s,a) = (1-\alpha) Q_k(s,a) + \\ &\alpha \left[r_t + \gamma^N r_{t+1} + \ldots + \gamma^{N-1} + \gamma^N \max_{a' \in A} Q_k(s', a') \right] \end{aligned} \tag{4}$$

Definition 2: Under the hierarchical design, subtask M can be defined as the following 3 tuples $\{A, R, T\}$; T is the basis for termination, and when M is executed to the state in T, M is terminated immediately. A is represented as a set of executable actions, A can include the M_i of the lower subtasks, if A contains M_i, it is also expressed in the form of 3 tuples $\{A_i, R_i, T_i\}$ in the M_i, until M, reaches $s' \in T, R$ is the pseudo-reward reward, which represents all the pseudo-return values obtained by the agent from the state s through multiple steps to $s' \in T$ and is used for the calculation of the upper-level return.

Definition 3: If the total task can be decomposed into a set of subtasks $\{M_0, M_1, \ldots, M_n\}$, then the overall strategy π can be decomposed into $\{\pi_0, \pi_1, \ldots, \pi_n\}$, where π_i is the strategy of the subtask M_i. Let $V^\pi(i,s)$ be the expected return function of the agent from the execution of the subtask M_i to the termination state based on the π_i in the state s, and if a is the original action, then $V^\pi(a,s) = R(a,s)$, $R(a,s)$ is the reward for executing action a in the state s.

The expression of the Q function is:

$$Q^\pi(i,s,a) = V^\pi(a,s) + C^\pi(i,s,a) \tag{5}$$

where the completion function $C^\pi(i,s,a)$is the cumulative discount return expectation based on the π_i call subtask M_i in state s:

$$C^\pi(i,s,a) = \sum_{s',N} \gamma^N P(s', s \mid N, a) V^\pi(i, s') \tag{6}$$

In state s, the expected return function for performing subtask i based on the $\pi_i$is expressed as:

$$V^\pi(i,s) = \begin{cases} Q^\pi(i,s,a) & \text{If i is a composite task} \\ R(i,s) & \text{If i is the original action} \end{cases} \tag{7}$$

Section 4 will describe in detail the various variables defined in Definition 1~Definition 3 based on the Airport Pedestrian Stratification Intensive.

3. Related Work

In the research related to terminal simulation technology, different scholars have different ideas and emphases to solve problems.

Zhang Yaping et al. [3] analyzed the passenger behavior in the departure hall through video data analysis of an airport in China, and established a departure passenger path model by improving the social force model, which simulated the path planning behavior of passengers in the public pedestrian space and service facility selection area of the departure hall at an acceptable accuracy level.

Mekic, Adin et al. [4] modeled the airport terminal considering the free time of passengers, analyzed how the allocation strategy of airport call broadcast and security check lanes would affect passengers' participation in free activities outside of the required processes of the airport, and analyzed the impact of the strategy on airport performance. Rozema, LM [5], on the other hand, took into account a different way of thinking and predicted the behavior of passengers through their observable characteristics at the airport, and established a framework composed of clustering and classification, which found behavioral categories through the clustered part, and classified them into various categories through the observable features of passengers. Some of the scholars believe that the check-in counter process and work efficiency are a major factor affecting the passenger service experience. AlSutan et al. [6] modeled the check-in system, dynamically allocated the check-in counters, and studied the impact of each parameter on the operating costs of check-in counter personnel and passenger service experience. Shakur, Ashab et al. [7] provided a way to optimize counter efficiency by simulating the check-in process and studying the check-in process.

Hierarchical intensive is an important direction of intensive chemistry research, and complex tasks in intensive chemistry will appear in high-dimensional state space and action space, resulting in dimensional explosion. In order to solve the above problems, some scholars have proposed the idea of hierarchical reinforcement learning (HRL). Using the semi-Markov decision-making process for modeling, the tasks are

established, each task contains a series of actions or tasks, and can be called by agents or other tasks, and the tasks of learning are decomposed through the mechanism of state abstraction, reducing the state dimension of each hierarchical space. Influential research on hierarchical reinforcement mainly includes the Option algorithm proposed by Sutton et al. [10], the HAM [11] (Hierarchy of abstract Machine) algorithm proposed by Parr et al., and the MAXQ algorithm proposed by Dietterich et al. [12].

4. Airport Pedestrian Model Setting based on Hierarchical Reinforcement

4.1. Sub-task Division

The processes that passengers have to go through in the terminal can be divided into two categories: necessary processes and non-essential processes, and the necessary processes refer to the necessary events that passengers have to go through before boarding, such as security checks, check-in, etc. Non-essential processes are actions that can be chosen by the traveler's own volition, such as shopping. The terminal flow chart is shown in the figure.

For the purposes of this article, the entire process from the time a pedestrian enters the gate to the time they leave the airport is considered to be $stage$. In this model, we divide the process that passengers may have to go through in the terminal according to two necessary processes, and divide the total process stage into three sub-processes, as shown in Figure 2. That is, $stage = \{stage_1, stage_2, stage_3\}$. $stage_1$=No pre-check-in process, $stage_2$=Check-in but no security check, $stage_3$=All processes after completing the security check and entering the waiting hall.

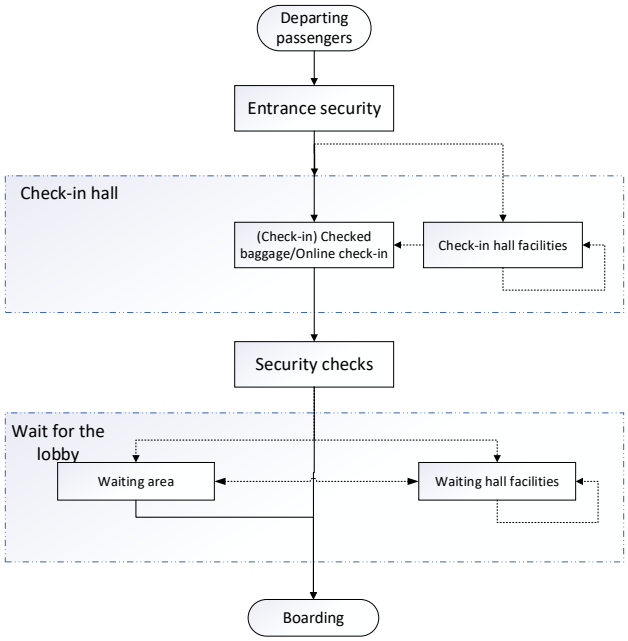

Figure 2. Flow chart of the pedestrian boarding process at the airport

In terms of task division for each sub-process, consider the relationship between macro and micro actions. For example, in the check-in hall, the possible macro actions of passengers can be divided into check-in or service, and then the service is specifically divided, and one of the multiple services may be selected. If the service is treated as a subtask, then the termination condition $s' \in T$ is to complete the service and leave the corresponding service area. Through this division, the macro options can be considered first, and then the micro actions can be further selected, which is conducive to reducing the state space and action space of intensive chemical training.

4.2. State Building

In intensive chemistry, a state is the environmental information that an agent perceives in the environment at a certain moment. However, this article is a custom environment, so using a hierarchical approach, the state of the agent is not the same for different subtasks at different stages. The state is mainly judged by the set of observations filtered by the subtasks in the current stage.

(1) Check-in sub-process ($stage_1$): Under the check-in sub-process, the degree of freedom of time, $freetime$ and the number of people at the $checkin$ counter together constitute the pedestrian state set, and the above factors are the direct factors affecting the choice made by pedestrians.

The degree of freedom of time is defined as the difference between the current time and the departure time of the pedestrian flight through discretization, the higher the degree of freedom of the pedestrian time in the terminal, the longer the pedestrian has free time, and vice versa, all the necessary processes need to be completed quickly.

The number of people at the check-in counter is a secondary factor affecting whether pedestrians choose to check in, and due to the large and uncertain number of people in the area, discretization is required in the training process.

We discretize the time freedom freetime and the number of people at the check-in counter, which has the advantage of making the state judgment more streamlined, which is conducive to training and designing predictive models. The degrees of freedom of time are divided into four discrete intervals, greater than 2.5 hours, between 2 hours and 2.5 hours, between 1 hour and 2 hours, and less than one hour. The number of possible destinations (including check-in counters, security checks and waiting areas) is divided into three zones: greater than 70, 50 to 70 and less than 50.

(2) Security sub-process ($stage_2$): Under the security sub-process, the status determination is determined by the time freedom freetime and the number of people in the security area. The number of people in the security screening area refers to the number of people who are waiting or under inspection under the security screening service, but it is still a secondary factor.

(3)Waiting sub-process ($stage_3$): The status determination under the waiting sub-process is determined by the *freetime* and the number of people in the pedestrian waiting area. The number of people in the pedestrian waiting area is the number of people in the area near the pedestrian target gate that can be used for pedestrians to sit and rest.

4.3. Action Settings

In the hierarchical design, the actions of the upper tasks are subtasks, and the subtasks are not actions that pedestrians can perform. In a larger sense, it is an expression of a trend, which can be understood as prompting pedestrians to make directional decisions. The setting of the action of a sub-task can be seen as the facility or necessary process that the person can choose to go to in a certain sub-task at a certain stage. The pedestrian action setting is affected by different stages and destinations, and the action sets can be selected differently in

different stages. At a stage, the action set can be represented as a subset.

The original actions include check-in, catering, shopping, toilets, security checks, and waiting for boarding, and the execution of the actions will directly act on the environment, while the behavior sub-tasks can be regarded as the action selection of a layer of sub-tasks, which will further select the original actions such as catering, shopping, and toilets.

4.4. Rewards

In the simulated environment of the airport, we divide all shopping, toilets, and food service stores into check-in hall service areas and waiting hall service areas according to the area, and each service facility can be regarded as a service area in a certain area. The image on the left shows the density visualization of a certain part of the terminal building according to the real number of people, while the picture on the right is the density visualization in the simulated environment. Whether the density is consistent can be regarded as the number of people in the real scene in a certain service facility in a certain range, and the number of people in the simulation scene is similar and in the same range. Then it can be considered that the choice of action is in line with the density. In the above figure, you need to see whether the density of the area shown on the right matches the density of the real area on the left.

We divide the number of people in the area into zones, where red dots indicate the highest density, blue dots indicate moderate density, and cyan dots indicate low density. If the number of simulated people is in the real interval during the simulation, it means that the number of people in the simulated area at the current moment is appro mately equal to the number of people in the real area, and vice versa.

5. Experiments

In this experiment, Anylogic simulation software is used as the simulation platform for airport terminals, Anylogic is a simulation software based on social force model, which uses the form of flow chart to create model logical relationships, which can be used to construct the airport environment for simulation experiments and provide corresponding data collection. Use pycharm as an IDE tool to write python code for agent training and corresponding data collection, and use Alpyne to complete the connection between the simulation environment and the reinforcement code.

5.1. Experimental Data

In order to ensure the authenticity of the experiment, this paper uses the area map of the T2 terminal of Chengdu Shuangliu Airport, and uses the flight information of the terminal for 6 hours on a certain day and the number of people per hour in the internal facilities of the terminal on the Anylogic simulation platform.

Terminal simulation environment construction: The data required to build the terminal simulation environment includes terminal CAD maps and performance parameters of each facility. The real CAD drawing of the airport is used as the environmental base map, and the walls and functional facilities area are created on it according to the real scale to complete the construction of the terminal simulation environment. In a simulated environment, all walls are considered impassable, and if there is already an occupant at the service point at the facility, there will be a queue at a fixed location. Pedestrians will be generated in front of the airport gate, pass through the check-in hall, enter the waiting hall through security, queue up for boarding before the scheduled departure time of the flight, and finally complete the overall process through the corresponding boarding gate.

Pedestrian arrival data construction: The data required for pedestrian arrival construction is all flight information contained in 6 hours, and the pedestrian arrival distribution can be appro mated as a Poisson distribution [15], so the number of flights will be generated at corresponding time intervals.

Reward design data: In order to make the simulation results close to the real world, the number of people in all facilities in the terminal with an interval of one hour is required as the reward criterion

5.2. Evaluation Indicators

In this method, certain evaluation indicators are needed to determine whether the prediction effect can be close to the real situation. In the simulation prediction, the visual evaluation of the simulation effect is reflected in the number of facilities in all terminals in a certain period of time and the number of corresponding facilities in the real situation. And for each pedestrian, they also prefer to have the shortest queuing time under the condition of meeting the number of people in the facility. Therefore, the evaluation indicators are as follows:

(a) Simulated coincidence rate. The simulated coincidence rate is defined as the ratio of the number of facilities f to the total number of facilities f that roughly conforms to the real situation over a certain period of time. The value range of the simulation coincidence rate is [0,1], the larger the value, the better the simulation effect, and vice versa, the worse the effect.

(b) Total queuing time. Total queuing time refers to the total queuing time of all pedestrians, which describes the time spent by pedestrians in all facilities, and the total queuing time always reaches a bottom value, indicating that pedestrians have reached the optimal learning.

5.2.1. Simulated Coincidence Rate

In our experiment, we used a 6-hour flight information table, set the facility time parameters, and connected the common Q-learning algorithm, the single-table hierarchical reinforcement algorithm and our hierarchical reinforcement algorithm. A total of 25,051 pedestrians were deployed in the simulated environment during the simulation of the facility simulation rate in the simulated environment. There are 60 check-in counters and 70 service facilities. Figure 3 shows the data pairs after iteration:

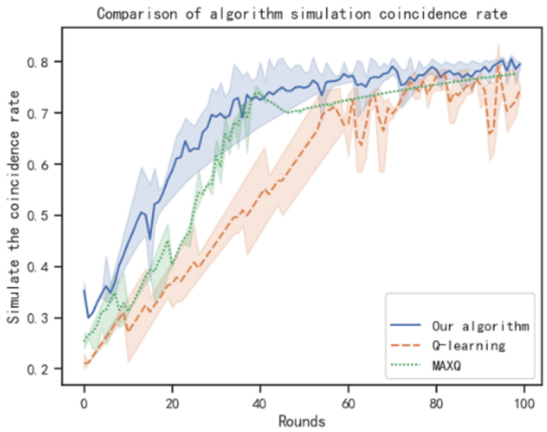

Figure 3. Comparison of algorithm simulation coincidence rate

As shown in the figure above, we can see that all algorithms in the first 20 rounds have achieved optimization results by updating their own strategies, and the simulation coincidence rate continues to increase. At the end of the round, the optimization effect is basically at its peak. Since both the algorithm and the Q-learning algorithm adopt a small probability exploration strategy, there are still some fluctuations when approaching the peak, while the single-table hierarchical reinforcement has little impact on the overall effect because multiple agents jointly maintain a table, so the overall upward trend is relatively flat. In the peak comparison, it can be seen that the simulation coincidence rate of the proposed algorithm is higher than that of the comparison algorithm.

5.2.2. Total Queuing Time

In the process of algorithm optimization, we still hope to make the queue waiting time of pedestrian agents shorter, because this is the case in real life, so the sum of the queuing time of all pedestrian agents is used as another evaluation index to reflect the efficiency of the algorithm. Under the original experimental conditions, the data comparison chart is shown in Figure 4 below:

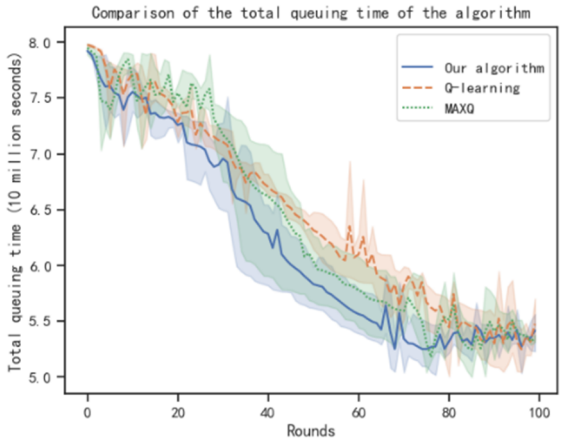

Figure 4. Comparison of the total queuing time of the algorithm

According to the above figure, it can be found that the optimization effect of the three algorithms is basically the same when reaching the final round, and the control of the queuing time is basically better under the condition of the same number of pedestrians, so that the pedestrian intelligence will consider the queue with fewer queuing people and shorter queuing time for service. However, it can be seen that the convergence speed of the proposed algorithm is better than that of other algorithms, and the queuing time reaches the optimal level within about 40 rounds

6. Conclusion

In view of the complex process in the terminal, this paper combines hierarchical reinforcement and multi-agent to train the multi-agent hierarchical reinforcement chemical in the complex environment of the terminal building by treating pedestrians as agents. The overall process is divided into different subtasks, and the pedestrian agent coordinates with each other at different levels, so that the agent's adaptability can be enhanced in complex environments. From the experimental results, this hierarchical reinforcement algorithm achieves a better pedestrian trajectory prediction effect and a faster convergence effect in the complex terminal environment.

References

[1] Canese, Lorenzo, Cardarilli, Gian, Carlo, Di, Nunzio, Luca, Fazzolari, Rocco, Giardino, Daniele, Re, Marco, Spano ,& Sergio. Multi-Agent Reinforcement Learning: A Review of Challenges and Applications[J].APPLIED SCIENCES-BASEL, 2021, 11 (11).

[2] Peysakhovich, Alexander, Lerer, & Adam. Prosocial learning agents solve generalized stag hunts better than selfish ones [J]. Ar v,2017.

[3] Zhang, Yaping,Li, Jialin,Kong, Dexuan, ng, aoqing,Luo, Qian, Mao, & Jian.Modeling and Simulation of Departure Passenger's Behavior Based on an Improved Social Force Approach: A Case Study on an Airport Terminal in China [J]. ADVANCES IN CIVIL ENGINEERING,2021,2021.

[4] Mekic, Adin, Mohammadi, Ziabari, Seyed, Sahand, Sharpanskykh, & Alexei. Systemic Agent-Based Modeling and Analysis of Passenger Discretionary Activities in Airport Terminals [J]. AEROSPACE, 2021,8(6).

[5] Rozema, LM. “Behavioural Classification of Passengers in an Airport Terminal.” (2017).

[6] Al-Sultan, A. T. (2018). Simulation and Optimization for Modeling the Passengers Check-in System at Airport Terminal. Review of Integrative Business and Economics Research, 7(1), 44.

[7] Shakur, Md. Ashab & Hasan, Muhammad. (2019). SIMULATION MODELING AND OPTIMIZATION OF THE CHECK-IN PROCESS OF AN INTERNATIONAL AIRPORT.

[8] Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6382–6393.

[9] Iqbal S, Sha F. Actor-Attention-Critic for Multi-Agent Reinforcement Learning[J]. 2018.

[10] Sutton R S, Precup D, Singh S. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning - ScienceDirect[J]. Artificial Intelligence, 1999, 112 (1–2):181-211.

[11] Parr R E. Hierarchical Control and Learning for Markov Decision Processes[D]. University of California, 1998: 36-47.

[12] Dietterich T G. An Overview of MAXQ Hierarchical Reinforcement Learning[M]. Abstraction, Reformulation, and Appro mation. Springer Berlin Heidelberg, 2000: 26-44.

[13] Silver D , Lever G , Heess N , et al. Deterministic Policy Gradient Algorithms[C]// International Conference on Machine Learning. PMLR, 2014.

[14] Lillicrap T P , Hunt J J , Pritzel A , et al. Continuous control with deep reinforcement learning[J]. Computer ence, 2015.

[15] Curcio, Duilio et al. “Passengers’ Flow Analysis And Security Issues In Airport Terminals Using Modeling & Simulation.” (2007).