Generation Method of Control Strategy for Aircrafts Based on Hierarchical Reinforcement Learning

Zeyuan Liu(✉), Qiucheng Xu, Yanyang Shi, Ke Xu, and Qingqing Tan

State Key Laboratory of Air Traffic Management System and Technology, Nanjing Research Institute of Electronic Engineering, Nanjing 210007, China
liuzeyuan@cetc.com.cn

Abstract. With the increasing density of air traffic and the complexity of the terminal sector, air traffic controllers will face more challenges and pressures in ensuring the safe and efficient operation of air traffic. In this work, an artificial intelligence (AI) agent is built to handle dense, complex and dynamic air traffic in the future. In this work, an artificial intelligence (AI) agent based on deep reinforcement learning is built to mimic air traffic controllers, such that the dense, complex and dynamic air traffic flows in terminal airspace can be handled sequentially and separated. To solve the problem, hierarchical reinforcement learning method is proposed, the flights choose agent and the flights action agent are achieved by DDQN. Results show that the built AI agent can guide 16 aircrafts safely and efficiently through Sector 01 of Nanjing Terminal, simultaneously.

Keywords: Terminal sector · Artificial intelligence agent · Aircraft control strategy · Hierarchical reinforcement learning

1 Introduction

With the rapid development of China's air transport industry, the number of flights is increasing at an average annual rate of over 10% in recent years [1]. What is more, by 2030, there will be more than 450 civil transport airports, and the volume of passenger traffic will reach 1.8 billion [2]. However, the rapid development of air transport would lead to the increasing challenges and pressures [3] for air traffic controllers to ensure the safe and efficient operation of air transport.

To deal with the challenges of current and future air traffic demands, Civil Aviation Administration of China (CAAC) has proposed the idea of Four Enhanced Air Traffic Management (ATM) Solutions in 2018 [4], including enhanced security, enhanced efficiency, enhanced intelligence and enhanced collaboration, where the enhanced intelligence is to suggest that the ATM should take new technologies, such as big data, blockchain, artificial intelligence (AI) and so on, to promote the operational effectiveness significantly. Therefore, some researchers are working on how to apply these new technologies to the aviation.

Supported by China National Key R&D Program (No. 2018YFE0208700).

Q. Liang et al. (Eds.): Artificial Intelligence in China, LNEE 854, pp. 109–116, 2022.
https://doi.org/10.1007/978-981-16-9423-3_14

Deep Reinforcement Learning (DRL) framework and algorithm is one of the most famous AI technologies, which has a great ability of dealing with continuous sequential decision-making problems [5, 6], and have been demonstrated to perform high level tasks and learn complex strategies, such as play the games of AlphaGo [7] and StarCraft-II [8].

Inspired by this, many researchers try to solve many difficult decision-making problems in ATM by using the deep-reinforcement Learning Algorithm.

In [9], the authors adopted a reinforcement learning method to predict the taxi-out time of the flight, and the predicted taxi-out time result is then compared with the actual taxi-out time to reduce the taxi-out time error. In [10], a Multi-agent system using Reinforcement Learning is developed for both simulation and daily operations to support human decisions, where two types of reward functions are proposed for air traffic flow management (ATFM) decision making to control safety separation of Ground Holding Problem (GHP) and Air Holding Problem (AHP).

In [12], K. Tumer and et al. proposed a multi-agent algorithm based on reinforcement learning for traffic flow management, where each agent is associated with a fix location and its goal is to set separation and speed up or slow down traffic flows to manage congestion. At last, the proposed method is tested on an air traffic flow simulator, FACET. In their following work [11], the authors proposed a distributed agent based solution where agents provide suggestions to human controllers, and an agent reward structure is designed well to allow agents to learn good actions in the indirect environment, such that the "Human-in-the-Loop" solution can be achieved.

M. Brittain and P. Wei [13] proposed a hierarchical deep reinforcement learning algorithm to build an AI agent, which takes the NASA Sector 33 app as the simulator. The well-trained AI agent can guide aircraft safely and efficiently through "Sector 33" and achieve required separation at the metering fix. And then, the authors [14] also proposed a deep multi-agent reinforcement learning framework to identify and resolve conflicts between aircrafts in a high-density, stochastic, and dynamic en route sector with multiple intersections. However, these works only considered the horizontal space separation and ignored the vertical separation of airspace.

In this work, an artificial intelligence (AI) agent based on deep reinforcement learning is built to mimic air traffic controllers, such that the dense, complex and dynamic air traffic flows in terminal airspace can be handled sequentially and separated. To simplify the problem, the complex three-dimensional terminal airspace is projected onto the vertical plane by dispersing state space and action space. And then, the typical reinforcement learning algorithm, double deep Q-network, is taken to realize the AI agent. Results show that the built AI agent can guide 6 aircrafts safely and efficiently through Sector 01 of Nanjing Terminal, simultaneously.

The remainder of the paper is organized as follows. Section 2 gives a simple example of ATC tasks in terminal sector and describes the problem definition. Section 3 shows the proposed algorithm for hierarchical reinforcement learning. Experimental results and conclusions are discussed in Section 4 and Section 5, respectively.

2 Problem Description

Air traffic control (ATC) is a service provided by ground-based air traffic controllers who direct aircraft on the ground and through controlled airspace, where the goal is to guide aircrafts to their runway for landing. The job of an air traffic controller is to prevent collisions of aircraft, safely and efficiently organize the flow of traffic and to provide support to pilots. Although traffic flow and efficiency are important factors, their primary goal is to guarantee safety of the aircraft. To accomplish this, air traffic controllers use traffic separation rules which ensure the distance between each pair of aircraft is above a minimum value all the time.

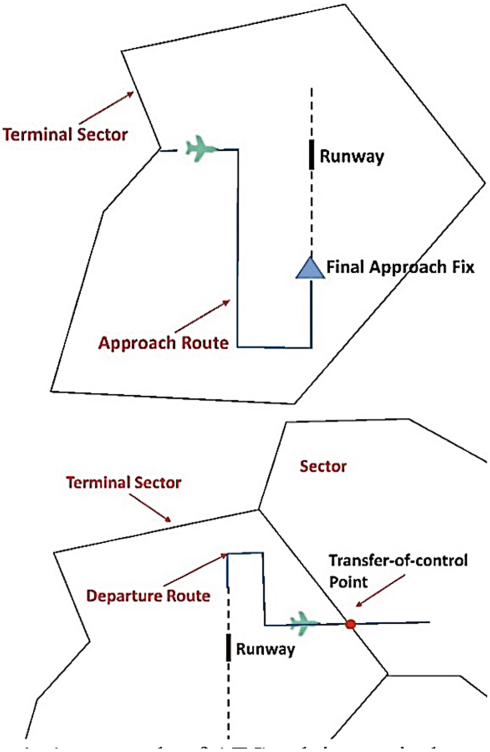


Fig. 1. An example of ATC task in terminal sector

In terminal sector, the ATC task includes two types, one is the approach control and the other is the departure control. Figure 1 gives an example of the ATC task in terminal sector. One ATC task is approach control, which is the job of directing aircrafts which are approaching an airport onto the final approach course at the correct altitude.

Our goal is to train an AI agent to perform basic Air Traffic Control tasks through hierarchical reinforcement learning. In this work, we use two deep reinforcement learning agent to accomplish the task, the flights choose agent and the flights action agent, respectively. Fig. 2 show the relationship between the two agents and the simulation environment. Their state space and action space are as follows.

(1) State space S, which contains all information about the environment and each element $s_t \in S$ can be considered a snapshot of the environment at time t. The state space of the flights choose agent is the latitude, longitude and altitude of all aircraft in the sector. The state space of flights action agent is the combination of any two flight positions within the current sector.
(2) Action space A, which is the set of all actions that AI agent could select in the environment. In this work, the flights choose agent selects a pair of flights in the current sector, and the flights action agent decides whether the two selected flights are descending height or ascending height.
(3) State transition, the agents get the state of the environment every 4 s and choose decision from a set of feasible decision options A. Corresponding to the decision, we can get the transition from a state s_i to another state s_j.
(4) Objective: The goal of the agent is to interact with the emulator by selecting actions in a way that maximises future rewards, where the selected action can maintain safe separation between aircraft and resolve conflicts for all aircraft in the sector by providing height adjustment, and all aircrafts arrival at the target positions with maximization of the cumulative reward from each transition.

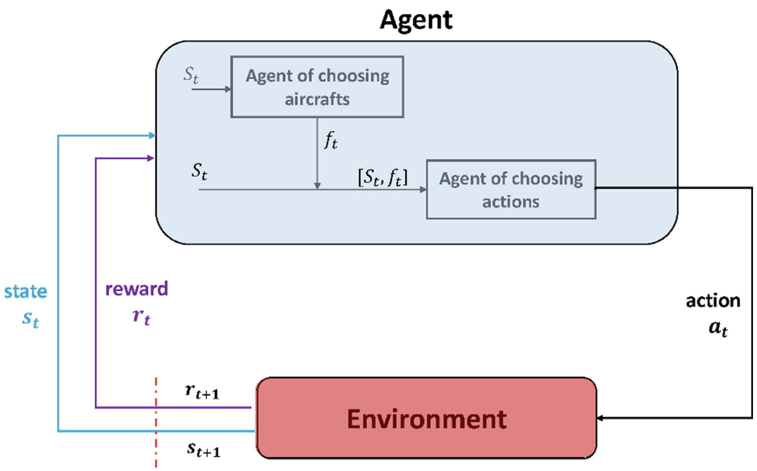


Fig. 2. The relationship between the two agents and the simulation environment

3 The Proposed Algorithm

In our research, we used two double deep Q-network (DDQN) to generate two different strategies, which are flights choose strategy and flights action strategy. The flights choose agent contains two fully-connected hidden layers, and the number of nodes in each layer are 256, 64, the flights action agent contains three fully connected layers, the number of nodes in each layer are 256, 128 and 128. **Algorithm 1** show the overall flow of our method.

During the training progress, ϵ-greedy search strategy is taken, and ϵ is decayed from 1.0 to 0.01, in the experiment, the max buffer length is set to 2000. During the agent's decision-making process, the flights choose agent first judges a pair of flights with potential conflicts according to the state of the environment. The environment selects the states of potential conflicts flights based on the action of flights choose agent and uses them as part of the input of flights action agent, then flights action agent gives the control strategies of these two flights.

The design of the reward function for the flights choose agent and flights action agent should be consistent with the goal of this paper, which are defined as follows:

$$r_{fca} = 1000/dis_{flights} + 10000/dis_{airport} \tag{3.1}$$

$$r_{faa} = \begin{cases} -\alpha * 0.1 & \textit{if action is raise height} \\ 0.05 & \textit{if action is maintain height} \\ \alpha * 0.1 & \textit{if action is descent height} \end{cases} \tag{3.2}$$

where $dis_{flights}$ presents the distance between two flights with potential conflicts, $dis_{airport}$ presents the minimum distance of these two flights from the airport. α is a flag that represents whether the aircraft is approaching or departing. If the aircraft is approaching, we set $\alpha = -1$, otherwise, $\alpha = 1$. Reward function r_{fca} and r_{faa} is calculated at each time-step. And, once safe separation is not satisfied, or aircraft overpasses the terminal sector boundary, or sector handover condition is not satisfied, reward r_{fca} and r_{faa} will minus 5. If all aircrafts reached their corresponding target positions, reward r_{fca} and r_{faa} will add 5.

In this work, the learning process in one episode is terminated when one of the following four situation is satisfied:

(1) All aircrafts reached their target positions $(x^i_{target}, y^i_{target}, h^i_{target})$ without collision, that is,

$$\sqrt{\left(x_i - x^i_{target}\right)^2 + \left(y_i - y^i_{target}\right)^2 + \left(h_i - h^i_{target}\right)^2} = 0, \forall i \tag{3.3}$$

(2) An aircraft overpasses the terminal sector boundary;
(3) Sector handover condition is not satisfied, that is,

$$\sqrt{\left(x_i - x^i_{target}\right)^2 + \left(y_i - y^i_{target}\right)^2} = 0 \, and \, h_i - h^i_{target} \neq 0, \forall i \tag{3.4}$$

(4) Collision is occurred between aircrafts, that is,

$$\sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(h_i - h_j\right)^2} < \delta, \forall i \neq j \tag{3.5}$$

At last, by training the proposed DDQN model until it converges, we can obtain the optimal control strategy for aircrafts in terminal sectors.

4 Experimental Results

The proposed AI agent construction method have been implemented in Python language on a 64-bit workstation (Intel 2.4 GHz, 256 GB RAM).

In this work, the simulator based on Sector 01 of Nanjing terminal is constructed as our air traffic control environment. Figure 3 gives an example of the constructed simulator environment, where there are 10 approach routes and 15 departure routes.

In the experiments, we considered 6 aircrafts with control to evaluate the performance of our reinforcement learning framework. By training the AI agent on around 1200 episodes and choosing a time-step of four seconds, we can obtain the optimal solution for this problem and Fig. 4 shows the experiment results. Figure 4(a) shows the height profile under agent control, and we can see that, these aircrafts can successfully reach their target positions using the control strategy generated by AI agent. Figure 4(b) shows the average loss during training, Fig. 4(c) shows the agents' average scores during training, which shows that the score increases with the number of training episodes. Figure 4(d) shows the number of conflicts during the training, as we can see in this fig, the number of conflicts is decrease to zero at the end of training, which demonstrates the effectiveness of the proposed method.

Algorithm 1 Hierarchical RL Agent

```
Initialize Flights Choose Agent $FCA$ and Flights action Agent $FAA$;
Initialize flights choose score $score1 = 0$ and flights action score $score2 = 0$, Initialize state $s_t$;
Initialize queue Flights Choose Replay Buffer $FCRB$, queue Flights action Replay Buffer $FARB$;
for $i = 1$ to n do:
    while not $is\_end$ do:
        $a_{fca} = FCA.ChooseAction(s_t)$;
        $s_{FlightAction} = ChooseState(s_t, a_{fca})$;
        $a_{faa} = FAA.ChooseAction(s_{FlightAction}, a_{fca})$;
        $s_{t+1}, r_{fca}, r_{faa}, is\_end = Environment.ExecuteAction(a_{faa})$;
        $score1 += r_{fca}, score2 += r_{faa}$;
        if $FCRB.size() > MaxBuffer$ and $FARB.size() > MaxBuffer$:
            update $FCRB$, $FARB$ parameters
        end if
    end while
end for
```

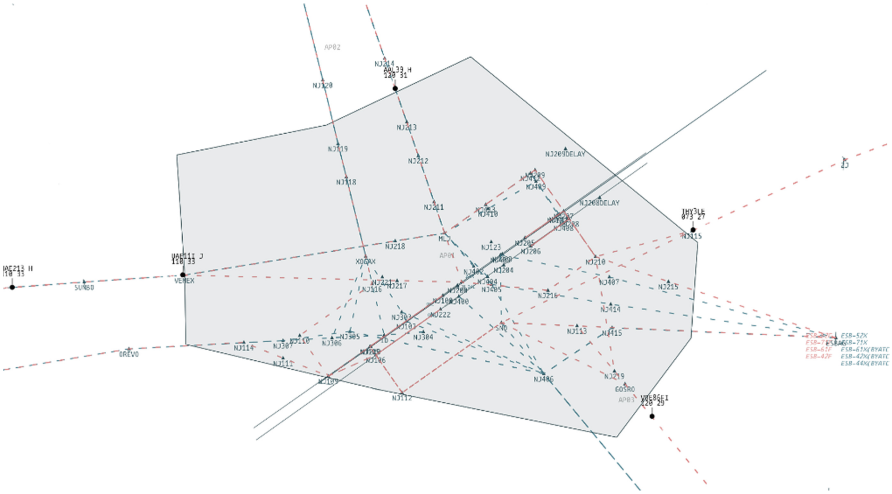
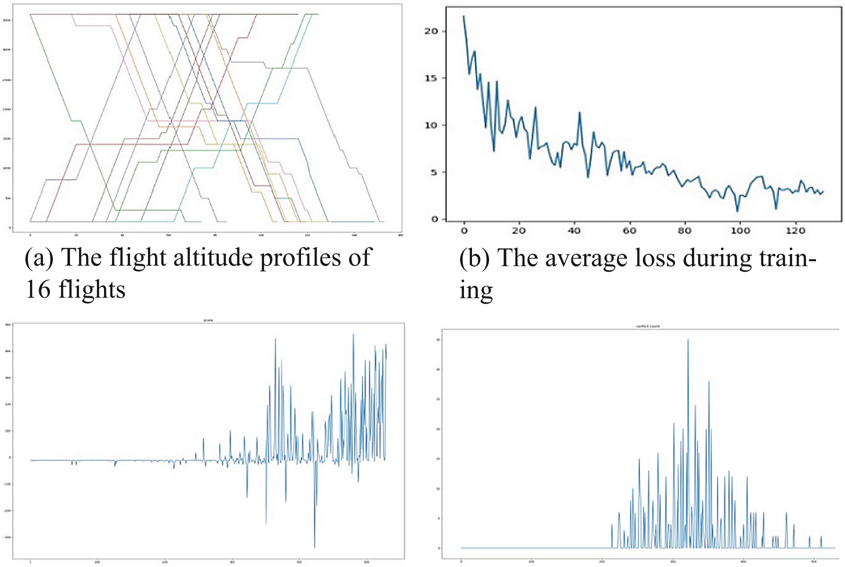


Fig. 3. The simulator environment based on Sector 01 of Nanjing terminal



(a) The flight altitude profiles of 16 flights

(b) The average loss during training

(c) The average score during training

(c) The number of conflicts during training

Fig. 4. The experiment results on Sector 01 of Nanjing terminal

5 Conclusion

In this work, two artificial intelligence (AI) agent based on deep reinforcement learning is built to mimic air traffic controllers, such that the dense, complex and dynamic air traffic flows in terminal airspace can be handled sequentially and separated. To solve the problem, the flights choose agent and flight action agent was built to formed a hierarchical structure. Results show that the built AI agent can guide 16 to 20 aircrafts safely and efficiently through Sector 01 of Nanjing Terminal, simultaneously.

References

1. Zhao, W.: The opportunities, challenges and obligations in internationalization of China civil aviation. Civil Aviation Management **09**, 6–11 (2017)
2. Yan, Y., Cao, G.: Operational concepts and key technologies of next generation air traffic management system. Command Inf. Syst. Technol. **9**(3), 8–17 (2018)
3. Ma, X., Xu, X., Yan, Y., et al.: Correlation analysis on delay propagation in aviation network. Command Inf. Syst. Technol. **9**(4), 23–28 (2018)
4. http://www.caac.gov.cn/XWZX/MHYW/201803/t20180315_55771.html
5. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction (2011)
6. Mnih, V., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (2015)
7. Deepmind: Alphago at the Future of go Summit, pp. 23–27. http://deepmind.com/research/alphago/alphago-china/ (May 2017)
8. Vinyals, O., et al.: StarCraft ii: A new challenge for reinforcement learning. arXiv preprint arXiv:1708.04782. (2017)
9. George, E., Khan, S.S.: Reinforcement learning for taxi-out time prediction: An improved q-learning approach. In: 2015 International Conference on Computing and Network Communications (CoCoNet), pp. 757–764. IEEE (2015)
10. Cruciol, L.L., de Arruda Jr, A.C., Weigang, L., Li, L., Crespo, A.M.: Reward functions for learning to control in air traffic flow management. Transp. Res. Part C Emerg. Technol. **35**, 141–155 (2013)
11. Agogino, A., Tumer, K.: Learning indirect actions in complex domains: action suggestions for air traffic control. Adv. Complex Sys. **12**(04n05), 493–512 (2009)
12. Tumer, K., Agogino, A.: Distributed agent-based air traffic flow management. In: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 1–8 (2007)
13. Brittain, M., Wei, P.: Autonomous aircraft sequencing and separation with hierarchical deep reinforcement learning. In: Proceedings of the International Conference for Research in Air Transportation (2018)
14. Brittain, M., Wei, P.: Autonomous separation assurance in an high-density en route sector: a deep multi-agent reinforcement learning approach. In: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), pp. 3256–3262. IEEE (2019)