ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

Research on multi UAV attack defense confrontation algorithm based on machine learning

Bo Liu, Xuan Li, Tianci Zheng, Wenyu Gao, Xinyu Zhang, Xiaoyan Wang

College of Materials Science and Engineering, Jilin University, Changchun 130022, China

Abstract: *In recent years, with the continuous development and wide application of unmanned technology, the use of unmanned aircraft in various fields such as agriculture, industry, disaster, leisure and military purposes has increased, and even played an irreplaceable role in certain fields. However, the functions of a single drone are very limited and cannot meet the increasing diversified needs of people. Among the many diverse needs of people, target tracking is a very important task that unmanned aircraft systems need to undertake in future application scenarios. Therefore, how to make multiple drones work together to complete the tracking of the same target has become an important research topic. However, so far, the path planning of many unmanned cooperative tracking targets has not been well resolved. This paper combines a DQN-based MADDPG (Multi-Agent Deep Deterministic Policy Gradient) algorithm to propose a path planning method for multiple UAVs to cooperatively track targets, which can dynamically plan and adjust the flight path of multiple cooperative UAVs in real time and get better tracking effect in a period of time.*

Keywords: *DQN, MADDPG, Multiple UAVs, Path Planning*

1. Introduction

Machine learning is an interdisciplinary subject, involving probability theory, statistics, approximation theory, convex analysis, algorithm complexity theory and other disciplines. It focuses on how computers simulate or realize human learning behaviors, so as to acquire new knowledge or skills, reorganize the existing knowledge structure, and constantly improve their own performance. It is the core of artificial intelligence and the fundamental way to make computer intelligent.

Unmanned aerial vehicle (UAV) has been widely used in military and civil fields. How to use UAV to get safe guidance and avoid dynamic obstacles is the key topic of many scholars. UAV confrontation is an important experimental link and research direction of machine learning. Due to the advantage of close combat, UAV plays an important role in the future national defense deployment and other fields. It can be used in all aspects of battlefield reconnaissance, electronic jamming and other electronic countermeasures, and plays an important role in the cooperative combat with manned aircraft and other weapons. Its great influence in the future high-tech local war cannot be ignored. Through the UAV reconnaissance and positioning of the enemy's key areas, we can obtain the accurate distribution of targets, and also implement signal jamming. While ensuring the enhancement of national defense and military strength, this technology can be applied to traffic, driverless, transportation industry, etc. to liberate manpower, and use machines to replace manpower to do work which is harmful for human beings.

The use of unmanned satellite aircraft in agriculture, industry, disaster, leisure and military purposes is increasing, and even plays an irreplaceable role in some fields. However, the function of a single UAV is very limited, which cannot meet the increasing diversity and characteristics of modern people. In people's various and characteristic needs, target tracking is a very important task for unmanned aircraft system in future application scenarios. Therefore, we also focuses on how to make UAVs work together to complete the tracking of the same target. So far, the path planning of multi UAV cooperative target tracking has not been well solved.

This project, which is based on and combined with the MADDPG (Multi-Agent Deep Deterministic Policy Gradient) of DQN (Deep Q-learning) algorithm, proposes a path planning method for multiple UAVs to track targets cooperatively. By integrating the MADDPG algorithm based on DQN and ROS+Gazebo, a simulation platform for UAV cluster tracking targets is built, which lays a solid foundation for the application of MADDPG algorithm in practice。

The main work of this project is as follows:

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

1) Install the complete desktop of ROS kinetic and the physical based simulation environment gazebo on the Linux operating system Ubuntu 16.04 LTS, and then build the simulation platform of ROS+Gazebo. Then build the simulation scene of multiple UAVs tracking targets the simulation platform.

2) Use Python to implement MADDPG algorithm, and install MPE (Multi-Agent Particle Environments) environment needed to run MADDPG algorithm.

3) The MAPPDG algorithm is integrated into the simulation scene of ROS+Gazebo to train multiple UAVs, so as to achieve the better effect of multi UAV cooperative target tracking.2. Methodology .

2. Methodology

2.1 MADDPG Algorithm

Reinforcement learning (RL) has been widely used to solve challenging problems in recent years, from games to robotics. In industrial applications, RL is emerging as a practical component in large-scale systems, such as data center cooling. Most of RL's success occurs in a single proxy domain where modeling or predicting the behavior of other participants in the environment is largely unnecessary. However, there are many important applications involving the interaction between multiple agents, among which the emerging behavior and complexity are generated by the co-evolution of agents. For example, multi robot control, discovery of communication and language, analysis of multiplayer games and social dilemmas are all carried out in the field of multi-agent. The variant of hierarchical reinforcement learning can also be regarded as a multi-agent system, in which multi-level is equivalent to multi-agent. In addition, it has recently been shown that multi-agent self-play is a useful training paradigm. It is very important to successfully extend RL to the environment with multiple agents for building artificial intelligence systems that can interact with human beings and each other effectively. Unfortunately, traditional reinforcement learning methods (such as Q-learning or strategy gradient) are not suitable for multi-agent environments.

Too simple policy gradient method is not effective in simple multi-agent settings. The goal in this section is to derive an algorithm that works in this case. However, we operate under the following constraints: (1) learning strategies can only use local information (that is, their own observations) when they are executed [2]; (2) we don't assume the distinguishable model of environmental dynamics like that; (3) we don't use any specific structure in the way of communication between agents (that is, there is no difference in communication channels). To meet the above requirements, a general multi-agent learning algorithm will be provided, which can be applied not only to cooperative scenarios with clear communication channels, but also to competitive scenarios and scenarios involving only physical interaction between agents.

Similar to [3], we achieve our goal by adopting a decentralized centralized training framework. Therefore, we allow the strategy to use additional information to simplify the training, as long as it is not used in the test. It's unnatural to do this through Q-learning, because Q-functions usually can't contain different information in training and testing. Therefore, we propose a simple extension of the actor-critical policy gradient method, in which critical is added additional information about other agent policies.

Every agent in MADDPG algorithm uses DDPG algorithm, and DDPG is essentially an actor-critical method, and the two most important functions are p_train () and q_train().The main purpose of p_train() is to build actor and Q_ Train is to establish critical. The training process of each agent is similar to that of a single DDPG algorithm, the difference is mainly reflected in the input of Critical: in the DDPG algorithm of a single agent, the critical input is a state-action pair information, but in MADDPG, the critical input of each agent can have additional information besides its own state-action information, such as the actions and communication messages of other agents.

The MADDPG algorithm flow chart is shown in Figure 1.

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

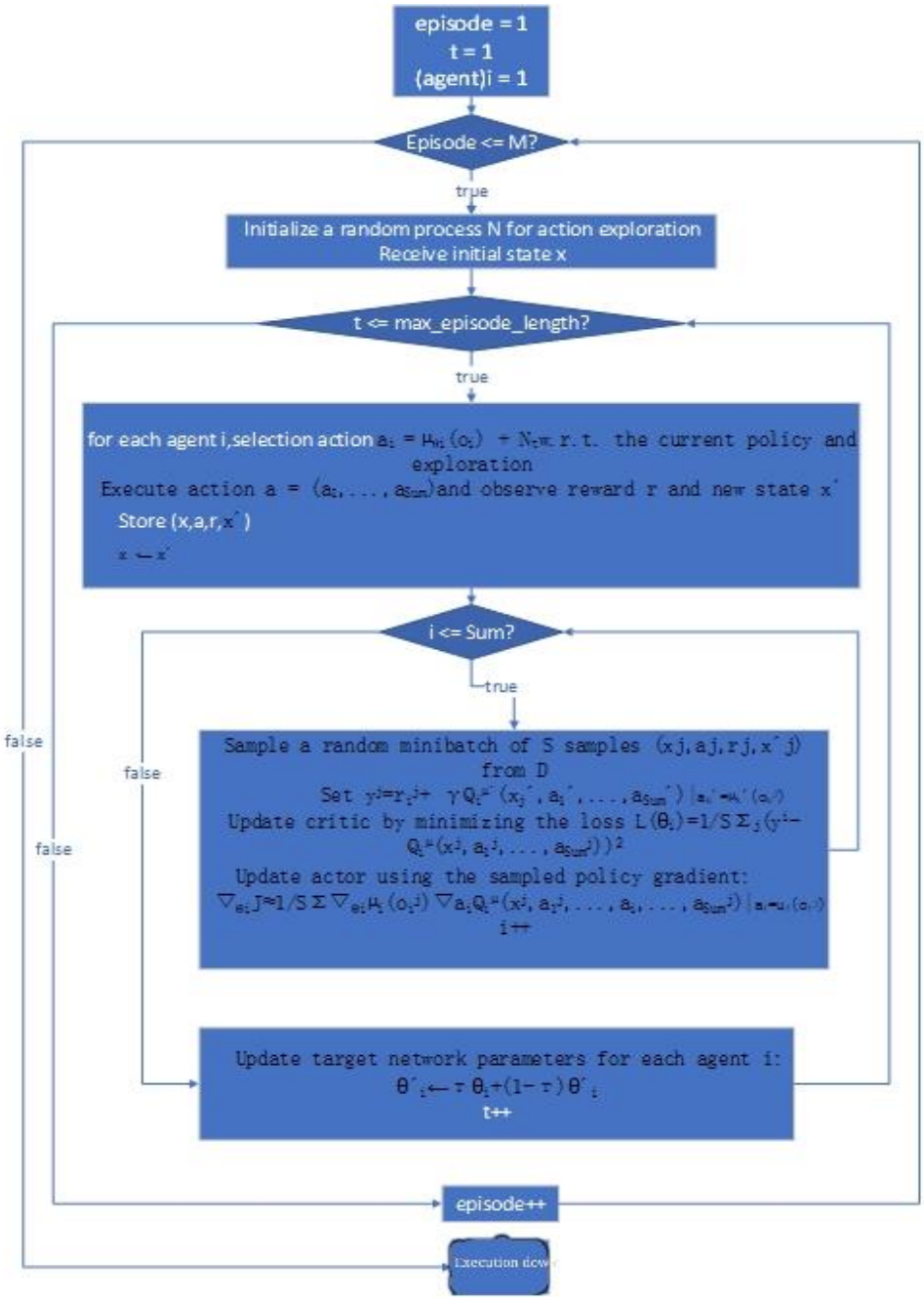

Figure 1: MADDPG algorithm

2.2 Agent communication in MADDPG

Because the environment state of UAV, not stable, is determined by the behavior of multiple agents, Q-learning algorithm is difficult to train, and the variance of policy gradient algorithm will become larger with the increase of the number of UAVs.[4]

When the strategy is trained, only the actor needs to interact with the environment, that is, only the green cycle is needed. The input of the actor is the environment's state S, and the output is the action a. In the process of training, critical needs to obtain the current environment state and the action taken by the actor, form the state action pair (s, a) as the input, and output the value V of the state action pair to evaluate the current action, and help the actor to improve the strategy.

Taking two agents as an example, the simultaneous input and output are shown in Figure 2

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

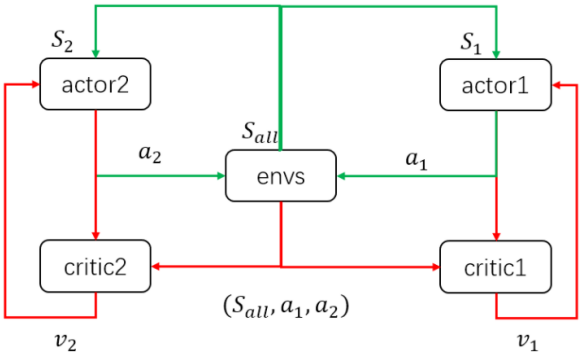

Figure 2

When the model is trained, only two actors interact with the environment, that is, only the green cycle is needed. The input state of each agent is different from that of a single agent. After the environment outputs the next full information state Sall, actor1 and actor 2 can only obtain some of the status information S1 and S2 that they can observe. During the training process, critic1 and critic2 can obtain the full information state, and at the same time, they can also obtain the strategic actions a1 and a2 adopted by two agents. That is, although actor cannot see all the information and do not know the strategies of other actors, each actor has a god perspective mentor, who can observe all the information and guide the corresponding actor optimization strategy.

The whole process is centralized training and decentralized execution. This improvement will theoretically alleviate the problem of unstable environment.

That is $P(s`|s, \alpha, \pi_{1,\ldots,}\pi_n) \neq P(s`|s, \alpha, \pi`_{1,\ldots,}\pi`_n)$ For any $\pi i \neq \pi' i \pi i \neq \pi i'$, it is turned to

$P(s'|s, a1, \ldots, aN, \pi1, \ldots, \pi N) = P(s'|s, a1, \ldots, aN) =$
$P(s'|s, a1, \ldots, aN, \pi'1, \ldots, \pi'N)$ $for\ any\ \pi i \neq \pi' i.$

3. Applying MADDPG in UAVCS

3.1 Introduce of UAVCS system

We build a simulation platform based on ROS kentic and gazebo, and the mature simulation platform gazebo provides simulation support. The overall framework of the system consists of 3D real-time display system and heterogeneous agent simulation system. 3D real-time display system is developed based on gazebo simulator, including real-time map scene display and real-time motion display of UAV. Heterogeneous agent system includes multi rotor UAV system. We call the whole system uavcs. We add color and texture to the simulation environment on this platform to generate a quadrotor UAV, and use it to combine with the later algorithm for simulation experiments.

Based on the original use method of MADDOG algorithm provided by OpenAI, this paper proposes the following ideas: to build a neural network for training UAV system, a total of four agents are established in the system, and each agent corresponds to two DDPG structures, one is Eval net, the other is target net. The state space and action space of UAV are introduced into the training model as parameters. After 10000 times of training, the experience will be stored in the experience pool for critical feedback and training. The combination process is shown in Figure 4.

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

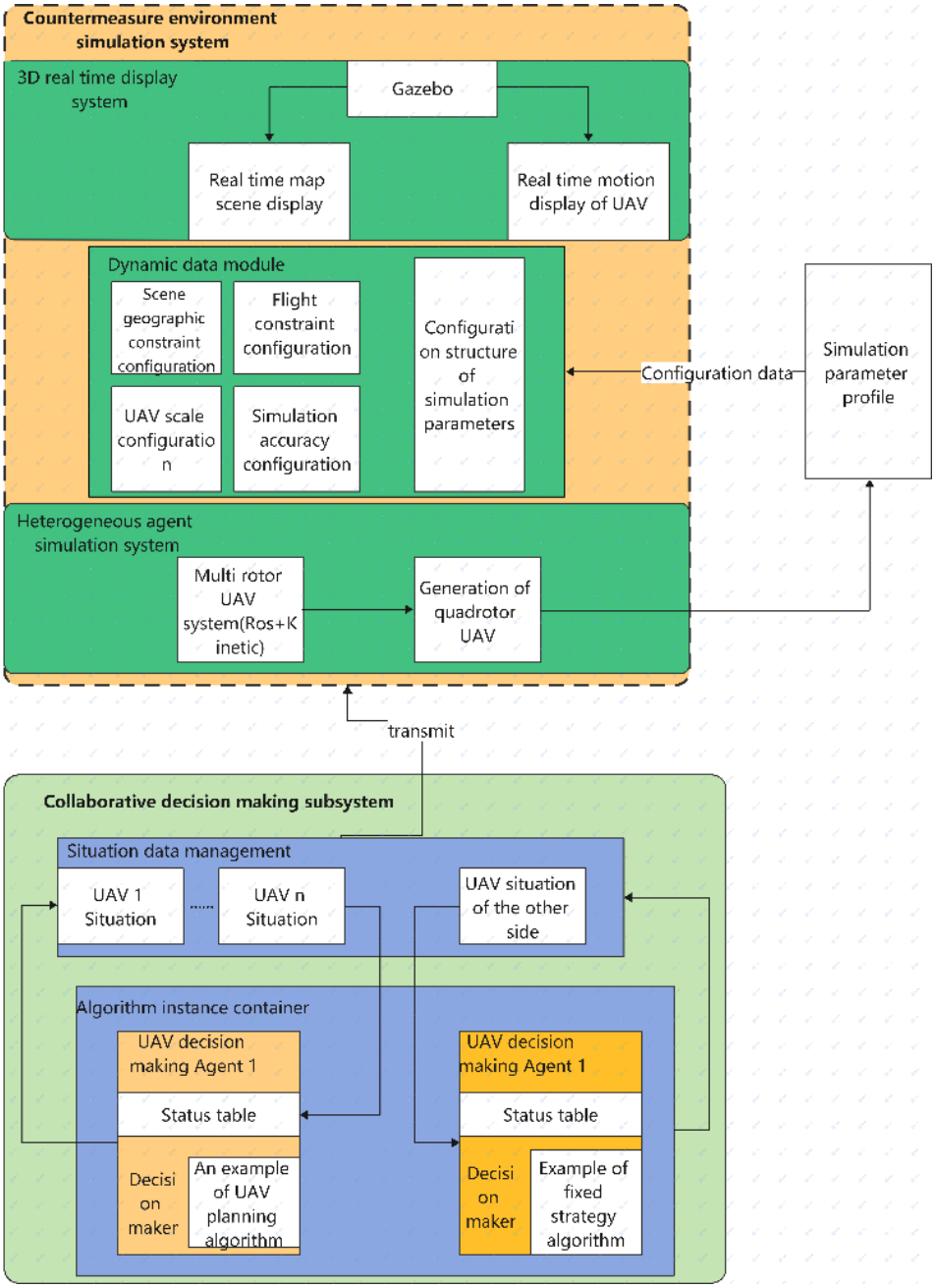

Figure 3: UAVCS

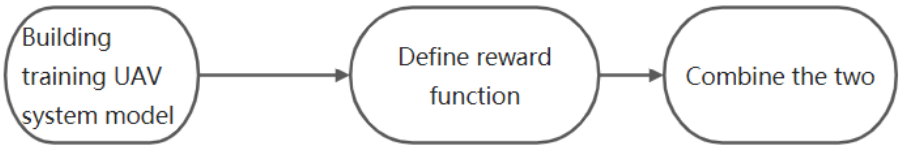

Figure 4: Algorithm

The complete simulation environment after startup is shown in Figure 5.

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

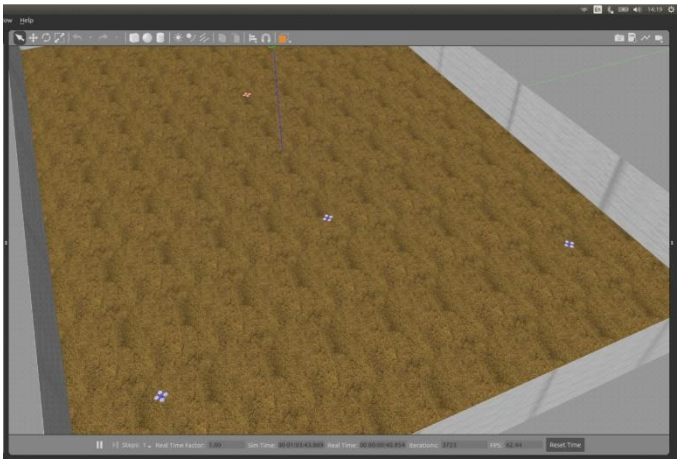

Figure 5: Multi-drone simulation scene based on ROS+Gazebo

3.2 Reward Function

For continuous state space and behavior space, UAV needs a long period of interaction with the environment after random initialization to reach the final state. At this time, the way of giving the corresponding return only after the UAV cluster reaches the final state has the defect of long return period, which easily leads to the failure of effective learning in reinforcement learning process, that is, there is the problem of sparse return.[5]

In order to solve the problem of sparse reward, we modify the learning goal of UAV cluster to increase effective reward[6], so as to speed up the learning speed, and construct the reward function of UAV in different situations to guide the learning direction of deep reinforcement learning.

The formula is as follows:

$$\begin{cases} r_1 = 100 \\ r_2 = -10 \\ r_3 = -100 \\ r_4 = d_{i\_t} - d_{i\_t}` + v_i * \cos(\beta) \end{cases} \tag{1}$$

d_{i_t} ——The distance between UAV and target at current time;

$d_{i\_t}`$ ——The distance between UAV and target next time;

β ——The angle between the speed direction of UAV and the target line at present;

v_i——The speed of the UAV at the current time.

In the simulation experiment, there are four types of task reward. When the UAV cluster completes the pursuit task, the UAV will be rewarded $r_1 = 100$; When the UAV cluster collides with the battlefield boundary, a negative return is given to the UAV that collides, that is $r_2 = -10$; When the UAV cluster does not complete the pursuit task, it will give a negative return to all UAVs, which is $r_3 = -100$, and end the current round of training; In the process of task execution, the guided return function of UAV cluster $r_4 = d_{i\_t} - d_{i\_t}` + v_i * \cos(\beta)$ is used to judge the return of UAV.

The return function of UAV cluster in formula (1) is represented by the change of distance between UAV and target, the velocity direction of UAV and the velocity of UAV. When the distance between the UAV and the target becomes smaller, the corresponding return function is positive; the return function is composed of the speed of the UAV and the speed direction. Under the same speed, the more the direction of the velocity vector points to the target, the higher the return of the UAV; similarly, when the speed direction of the UAV points to the target, the greater the return of the UAV The higher the speed is, the higher the negative return is when the speed direction of UAV is far away from the target [7].

As the UAV cluster starts from the initial state, it takes a long time to reach the target state. If it can not get the effective return of the environment in the long-time intermediate state, it is easy to cause the gradient disappear in the algorithm training process, which leads to the training process can not converge. When the UAV cluster adopts the above guided return function, a value return corresponding to the current value will be generated according to any UAV state in the training process, so as to guide the

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

UAV cluster to gradually transfer to the target state. Therefore, formula (1) can accurately reflect the behavior benefits of UAV. The training results of the algorithm show that the guided return function can better solve the sparse return problem in deep reinforcement learning

4. Results and discussion

4.1 Training result

The result of training 1000 episodes is shown in Figure 6.

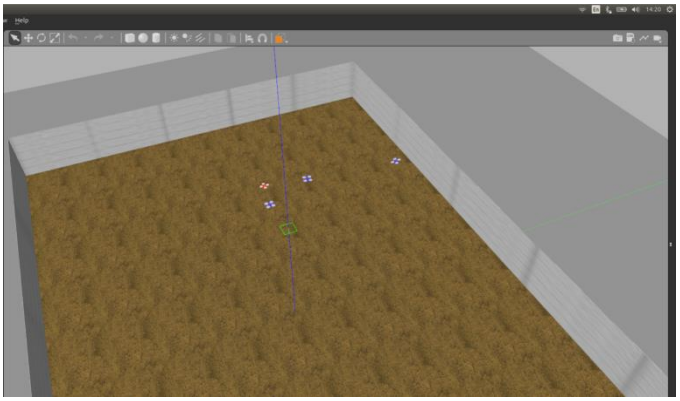

Figure 6: Results after training 1000 episodes

The final tracking effect after training 10000 episodes is shown in Figure 7

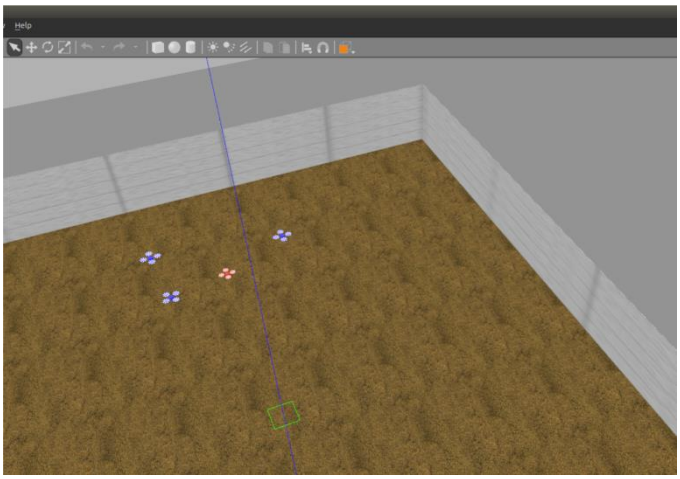

Figure 7: Results after training 10000 episode

The training results show that the tracking effect can be basically achieved after 1000 iterations.

4.2 Discussion

In this paper, the MADDPG algorithm and ROS+Gazebo simulation platform are successfully combined to build a simulation scene for training multi UAVs system with MADDPG algorithm, and a new scheme for planning flight path for multi UAV system is explored. The main work is as follows:

1) We successfully build the platform combined by Ubuntu 16.04 LTS, ROS+Gazebo, MPE and deep reinforcement learning environment, where we can build a variety of UAV cluster collaborative work scenarios.

2) The MADDPG algorithm is successfully implemented in Python, and the MADDPG algorithm and ROS are integrated together. An artificial intelligence system based on DQN algorithm is constructed to train UAV clusters, and a new way to plan multiple UAV routes is found.

The deficiency of this paper is that, subject to the overall academic level of the team, our team can only build a scene with a small number of UAVs, and use this scene for simple simulation. We hope that

ISSN 2616-5775 Vol. 4, Issue 3: 52-59, DOI: 10.25236/AJCIS.2021.040308

with our continuous learning of professional knowledge, we can optimize on the basis of MADDPG, so as to achieve better simulation training effect.

5. Conclusion

The main research content of this paper is the path planning of multi UAV to track targets. By integrating the MADDPG algorithm and ROS + gazebo, we can build the simulation platform of Ubuntu 16.04 LTS + ROS + Gazebo + deep reinforcement learning environment + MPE to create a variety of UAV cluster collaborative work scenarios. A new scheme of planning flight path for multi UAV system is explored. The MADDPG algorithm is successfully implemented in Python language, and the MADDPG algorithm and ROS are integrated together. The artificial intelligence system is constructed to train UAV cluster.

After the training, the UAV cluster can perform the pursuit task well. At the same time, it is verified that the model can be directly applied to cluster pursuit task composed of more UAVs without changing the network model and state space structure. The simulation results show that the MADDPG algorithm can solve the good behavior strategy for the pursuit task of UAV cluster, which reflects the tremendous potential of reinforcement learning algorithm based on artificial neural network in improving the generalization ability of UAV cluster command decision model

References

[1] Volodymyr Mnih, Koray Kavukcuoglu,David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, et al. Human-level control through deepreinforcement learning.Nature, 518(7540): 529–533, 2015.
[2] WEI H. Research of UCAV air combat based on reinforcement learning[D].Harbin :Harbin Institute of Technology,2015(in Chinese).
[3] I. Mordatch and P. Abbeel. Emergence of grounded compositional language in multi-agent populations. arXiv preprint arXiv:1703.04908, 2017.
[4] J. N. Foerster, Y. M. Assael, N. de Freitas, and S. Whiteson. Learning to communicate with deep multi-agent reinforcement learning. CoRR, abs/1605.06676, 2016.
[5] LUO D L,XU Y,ZHANG J P.New progresses on UAV swarm confrontation[J]. Science & Technology Review,2017,35(7) :26-31(in Chinese).
[6] ZHANG Yaozhong, XU Jialin, YAO Kangjia, LIU Jieling. Pursuit missions for UAV swarms based on DDPG algorithm [J]. ACTA AERONAUTICAET ASTRONAUTICA SINICA, 2020, 41(10): 324000-324000.
[7] Lowe R, Wu Y, Tamar A, et al. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments [J]. 2017.