Hypersonic Vehicle Control Based on Deep Reinforcement Learning

Xianlong Ma, Weijun Hu(✉), and Zhiqiang Gao

Northwestern Polytechnical University, Xi'an 710072, China
maxianlong@nwpu.edu.cn, huweijun2021@126.com, 1256113589@qq.com

Abstract. The hypersonic vehicle is an important combat force in the future battlefield. At present, traditional guidance and control need to be designed prior to achieving the operational mission, lacking the ability of independent decision-making and rapid response, and not able to adapt to the development needs of complex battlefield situations in the future. With the rapid development of artificial intelligence, the decision-making ability of deep reinforcement learning has been applied in many aspects. In this paper, for the decision-making problem of autonomous flight maneuvering control of hypersonic aircraft, the deep deterministic policy gradient algorithm is used to design an autonomous flight maneuvering control decision-making algorithm to achieve the trajectory planning for the vertical climb and cruise tasks of the aircraft. Through the simulation test, the autonomous flight from the random initial position to the target position is realized, which proves that the training results have certain generalization. In the end stage, the longitudinal climbing section is extended to three-dimensional space, and the training simulation is carried out, showing the feasibility of the algorithm in the actual situation.

Keywords: Hypersonic vehicle · DDPG · Longitudinal climb · Cruise control

1 Introduction

Hypersonic flight Vehicle (HFV) has fast flight speed, high maneuverability and strong defense penetration ability. In the military field, it is an important weapon and equipment for accomplishing future space operations and global rapid strikes. With the gradual maturity of high-thrust rockets, high-temperature-resistant special materials and scrambling engines, the development of hypersonic vehicles has entered a new stage and has higher requirements for guidance and control systems [1].

There are currently a number of modern control methods for hypersonic vehicle control. Literature [2] used an inversion control structure to solve the trajectory optimal tracking control problem. Literature [3] proposed the use of nonlinear dynamic inverse control method to accomplish the decoupling of velocity and altitude channels, and to achieve precise tracking of altitude and velocity commands. Literature [4] used the idea of trajectory linearization to complete hypersonic nonlinear attitude tracking control. Literature [5] used a robust control method based on signal compensation to complete

Chinese Society of Aeronautics and Astronautics: CASTYSF 2022, LNEE 972, pp. 47–59, 2023.
https://doi.org/10.1007/978-981-19-7652-0_6

the longitudinal control of the aircraft. Literature [6] proposed the backstep sliding mode control of hypersonic vehicle based on cerebellar neural network to achieve longitudinal height and velocity control. Literature [7] proposed a new type of iterative learning control based on sliding mode control to achieve aircraft attitude control. Literature [8] used a middle-aged high-end sliding mode to design a limited-time controller to complete the aircraft attitude control problem. The current modern control methods cannot fully meet the high control accuracy and strong robustness requirements of the control system, so the intelligence and autonomy of hypersonic vehicles are the necessary trends in the development of guidance and control technology [9].

In this paper, for the autonomous flight of hypersonic vehicle, deep reinforcement learning algorithm is used to study its control decision-making problem. In the first place, the framework of the DDPG algorithm and its algorithm training process are established, and the design of the state space that meets the task requirements is selected. On this basis, different reward functions are established for different task learning problems to solve the sparse reward problem of deep reinforcement learning. Verification of the feasibility of deep reinforcement learning in the control problem of aircraft through the simulation test of the agent.

2 Problem Description

2.1 Hypersonic Vehicle Modeling

The atmospheric model for hypersonic vehicle modeling adopts the USSA76 standard atmospheric model developed by the United States in 1976 [10].

The motion equations of the aircraft are divided into the center of mass motion equations and the motion equations rotating around the center of mass. In this paper, the instantaneous equilibrium assumption is adopted, the aircraft is regarded as a mass point, and ignoring the attitude changes of the aircraft. Assuming that the aircraft maneuvers at a fixed speed in the flat, the simplified motion model is shown as below:

$$\begin{cases} \dfrac{d\theta}{dt} = \dfrac{g}{V}\left(n_y - \cos\theta\right) \\ \dfrac{dx}{dt} = V\cos\theta \\ \dfrac{dy}{dt} = V\sin\theta \end{cases} \tag{1}$$

Considering that the aircraft maneuvers in three-dimensional space in practice, the training of the longitudinal plane climb segment model is extended to three-dimensional space, and the trajectory declination, lateral motion coordinates and normal overload will be considered in the aircraft motion model. It is still assumed that the aircraft speed

is fixed, the simplified motion model of the aircraft displays as following:

$$\begin{cases} \dfrac{d\theta}{dt} = \dfrac{g}{V}(n_y - \cos\theta) \\ \dfrac{d\psi_v}{dt} = -\dfrac{g}{V\cos\theta} n_z \\ \dfrac{dx}{dt} = V\cos\theta\cos\psi_v \\ \dfrac{dy}{dt} = V\sin\theta \\ \dfrac{dz}{dt} = -V\cos\theta\sin\psi_v \end{cases} \tag{2}$$

2.2 Reinforcement Learning Method

Reinforcement learning is a branch of machine learning [11]. Figure 1 explains the basic principle of reinforcement learning. Firstly, the agent interacts with the surrounding environment through an action to obtain a reward and a new state, and repeat until the end state. The interaction continues a lot of data is generated, and the reinforcement learning algorithm uses this data to improve the action strategy. Then it uses the new strategy to interact with the environment, generates new data, and uses the new data to optimize the action strategy. After several iterations, the agent can learn to get the maximum optimal strategy for return.

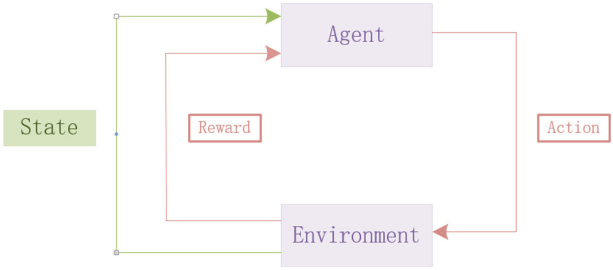


Fig. 1. Reinforcement learning basic framework

Deep Deterministic Policy Gradient (DDPG) [12] is a model-free policy, and Actor-Critic-based policy search method that can be used to solve continuous action space problems. The avoidance strategy means that the action strategy for generating data is not the same strategy as the evaluation and improvement strategy. The action strategy is a random strategy, and noise is added to the output of the strategy network to ensure sufficient exploration; the evaluation strategy is a deterministic strategy. DDPG integrates the successful experience of Deep Q-learning (DQN), that is, experience replay and setting up a separate network, which solves the correlation between data and the problem that the A-C algorithm is difficult to converge. Figure 2 shows the process framework of the DDPG algorithm.

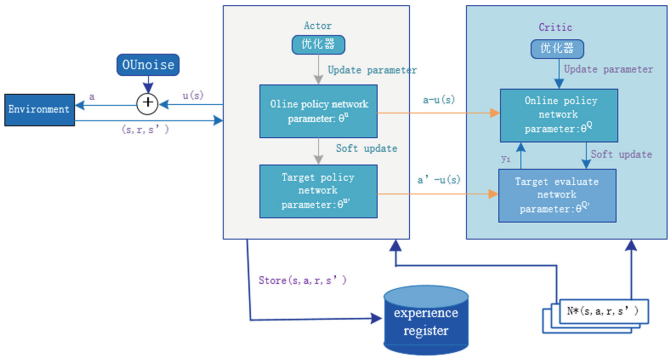


Fig. 2. DDPG algorithm framework

DDPG consists of Online policy network $\mu(s; \theta^{\mu})$, target policy network $\mu'(s; \theta^{\mu'})$, online evaluation network $Q(s, a; \theta^{Q})$, target evaluation network $Q'(s, a; \theta^{Q'})$ and the experience register $R = \{s, a, s', r\}$ consists of five parts. Where s is the current system state, a is the action, s' is the state of the system at the next moment, r is the current return, and θ^{μ}, $\theta^{\mu'}$, θ^{Q}, $\theta^{Q'}$ are the network trainable parameters, respectively. The algorithm process is as follows: firstly, the online decision network outputs action according to the current state, and then acts on the environment after adding noise. The environment feedback gets the reward and the next state, and then the current state, action, reward and next state are stored in the experience cache. Then, the data is randomly sampled from the experience buffer, the online policy network and the online evaluation network are updated by the optimization algorithm, and finally the target evaluation network and the target evaluation network are soft-updated.

The online evaluation network update adopts the method of TD target.

$$y_t = r + \gamma Q'\left(s', \mu'\left(s; \theta^{\mu'}\right); \theta^{Q'}\right) \tag{3}$$

where γ is the reward discount factor.

The loss function of the online evaluation network shows as following:

$$loss = \left(y_t - Q\left(s, a; \theta^{Q}\right)\right)^2 \tag{4}$$

The online evaluation network uses the back-propagation method to update the parameters θ^{Q} according to Eq. (6). a'-u(s).

The online decision network gradient update formula is as follows:

$$\nabla_{\theta^{\mu}} J = \frac{1}{M} \sum_{i=1}^{M} \nabla_a Q\left(s, a \middle| \theta^{Q}\right)\big|_{s=s_i, a=a_i} \nabla_{\theta^{\mu}} \mu\left(s \middle| \theta^{\mu}\right)\big|_{s_i} \tag{5}$$

However, it is difficult to implement this theoretical function in engineering. The goal of the decision network is to maximize the value function $Q(s, a)$ of the output action, so the loss function $loss = -Q(s, \mu(s; \theta^{\mu}))$ can be used to update parameter θ^{μ}.

The target network uses a soft update method to update the network parameters $\theta^{\mu'}$ and $\theta^{Q'}$.

$$\begin{cases} \theta^{\mu'} = (1-\tau)\theta^{\mu'} + \tau\theta^{\mu} \\ \theta^{Q'} = (1-\tau)\theta^{Q'} + \tau\theta^{Q} \end{cases} \tag{6}$$

where τ is the update parameter, which is generally taken relatively small, that is, the parameter is updated a little each time, the training is more stable, and the convergence is better.

3 Problem Description

3.1 Parameter Space Settings

Aiming at the problem of trajectory planning of hypersonic aircraft, this paper regards the aircraft as a mass point, and establishes a system of motion equations for the center of mass. The basic state parameters selected in this paper are as follows:

$$s = \left[V_x, V_y, ex, ey\right] \tag{7}$$

where V_x and V_y are the speed components of the aircraft, including the magnitude and direction of the speed; ex and ey are the distance error informations between the aircraft and the target point, which is beneficial to improve the generalization of the neural network.

According to the definition of the control quantity in the established motion model, the normal overload n_y is selected as the action parameter. Limited by the lift, drag, engine thrust and structural strength of the aircraft, there is a maximum overload limit $n_y \in [-N_{max}, N_{max}]$. N_{max} is the maximum overload.

The three-dimensional climbing state space is defined as $s = [\theta, \psi_v, x, y, z]$, the action space is defined as $a = [n_y, n_z]$, and the maximum overload constraint is satisfied $n_y, n_z \in [-N_{max}, N_{max}]$.

3.2 Reward Function Settings

For the aircraft climbing maneuver task in the longitudinal plane, this paper designs the termination reward function and the flight evaluation reward function, and uses the positive reward to promote the learning of the agent.

Termination reward function is following. The goal of the aircraft maneuvering decision is to guide to the target point, and the conditions that define the completion of the guidance task are as follows:

$$|X_{end} - X_T| \le D_{\min} \tag{8}$$

where X_{end} is the position of the aircraft at the end time, X_T is target x location, D_{min} is the maximum allowable off-target amount.

Set the task completion flag to done, and the description of the termination reward is shown as follows:

$$r_{termin} = \begin{cases} 10 & \text{if done} \\ 0 & \text{else} \end{cases} \tag{9}$$

Flight evaluation reward function is listed as following. The autonomous guidance of the aircraft is a difficult exploration problem. Only setting the sparse return such as the termination reward is difficult to learn, and the algorithm is even more difficult to converge. Therefore, setting the flight evaluation reward function can better guide the aircraft to the target position. The specific form is as follows.

$$r_{position} = \begin{cases} \dfrac{D_t - D_{t+1}}{\Delta T v_{\max}} & D_t > D_{t+1} \\ \dfrac{2(D_t - D_{t+1})}{\Delta T v_{\max}} & D_t \leq D_{t+1} \end{cases} \tag{10}$$

where D_t and D_{t+1} are the distance between the aircraft and the target point at t and $t + 1$ time, respectively; ΔT is the decision-making cycle; v_{max} is the maximum flight speed of the aircraft.

Overall, the total reward function R used for climbing and diving training is shown below:

$$R = r_{termin} + r_{position} \tag{11}$$

For the fixed altitude cruise task, that is, the stable flight height and the ballistic inclination, so this paper designs a reward function for the altitude deviation and the ballistic inclination deviation. y_T is target y location.

$$\begin{cases} r_{\Delta y} = -\dfrac{|y - y_T|}{\Delta y_{\max}} \\ r_{\Delta\theta} = -\sin|\theta| \end{cases} \tag{12}$$

The total reward function used in the cruise task at fixed altitude is shown below:

$$R = r_{\Delta y} + r_{\Delta\theta} \tag{13}$$

The three-dimensional space climbing reward function adds a reward function $r_{\Delta z}$ to the original reward function in the longitudinal plane to eliminate the z-direction error, and the specific expression is as follows. z_T is target z location.

$$r_{\Delta z} = -\frac{|z - z_T|}{\max|z - z_T|} \tag{14}$$

Table 1. Network Hyperparameter Settings

Hyperparameter	Font size and style
Policy network learning rate	0.00001
Evaluate network learning rate	0.0001
reward decay factor	0.99
Soft update parameters	0.001
batch sample size	128
Experience buffer size	100000

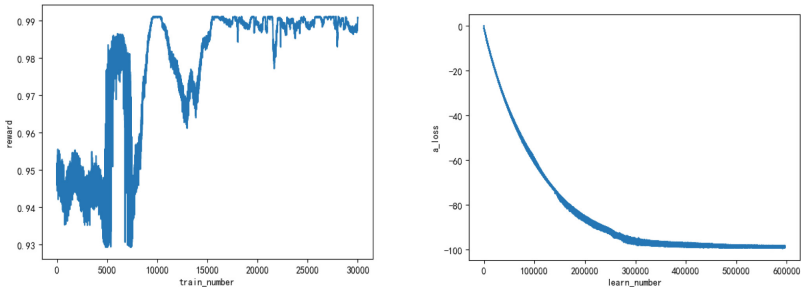
4 Emulation Proof

4.1 Analysis of Training Results

Network Hyperparameter Settings is following (Table 1).

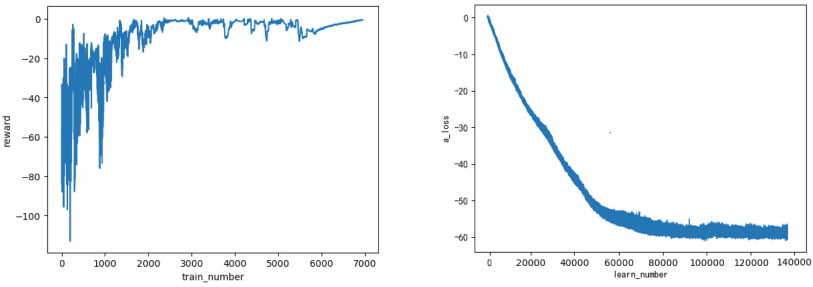
In order to reduce the size of the exploration space and accelerate the learning of the neural network, the size of the aircraft flight airspace is set to 70 km × 15 km, and the normalization method is used to integrate the input state data and action data into a dimensionless quantity to prevent the gradient explosion of the neural network. In the DDPG algorithm, in order to enable the agent to find a better strategy, exploration noise is added to the actions output by the actor network. This paper considers adding a simpler and easier to implement Gaussian noise.

After about eight hours of training, the algorithm is turning stable, the reward function gradually rises to a stable value, the loss function of the policy network gradually decreases and becomes stable, and the neural network tends to converge, which indicates that the maneuvering strategy of the aircraft is gradually optimized and tends to stabilize. The following figures are the reward functions and loss graphs for different tasks (Figs. 3, 4 and 5).



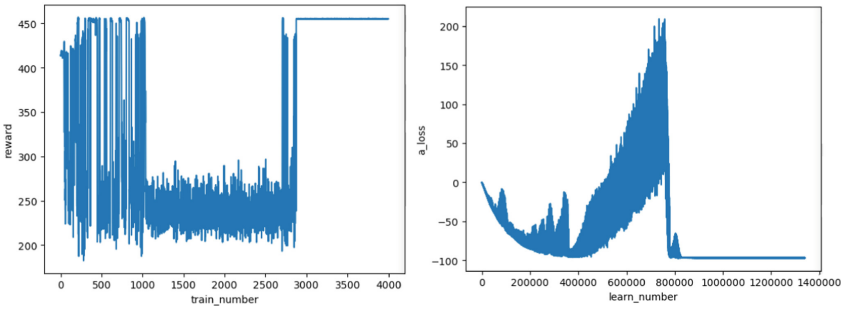
reward function curve Policy Network loss curve

Fig. 3. Climbing section training convergence process curve



reward function curve Policy Network loss curve

Fig. 4. The curve of the training convergence process in the cruise segment



reward function curve Policy Network loss curve

Fig. 5. Convergence process curve of 3D climbing training

4.2 Longitudinal Plane Simulation Test

Climb Simulation Test

Taking the trained climbing strategy network, and carrying out a large number of simulation tests, the initial position of the aircraft is $x = 0$, and the height y is randomly generated within 5000–10000 m. A total of 1000 tests are carried out. The end point distribution of the aircraft is shown in Fig. 6.

The aircraft successfully flew 803 times within the target range required by the mission, with a success rate of 80.3%, including 165 times within 100 m and 174 times within 50 m, effectively completing the preset mission requirements. The simulation test results are as follows (Table 2 and Fig. 7).

From the simulation results of the four representative samples, it can be seen that the aircraft can fly from the starting point to the given target point. In the untrained airspace, the strategy network also has good performance and good generalization ability.

Constant Altitude Cruise Simulation Test

If taking the trained aircraft's constant-altitude cruise strategy model for simulation testing, and randomly generate the initial ballistic inclination in the range of $-2°$ to $2°$ to reflect the performance of the trained strategy network. The ten random test flight

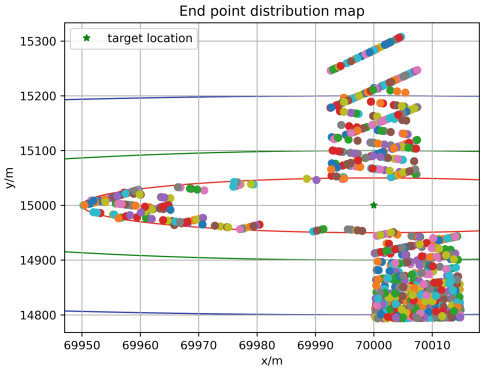


Fig. 6. Climbing simulation test shooting chart

Table 2. Four representative sample simulation test results.

Initial position (x,y)/km	Distance from target point/m	Flight duration t/s
(0,5)	137.54	47.26
(0,7)	202.81	47.03
(0,9)	47.32	46.85
(0,10)	308.40	46.84

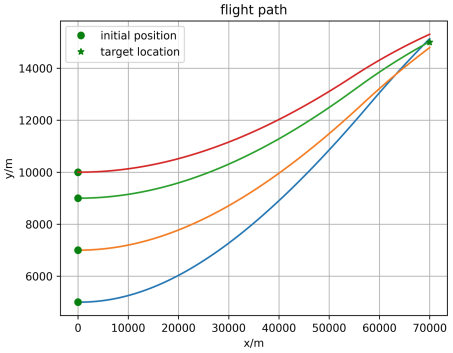


Fig. 7. Four representative sample flight trajectories

trajectories are shown in Fig. 8. It can be seen from the figure that starting from a random ballistic inclination, it can return to the cruising altitude within 20 s and continue to maintain a high cruise, and the maximum altitude difference does not exceed 150 m.

Two groups of representative samples are selected, and the initial ballistic inclination is $-2°$ and $2°$, the change of ballistic inclination and overload during flight are shown in Fig. 9.

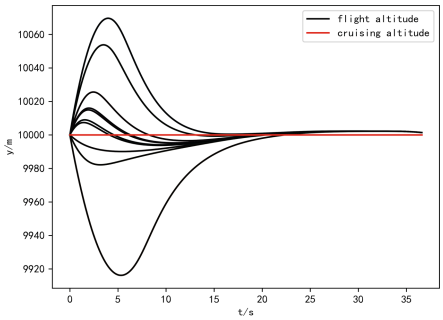


Fig. 8. Random ballistic inclination test flight height change curve

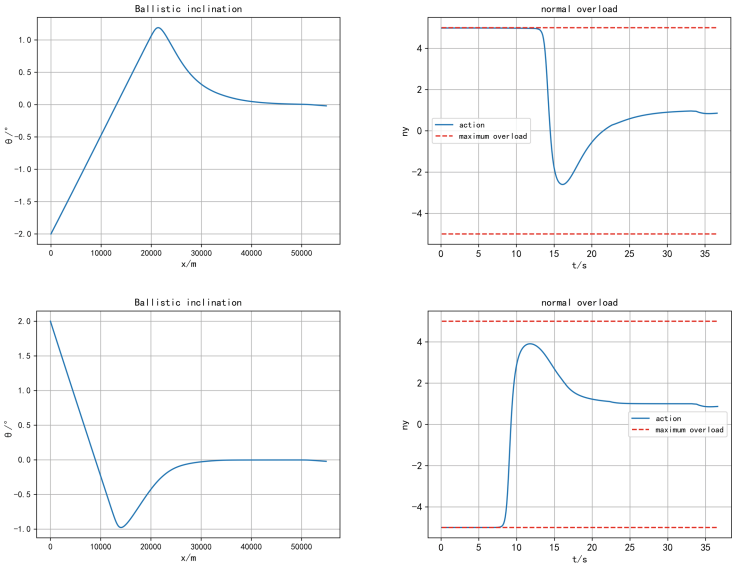


Fig. 9. Change curve of ballistic inclination and overload during initial angle flight

It can be seen from the figure that the aircraft immediately makes a decision based on the current ballistic inclination, executes the maximum overload to quickly correct the current ballistic inclination, then gradually reduces the maneuvering overload to smoothly return the aircraft to the predetermined cruise altitude, and finally maintains a constant overload and constant altitude cruise.

In order to verify the generalization performance of the policy network, set the flight task: keep cruising from the current cruising altitude of 11 km to the target altitude of 12 km, and the results are shown in Fig. 10.

By the given data, it is obvious that the aircraft maintains a cruise at an altitude of 12 km for about 50 s, which proves that the network has a certain generalization ability.

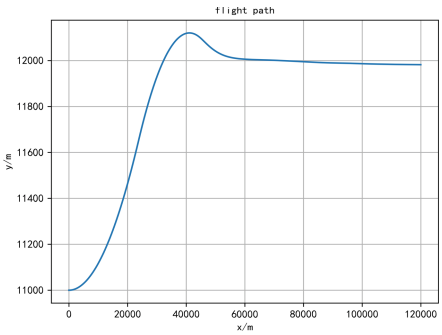


Fig. 10. The generalization test results of the cruise policy network

3D Space Climb Training Test

The trajectory planning is shown in Fig. 11. By the given figure, we can conclude that due to the fast flight speed and the large turning radius, the flight with the maximum normal overload is required for the whole journey. However, the decision-making ability of the policy network also needs to be further improvements.

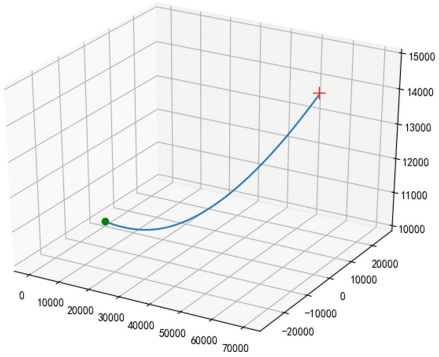


Fig. 11. Schematic diagram of ballistic planning

Loading the converged neural network into the algorithm model and perform 1000 random tests, the initial position of the aircraft is randomly generated within the range of 2 km × 5 km on the plane, and the position of the target point is fixed as (70 km, 15 km, 0). The horizontal coordinate is the number of tests and the vertical coordinate is the distance between the end position of the aircraft and the target point.

Figure 12 shows that 933 times the flight of the 1000 tests reached the target within 100 m, which means a success rate of 93.3% and it effectively completed the preset task. The result proves that the trained policy network enables the aircraft to have a certain autonomous flight ability and a certain generalization ability.

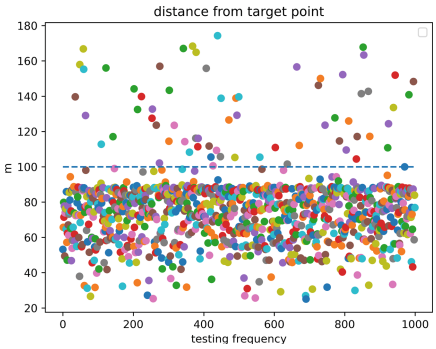


Fig. 12. Random test results

5 Conclusion

This paper conducts simulation training on three maneuvering situations of climbing, cruising. Finally, in order to further study the decision-making model in the three-dimensional space, the vertical plane climbing motion is extended to the three-dimensional space, and the simulation training is continued. The conclusions of this paper are as follows:

(1) State space, action space and different reward functions are designed for climbing, cruising The reward function avoids the sparse reward problem in deep reinforcement learning and enables the aircraft to learn an optimal strategy.
(2) For different tasks, the use of the DDPG algorithm proposed in this paper to make autonomous flight maneuver decisions for hypersonic aircraft, which can provide the ability that the aircraft learn a set of strategies to guide the aircraft to fly to the target point without any prior knowledge. Additionally, since the terminal distance error could meet the mission requirements, the autonomy of the maneuvering control of the aircraft is effectively improved, and the simulation test proves that the algorithm has a certain generalization.

References

1. Na, L., Mengmeng, F.: A review of research advances in hypersonic vehicle control technology. Aerodyn. Missile J. (12), 16–21+62 (2019)
2. Qiang, Q., Xiangwei, B., Baoxu, J.: Adaptive dynamic programming for hypersonic flight vehicle based on backstepping control. Tact. Missile Technol. (05), 102–112 (2021)
3. Wenjun, Y., Ke, Z., Minghuan, Z., et al.: ESO based nonlinear dynamic inversion control for hypersonic flight vehicle. J. Northwest. Polytech. Univ. **34**(05), 805–811 (2016)
4. Xingling, S., Jun, L., Dongguang, L.: Nonlinear attitude control for hypersonic vehicles based on trajectory linearization. Unmanned Syst. Technol. **3**(03), 56–66 (2020)
5. Jiansong, Z., Qinghua, M., Haiqing, L., et al.: Robust longitudinal control method for hypersonic vehicle. J. Project. Rockets Missiles Guid. **40**(02), 19–22 (2020)

6. Daqing, H., Dingguo, J.: Backstepping sliding mode neural network control system for hypersonic vehicle. Opt. Precis. Eng. **27**(11), 2392–2401 (2019)
7. Xiaodong, L., Hao, Z., Jianguo, G., et al.: Iterative learning control combination with adaptive sliding mode technique for a hypersonic vehicle. J. Northwest. Polytech. Univ. **37**(06), 1120–1128 (2019)
8. Guo, J.G., Lu, N.B., Zhou, J., et al.: Fuzzy control of finite time attitude coupling in hypersonic vehicles. Acta Aeronaut. Astronaut. Sin. **41**(11), 41–50 (2020)
9. Yangwang, F.A.N.G., Dong, C.H.A.I., Donghui, M.A.O., et al.: Status and development trend of the guidance and contol for air-breathing hypersonic vehicle. Acta Aeronaut. Astronaut. Sin. **35**(07), 1776–1786 (2014)
10. Bingwei, Y.: Formulization of standard atmospheric parameters. J. Astronaut. (01), 83–86 (1983)
11. Li, Y.: Deep reinforcement learning: an overview (2017). arXiv:1701.07274
12. Lillicrap, T.P., Hunt, J.J., Pritzel, A., et al.: Continuous control with deep reinforcement learning (2015). arXiv:1509.02971