IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 1 July 2023, accepted 16 July 2023, date of publication 19 July 2023, date of current version 24 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3297095

RESEARCH ARTICLE

Autonomous Maneuver Decision-Making Through Curriculum Learning and Reinforcement Learning With Sparse Rewards

YUJIE WEI[1,2], HONGPENG ZHANG[1], YUAN WANG[1], AND CHANGQIANG HUANG[1]

[1]Institute of Aeronautics Engineering, Air Force Engineering University, Xi'an 710038, China
[2]Air Force Xi'an Flying College, Xi'an 710300, China

Corresponding author: Yuan Wang (wangy_af@163.com)

This work was supported in part by the Natural Science Foundation of Shaanxi Province under Grant 2022JQ-584.

ABSTRACT Reinforcement learning is an effective approach for solving decision-making problems. However, when using reinforcement learning to solve maneuver decision-making with sparse rewards, it costs too much time for training, and the final performance may not be satisfactory. In order to overcome the shortcomings, the method for maneuver decision-making based on curriculum learning and reinforcement learning is proposed. First, three curricula are designed to address the maneuver decision-making problem: angle curriculum, distance curriculum and hybrid curriculum. They are proposed according to the intuition that closer destinations are easier to arrive at. Then, they are used to train agents and compared with the original method without any curriculum. The training results show that angle curriculum can increase the speed and stability of training, and improve the performance of maneuver decision-making; distance curriculum can increase the speed and stability of agent training; hybrid curriculum is not better than the other curricula, because it makes the agent get stuck at the local optimum. The simulation results show that after training, the agent can handle the situations where targets come from different directions, and the maneuver decision-makings are rational, effective, and interpretable, whereas the method without curriculum is invalid.

INDEX TERMS Maneuver decision-making, curriculum learning, reinforcement learning, sparse rewards.

I. INTRODUCTION

Autonomous air combat maneuver decision-making refers to that the computer alters the control quantities according to states (such as flight speed, altitude, azimuth, and distance between both sides of air combat), so that the aircraft can occupy a valuable position and then attack the target or escape. At present, the methods for autonomous air combat maneuver decision-making are investigated by simulations.

Hu et al. [1] discretized the state space of air combat into a 13 dimensional space for dimension reduction and designed 15 discrete actions to reduce the difficulty of training. With a reward function based on the situation assessment and the final combat gain, the hybrid autonomous maneuver decision strategy was proposed which can realize the capability of obstacle avoidance, formation and confrontation. Dantas et al. [2] compared supervised learning methods using reliable simulated data to evaluate the most effective moment for launching missiles during air combat. They found that the simulated data can improve the flight quality in beyond-visual-range air combat and increase the effectiveness of offensive missions to hit a particular target. Yang et al. [3] constructed a basic maneuver library for the proximal policy optimization algorithm and a reward function with situation reward shaping in order to increase the convergence rate of training. The simulation results shown that the agent with the proposed method can defeat the enemy. Fan et al. [4] proposed a maneuver decision-making method based on asynchronous advantage actor critic algorithm [5], which incorporates a two-layer reward mechanism of internal rewards and sparse rewards [6],

The associate editor coordinating the review of this manuscript and approving it for publication was Yiming Tang.

[7], [8]. This method can reduce the correlation between samples through multi-threading asynchronous learning.

Jung et al. [9] proposed SAC-LSTM algorithm for maneuver generation in within visual range air-to-air combat under the conditions of a partially observable environment. Bayesian inference is used for maneuver decision-making in [10]. This method uses discrete action space. The experimental results shown that the method can improve the performance. Zhang et al. [11] used Monte Carlo tree and self-play to train the air combat agent for maneuver decision-making.

Air combat maneuver decision-making with sparse rewards is an interesting but difficult problem. Sparse rewards mean that the agent can acquire a reward only when the air combat ends, and the agent cannot acquire any reward before the ending of the air combat. Since the reward signals are sparse, it is difficult for the agent to obtain effective training samples, which may lead to poor training performance. Therefore, it is a difficult problem. On the other hand, sparse rewards can avoid designing reward functions, which is a time-consuming and laborious procedure.

The goal of this paper is to train the agent by reinforcement learning (RL) with sparse rewards in order to enable the agent to cope with targets from different directions (that is, the azimuth range of the target is in [−180°, 180°]). However, we find that reinforcement learning cannot solve the problem of maneuver decision-making. Therefore, on the basis of the intuitions that: if the azimuth of the target is smaller, it is easier for the agent to hit the target; if the distance between the target and the agent is less, it is easier for the agent to hit the target, we propose three curricula for reinforcement learning and evaluate them in order to find an effective curriculum. After training, the curriculum of which the number of win is most is chosen for simulations. The main contributions of this article are as follows: 1. Existing studies discretized action space to reduce the complexity of maneuver decision-making, which is not practical since the real action space is continuous. However, this method uses continuous action space for maneuver decision-making. 2. Existing studies use missile attack zones instead of miss distance, which is a simplified and unrealistic criterion. However, this method uses miss distance as the criterion for air combat instead of the missile attack zone. 3. Existing studies uses handcrafted reward functions, such as angle reward functions and distance reward functions. However, this method does not require any handcrafted reward function, and only uses sparse rewards. 4. According to the characteristics of air combat, three different curricula are designed to train agents, namely, angle curriculum, distance curriculum and hybrid curriculum. 5. Proximal policy optimization (PPO) [12] is combined with the above three curricula, and sparse rewards are used as reward signals. 6. The ablation studies are conducted to investigate the three curricula to find the most effective one. 7. The simulation experiments are conducted to verify the effectiveness of the method.

II. RELATED WORK

Elman pointed out that successful learning may depend on starting from small things [13]. To test the neural network's ability to learn and express the relationship between parts and the whole, Elman trained the neural network to process complex sentences. These networks can learn to solve the task only when they are forced to start from strict memory restrictions, which is equivalent to limiting the range of data provided to the network at the initial learning stage. Bengio et al. formalized such training strategies in machine learning and named them as curriculum learning [14]. For example, when people and animals are provided with examples in a meaningful order (such as gradually increasing the difficulty or quantity) rather than random order, they can learn better.

Jiang et al. proposed self-paced curriculum learning [15], which takes into account both the prior knowledge before the training and the learning process during the training. Based on the generalized boundary criterion of the task order [16], Pentina et al. optimized the average expected classification performance on all tasks, and solved the problem of multiple task curriculum learning by finding the best task order. Sachan and Xing et al. sorted the samples according to the complexities [17], so that the simpler samples can be used in the learning algorithm earlier, and the more difficult samples can be used later. They proposed seven heuristic methods to improve curriculum learning, and compared these methods on four non-convex question answering models. The experimental results shown that these methods can improve the performance. Alex et al. introduced a method of automatically selecting curricula according to the growth of prediction accuracy and the growth of the network complexity to maximize the learning efficiency [18]. They used the proposed method to train LSTM networks [19] and the experimental results shown that the method can significantly accelerate the learning speed. Jiang et al. proposed a new curriculum learning method called MentorNet [20]. The method dynamically learned a data-driven curriculum to overcome the overfitting problem of damaged labels. The experimental results shown that the method can improve the generalization performance of deep networks trained on corrupted data. Zhou and Bilmes proposed a minimax curriculum learning (MCL) to adaptively select subset sequences of training in a series of stages of machine learning [21]. The results shown that MCL achieves better performance and uses fewer samples for training both shallow and deep models while achieving the same performance. Platanios and Stretcu proposed a curriculum learning framework for neural machine translation [22], which determines the training samples displayed to the model at different times during training according to the estimated difficulty of the samples and the current ability of the model. The experimental results shown that the proposed framework can reduce training time, reduce the demand for professional heuristic methods and large quantities of data, and make the overall performance better.

Inspired by human learning, Stretcu and Platanios proposed a novel curriculum learning method, which decomposes challenging tasks into easier intermediate sequences for pre-training the model before processing the original tasks [23], [24]. They trained the model at each level of the hierarchy from coarse labels to fine labels, so as to transfer the acquired knowledge at these levels. The results shown that the classification accuracy of the method is improved by 7%. Zhao et al. put forward a formulation of multitask learning [25]. The formulation learned the relationship between tasks represented by a task covariance matrix and the relationship between features represented by a feature covariance matrix. Li et al. proposed a competence-aware curriculum for visual concept learning by question and answer manner [26]. The method included a neural symbol concept learner for learning visual concepts and a multi-dimensional Item Response Theory model for guiding the learning process with adaptive curriculum. The experimental results shown that the proposed method achieved the most advanced performance with excellent data efficiency and convergence speed through the competence-aware curriculum. Wu and Dyer studied the impact of limited training time budget and noisy data on curriculum learning [27]. The experimental results shown that, under the condition of limited training time budget or noise data, curriculum learning rather than anti-curriculum learning can indeed improve the learning performance. In order to improve the ability of convolutional neural network (CNN) [28], [29], [30] of representing shape and texture information at the same time, Sinha et al. proposed to gradually increase the amount of texture information available in training by reducing the standard deviation of the Gaussian kernel [31]. This training scheme significantly improved the performance of CNN in various image classification tasks without adding additional trainable parameters or auxiliary regularization targets. Weinshall et al. provided theoretical research on curriculum learning when using stochastic gradient descent to optimize convex linear regression loss [32], and proved that the convergence rate of the ideal curriculum learning method monotonously increased with the difficulty of the examples. In order to analyze the impact of curriculum learning on the training of deep convolution network for image recognition, Cohen and Weinshall proposed two methods [33], namely, transfer learning and bootstrapping, to sort training examples by difficulty, and used different pace functions to guide the sampling.

These studies indicate advantages of applying curriculum learning to deep learning: curriculum learning can accelerate the training speed and enhance the performance of the models, which inspire us to use curriculum learning to address the problem of maneuver decision-making, since we find that existing methods cannot solve this problem.

In addition to solving problems in deep learning, curriculum learning is also used to solve problems in RL [34], [35]. Fournier et al. proposed a curriculum learning method based on accuracy [36]. The method was based on deep deterministic policy gradient algorithm [37], [38], which adaptively selected the requirements of accuracy and automatically generated curricula with increasing difficulty. The results shown that the method can improve the learning efficiency. Narvekar et al. proposed a method to automatically generate task sequences for special curricula in RL [39]. The method used the heuristic function in [40] to recursively decompose difficult tasks, and selected tasks that results in the greatest change in policy. Florensa et al. proposed an reverse curriculum generation method to solve the problem of RL [41]. The method used reverse learning to solve difficult robot operation tasks without any prior knowledge. The robot was trained to reach a given target state from a nearby initial state at first. Then, the robot was trained to solve the task from a further initial state. Racaniere and Lampinen [42] proposed an automatic curriculum generation method using a setter-solver paradigm for reinforcement learning algorithms in sparsely rewarding environments. The authors demonstrated the success of our approach in rich but sparsely rewarding 2D and 3D environments. Rane used curriculum learning to solve tasks with sparse rewards in deep RL [43]. The experimental results shown that curriculum learning can improve the convergence speed of training and make the performance better.

These studies indicate the application of curriculum learning in reinforcement learning, which inspire us that we need to design appropriate curriculum according to the characteristics of maneuver decision-making to ensure the curriculum we designed is effective and efficient.

III. METHOD

A. MODELS

The model of the aircraft [1] is shown in (1):

$$\begin{cases} \dot{x} = v\cos\gamma\cos\psi \\ \dot{y} = v\cos\gamma\sin\psi \\ \dot{z} = v\sin\gamma \\ \dot{v} = g(n_x - \sin\gamma) \\ \dot{\gamma} = \dfrac{g}{v}(n_z\cos\mu - \cos\gamma) \\ \dot{\psi} = \dfrac{g}{v\cos\gamma}n_z\sin\mu \end{cases} \tag{1}$$

where x, y, and z are three-dimensional coordinates of the aircraft. γ is the pitch angle, ψ is the yaw angle. v represents the aircraft velocity. g is the gravitational acceleration. μ, n_x, and n_z are control parameters. The missile model is [24]:

$$\begin{cases} \dot{x}_m = v_m\cos\gamma_m\cos\psi_m \\ \dot{y}_m = v_m\cos\gamma_m\sin\psi_m \\ \dot{z}_m = v_m\sin\gamma_m \end{cases} \tag{2}$$

where x_m, y_m, and z_m are the coordinates of the missile. v_m is the velocity. γ_m is the pitch angle, and ψ_m is the yaw angle.

$$\begin{cases} \dot{v}_m = \dfrac{(P_m - Q_m)g}{G_m} - g\sin\gamma_m \\ \dot{\psi}_m = \dfrac{n_{mc}g}{v_m\cos\gamma_m} \\ \dot{\gamma}_m = \dfrac{n_{mh}g}{v_m} - \dfrac{g\cos\gamma_m}{v_m} \end{cases} \tag{3}$$

where n_{mc} and n_{mh} are overloads. P_m and Q_m are thrust and air resistance. G_m is the mass [33]:

$$P_m = \begin{cases} P_0 & t \le t_w \\ 0 & t > t_w \end{cases} \tag{4}$$

$$Q_m = \frac{1}{2}\rho v_m^2 S_m C_{Dm} \tag{5}$$

$$G_m = \begin{cases} G_0 - G_t t & t \le t_w \\ G_0 - G_t t_w & t > t_w \end{cases} \tag{6}$$

where $t_w = 12.0$ s, $\rho = 0.607$, $S_m = 0.0324$, $C_{Dm} = 0.9$. P_0 is the average thrust, G_0 is the initial mass, G_t is the rate of flow of fuel. K is the guidance coefficient. The two overloads n_{mc} and n_{mh} are:

$$\begin{cases} n_{mc} = K \cdot \dfrac{v_m \cos\gamma_t}{g}[\dot{\beta} + \tan\varepsilon \tan(\varepsilon+\beta)\dot{\varepsilon}] \\ n_{mh} = \dfrac{v_m}{g}\dfrac{K}{\cos(\varepsilon+\beta)}\dot{\varepsilon} \end{cases} \tag{7}$$

$$\begin{cases} \beta = \arctan(r_y / r_x) \\ \varepsilon = \arctan(r_z \Big/ \sqrt{r_x^2 + r_y^2}) \end{cases} \tag{8}$$

$$\begin{cases} \dot{\beta} = (\dot{r}_y r_x - r_y \dot{r}_x) \Big/ (r_x^2 + r_y^2) \\ \dot{\varepsilon} = \dfrac{(r_x^2 + r_y^2)\dot{r}_z - r_z(\dot{r}_x r_x + \dot{r}_y r_y)}{R^2\sqrt{r_x^2 + r_y^2}} \end{cases} \tag{9}$$

where β and ε are the yaw angle and pitch angle of line-of-sight, respectively. The line-of-sight vector is the distance vector r, where $r_x = x_t - x_m$, $r_y = y_t - y_m$, $r_z = z_t - z_m$, and $R = \|r\| = \sqrt{r_x^2 + r_y^2 + r_z^2}$. If the distance between the missile and the target is less than 12 m, the target is hit. If the missile fails to hit the target after 120 s, the target is regarded as missed. In the midcourse guidance stage, the target is missed if its azimuth relative to the aircraft exceeds 60°. The maximum time of simulation is 200 s.

B. PROXIMAL POLICY OPTIMIZATION

The problem of RL is generally described by Markov Decision Process [5]. Considering the tuple $\{\boldsymbol{S}, \boldsymbol{A}, p, \gamma, T\}$, where is $\boldsymbol{S}$ the state space, $\boldsymbol{A}$ is the action space, p is the transition function, γ is the discount factor and T is the horizon. $\pi_\theta(a_t|s_t)$ is the policy. The main goal of RL is to maximize the discounted sum of rewards $E_\pi[\sum_{k=0}^{T}\gamma^k r_{t+k}]$. In air combat, this goal equals to maximize the number of win. To address the problem of RL, several methods have been proposed, such as policy gradient algorithm [43], deterministic policy gradient algorithm [37], [38], actor-critic algorithm [5], trust region policy optimization (TRPO) [44] and PPO [15].

TRPO proves that its return after policy update is non-decreasing and avoids storing a dense Hessian matrix or policy gradients in order to reduce computational cost. However, its policy updates is not stable. Therefore, PPO clips the probability ratio of the policy to punish the policy changes that make the probability ratio far from 1, making the learning more stable. The objective of PPO is:

$$L^{CLIP}(\theta) = \mathrm{E}\left[\min(r_t(\theta)A_t, \mathrm{clip}(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t)\right]$$
$$P_r(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \tag{10}$$

where, θ_{old} is the parameters of the current policy, θ is the corresponding undated parameters and A_t is the advantage function. The self-play is used for training agents: First, the agent generates transitions (s_t, a_t, and r_t) by fight against itself, that is, self-play. Second, the transitions are saved. Then, sample transitions according to the probabilities and train neural networks from the transitions. Finally, perform self-play again with the trained neural networks.

C. CURRICULUM LEARNING

Since the initial azimuth angle is randomly sampled from [−180°, 180°], there are few valuable samples and the agent trained over these samples can hardly make useful decisions. Meanwhile, large initial azimuth makes the training slow (that is, the training time is longer when the number of win reaches a certain value) and the performance may not be satisfactory (that is, the number of win is few). The main reason is that in air combat, the agent needs to reduce the line-of-sight at first, thus, the airborne radar can continuously detect the target after the missile is launched, so as to avoid missing the target in the midcourse guidance phase. For example, when the range of initial azimuth angle is [−45°, 45°], it is easier for the agent to obtain samples of hitting the target, thus, the agent trained over these samples can make effective maneuver decisions. However, when the initial azimuth range is [−180°, 180°], it is much more difficult for the agent to detect and track the target and a large number of useless samples are generated, from which the agent can hardly obtain useful information. Therefore, to solve these problems, we propose to use curriculum learning to improve the probability of the agent obtaining effective samples and enable the agent to make effective maneuver decisions by training.

Curriculum learning [21] is inspired by the process of human learning in the real world: humans do not learn difficult tasks from scratch, on the contrary, they start from simpler tasks and gradually learn to solve more difficult tasks. For example, a student begins to learn addition, subtraction, multiplication and division, function and limit at first, then, the student tries to learn differential and integral calculus. Inspired by human learning, curriculum learning is a learning strategy for machine learning, that is, first learn to solve simple tasks, then improve the difficulty and learn to solve these more difficult tasks.

We combine the characteristics of air combat with curriculum learning and propose three different curricula to investigate the impact of different curricula on the training process and the performance of agents to find the valid method. The agent training strategies based on curriculum learning are as follows: 1. Curriculum design. First, design the difficulties of the curricula, and determine the condition for curriculum transfer (namely, increasing the curriculum

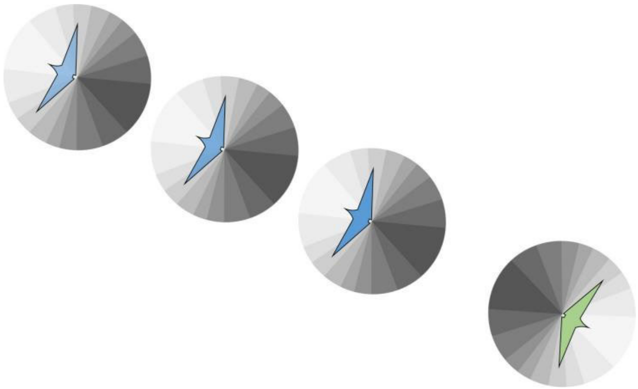


FIGURE 1. Angle curriculum.

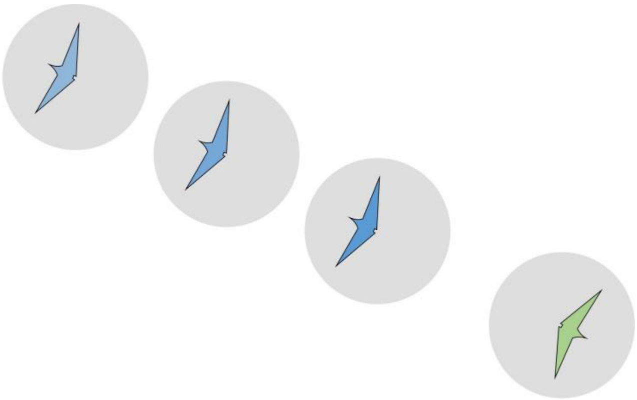


FIGURE 2. Distance curriculum.

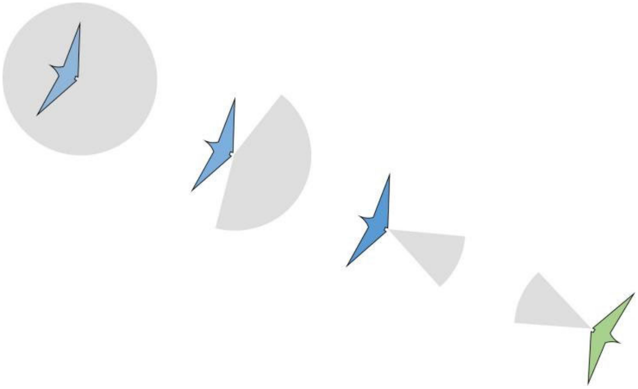


FIGURE 3. Hybrid curriculum.

difficulty). 2. Curriculum learning. Train the agent in the initial curriculum and test the agent. If the test results meet the condition for curriculum transfer, the agent can be trained in the next curriculum. The three different curricula are shown below.

1) ANGLE CURRICULUM

In the air combat, the larger azimuth of the target means the more difficult it is for the agent to hit the target. For example, if the azimuth is 180°, which means that the target is directly behind the agent, the radar of the agent cannot detect the target, so the agent needs to maneuver to detect the target. However, if the azimuth is 0°, the target is in front of the agent. Thus, the agent can launch the missile without any maneuver and as long as the target is within the detection range of the radar, the missile may not miss the target in the midcourse guidance phase. In addition, the smaller azimuth means that it takes more time for the target to escape from the detection of the radar. In conclusion, the smaller azimuth of the target, the less difficult it is for the agent to defeat the target; the larger azimuth of the target, the more difficult it is for the agent to defeat the target. Therefore, as shown in Fig.1, the angle curriculum is designed according to the initial azimuth of the target, which varies in: $[-18°, 18°]$, $[-36°, 36°]$, …, $[-180°, 180°]$. Namely, the angle range of the next curriculum is 36° greater than the angle range of the previous curriculum, and finally the initial azimuth range of the target is $[-180°, 180°]$. Meanwhile, the initial distance is randomly and uniformly sampled from the interval of [50,000 m, 150,000 m].

Fig.1 shows the schematic diagram of angle curriculum. The blue plane and the green plane represent the both sides of air combat. The lighter the color of the blue aircraft, the farther away it from the green aircraft. The grey sector represents the range of the initial azimuth angle of the target during the training. The larger the angle of the sector, the larger the range of the initial azimuth of the target. The agent is trained and tested in the minimum range of the initial azimuth first. The condition for curriculum transfer and the method for testing the agent are shown in Section IV. If the agent is able to pass the test, it can be trained in the next curriculum, otherwise continue to train the agent in current curriculum.

2) DISTANCE CURRICULUM

The smaller the distance between the two sides, the less time it takes for the midcourse guidance of the missile, thus, the less difficult it is for the agent to defeat the target. Therefore, as shown in Fig.2, distance curriculum is designed according to the initial distance between the two sides of air combat, which varies in: [50,000 m, 60,000 m], [50,000 m, 70,000 m], …, [50,000 m, 150,000 m], that is, the distance range of the next curriculum is 10,000 m larger than that of the previous curriculum. Meanwhile, the initial azimuth of the target is randomly and uniformly sampled from the interval of $[-180°, 180°]$.

3) HYBRID CURRICULUM

Hybrid curriculum combines angle curriculum with distance curriculum. In hybrid curriculum, the initial azimuth of the target and the initial distance between the both sides are increased simultaneously. Namely, the angle range of the next curriculum is 36° greater than that of the previous curriculum, and the distance range of the next curriculum is 10,000 m greater than that of the previous curriculum. Fig.3 shows the schematic diagram of hybrid curriculum.

TABLE 1. Air combat states.

Symbol	Quantity	Formula
ψ	yaw angle	$\psi=\psi_0+\int\frac{g}{v\cos\gamma}n_z\sin\mu\,\mathrm{dt}$
γ	pitch angle	$\gamma=\gamma_0+\int\frac{g}{v}(n_z\cos\mu-\cos\gamma)\,\mathrm{dt}$
v	velocity	$v=v_0+\int g(n_x-\sin\gamma)\,\mathrm{dt}$
z	altitude	$z=z_0+\int v\sin\gamma\,\mathrm{dt}$
d	distance between the two sides	$d=\lVert r_1-r_2\rVert$
f_1	launch missile	0 or 1
ψ_1	yaw angle of the missile	$\psi_m=\psi_{m0}+\int\frac{n_{mc}g}{v_m\cos\gamma_m}\,\mathrm{dt}$
γ_1	pitch angle of the missile	$\gamma_m=\gamma_{m0}+\int\frac{n_{mh}g}{v_m}-\frac{g\cos\gamma_m}{v_m}\,\mathrm{dt}$
d_1	distance between the missile and the other side	$d_1=\lVert r_{m1}-r_2\rVert$
β	heading crossing angle	$\beta=\arccos(\frac{v_1\cdot v_2}{\lVert v_1\rVert\lVert v_2\rVert})$
f_2	launch missile from the other side	0 or 1

D. AIR COMBAT STATES AND ACTIONS

As shown in Table 1, the air combat state is a one-dimensional vector with 11 elements: ψ, γ, v, z, d, f_1, ψ_1, γ_1, d_1, β, f_2, the actions are μ, n_x, n_z and launch.

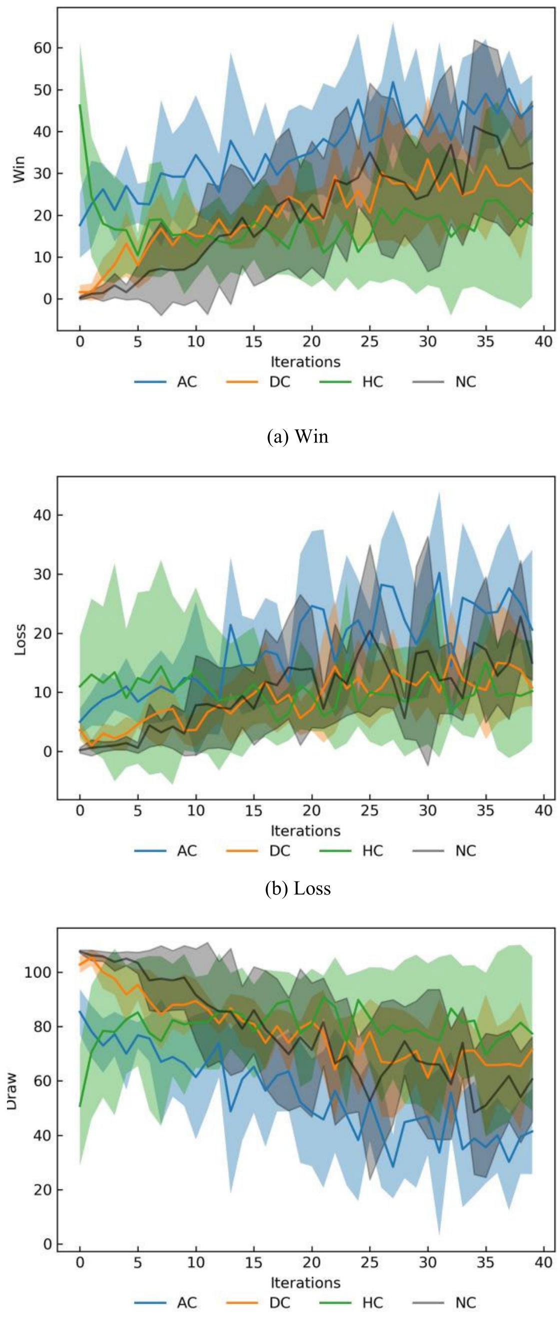
IV. EXPERIMENTS AND RESULTS

In order to verify the performance of the training results of the three different curricula and the performance of the trained agents, ablation studies and simulations are performed in this section. For the three different curricula and the original method without curriculum, we perform five times of independent training. Each training consists of forty iterations. Batchsize is 1024, optimizer is Adam. Actor learning rate and critic learning rate are 0.0002 and 0.001, respectively. The layer of the neural network is 2 and the units are 256. The activation is Tanh and the discount factor is 1.

A. ABLATION STUDIES

In this section, the four different methods are compared: angle curriculum (AC), distance curriculum (DC), hybrid curriculum (HC), and no curriculum (NC, the original method without any curriculum, namely, baseline). The training processes of the four methods at each iteration can be seen in Fig.4. The solid line represents the mean of the number of win, loss or draw of the corresponding curriculum, and the shaded part represents the standard deviation of the number of win, loss or draw.

As shown in Fig.4, some shaded parts are negative. However, this phenomenon does not mean that the number of win or loss are negative. It mainly because that the mean values are less than the corresponding standard deviations, which results in the negative shaded parts. Obviously, the more win and the fewer draw represent the better training performance. At the same time, more loss can also indicate that the training performance is better, because both the win and the loss are the results of the agent itself.



(a) Win

(b) Loss

(c) Draw

FIGURE 4. Win, loss and draw.

As shown in Fig.4a, during the training, the wins of angle curriculum, distance curriculum and no curriculum are gradually increasing, and the win of angle curriculum is always the most. The win of distance curriculum rises faster than

TABLE 2. The number of win and loss between different methods at different iterations.

Iteration	AC vs DC	AC cs HC	AC vs NC
10	14-0	24-6	35-7
20	7-2	14-18	57-10
30	13-9	16-11	56-16
40	13-2	25-7	45-1

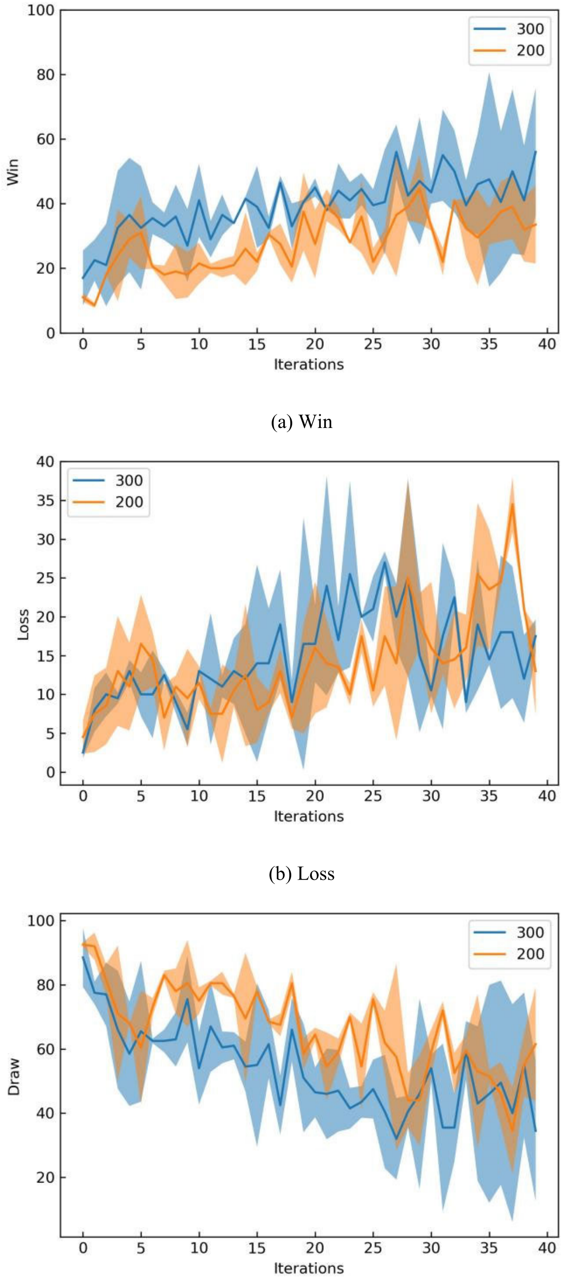
that of no curriculum, and the standard deviation of distance curriculum is less. At the beginning of the training, the win of hybrid curriculum is much, and then it suddenly decreases and remains almost unchanged. It can be seen from Fig.4b that the loss of angle curriculum, distance curriculum and no curriculum are gradually increasing, whereas the loss of hybrid curriculum has no obvious trend of increasing or decreasing. It can be seen from Fig.4c that during the training, the draws of angle curriculum, distance curriculum and no curriculum are decreasing, and the draw of angle curriculum is less. The draw of hybrid curriculum is the least at the beginning of the training, and then it increases and remains almost unchanged. The main difference between no curriculum and the other plots is the shaded parts as shown in Fig.4. Especially in initial iterations (1st-15th): First, the area of the shaded parts of no curriculum is much more than angle curriculum and distance curriculum (especially in Fig.4(a)), which means that no curriculum is more unstable. Second, the line of angle curriculum is always above the others, which means that the performance of angle curriculum is better.

To investigate the robustness of the method, we test neural network architectures with 200 units and 300 units. The neural networks are trained by AC and for each network, we perform two times of independent training. In Figure 5, 300 and 200 are the number of hidden units.

As shown in Figure 5, even the neural network architecture changes, the number of win still increases during training, which means that the method is effective and stable. On the other hand, the agents are trained for 40 iterations because the number of win becomes stable in the latter several iterations as shown in Figure 5, therefore, we think there is no need to continue and we stop the training. Actually, we can also stop the training earlier. Overfitting was not observed during training, because if overfitting exists, the number of win will decrease severely.

B. SIMULATION EXPERIMENTS

In this section, air combat simulation experiments are used to verify the actual performance of the agent after curriculum learning. The initial height of the both sides of air combat is 10 000 m, and the initial yaw angle is 0°. The initial distance between the two sides is uniformly and randomly sampled from [50 000 m, 100 000 m], the initial velocity is uniformly and randomly sampled from [250 m/s, 400 m/s], and the initial yaw angle is uniformly and randomly sampled from [−180°, 180°].



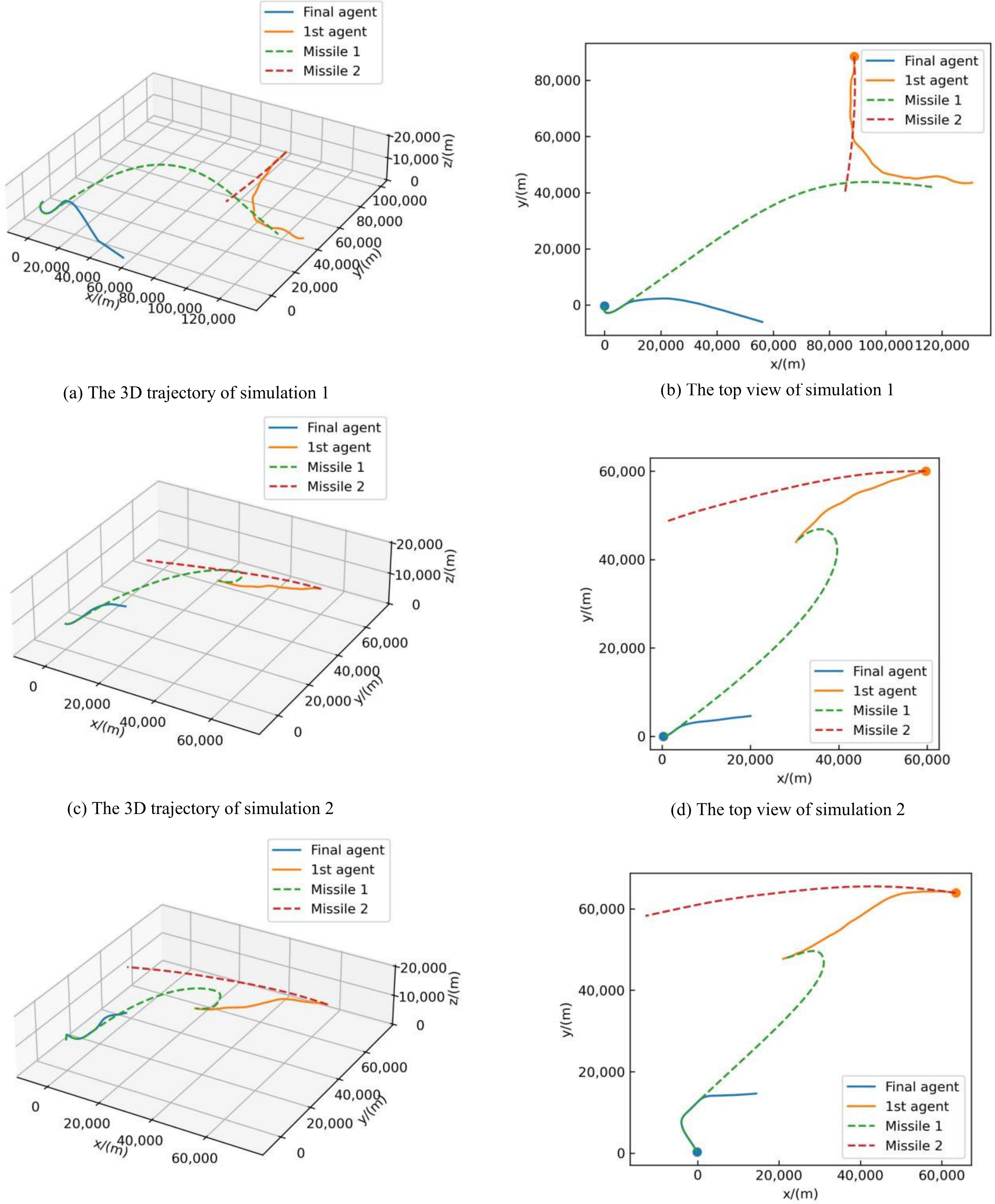
(a) Win

(b) Loss

(c) Draw

FIGURE 5. Training results of different architectures.

Although the curriculum can increase the speed of convergence, but it may or may not improve the final model performance. Therefore, we compare different agents of different iterations to test the performance of different methods. Concretely, 100 simulations are preformed between different methods at different iterations. The initial states of these simulations are random, and the number of win and lose of these simulations are recorded and listed in Table 2.



(a) The 3D trajectory of simulation 1

(b) The top view of simulation 1

(c) The 3D trajectory of simulation 2

(d) The top view of simulation 2

(e) The 3D trajectory of simulation 3

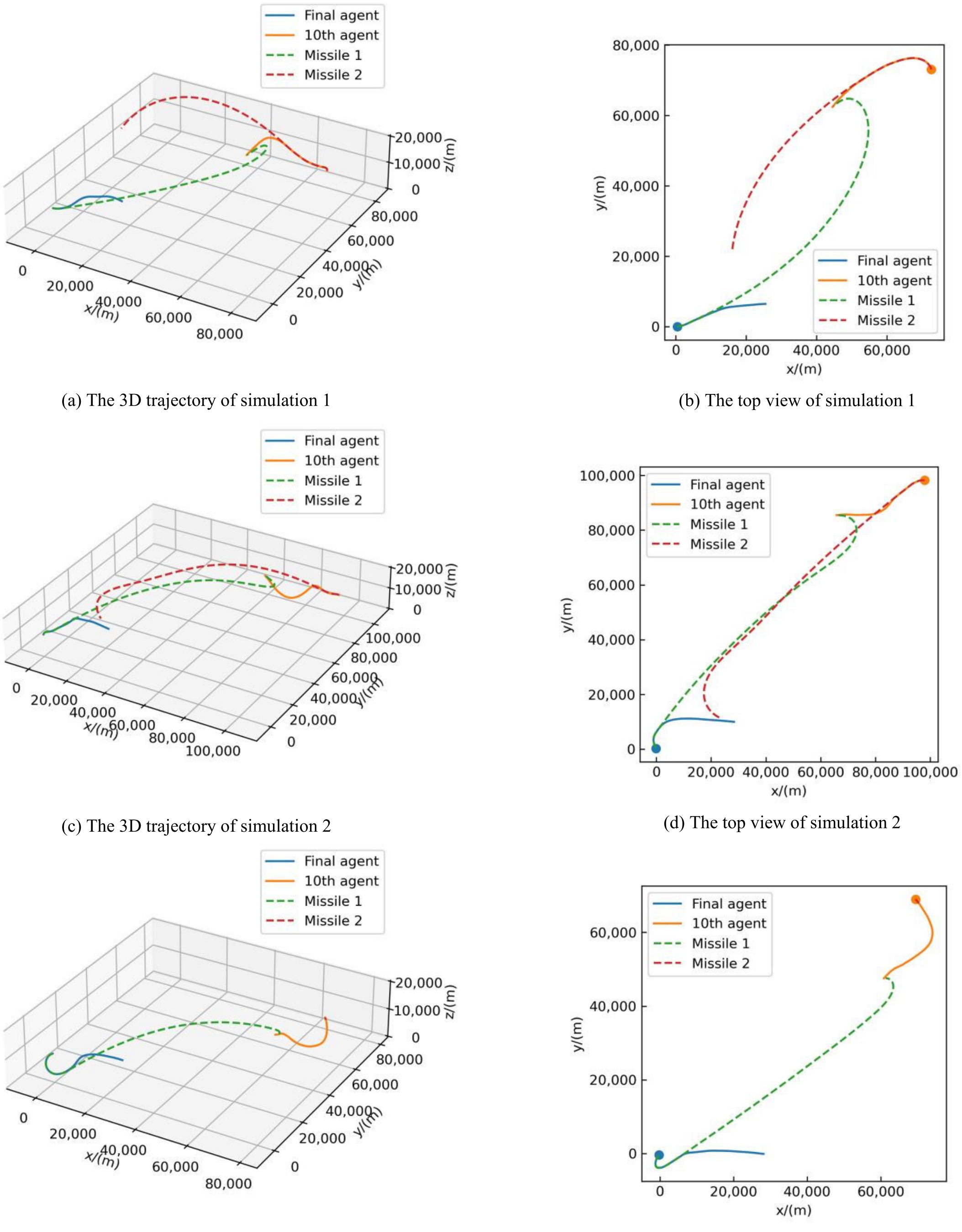
(f) The top view of simulation 3

FIGURE 6. **Air combat results of the first agent and the final agent.**

As shown in Table 2, the number of win of AC is almost always more than that of the other methods. Especially, it is much more than the number of win of NC, which means that curriculum learning is the essential part.

Since the number of win of AC increase faster and are more than the other methods, air combat simulation experiments are performed by agents trained with AC. The final agent (i.e. the fortieth agent) fights against the 1st, 10th and 25th agent in three different initial situations respectively. The results are shown in Fig.6, Fig.7 and Fig.8.

Fig.6 shows the simulation results of the first agent and the final agent in three different initial situations, namely, simulation 1, simulation 2 and simulation 3. The blue solid line represents the trajectory of the final agent, and the orange solid line is the trajectory of the first agent; the green dotted line is the trajectory of the missile of the final



(a) The 3D trajectory of simulation 1

(b) The top view of simulation 1

(c) The 3D trajectory of simulation 2

(d) The top view of simulation 2

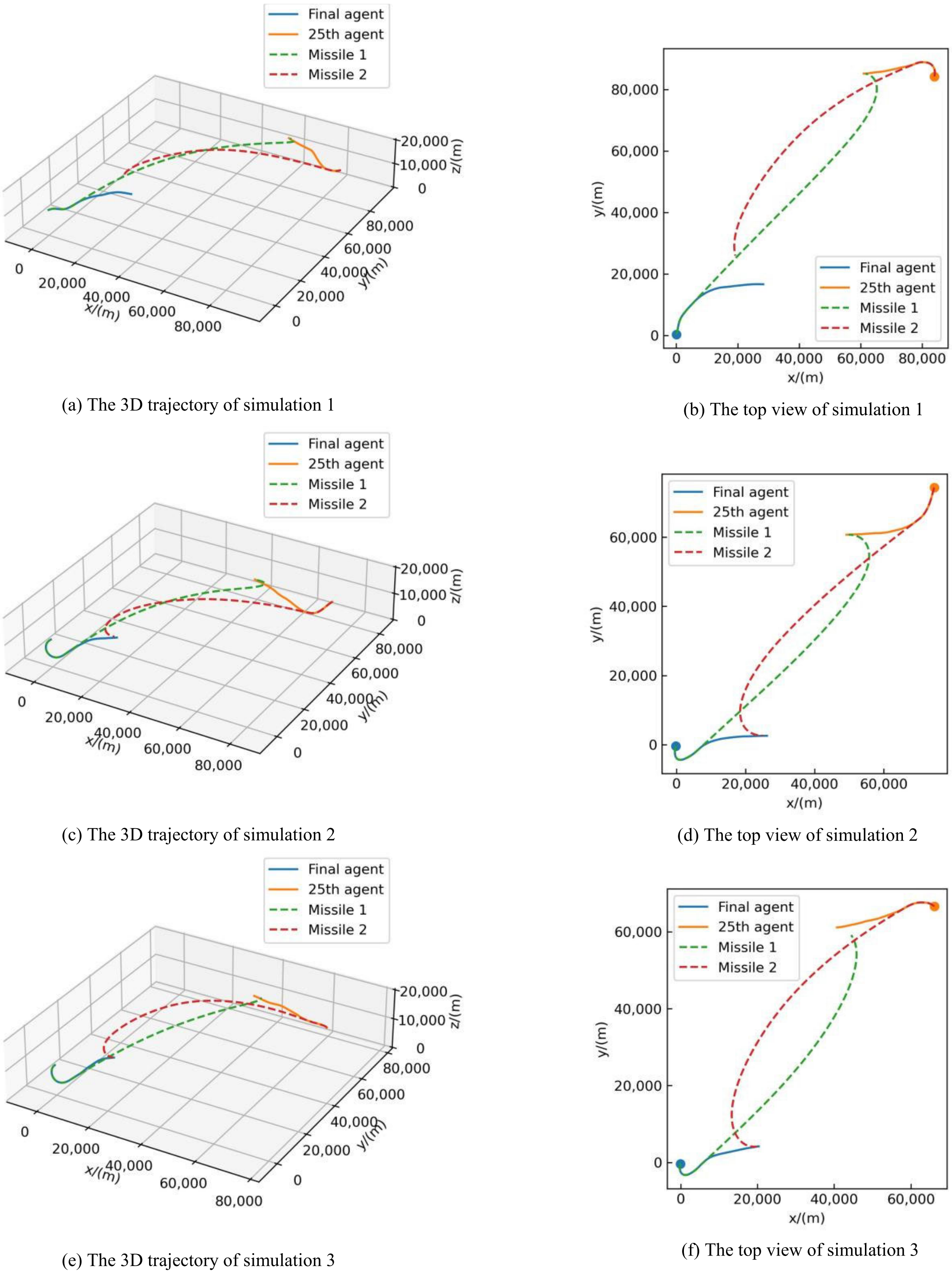
(e) The 3D trajectory of simulation 3

(f) The top view of simulation 3

FIGURE 7. Air combat results of the tenth agent and the final agent.

agent, and the red dotted line is the trajectory of the missile of the first agent. Fig.6a, Fig.6c and Fig.6e are the three-dimensional trajectory of the three simulations respectively. Fig.6b, Fig.6d and Fig.6f are the corresponding top views.

In simulation 1, the initial yaw angle of the final agent is $-89.7°$, and the initial yaw angle of the first agent is $-85.6°$. It can be seen that at the beginning of the air combat, the final agent is pursued by the first agent, that is, the final agent is at a disadvantage. As shown in Fig.6b, the first agent launches the missile immediately and flies forward for a period of time. Then, because the final agent exceeds the detection range of the radar, the missile of the first agent misses the target. After that, the first agent chooses to turn left to get rid of the attack of the missile of the final agent. However, the final agent flies to the left to ensure that the target can be detected by the radar, and then launches the missile. Finally, the simulation ends because the maximum simulation time is reached. At this time, the missile of the final agent has not missed the target, and the distance between the first agent and the missile is less



(a) The 3D trajectory of simulation 1

(b) The top view of simulation 1

(c) The 3D trajectory of simulation 2

(d) The top view of simulation 2

(e) The 3D trajectory of simulation 3

(f) The top view of simulation 3

FIGURE 8. Air combat results of the twenty-fifth agent and the final agent.

than 20 000 m, thus, the radar does not need to detect the target.

In simulation 2, the initial yaw angle of the final agent is 0°, and the initial yaw angle of the first agent is 180°. It can be seen that at the beginning of the air combat, both sides are of equal status. As shown in Fig.6d, the first agent keeps trying to fly to the final agent and launches the missile at the beginning. The final agent first turns to the target and launches the missile after some time. In addition, the final agent does not always approach the target, but first approaches the target, because this maneuver can increase the probability of hitting the target; and then stays away from the target, because this maneuver can reduce the probability of being hit by the target's missile. Finally, the missile of the final agent hits the first agent, so the final agent wins.

In simulation 3, the initial yaw angle of the final agent is 129.5°, and the initial yaw angle of the first agent is 171.2°. As shown in Fig.6f, the first agent launches the missile at the beginning of the simulation and flies towards the final agent to avoid missing the target in the midcourse guidance phase. Because of the large line of sight angle, the final agent flies to the target first and then launches the missile, rather than launching the missile at the beginning as the first agent did. Because of the large line of sight angle when the first agent launches the missile, it is difficult for the missile to hit the target. However, the final agent reduces the line of sight angle by maneuvers, and then launches the missile, so the missile can hit the target.

Fig.7 shows the simulation results of the tenth agent and the final agent in three different initial situations. In simulation 1, the initial yaw angle of the final agent is 4.6°, and the initial yaw angle of the tenth agent is 103.1°. At the beginning, the line of sight angle of the tenth agent is larger than that of the final agent, that is, the tenth agent is at a disadvantage. As shown in Fig.7b, unlike the first agent, the tenth agent does not launch the missile first, but instead flies towards the final agent for a period of time before launching the missile. Obviously, if the tenth agent launches the missile at the beginning of air combat, the missile will miss the target because the target is outside the detection range of the radar, which means that after some time of training, the decision-making ability of the agent has been improved. At the same time, the final agent does not launch the missile at the beginning of air combat, but adjusts the flight direction and flies towards the tenth agent for a period of time before launching the missile. Finally, the missile of the tenth agent has not hit the final agent, but its missile has already hit the tenth agent.

In simulation 2, the initial yaw angle of the final agent is 113.8°, and the initial yaw angle of the tenth agent is 175.6°. At the beginning, the line of sight angle of the final agent is larger than that of the tenth agent, which means that the final agent is at a disadvantage. As shown in Fig.7d, the both sides fly towards each other for a period of time before launching the missile. In addition, they do not always approach each other, but gradually approach at first and then gradually stay away from each other. This is because that being too close to the target can increase the likelihood of being attacked by the target's missile, and when the missile is in the terminal guidance phase, the agent do not need to track the target. Therefore, in order to reduce the probability of being hit by the target's missile, the agent chooses to stay away from the target. Finally, the missile hits the tenth agent first, so the final agent wins.

In simulation 3, the initial yaw angle of the final agent is −128.7°, and the initial yaw angle of the tenth agent is 47.0°. As shown in Fig.7f, the tenth agent launches the missile first. Since the target is outside the detection range of the radar, the missile misses the target. This phenomenon demonstrates that although the performance of the tenth agent is better than that of the first agent, it is still not good enough. In the end, the missile hits the tenth agent first, so the final agent wins.

Fig.8 shows the simulation results of the twenty-fifth agent and the final agent in three different initial situations. In simulation 1, the initial yaw angle of the final agent is 74.0°, and the initial yaw angle of the twenty-fifth agent is 70.1°. Thus, the twenty-fifth agent is pursued by the final agent, that is, the twenty-fifth agent is at a disadvantage. As shown in Fig.8b, the twenty-fifth agent first flies to the left and rear, enabling the radar to detect the final agent and then launches the missile. Meanwhile, the final agent flies to the right first, and then launches the missile. Because the twenty-fifth agent was at a disadvantage at the beginning, the missile hits the twenty-fifth agent first. At this time, the missile of the twenty-fifth agent has not hit the final agent, so the final agent wins.

In simulation 2, the initial yaw angle of the final agent is −107.0°, and the initial yaw angle of the twenty-fifth agent is −107.6°. Thus, the final agent is pursued by the twenty-fifth agent, that is, the final agent is at a disadvantage. As shown in Fig.8d, the final agent changes its flight direction immediately to avoid the disadvantageous situation. Compared with simulation 1, the missile of the twenty-fifth agent at the end of the air combat is closer to the final agent. As the missile hits the twenty-fifth agent first, the final agent wins.

In simulation 3, the initial yaw angle of the final agent is -105.9°, and the initial yaw angle of the twenty-fifth agent is 135.8°. The final agent is at a disadvantage. As shown in Fig.8f, similar to simulation 2, the final agent first changes the flight direction and then launches the missile. Because the final agent was at a disadvantage at the beginning of the air combat, the missile of the twenty-fifth agent hits the final agent first, but its missile has not yet hit the twenty-fifth agent, so the twenty-fifth agent wins.

V. DISCUSSION

According to the ablation studies in Section IV-A, angle curriculum is the best, hybrid curriculum is the worst, and distance curriculum is better than hybrid curriculum but worse than angle curriculum. Although the number of win of no curriculum is slightly more than that of distance curricula in the late stage of training, the training of it is slower and more unstable. However, angle curriculum can not only accelerate the training, but also significantly improve the number of win. Hybrid curriculum is useless, which is mainly because that the agent gets stuck at local optimum in the initial curriculum learning, resulting in the failure of curriculum transfer.

Therefore, in curriculum learning, the curriculum design depends on professional knowledge and common sense of the certain fields. The designed curriculum may not only improve the training speed and performance, but also cause overfitting and failure. For example, although the hybrid curriculum is consistent with the common sense of the air combat, it is invalid. Overall, appropriate curricula can accelerate the training and provide better performance whereas improper curricula can damage the training.

According to the simulation experiments in Section IV-B, the decision-making ability of the agent is gradually enhanced during the training, which indicates that sparse rewards are effective for air combat agents. On the other hand, the agent approaches the target first and then stay away from the target, which is consistent with the characteristics of missile. Because the agent needs to ensure that it can keep the target detected continuously in the midcourse guidance stage, it chooses to approach the target first. After that, if the agent continues to approach the target, it will be hit by the target's missile more likely.

VI. DISCUSSION

The purpose of this paper is to propose an effective RL method for air combat maneuver decision-making with sparse rewards. However, RL usually needs much training time and results in a failure of accomplishing the task.

In order to solve these problems, the method based on curriculum learning and RL is proposed. First, three curricula are designed: angle curriculum, distance curriculum and hybrid curriculum. Then, the curricula are used to train agents for air combat maneuver decision-making. The training results indicate that the performance of angle curriculum is the best, which can not only improve the speed and stability of training, but also improve the performance of the agent; distance curriculum can improve the speed and stability of training; hybrid curriculum is invalid, because it makes the agent get stuck at local optimum at the initial stage of training, which leads to the failure of curriculum transfer. The simulation results show that the proposed method can produce effective agents, and the maneuver decisions made by the agent are consistent with the characteristics of missile. In addition, the initial situation is very important, because a weaker agent in a dominant situation may defeat a stronger agent in a inferior situation.

In future, we need to improve the decision-making ability to ensure that agents can win even when they are at disadvantages.

REFERENCES

[1] J. Hu, L. Wang, T. Hu, C. Guo, and Y. Wang, "Autonomous maneuver decision making of dual-UAV cooperative air combat based on deep reinforcement learning," *Electronics*, vol. 11, no. 3, p. 467, Feb. 2022.

[2] J. P. A. Dantas, A. N. Costa, F. L. L. Medeiros, D. Geraldo, M. R. O. A. Maximo, and T. Yoneyama, "Supervised machine learning for effective missile launch based on beyond visual range air combat simulations," 2022, *arXiv:2207.04188*.

[3] K. Yang, W. Dong, M. Cai, S. Jia, and R. Liu, "UCAV air combat maneuver decisions based on a proximal policy optimization algorithm with situation reward shaping," *Electronics*, vol. 11, no. 16, p. 2602, Aug. 2022.

[4] Z. Fan, Y. Xu, Y. Kang, and D. Luo, "Air combat maneuver decision method based on A3C deep reinforcement learning," *Machines*, vol. 10, no. 11, p. 1033, Nov. 2022.

[5] V. R. Konda and V. S. Borkar, "Actor-critic–type learning algorithms for Markov decision processes," *SIAM J. Control. Optim.*, vol. 38, no. 1, pp. 94–123, 1999.

[6] K. H. Chen, D. Du, and P. Zhang, "Monte–Carlo tree search and computer go," in *Proc. Adv. Inf. Intell. Syst.*, vol. 251, 2009, pp. 201–225.

[7] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, Oct. 2017.

[8] A. Fawzi, M. Balog, A. Huang, T. Hubert, B. Romera-Paredes, M. Barekatain, A. Novikov, F. J. R. Ruiz, J. Schrittwieser, G. Swirszcz, D. Silver, D. Hassabis, and P. Kohli, "Discovering faster matrix multiplication algorithms with reinforcement learning," *Nature*, vol. 610, no. 7930, pp. 47–53, Oct. 2022.

[9] J. H. Bae, H. Jung, S. Kim, S. Kim, and Y.-D. Kim, "Deep reinforcement learning-based air-to-air combat maneuver generation in a realistic environment," *IEEE Access*, vol. 11, pp. 26427–26440, 2023.

[10] H. Changqiang, D. Kangsheng, H. Hanqiao, T. Shangqin, and Z. Zhuoran, "Autonomous air combat maneuver decision using Bayesian inference and moving horizon optimization," *J. Syst. Eng. Electron.*, vol. 29, no. 1, pp. 86–97, Feb. 2018.

[11] H. Zhang, H. Zhou, Y. Wei, and C. Huang, "Autonomous maneuver decision-making method based on reinforcement learning and Monte Carlo tree search," *Frontiers Neurorobot.*, vol. 16, Oct. 2022, Art. no. 996412.

[12] L. Engstrom, A. Ilyas, S. Santurkar, D. Tsipras, F. Janoos, L. Rudolph, and A. Madry, "Implementation matters in deep policy gradients: A case study on PPO and TRPO," 2020, *arXiv:2005.12729*.

[13] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, no. 1, pp. 71–99, Jul. 1993.

[14] Y. Bengio, J. Louradour, and R. Collobert, "Curriculum learning," in *Proc. Int. Conf. Mach. Learn.*, Aug. 2009, pp. 41–48.

[15] L. Jiang, D. Meng, Q. Zhao, S. Shan, and A. Hauptmann, "Self-paced curriculum learning," in *Proc. AAAI Conf. Artif. Intell.*, Dec. 2016, pp. 2694–2700.

[16] A. Pentina, V. Sharmanska, and C. H. Lampert, "Self-paced curriculum learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 5492–5500.

[17] M. Sachan and E. Xing, "Easy questions first? A case study on curriculum learning for question answering," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Aug. 2016, pp. 453–463.

[18] G. Alex, G. Marc, J. Bellemare, R. Munos, and K. Kavukcuoglu, "Automated curriculum learning for neural networks," in *Proc. Int. Conf. Mach. Learn.*, Aug. 2017, pp. 1131–1320.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[20] L. Jiang, Z. Y. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2018, pp. 2304–2313.

[21] T. Zhou and J. Bilmes, "Minimax curriculum learning: Machine teaching with desirable difficulties and scheduled diversity," in *Proc. Int. Conf. Learn. Res.*, May 2018, pp. 1459–1466.

[22] E. A. Platanios and O. Stretcu, "Competence-based curriculum learning for neural machine translation," in *Proc. Int. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, Jun. 2019, pp. 1162–1172.

[23] O. Stretcu and E. A. Platanios, "Coarse-to-fine curriculum learning for classification," in *Proc. Int. Conf. Learn. Represent.*, Apr. 2020, pp. 116–122.

[24] O. Stretcu, E. A. Platanios, T. M. Mitchell, and B. Póczos, "Coarse-to-fine curriculum learning," 2021, *arXiv:2106.04072*.

[25] H. Zhao, O. Stretcu, A. J. Smola, and G. J. Gordon, "Efficient multitask feature and relationship learning," in *Proc. Conf. Uncertainty Artif. Intell.*, Jul. 2019, pp. 777–787.

[26] Q. Li, S. Y. Huang, Y. N. Hong, and S. C. Zhu, "A competence-aware curriculum for visual concepts learning via question answering," in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2020, pp. 141–157.

[27] X. Wu and E. Dyer, "When do curricula work," in *Proc. Int. Conf. Learn. Represent.*, May 2021, pp. 292–307.

[28] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[29] S. Lawrence, C. L. Giles, A. Chung Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jan. 1997.

[30] S. C. Turaga, J. F. Murray, V. Jain, F. Roth, M. Helmstaedter, K. Briggman, W. Denk, and H. S. Seung, "Convolutional networks can learn to generate affinity graphs for image segmentation," *Neural Comput.*, vol. 22, no. 2, pp. 511–538, Feb. 2010.

[31] S. Sinha, A. Garg, and H. Larochelle, "Curriculum by smoothing," 2020, *arXiv:2003.01367*.

[32] D. Weinshall, G. Cohen, and D. Amir, "Curriculum learning by transfer learning: Theory and experiments with deep networks," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2018, pp. 5238–5246.

[33] G. Cohen and D. Weinshall, "On the power of curriculum learning in training deep networks," in *Proc. Int. Conf. Mach. Learn.*, Jun. 2019, pp. 2535–2544.

[34] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, May 1992.

[35] D. P. Bertsekas, *Reinforcement Learning and Optimal Control*, 1st ed. Beijing, China: Tsinghua Univ. Press, 2019, ch. 1, pp. 50–119.

[36] P. Fournier, O. Sigaud, M. Chetouani, and P.-Y. Oudeyer, "Accuracy-based curriculum learning in deep reinforcement learning," 2018, *arXiv:1806.09614*.

[37] W. Shi, S. Song, C. Wu, and C. L. P. Chen, "Multi pseudo *Q*-learning-based deterministic policy gradient for tracking control of autonomous underwater vehicles," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3534–3546, Dec. 2019.

[38] G. Barth-Maron, M. W. Hoffman, D. Budden, W. Dabney, D. Horgan, T. B. Dhruva, A. Muldal, N. Heess, and T. Lillicrap, "Distributed distributional deterministic policy gradients," 2018, *arXiv:1804.08617*.

[39] S. Narvekar, J. Sinapov, and P. Stone, "Autonomous task sequencing for customized curriculum design in reinforcement learning," in *Proc. Int. Joint Conf. Artif. Intell.*, Jul. 2016, pp. 2536–2542.

[40] S. Narvekar, J. Sinapov, and M. Leonetti, "Source task creation for curriculum learning," in *Proc. Int. Conf. Auton. Agents Multiagent Syst.*, May 2016, pp. 566–574.

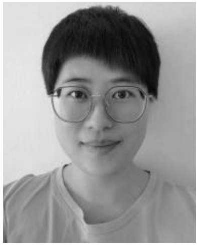
[41] F. Carlos and H. David, "Reverse curriculum generation for reinforcement learning," in *Proc. Conf. Robot Learn.*, Nov. 2017, pp. 1–14.

[42] S. Racaniere and A. Lampinen, "Automated curricula through setter-solver interactions," in *Proc. Int. Conf. Learn. Represent.*, Apr. 2020, pp. 1–18.

[43] S. Rane, "Learning with curricula for sparse-reward tasks in deep reinforcement learning," Ph.D. dissertation, Dept. Elect. Eng., Massachusetts Inst. Technol., Cambridge, MA, USA, 2020.

[44] Y. Liu and K. Zhang, "An improved analysis of (variance-reduced) policy gradient and natural policy gradient methods," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2020, pp. 7624–7636.

[45] T. Kurutach, I. Clavera, Y. Duan, A. Tamar, and P. Abbeel, "Model-ensemble trust-region policy optimization," 2018, *arXiv:1802.10592*.



YUJIE WEI was born in China, in 1989. She received the B.S. degree from Xi'an Jiaotong University, in 2010, and the M.S. degree from the Equipment Academy, in 2013. She is currently pursuing the Ph.D. degree with the Air Force Engineering University.

She has been a Teacher with the Air Force Xi'an Flight College, since 2013, where she was promoted as a Lecturer, in 2016. She has authored and coauthored more than 30 journal articles and ten textbooks. Her current research interests include the command and guidance of airborne early warning aircraft, UAV maneuver decision-making, airborne radar early warning detection, and cooperative operations.

HONGPENG ZHANG was born in 1995. He received the bachelor's degree from Air Force Engineering University, Xi'an, China, in 2017. He is currently pursuing the master's degree in unmanned aircraft combat systems and technology with Air Force Engineering University. His current research interests include autonomous air combat for unmanned combat aerial vehicles and machine learning, including deep learning and its application on unmanned aerial vehicle combat systems.

YUAN WANG was born in 1990. He received the Ph.D. degree in unmanned aircraft combat systems and technology from Air Force Engineering University, Xi'an, China, in 2018.

He is currently a Doctor Tutor and an Assistant Professor with the College of Aeronautics and Astronautics Engineering, Air Force Engineering University. His current research interests include weapon systems, and application engineering and optimal control.

CHANGQIANG HUANG was born in Jiangshu, in 1961. He received the Ph.D. degree in navigation, guidance and control from Northwestern Polytechnical University, Xi'an, China, in 2006. He is currently a Professor and a Doctoral Tutor with Air Force Engineering University. His current research interests include autonomous air combat for unmanned combat aerial vehicles and artificial intelligence, including knowledge extraction, big data application, and air combat simulation systems.