Task-Allocation-Driven Multi-Agent Reinforcement Learning for Cooperative Evasion Guidance of High-Speed Aerial Vehicles

Dong Zhao*, Chida Liu, Can Liu, Jianguo Liu, Jingfan Guo and Tian Yan

Northwestern Polytechnical University, Xi'an, Shaanxi Province, P.R. China

Abstract: Aiming at the cooperative guidance and control problem of multi-agent systems in complex dynamic environments, this paper proposes an intelligent cooperative maneuvering guidance strategy integrated with role assignment design. High-speed vehicles are divided into Supportive Agents and Primary Mission Agents: through role-cooperative design, Supportive Agents actively maneuver to divert external disturbances from critical paths, while Primary Mission Agents ensure the accurate achievement of terminal mission objectives through autonomous decision-making under the premise of safety guarantee. Based on the multi-agent Soft Actor-Critic (SAC) framework, this paper presents an improved CD-MASAC (Curriculum-Driven Multi-Agent Soft Actor-Critic for Robust Cooperative Guidance Under Target Constraints) algorithm. By introducing a curriculum learning strategy and a dynamic learning rate adjustment mechanism, the training efficiency and convergence stability under complex constraints are significantly enhanced. Furthermore, a control loop with the desired axial velocity as the output is designed; by adjusting the flight rate in real time, the variable-speed capability of the vehicle is fully utilized, which not only satisfies terminal trajectory constraints but also effectively reduces energy consumption during maneuvering and improves flight sustainability. Simulation results demonstrate that the proposed strategy exhibits strong robustness and high control accuracy under significant environmental uncertainties, providing a universal guidance and control scheme for future highly autonomous aerial systems.

Keywords: Intelligent Aeronautical Systems, Multi-agent Cooperative Guidance, Autonomous Flight Decision-making, Adaptive Control, Real-time Trajectory Management.

INTRODUCTION

With the rapid development of advanced aerial autonomous technologies and the increasing complexity of flight environments, high-speed vehicles are facing increasingly severe environmental constraints and dynamic challenges when performing complex missions. Traditional single-vehicle guidance and decision-making methods often struggle to balance system resilience and execution efficiency when dealing with multiple uncertainties and safety-critical path planning.To achieve a higher level of aerial autonomy, it is particularly important to investigate intelligent guidance laws with conflict resolution capability and cooperative maneuver planning. This requires not only that vehicles can achieve autonomous avoidance under dynamic disturbances, but also that high-precision constraint satisfaction of terminal mission states can be realized through risk sharing among multiple vehicles on the premise of ensuring flight safety [1-3].

In the existing evasion guidance law design, there are mainly two categories: classical game-based evasion strategies and intelligent game-based evasion strategies.

*Address correspondence to this author at Northwestern Polytechnical University, Xi'an, Shaanxi Province, P.R. China;
E-mail: dongzhao@mail.nwpu.edu.cn

Classical game-based evasion strategies generate evasion guidance laws to evade pursuers using classical methods such as differential games and optimal control, on the basis of simultaneously considering the capabilities of both offensive and defensive sides.

Studies [3-8] attempt to solve the pursuit-evasion game problem using differential game methods and have improved these methods to address their shortcomings in such games. Studies [9-14], on the other hand, seek to resolve the aircraft pursuit-evasion game problem through optimal control methods. They optimize from the perspectives of energy consumption, miss distance, and stability in pursuit-evasion games to derive evasion guidance laws. However, classical game-based maneuvers rely heavily on information about both offensive and defensive sides, making them highly challenging in engineering practice. Additionally, the speed of generating maneuver commands using classical game-based methods increases exponentially with the number of aircraft on both sides, which is unfavorable for evading coordinated pursuit by multiple pursuers using such methods.

Unlike classical game-based evasion strategies, intelligent game-based evasion strategies acquire the pursuers' motion information through external data links or on-board sensors. The flight control system, based on the characteristics of the guidance method

Received on 27-01-2026 Accepted on 10-03-2026 Published on 16-03-2026

E-ISSN: 3083-3450/2026

for the encounter between the two sides, uses intelligent algorithms to generate the maneuver patterns and directions required to evade the pursuers. It adopts a closed-loop maneuver scheme of "pursuer motion – situational awareness – intelligent maneuver strategy generation – maneuver control implementation" to achieve timely and on-demand maneuvering, thereby increasing the miss distance and improving the evasion probability. Studies [15-28] consider the pursuit-evasion game problem as a Markov Decision Process (MDP) and obtain evasion guidance laws for different scenarios using single-agent and swarm intelligence algorithms, enhancing the intelligence level of evading aircraft.

Intelligent algorithms can broaden the thinking on maneuver evasion and leverage intelligent empowerment to generate evasion strategies that traditional methods cannot achieve. Nevertheless, relying solely on intelligent algorithms to generate evasion strategies in complex scenarios makes it difficult to guarantee effectiveness. In high-dynamic and strongly adversarial complex environments, it is necessary to combine specific problems and introduce existing multi-agent collaborative task design to improve the effectiveness of intelligent evasion strategies in actual offensive and defensive confrontations.

Aiming at the cooperative evasion problem of high-speed aircraft in multi-agent game confrontation scenarios, this paper uses artificial intelligence algorithms such as multi-agent reinforcement learning and curriculum reinforcement learning to generate game evasion strategies, and achieves successful evasion in multi-agent game confrontation scenarios through online orbital game maneuvers. The main research contents and innovations are as follows:

(1) For scenarios involving multiple pursuers, high-speed aircraft are divided into supporter and leader according to task types, and collaborative task planning is designed to achieve cooperative evasion.

(2) To address the cooperative evasion problem between supporter and leader, curriculum learning and dynamic learning rate design are introduced to improve the MASAC algorithm. An intelligent evasion strategy is designed based on the improved CD-MASAC algorithm to realize cooperative evasion between supporter and leader and enhance the agent training effect.

(3) A control strategy using the desired axial velocity command to regulate the axial motion of high-speed aircraft is designed. This avoids the invalidation of the agent's axial acceleration output when the high-speed aircraft reaches the speed limit, and further leverages the advantages of the axial variable speed capability of high-speed aircraft.

2. PROBLEM DESCRIPTION

2.1. Pursuit-Evasion Game Model

This paper designs two high-speed vehicles to cooperatively evade multiple incoming pursuers. Addressing the task scenario of multi-body cooperative evasion with introduced target constraints, the relative motion relationship is shown in Figure **1**.

Where T , E is the high-speed vehicle (Remark 1), and $M_1 - M_3$ is the pursuer. The flight state of each vehicle is described by the ballistic deflection angle ψ , velocity V , and overload n; the position information is

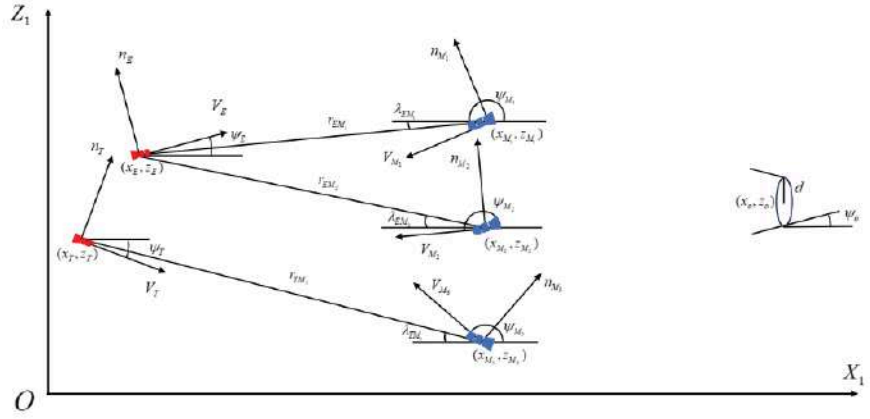

Figure 1: Geometric representation of multi-missile confrontation scenario.

described by the coordinates X_1OY_1 in the inertial coordinate system (x, y). The subscripts T, E and M denote the high-speed vehicle and the pursuer, respectively. Here, r represents the distance between the two sides of the game, and λ represents the line-of-sight (LOS) angle between the two sides. The center coordinates of the target constraint area are (x_o, y_o), the radius is d, and the ballistic deflection angle limit is $(-\psi_o, \psi_o)$.

2.1.1. 3-DOF Model of both Players

Considering the kinematic and dynamic characteristics of the high-speed vehicle and the pursuer during flight, the three-degree-of-freedom (3-DOF) kinematic and dynamic equations for the high-speed vehicle and the pursuer are established as follows:

$$\begin{cases} \dfrac{\mathrm{d}x_i}{\mathrm{d}t} = V_i \cos\theta_i \cos\psi_i \\ \dfrac{\mathrm{d}y_i}{\mathrm{d}t} = V_i \sin\theta_i \\ \dfrac{\mathrm{d}z_i}{\mathrm{d}t} = -V \cos\theta_i \sin\psi_i \end{cases} \tag{1}$$

$$\begin{cases} \dfrac{dV_i}{dt} = g(n_{xi} - \sin\theta_i) \\ \dfrac{d\theta_i}{dt} = \dfrac{g}{V_i}(n_{yi} - \cos\theta_i) \\ \dfrac{d\psi_i}{dt} = -\dfrac{g}{V_i \cos\theta_i} n_{zi} \end{cases} \tag{2}$$

Where: $i \in \{E, T\}$, Since the attack-defense game occurs in an equal-altitude plane, it can be considered that the initial flight path angle of both sides is 0°, i.e., $\theta = 0$, and the normal overload always balances the gravitational acceleration during the process. Therefore, for the aforementioned high-speed vehicle and pursuer, we have $\dfrac{d\theta}{dt} = 0$.

2.1.2. Relative Motion Equations of both Players

Combining the kinetic models of both sides in Eq. (1) and Eq. (2) and the relative motion relationship shown in Figure **1**, the relative kinematic equations of the confrontation process between the high-speed vehicle and the pursuer can be obtained:

$$\begin{cases} \dot{r}_{iM} = -V_i \cos(\psi_{Vi} - \lambda_{iM}) + V_M \cos(\psi_{VM} + \lambda_{iM}) \\ \dot{\lambda}_{iM} = \left(V_i \sin(\psi_{Vi} - \lambda_{iM}) - V_M \sin(\psi_{VM} + \lambda_{iM})\right) / r_{iM} \\ \ddot{r}_{iM} = n_{zi} \sin(\psi_{Vi} - \lambda_{iM}) + n_{zM} \sin(\psi_{VM} + \lambda_{iM}) + r_{iM}\dot{\lambda}_{iM}^2 \\ \ddot{\lambda}_{iM} = \left(n_{zE} \cos(\psi_{Vi} - \lambda_{iM}) + n_{zM} \cos(\psi_{VM} + \lambda_{iM}) - 2\dot{r}_{iM}\dot{\lambda}_{iM}\right) / r_{iM} \\ \dot{\psi}_{Vi} = n_{zi} / V_i \\ \dot{\psi}_{VM} = -n_{zM} / V_M \end{cases} \tag{3}$$

Where M is the pursuer and i is the high-speed vehicle. $i \in \{E, T\}$

2.1.3. Pursuer Guidance Law

In the research of this paper, the pursuer has formed a retro-trajectory (inverse) pursuit situation. Considering the pursuit strategy of the pursuer in the terminal guidance phase, the Proportional Navigation Guidance (PNG) law in the plane is selected as the guidance law for the pursuer. Its lateral overload command is as follows:

$$n_{zcM} = -NV_r\dot{\lambda}_{iM} \tag{4}$$

Where: n_{zcM} is the guidance command of the pursuer; N is the navigation coefficient; V_r is the closing speed of the opposing sides; $\dot{\lambda}_{iM}$ is the angular velocity of the line of sight (LOS).

2.1.4. Autopilot Model

Considering that the evasion of the pursuer by the high-speed vehicle studied in this paper is a short-distance, high-dynamic process, and the evasion problem research in this paper adopts a three-degree-of-freedom particle motion model, the autopilot model of the vehicle is designed as a first-order dynamic element. The relationship between the overload command and the overload response can be expressed as:

$$\frac{n_{zi}(s)}{n_{zci}(s)} = \frac{1}{1 + \tau_i s} \tag{5}$$

Where: n_{zci} is the guidance command of the high-speed vehicle; n_{zi} is the actual flight overload of the high-speed vehicle after passing through the first-order inertial link control system; τ_i is the response time constant of the first-order dynamic characteristic.

2.1.5. Axial Velocity Control Loop

The difference between the desired axial velocity and the actual axial velocity is selected as the input of the velocity control loop, and the output is the axial

overload. The velocity control loop in this paper is as follows:

$$\begin{cases} n_{xi} = n_{x,\max} \cdot sign(\Delta v_i), \left|\Delta v_i\right| \geq \Delta v_{ih} \\ n_{xi} = K_p \Delta v_i + K_i \int \Delta v_i \, dt + K_d \dfrac{d\Delta v_i}{dt}, \left|\Delta v_i\right| < \Delta v_{ih} \end{cases} \tag{6}$$

Where: Δv_i is the difference between the axial desired velocity and the actual velocity of the high-speed vehicle. When the difference between the desired axial velocity and the actual axial velocity is large, the axial overload is maximized to catch up with the axial desired velocity as quickly as possible; when the difference between the two is small, the axial overload is precisely controlled by PID control. Δv_{ih} is a design value that determines the switching node of the axial control loop; K_p , K_i and K_d are the proportional, integral, and differential coefficients in the PID control.

The design of this velocity control loop aims not only to improve guidance accuracy, but also to support the goal of sustainable flight. By taking the desired axial velocity as the output command, the agent can adjust its rate in real time according to current requirements and implement refined trajectory management. This strategy avoids ineffective oscillations of the actuators at the edge of acceleration saturation, thereby reducing control effort and energy loss during maneuvering.

2.2. Pursuit-Evasion Game Problem

The overall mission of high-speed vehicles is defined as a typical two-stage autonomous control process. Stage 1: Safe maneuvering and conflict resolution. Its core objective is to avoid dynamic disturbances through cooperative capabilities, ensuring the survival and safety of the system. Stage 2: Mission accomplishment and terminal constraint satisfaction. On the premise of safety guarantee, the vehicle states are adjusted to meet the predefined constraints of the terminal mission through autonomous overload regulation. This two-stage design forms a general paradigm for intelligent aerial autonomous control.

Assuming that the effective killing radius of the pursuer against the high-speed vehicle is r_M , the condition for successful evasion by the high-speed vehicle is:

$$r_{TM,\min} > r_M \tag{7}$$

After completing the evasion of the pursuer, the high-speed vehicle needs to adjust its flight trajectory through its own overload to facilitate the subsequent strike task:

$$\begin{cases} \left|z_T\right| < d \\ \left|\psi_T\right| < \psi_o \end{cases}, x_T = x_o \tag{8}$$

In the process of attack-defense game confrontation, the high-speed vehicle needs to satisfy its own lateral maximum overload and axial maximum velocity limits:

$$\begin{cases} V_i \in [V_{\min}, V_{\max}] \\ n_{zi} \in [-n_{z,\max}, n_{z,\max}] \end{cases} \tag{9}$$

In summary, the multi-body pursuit-evasion game problem can be formulated as Problem 1.

Problem 1 (Multi-body Pursuit-Evasion Game Problem): In the pursuit-evasion game scenario of Figure **1**, the pursuer adopts (4) as the guidance law. Train an intelligent algorithm to satisfy its own capability constraints (9), complete the evasion of the pursuer (7), and subsequently adjust the flight route (8) to facilitate the subsequent task.

Assumption 1: During the multi-body pursuit-evasion game, both sides maintain a constant altitude. (Remark 2)

Remark 1: This paper considers assigning different tasks to two high-speed vehicles to cooperatively evade the pursuer. According to the different tasks, the high-speed vehicles are divided into Decoy Vehicle (E) and Attacking Vehicle (T) respectively. The attacking vehicle acts as the leader, while the decoy vehicle serves as the supporter.

Remark 2: The maneuvering flight of high-speed vehicles can be divided into longitudinal and lateral maneuvers according to the direction of the maneuver. The two maneuvers can exist separately or simultaneously. Since the change of altitude will produce large energy loss, and in the short-range high-dynamic confrontation process, the altitude change produced by the longitudinal maneuvering capability of the high-speed vehicle platform is too limited compared to the maneuvering range in the lateral plane. Therefore, this paper prioritizes choosing lateral maneuvers to complete evasion.

3. DESIGN OF MULTI-AGENT COOPERATIVE GUIDANCE LAW INTEGRATING TASK ALLOCATION AND CL-DL-MASAC ALGORITHM

This paper proposes an improved multi-agent reinforcement learning-based intelligent maneuvering

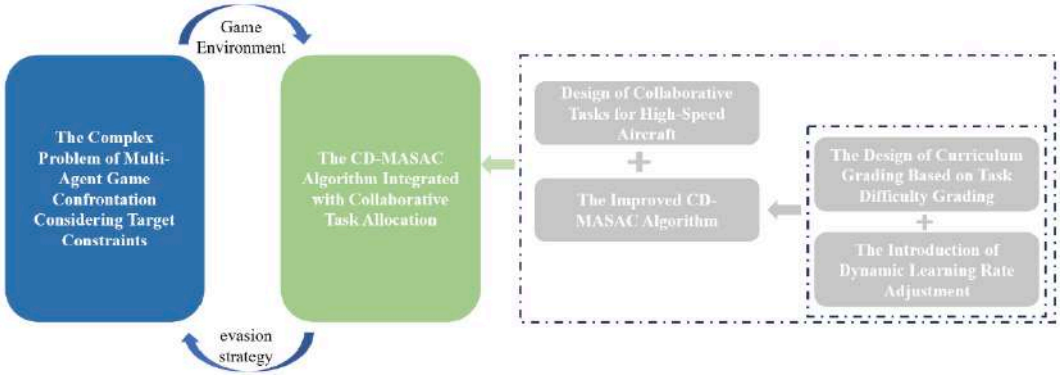

Figure 2: Block Diagram of Intelligent Maneuver Strategy Design.

strategy integrated with task allocation for multi-agent game confrontation scenarios, as shown in Figure **2**.

The algorithmic framework proposed in this paper, namely the task-assignment-driven multi-agent reinforcement learning mechanism, exhibits strong generality. This framework integrates cooperative role assignment, curriculum learning-driven training paradigm, and dynamic learning rate adjustment techniques. As a standardized solution, it can be directly extended to safety-critical aerial systems such as UAV swarm cooperative navigation and automated traffic management in complex constrained airspace, rather than being limited to a single adversarial scenario.

3.1. Design of Cooperative Tasks for High-Speed Aircraft

Considering the multi-agent game confrontation scenario illustrated in Figure **1** of Chapter 2, multiple pursuers approaching from different directions compress the maneuvering space of the high-speed aircraft. To enhance the survivability of the high-speed aircraft in game confrontation, it is proposed to assign different tasks to the high-speed aircraft to achieve cooperative evasion against the pursuers.

Given that the high-speed aircraft in this paper possess axial acceleration capability, the axial acceleration of the high-speed aircraft can be designed to enable the aircraft to advance forward for decoying the enemy. The supporter is selected as the decoy; upon detecting the incoming pursuers, it accelerates forward to lure the pursuers into altering their trajectories. The leader acts as the evader. Taking advantage of the favorable evasion conditions created by the decoy, it makes full use of its own lateral maneuverability and acceleration capability to evade the incoming pursuers. Meanwhile, in consideration of the leader's subsequent strike mission, the leader is required to reach the designated flight path after completing the evasion maneuver. The specific process is illustrated in the following figure:

Assumption 2: For the aforementioned scenario, it is assumed that the defensive side deploys three pursuers to pursue the high-speed aircraft. Regarding the task allocation problem of the pursuers, considering that the defensive side is unaware of the task allocation of the high-speed aircraft, to ensure the successful pursuit of the high-speed aircraft, the allocation strategy of the pursuers is as follows: the edge pursuers always track the nearest high-speed aircraft in

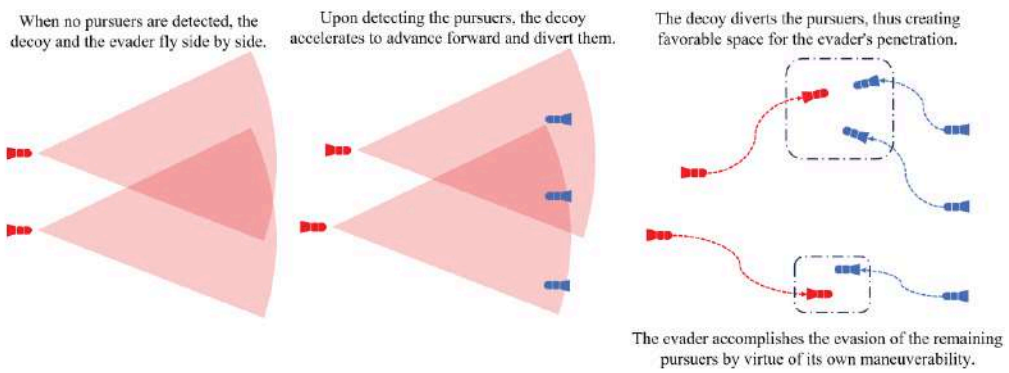

Figure 3: Flow Chart of Cooperative Maneuver Evasion for Decoys and Evaders.

the initial state, while the middle pursuer tracks the more threatening high-speed aircraft.

Remark 3: The faster the speed of the high-speed aircraft, the more difficult it is for the pursuers to intercept it. Therefore, when the decoy accelerates forward, the pursuers will consider the decoy to be faster and more threatening, thereby allocating more pursuers to intercept the decoy.

Remark 4: The realization of the cooperative task design among high-speed aircraft relies on multi-agent reinforcement learning. The task design for different high-speed aircraft is achieved by designing different observables and reward functions; for details, refer to Section 3.2.

3.2. Markov Decision Model

To solve the multi-agent pursuit-evasion game problem of high-speed aircraft via the multi-agent deep reinforcement learning method, it is first necessary to model this problem as a Markov Decision Process (MDP). An pursuit-evasion game environment is constructed based on the problem, where agents interact with the environment through states, actions, and rewards. A Markov Decision Process is typically defined by a five-tuple $<S,A,P,R,\gamma>$, where S denotes the finite state space, A denotes the finite action space, P denotes the state transition matrix, R denotes the reward function, and γ denotes the discount factor.

3.2.1. State Space

The state space is required to contain all information necessary for the decoy to lure pursuers and the evader to evade pursuit. Considering that the attack-defense confrontation process of high-speed aircraft evading pursuers involves online autonomous generation of maneuver commands under high-dynamic games, it is necessary to take into account the seeker constraints of the high-speed aircraft themselves. Moreover, from the perspective of algorithm convergence, normalization is performed on the state variables to ensure that the state magnitudes are basically consistent, thereby reducing the observation difficulty for agents.

$$\boldsymbol{s}_E=\begin{bmatrix} x_E,z_E,x_T,z_T,x_{M_x},z_{M_x},\psi_{VE}, \\ r_{EM_x},\lambda_{EM_x},\dot{\lambda}_{EM_x},V_E,V_{M_x} \end{bmatrix}^T, x\in\{1,2\} \tag{10}$$

$$\boldsymbol{s}_T=\begin{bmatrix} x_T,z_T,x_o,z_o,x_{M_y},z_{M_y},\psi_{VT}, \\ r_{TM_y},\lambda_{TM_y},\dot{\lambda}_{TM_y},V_T,V_{M_y} \end{bmatrix}^T, y\in\{2,3\} \tag{11}$$

Where: x_i,z_i denote the coordinates of the high-speed aircraft, where $i\in\{E,T\}$; x_M,z_M denote the coordinates of the pursuer; x_o,y_o denote the coordinates of the center of the target area; r_{iM} denote the relative distances between the high-speed aircraft and the pursuer, respectively; λ_{iM} denote the line-of-sight angles between the high-speed aircraft and the pursuer; $\dot{\lambda}_{iM}$ denote the line-of-sight angular rates between the high-speed aircraft and the pursuer, respectively; V_i denote the flight speeds of the high-speed aircraft; V_M denote the flight speed of the pursuer, respectively. During the learning process, normalization is performed on each observable, which facilitates the agents to perceive environmental information.

Perturbations are added to the observables to simulate the effect of sensor noise in real adversarial processes. The observables of the high-speed aircraft regarding its own state are accurate, whereas perturbation terms are introduced to those associated with the interceptors. The magnitude of the perturbations is set according to the relative distance; during the adversarial process, the perturbation terms decrease as the distance reduces, thus simulating the impact of distance variation on sensor observations.

3.2.2. Action Space

Considering that the modeling of the three-agent pursuit-evasion game for high-speed aircraft is conducted in a two-dimensional plane, the outputs of the actuators should be the lateral overload and axial velocity commands of the high-speed aircraft. In combination with the inherent performance constraints of the high-speed aircraft, it can be obtained that:

$$n_i\in[-n_{i\max},n_{i\max}] \tag{12}$$

$$V_i\in[V_{i\min},V_{i\max}] \tag{13}$$

Where: n_i denotes the normal overload of the high-speed aircraft; $n_{i\max}$ denotes the maximum available normal overload of the high-speed aircraft; V_i denotes the axial velocity of the high-speed aircraft; $V_{i\min}$ denotes the minimum axial velocity of the high-speed aircraft; $V_{i\max}$ denotes the maximum axial velocity of the high-speed aircraft.

Remark 4: In existing methods, the axial acceleration of high-speed aircraft is usually considered as the output. However, since there exist

constraints on the axial velocity of high-speed aircraft, invalid outputs of axial acceleration will occur when the velocity reaches its limit. Thus, this paper adopts the desired velocity of high-speed aircraft as the output.

3.2.3. Reward Function

The design of the reward function is a key focus and difficulty in reinforcement learning, as it directly determines whether the training can succeed and whether the final intelligent evasion strategy can be obtained. The reward function can be divided into the process reward function and the terminal reward function. Among them, the terminal reward directly determines the success of the training, while the process reward guides the agents to acquire key actions under different states through interaction with the environment, thus leading to successful training.

For the supporter, its main task is to lure the pursuers and create evasion space for the leader. In view of its task, the reward function is designed as follows:

$$R_E = r_{E1} + r_{E2} + r_{E3} + r_{E4} \quad (1)$$

$$\begin{cases} r_{E1} = \varepsilon_1 \ln(V_E / V_T) \\ r_{E2} = \varepsilon_2 \ln(|\lambda_{EM1} - \psi_{M1}| / |\lambda_{TM1} - \psi_{M1}|) \\ r_{E3} = \varepsilon_3 \ln(|\lambda_{EM2} - \psi_{M2}| / |\lambda_{TM2} - \psi_{M2}|) \\ r_{E4} = \begin{cases} 10, & \text{sucess} \\ -10, & \text{fail} \end{cases} \end{cases} \quad (15)$$

Where: r_{E1}, r_{E2}, r_{E3} belong to process rewards: r_{E1} corresponds to the supporter's task of advancing to lure the enemy: a reward is granted if the supporter's speed is higher than that of the leader; otherwise, a penalty is imposed. r_{E2}, r_{E3} correspond to whether pursuers $M1, M2$ are more biased toward the supporter during flight, respectively: a reward is granted if pursuers $M1, M2$ are more biased toward the supporter; otherwise, a penalty is imposed. r_{E4} belongs to the terminal reward: a reward is granted if the supporter successfully lures the enemy; otherwise, a penalty is imposed.

For the leader, its main task is to evade the remaining pursuers and reach the target area by utilizing its own maneuverability based on the evasion environment created by the supporter. In view of its task, the reward function is designed as follows:

$$R_T = R_{T_1} + R_{T_2} \quad (16)$$

Where: R_{T_1} denotes the reward function design for the leader's maneuver evasion phase task, and R_{T_1} denotes the reward function design for the leader's trajectory adjustment phase task.

In the maneuver evasion phase, the main task of the leader is to evade the pursuit of the pursuers by utilizing its own maneuverability. The reward function R_{T_1} for this phase is designed as follows:

$$R_{T_1} = r_{T_1 1} + r_{T_1 2} + r_{T_1 3} + r_{T_1 4} \quad (17)$$

$$\begin{cases} r_{T_1 1} = c_1 \ln(\dot{\lambda}_{TM3}) + c_2 \\ r_{T_1 2} = c_3 e^{-\frac{r_{TM3}}{100}} \\ r_{T_1 3} = c_4 \ln(r_{TM3}(t_f) - (\varsigma - 1)) \\ r_{T_1 4} = \begin{cases} 10, & \text{sucess} \\ -10, & \text{fail} \end{cases} \end{cases} \quad (18)$$

Where: $r_{T_1 1}, r_{T_1 2}$ belong to process rewards: $r_{T_1 1}$ Corresponding to the leader increasing the line-of-sight angular rate of the pursuer relative to itself, so as to disrupt the guidance effect of the pursuer's proportional navigation guidance rate. $r_{T_1 2}$ Corresponding to the process reward related to the distance between the leader and the pursuer during the phase. $r_{T_1 3}, r_{T_1 4}$ belong to terminal rewards: A reward is granted if the leader successfully evades, otherwise a penalty is imposed. Meanwhile, a reward associated with the final miss distance is introduced to guide the leader to maximize the final miss distance.

In the trajectory adjustment phase, the main task of the leader is to adjust its own flight trajectory via its maneuverability, so as to ensure it has the capability to complete the subsequent strike mission. The reward function R_{T_2} for this phase is designed as follows:

$$R_{T_2} = r_{T_2 1} + r_{T_2 2} \quad (19)$$

$$\begin{cases} r_{T_2 1} = c_5 e^{-\frac{|z_T - z_o|}{100}} + c_6 e^{-\frac{|\psi_T|}{\psi_o}} \\ r_{T_2 2} = c_7 \ln(|z_T(t_f) - d'|) + c_8 \ln(|\psi_T(t_f) - \psi_o|) \end{cases} \quad (20)$$

Where: $r_{T_2 1}$ belongs to the process reward: The first term of $r_{T_2 1}$ corresponds to the deviation degree of the

leader's position from the predetermined flight path during the phase: the closer the leader is to the predetermined flight path, the greater the reward. The second term of $r_{T_2 1}$ corresponds to the deviation degree of the leader's trajectory deflection angle during the phase, guiding the leader's trajectory deflection angle to tend to the predetermined state. $r_{T_2 2}$ belongs to the terminal reward: A reward is granted if the leader successfully reaches the target area and its flight state meets the constraints; otherwise, a penalty is imposed. Both the reward and penalty are related to the deviation degrees of the leader's position and trajectory deflection angle.

Furthermore, the values of different parameters in the reward function are designed for the specific scenario. By simply simulating the game-theoretic adversarial process, the impact of the value selection of different parameters on the reward function during the training process is observed, and the accurate values of the parameters are determined. It is required that the process reward be less than the terminal reward; moreover, when the pursuers are successfully evaded, the terminal reward should exhibit a significant step change, so as to guide the agents to learn the desired cooperative evasion guidance law.

3.3. CD-MASAC Algorithm (C: Curriculum Learning; D: Dynamic Learning Rate)

3.3.1. MASAC Algorithm

To address the challenging problem of cooperative game confrontation for high-speed aircraft, which features high dynamics, strong antagonism, and a complex decision-making space, this study adopts the Multi-Agent Soft Actor-Critic (MASAC) algorithm as the solution. MASAC is an extension of the Soft Actor-Critic (SAC) algorithm in the field of multi-agent systems. Its core advantage lies in encouraging agents to conduct more thorough exploration by maximizing the entropy regularization term, which is crucial for discovering excellent and even optimal solutions in the vast maneuvering strategy space of high-speed aircraft. The SAC algorithm itself is designed to handle problems with continuous action spaces, which is naturally compatible with the overload command control of missiles. By treating other agents as part of the environment, MASAC can effectively tackle the inherent instability of multi-agent systems.

3.3.1.1. Maximum Entropy Objective

Unlike traditional reinforcement learning, which only maximizes the cumulative reward, the objective function of MASAC explicitly incorporates the policy entropy term. This prompts agents to maintain the randomness and diversity of actions as much as possible while pursuing high rewards. Its objective function is defined as follows:

$$J(\pi) = E_{\tau\sim\pi}[\sum_{t=0}^{T}\gamma^t(R(s_t,a_t)+\alpha H(\pi_i(\cdot\,|\,o_{i,t})))] \tag{21}$$

Where: $R(s_t,a_t)$ denotes the joint reward; γ denotes the discount reward; α denotes the temperature coefficient; $H(\pi_i(\cdot\,|\,o_{i,t}))$ represents the entropy of the policy of agent i under its local observation $o_{i,t}$. The MASAC algorithm drives more thorough exploration by maximizing the entropy, thereby avoiding premature convergence to suboptimal policies and enhancing the robustness of the policy.

3.3.1.2. Soft Policy Iteration

The training process of MASAC follows the framework of soft policy iteration, which mainly consists of two alternating steps: soft policy evaluation and soft policy improvement. Through the alternating iteration of these two steps, MASAC can efficiently learn a multi-agent policy that can not only complete cooperative tasks but also possess high exploration capability and robustness.

Soft Policy Evaluation: A centralized critic network is adopted to estimate the soft Q-value of the current policy. The critic is updated by minimizing the soft Bellman residual, and its loss function is defined as follows:

$$L(\theta_Q) = E[(Q(s,a;\theta_Q)-y)^2] \tag{22}$$

Where the calculation method for the target value y is as follows:

$$y = r+\gamma(Q^{target}(s',a';\theta_{Q^{target}})-\alpha\log\pi(a_i'\,|\,o_i')) \tag{23}$$

Since the critic can access the global state s and joint action a, the evaluation is more accurate and stable.

Soft Policy Improvement: Each decentralized actor updates its policy based on the soft Q-values provided by the critic. Its objective is to maximize the expected return and entropy of the policy. The loss function of actor i is defined as follows:

$$L(\theta_{\pi_i}) = E_{o_i,a_i}[\alpha\log\pi_i(a_i\,|\,o_i;\theta_{\pi_i})-Q(s,a_1,a_2,\cdots,a_N)] \tag{24}$$

By minimizing this loss function, the policy π_i is updated to select actions that yield higher Q-values and greater entropy.

3.3.1.3. Training Paradigm

The MASAC algorithm adopts the training paradigm of Centralized Training with Decentralized Execution (CTDE). CTDE allows the use of global information (e.g., the joint states and joint actions of all agents) to train one or more centralized critic networks during the training phase. This centralized critic can achieve better credit assignment, understand the impact of individual behaviors on the overall task objectives, and thus facilitate the learning of collaborative behaviors among agents. In the execution phase, each high-speed aircraft agent makes decisions solely based on its local observation information, which conforms to the constraints on information acquisition in real battlefield environments.

3.3.2. Improved MASAC Algorithm Combined with Curriculum Learning and Dynamic Learning Rate Adjustment (CD-MASAC)

Given that in the complex scenario of multi-agent cooperative game confrontation with target constraints, the vanilla MASAC algorithm faces high learning difficulty and a difficult-to-balance trade-off between convergence and learning speed, this paper proposes to integrate curriculum learning and dynamic learning rate adjustment to address the above issues.

3.3.2.1. Curriculum Learning Setup based on Task Difficulty Classification

Aiming at the complex scenario of multi-agent cooperative game confrontation with target constraints, this paper combines curriculum learning with deep reinforcement learning to design an intelligent evasion strategy for high-speed aircraft. Based on the idea of curriculum learning, according to the differences in pursuers' own capabilities and target constraints, the multi-agent cooperative game confrontation problem with target constraints is transformed into a series of sub-curricula with varying evasion difficulty and stringency of target constraints, which effectively reduces the training difficulty.

As shown in Figure **4**, three sub-curricula are introduced to guide the agents to gradually master the problem-solving methods. By ignoring target constraints and setting the pursuers' own capabilities from weak to strong, the agents are guided to focus on learning maneuver evasion strategies against pursuers; subsequently, weakened target constraints and complete target constraints are gradually introduced to guide the agents to gradually master the precision guidance capability of high-speed aircraft.

The introduction of curriculum learning enables the agent to achieve a smooth transition from simple tasks to difficult ones. This not only improves the success rate but also reduces the computational overhead and power cost required for training by shortening the number of iterations for convergence, which aligns with the development trend of green aviation technologies.

3.3.2.2. Dynamic Learning Rate Adjustment

In reinforcement learning, the magnitude of the learning rate affects the learning speed and convergence speed during the learning process. When the learning rate is large, the agents will conduct more aggressive exploratory learning, making it easy to explore problem-solving methods, but there is a

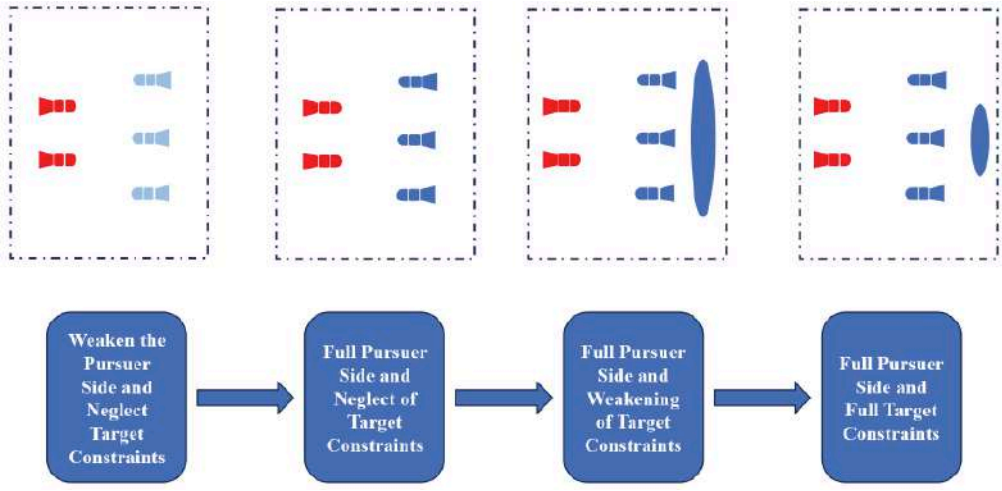

Figure 4: Difficulty Settings of Each Curriculum in Curriculum Learning.

problem of difficulty in achieving stable convergence; the opposite is true when the learning rate is small.

To balance the learning speed and convergence of the agents, this paper adopts a dynamic learning rate mechanism. By setting dynamically changing learning rates in the initial exploration phase and the later convergence phase, the agents can conduct rapid exploratory learning in the early stage and quickly find potential solutions to the problem; subsequently, by reducing the learning rate, the agents can stably converge to better results and avoid oscillation.

3.3.3. Pseudocode of the CD-MASAC Algorithm

Input: Maximum number of training episodes M, maximum number of steps per episode T. Set of curriculum difficulty levels $C = \{L_1, L_2, L_3, L_f\}$. Initial exploration learning rate a_h, convergence learning rate a_l, switching threshold E_s. Agent set N (including supporter and leader). Batch size B, soft update coefficient τ, discount factor γ.

Initialization: Initialize the Actor network π_ϕ and Critic network Q, as well as the target Critic network Q_{target}, for each agent i. Initialize the experience replay buffer D.

The pseudocode flow of CD-MASAC is presented as follows:

1. For episode $e = 1$ to M do:
2. // Improvement 1: Curriculum learning configuration
3. Select the current environmental difficulty level L from set C based on the current success rate of episode e (where L defines the pursuer's capability and the stringency of target constraints)
4. // Improvement 2: Dynamic learning rate adjustment
5. If $e < E_s$, set the learning rate to a_h (exploration phase)
6. Else, set the learning rate to a_l (convergence phase)
7. Initialize the environmental state
8. For step $t = 0$ to T do:
9. For each agent, sample an action according to the current policy
10. Note: Actions consist of lateral overload and desired axial velocity commands
11. Execute the joint actions and obtain the new state and joint reward
12. Note: The reward function is calculated separately for different roles (supporter/leader)
13. Store the tuple in the experience replay buffer D
14. If the number of samples in D meets the requirement:
15. Randomly sample B samples from D
16. For each agent do:
17. Calculate the target Q value
18. Update the Critic network parameters: minimize the loss
19. Update the Actor network parameters: maximize the entropy and expected return using the current dynamic learning rate
20. Automatically adjust the entropy coefficient
21. Perform soft update on the target network
22. End If
23. Update the state
24. If the termination condition is met (evasion success or failure), break the loop
25. End For
26. End For

Remark 5: Compared with existing research on pursuit-evasion games based on traditional methods and intelligent methods, the innovations of this paper are as follows:

1. In the multi-agent game confrontation problem, collaborative task allocation among high-speed aircraft is introduced to achieve maneuver evasion against pursuers and subsequent tasks.
2. In axial motion, the desired axial velocity is considered as the output, and the axial velocity control loop is used to control the axial velocity of the high-speed aircraft to converge to the desired axial velocity, avoiding the problem of

invalid output of axial acceleration when the velocity reaches the limit.

3. Curriculum learning and dynamic learning rate adjustment are simultaneously introduced to optimize the learning process of the MASAC algorithm. By classifying the task difficulty of the multi-agent game confrontation problem with target constraints, different sub-curricula are set to guide the agents to gradually master the problem-solving methods, and dynamic learning rate adjustment is introduced to balance the learning speed and convergence speed.

4. SIMULATION VERIFICATION

This chapter conducts simulation verification on the effectiveness of the leader's intelligent maneuvering strategy based on the CD-MASAC algorithm in the scenario of evading pursuers under target constraints. Firstly, it presents the relevant information used in the simulation verification; secondly, it verifies the effectiveness of the intelligent maneuvering strategy based on the CD-MASAC algorithm designed in this paper in this scenario through numerical simulation; finally, it further clarifies the generalization of the algorithm through Monte Carlo simulation analysis.

In the numerical simulation of this section, the neural network design is as follows:

Each agent adopts the same network training parameters, and the corresponding settings of network training parameters are shown in the table below.

The initial situation information of the high-speed aircraft and pursuers is shown as follows:

Table 1: Neural Network Structure

Network Component	Input Layer	Hidden Layer 1	Hidden Layer 2	Output Layer	Hidden Layer Activation Function	Output Layer Activation Function
Supporter-Actor	State Vector S_E	256 Neurons	256 Neurons	Action Vector	ReLU	Tanh
Supporter -Critic	Joint State-Action	256 Neurons	256 Neurons	Q-Value	ReLU	Linear
Leader-Actor	State Vector S_T	256 Neurons	256 Neurons	Action Vector	ReLU	Tanh
Leader-Critic	Joint State-Action	256 Neurons	256 Neurons	Q-Value	ReLU	Linear

Table 2: Network Training Parameters

Parameter	Value
Experience Replay Buffer Capacity	50000
Mini-Batch Sampling Size	256
Policy Network Exploration Learning Rate	$3x10^{-4}$
Policy Network Convergence Learning Rate	$1x10^{-6}$
Critic Network Exploration Learning Rate	$5x10^{-4}$
Critic Network Convergence Learning Rate	$5x10^{-6}$
Initial Exploration Rate	0.3
Decay Coefficient	$1x10^{-5}$
Discount Factor	0.9
Inertia Coefficient	0.99
Soft Update Frequency	0.001
Sampling Time	0.1
Regression Network Learning Rate	0.01
Regression Network Target Minimum Error	0.001
Regression Network Minimum Performance Gradient	1*10-6

Table 3: Initial Situation Information of Game Confrontation

Parameter Name	Parameter Value	Parameter Name	Parameter Value
Leader Initial Position	$(0\text{km}, 25\text{km}, -0.15\text{km})$	Central Pursuer Initial Position	$(30\text{km}, 25\text{km}, 0\text{km})$
Supporter Initial Position	$(0\text{km}, 25\text{km}, 0.15\text{km})$	Edge Pursuer Initial Position	$(30\text{km}, 25\text{km}, \pm 0.3\text{km})$
High-Speed Aircraft Speed	5.5Ma~6.5Ma	Pursuer Speed	3.5Ma
High-Speed Aircraft Initial Trajectory Deflection Angle	0rad	Pursuer Initial Trajectory Deflection Angle	π rad
First-Order Time Constant of High-Speed Aircraft Autopilot	0.5	First-Order Time Constant of Pursuer Autopilot	0.5
Maximum Available Overload of High-Speed Aircraft	2	Maximum Available Overload of Pursuer	6
Center Position of Target Constraint Region	$(40\text{km}, 25\text{km}, 0\text{km})$	Radius of Target Constraint Region	40m
Trajectory Deflection Angle Constraint Magnitude	1°	Proportional Navigation Guidance Coefficient	4

4.1. CD-MASAC Algorithm Training

In this section, based on the above-mentioned attack-defense confrontation situation information, agent training is conducted respectively with the vanilla MASAC algorithm, CL-MASAC algorithm, and CD-MASAC algorithm. The simulation training results are shown as follows:

As shown in Figure **5**, in the first 500 episodes of the CD-MASAC agent training process, the agent completed the curriculum learning of Difficulty 1, 2, and 3 in sequence, and then entered the scenario with the final difficulty for reinforcement learning. During episodes 500 to 5000, the agent explored effective problem-solving methods by maintaining a relatively high learning rate; around episode 5000, the learning rates of the action network and the critic network were reduced to accelerate the stable convergence of the agent.

Figure **6** depicts the training process of the CL-MASAC algorithm. A comparison with the training process diagram of the improved CD-MASAC algorithm shows that although the average reward maintained a relatively high value for a period, the excessively high learning rate led to a collapse phenomenon in the later stage, making it impossible to stably converge to the desired results.

Comprehensive comparison shows that the introduction of dynamic learning rate adjustment can further improve the convergence speed of the algorithm. Given the excessively high training difficulty, the MASAC method is challenging to train successfully without introducing curriculum learning. Therefore, this

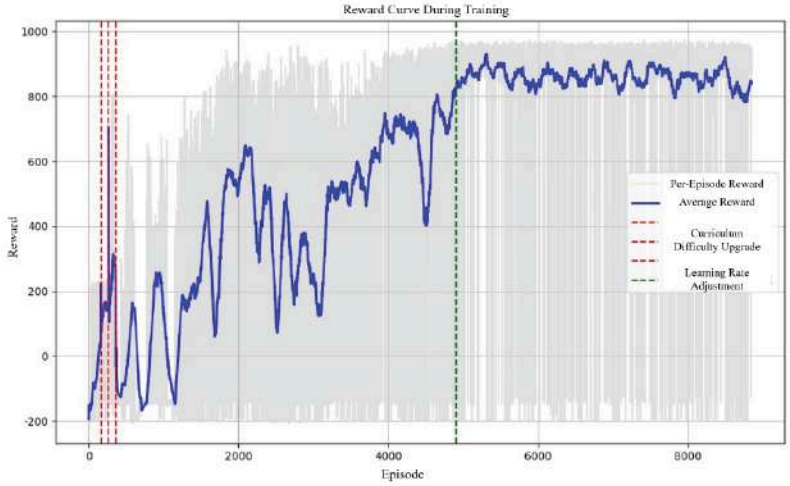

Figure 5: CD-MASAC Agent Training Process Diagram.

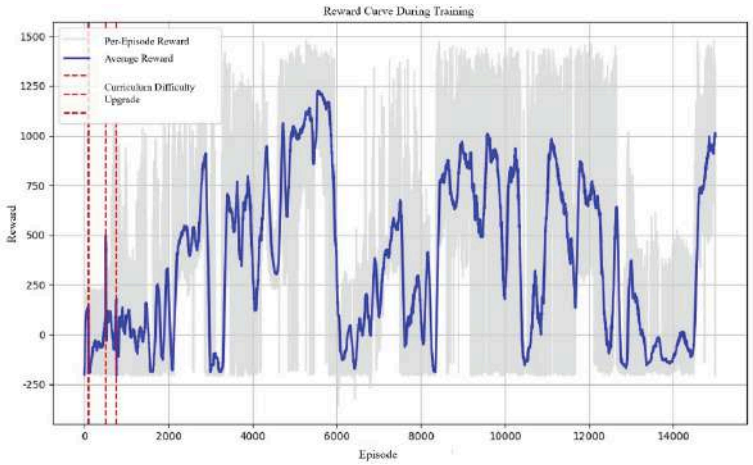

Figure 6: CL-MASAC Agent Training Process Diagram.

method is not included in the comparative analysis of this section.

4.2. Monte Carlo Simulation Analysis of the CD-MASAC Intelligent Maneuvering Strategy

To evaluate the effectiveness and robustness of the proposed integrated strategy of cooperative evasion and terminal guidance based on the CD-MASAC algorithm, this study designs and implements a series of Monte Carlo simulation experiments. By applying random perturbations to the initial states of the pursuers, the performance of the trained agent strategy in uncertain environments is systematically verified.

The Monte Carlo simulation is divided into three independent test groups, where perturbations are applied to one of the three pursuers respectively. In each test group, random perturbations are simultaneously imposed on the initial X-axis position and initial trajectory deflection angle of the designated pursuer, with ranges of ±1000 meters and ±5 degrees respectively. Each test group independently conducts 500 simulations, resulting in a total of 1500 simulations. To evaluate the statistical reliability of the simulation results, this paper calculates the confidence intervals for each performance metric at a 95% confidence level. The main evaluation metrics are as follows:

- Maneuver Evasion Success Rate in Phase 1: The probability that the leader successfully evades the pursuers and enters Phase 2.
- Terminal Hit Accuracy in Phase 2: The Y-axis position deviation and trajectory deflection angle deviation when the leader reaches the target area after successfully evading the pursuit.

4.2.1. Phase 1: Analysis of the Intelligent Evasion Strategy

As shown in Figure **7**, this figure illustrates the maneuver evasion performance of the agent in Phase 1 under joint perturbations applied to the initial position and deflection angle of a single pursuer. In the figure, green dots represent successful evasion of pursuit (with the minimum miss distance greater than 5 meters), while red dots represent failure.

It can be concluded from the three figures and the table that the proposed intelligent evasion strategy achieves an extremely high evasion success rate: across all test groups, regardless of which pursuer is subjected to disturbances, the agent's maneuvering evasion success rate consistently exhibits an extremely high level, reaching 98.4%, 93.3%, and 92.9% respectively at the 95% confidence level. This demonstrates that the trained cooperative strategy is not an overfitted solution tailored to deterministic scenarios, but rather a robust strategy with generalization capability. Furthermore, it indicates that the intelligent evasion strategy possesses a certain degree of adaptability to disturbances.

4.2.2. Phase 2: Accuracy Analysis of the Guidance Strategy

Figures **8** and **9** further analyze the terminal hit accuracy of the leader in Phase 2 after the completion of maneuver evasion against pursuers in Phase 1.

It can be analyzed that the intelligent maneuvering strategy achieves a high terminal hit rate: as shown in Figure **8** and Table **5**, among all cases of successful evasion from pursuers, the proportion of leaders that ultimately meet the target constraints is extremely high, reaching 99.1%, 100.0%, and 100.0% respectively at

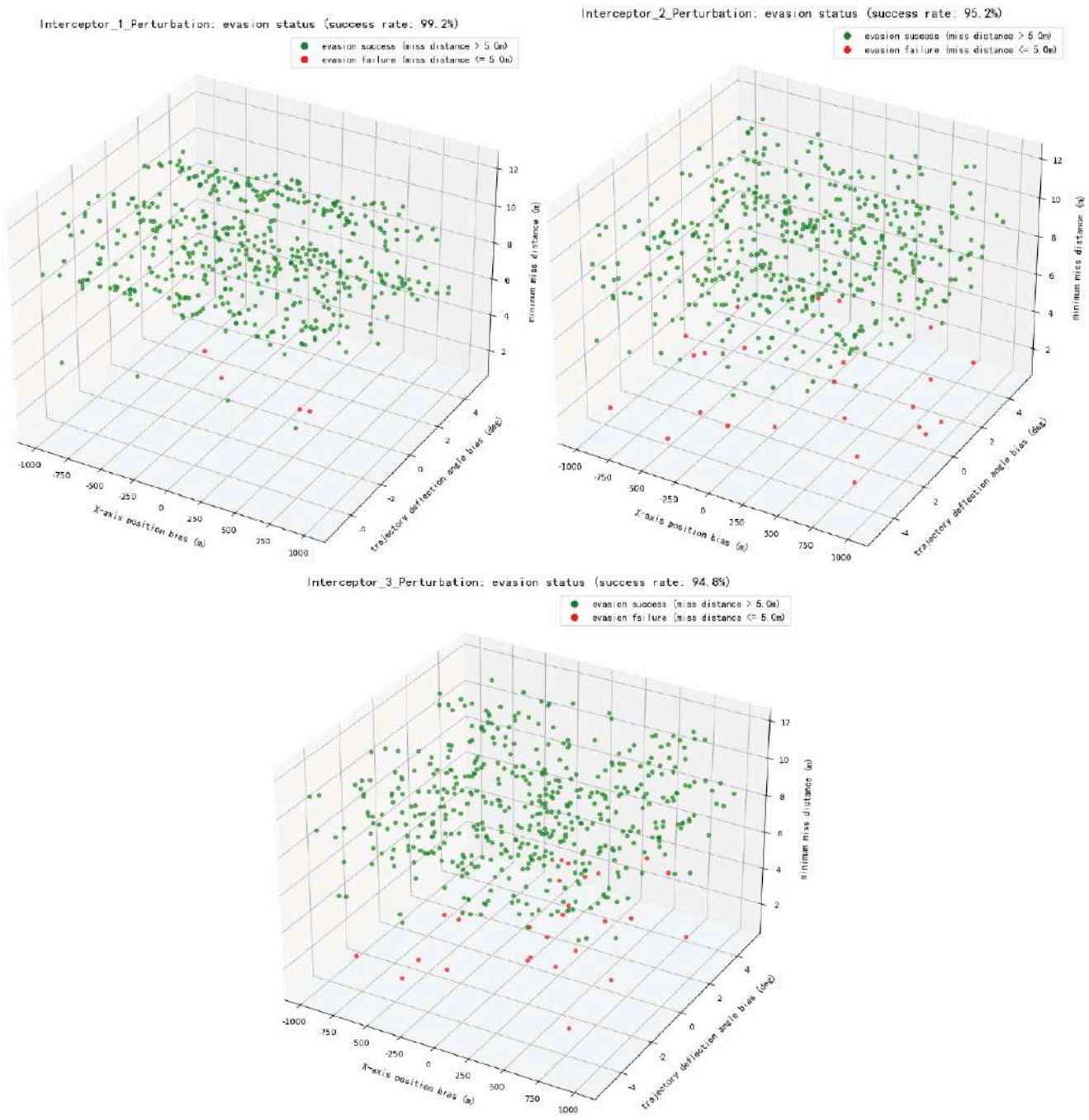

Figure 7: 3D Scatter Plot of Evasion Performance in the First Phase Under Different Pursuer Disturbances.

Table 1: Statistical Results of Monte Carlo Simulation (95% Confidence Interval)

Test Group	Sample Size	Evasion Success Rate	95% Confidence Interval
1	500	99.2%	[98.4%,100%]
2	500	95.2%	[93.3%,97.1%]
3	500	94.8%	[92.9%,96.7%]

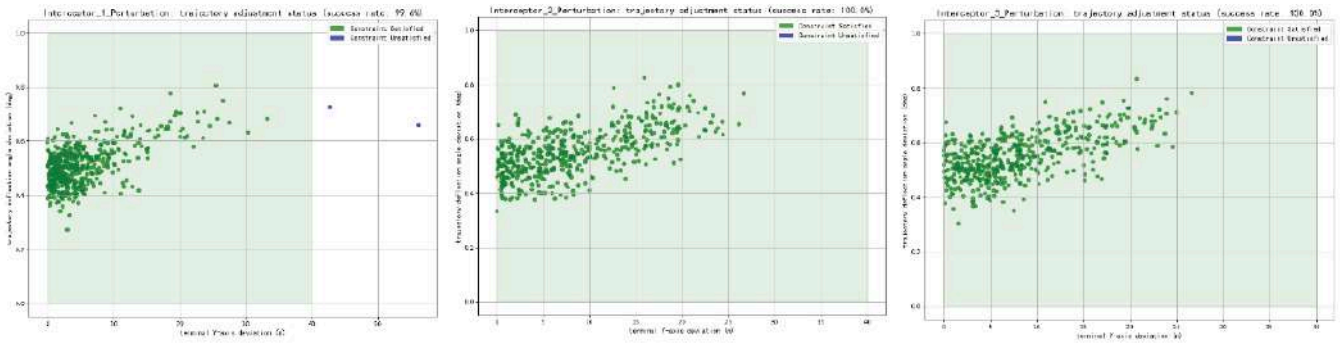

Figure 8: 2D Scatter Plot of Terminal Hit Accuracy After Successful Evasion Under Different Disturbances.

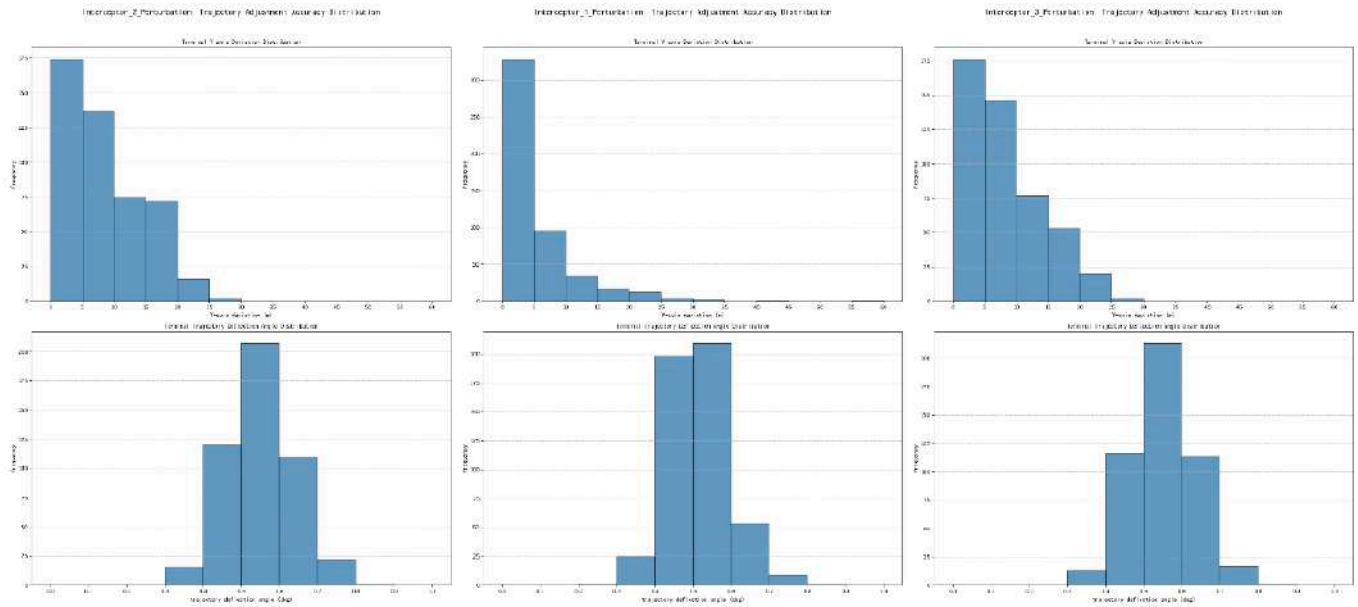

Figure 9: Histogram of Terminal Hit Error Distribution After Successful Evasion of Pursuit Under Different Disturbances.

Table 2 Statistical Analysis of Terminal Hit Rate (95% Confidence Level)

Test Group	Sample Size	Terminal Hit Rate	95% Confidence Interval
1	500	99.6%	[99.1%,100%]
2	500	100%	[100%,100%]
3	500	100%	[100%,100%]

the 95% confidence level. This indicates that the agents have not only learned how to evade pursuers but also can ensure proceeding to the second phase in a posture conducive to executing the terminal strike mission. As can be seen from the bar chart in Figure **9**, among all target hit cases, the Z-axis deviation of the vast majority of samples is concentrated within 5 meters, and the ballistic deflection angle deviation is concentrated within 0.6 degrees. This far exceeds the constraint boundaries of 40 meters and 1 degree set by the mission, demonstrating the high control accuracy of the proposed strategy.

In summary, the results of 500 Monte Carlo simulation experiments fully verify the robustness and high precision of the integrated cooperative evasion and terminal guidance strategy based on the CD-MASAC algorithm. This agent strategy can not only maintain a high evasion success rate in complex adversarial environments with significant uncertainties, but also complete the terminal strike mission in Phase

2 with high precision after fulfilling the task in Phase 1, demonstrating the integrity and continuity of the method. The evasion maneuver of the agent in Phase 1 is not achieved at the expense of the strike accuracy in Phase 2, but rather an organic unity of the two.

4.3. Simulation Analysis of the Evasion Process for the Improved CD-MASAC Intelligent Maneuvering Strategy

To verify the effectiveness of the proposed integrated strategy based on the CD-MASAC algorithm, this section conducts an in-depth analysis of the single-point simulation results under the typical scenario of head-on pursuit. By analyzing the flight trajectory, key performance indicators, and control commands, this section aims to reveal the strategic characteristics and intelligence level of the policies generated by the agent when completing the complex two-phase mission. The simulation results are shown as follows:

Figure **10** shows the complete flight trajectory of this single-point simulation. From a macro perspective, the leader and supporter executed effective large-scale cooperative maneuvers during the encounter with the three pursuers. After successfully breaking through the defensive zone formed by the pursuers, the leader's trajectory smoothly transitions to the target area located 40 km away and ultimately satisfies the terminal constraints, which verifies the integrity and continuity of the proposed method in accomplishing the two-phase mission. As can be seen from the enlarged view of Figure **7**, the leader finally reaches the target area with a miss distance of -0.3 meters in the Z-axis

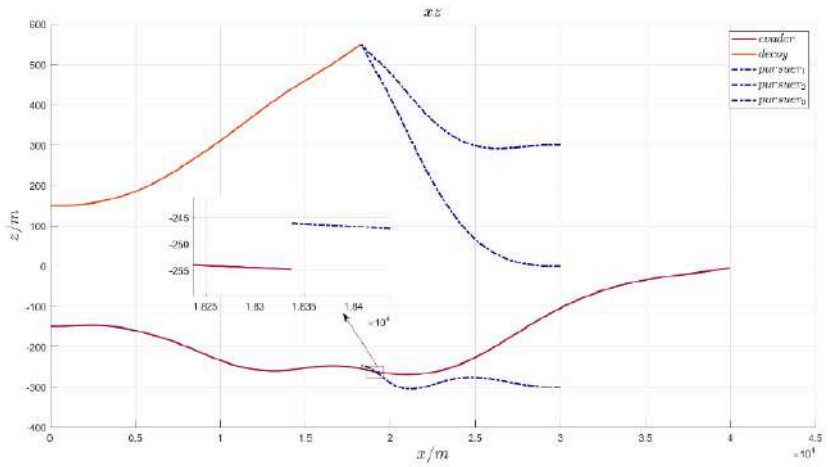

Figure 10: Game Confrontation Trajectory Diagram.

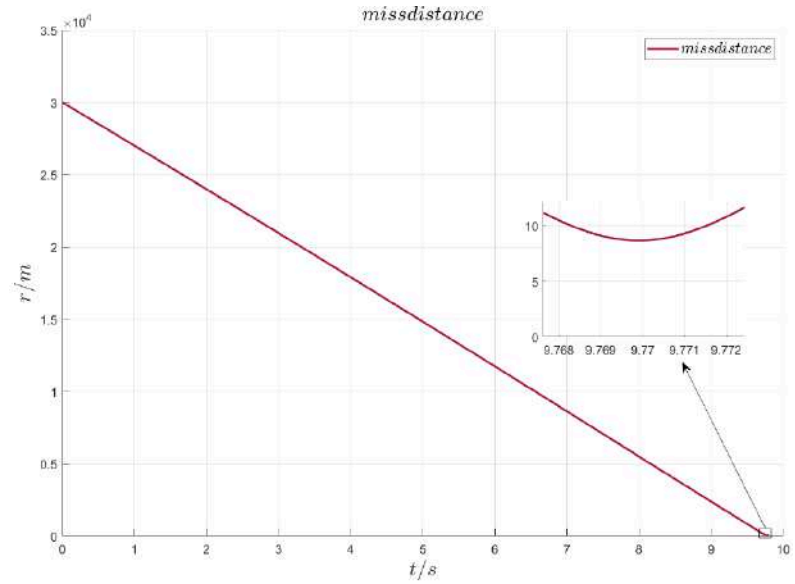

Figure 11: Miss Distance Variation Diagram.

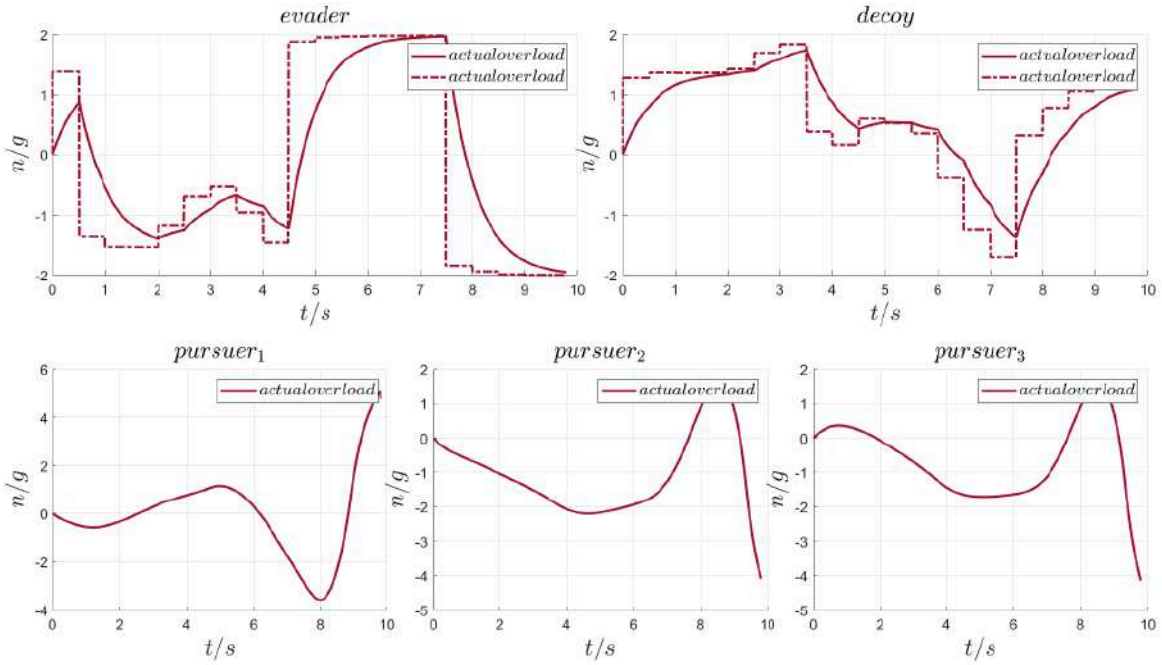

Figure 12: Overload Variation Diagram.

and a trajectory deflection angle deviation of -0.5 degrees. Both indicators are far lower than the preset constraint boundaries of 40 meters and 1 degree, demonstrating the high terminal control accuracy of the strategy.

The core objective of Phase 1 is to achieve successful evasion of pursuers. Figure **11** details the distance variation between the leader and its designated target pursuer. The enlarged view indicates that at the closest approach point around 9.8 seconds, the minimum miss distance between the two sides reaches 11.40 meters. This value is significantly higher than the 5-meter threshold for successful evasion, which confirms the full effectiveness of the evasion strategy generated by the agent.

Figure **12** reveals the control strategies of all parties during the game. By observing the acceleration overload curves of the leader and supporter, it can be seen that both performed large-amplitude maneuvers in opposite directions during the initial phase (approximately 0–4 seconds). This cooperative strategy greatly disrupts the pursuers' defensive formation, creating favorable conditions for subsequent successful evasion. The overload commands of the leader exhibit typical Bang-Bang control characteristics in multiple time periods. This indicates that through learning, the agent independently discovered the evasion strategy, i.e., performing evasion with maximum available maneuverability, thereby efficiently depleting the pursuers' overload capacity and increasing the miss distance.

Figure **13** illustrates the axial velocity control strategy of the high-speed aircraft throughout the mission, which is the key to achieving high-precision strikes with the proposed method. During the maneuver evasion phase, both the leader and supporter adopted a velocity control strategy of accelerating first and then decelerating. When engaging in high-intensity maneuvering confrontations with pursuers, the agent issued acceleration commands to maintain a high Mach number, ensuring its own velocity advantage. After completing the maneuver evasion, the agent issued continuous deceleration commands to the leader, reducing its velocity smoothly from approximately Mach 6.5 to Mach 5.75. Through deceleration, the agent gained a longer adjustment time, implementing a tactic of trading velocity for higher precision. As shown in Figure **13**, after completing safe obstacle avoidance, the system stabilizes the trajectory through intelligent deceleration. Such refined axial velocity control avoids ineffective overload output caused by excessive maneuvering. By reducing unnecessary control cost, this strategy not only extends the service life of the actuators but also decreases energy consumption during the flight, thus achieving more efficient trajectory management.

In summary, the single-point simulation results strongly verify the effectiveness of the proposed CD-

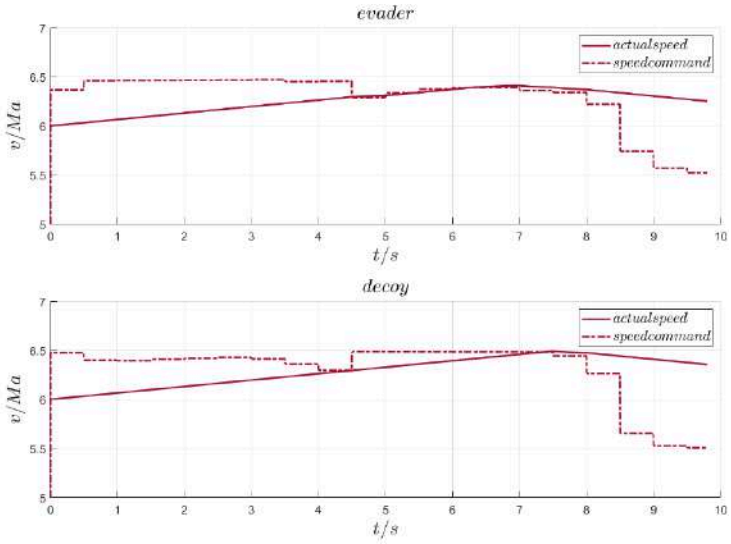

Figure 13: Velocity Variation Diagram.

MASAC method. The agent has not only learned to complete the highly challenging Phase 1 evasion mission using the Bang-Bang maneuver strategy, but also achieved deceleration and stabilization through intelligent velocity management, ultimately realizing a seamless transition to accomplish the Phase 2 terminal guidance mission with high precision. The entire process demonstrates that the proposed intelligent guidance law exhibits remarkable performance in solving this complex multi-phase, multi-constraint, multi-agent game problem.

5. CONCLUSION

For the multi-agent cooperative guidance problem in complex dynamic airspace, this paper proposes an intelligent autonomous control framework based on role-cooperative allocation and the CD-MASAC algorithm. By introducing the supporter–mission executor cooperative mechanism, the system can maintain excellent safe maneuvering performance even under significant environmental disturbances. The CD-MASAC algorithm not only ensures obstacle avoidance safety, but also guarantees high-precision satisfaction of terminal target constraints through fine axial management, achieving the organic unity of the safety phase and the mission phase. Curriculum learning and the adaptive learning rate mechanism significantly reduce the algorithm's training resource consumption, while the real-time velocity regulation strategy effectively decreases energy loss caused by ineffective maneuvers, demonstrating its application value in the field of sustainable aviation technology. The proposed CD-MASAC framework is not only applicable to high-speed maneuvering scenarios, but its core logic can be directly extended to intelligent aviation and sustainable flight fields, including UAV swarm cooperative collision avoidance, emergency conflict resolution, and safe terminal guidance in constrained spaces. This study provides important methodological support for future highly autonomous, robust, and energy-efficiency-aware intelligent aircraft systems.

The results of simulation verification fully demonstrate that the introduction of curriculum learning and dynamic learning rate adjustment reduces the learning difficulty. Meanwhile, the intelligent maneuvering evasion strategy obtained by integrating task allocation with reinforcement learning via the reward function can give full play to the cooperative evasion capability of supporters for attracting enemies and leaders for breaking through the interception, with the successful evasion rate of leaders reaching 92.9%. In addition, after considering the axial variable-speed capability of high-speed aircraft, the agents assist the maneuvering evasion and precise guidance tasks through real-time speed adjustment, and the achievement rate of trajectory constraints reaches 99.1%. The proposed method has high effectiveness and robustness. On the basis of ensuring that high-speed aircraft successfully evade pursuers, it also takes into account the subsequent target constraints, thus realizing effective evasion in the complex scenario of multi-agent game confrontation with target constraints incorporated.

In future research, further enrichment of the cooperative tactics among high-speed aircraft can be considered to enhance their cooperative evasion

capability. In addition, the six-degree-of-freedom model can be introduced for agent training and simulation, so as to be more in line with specific engineering practices.

REFERENCES

[1] Wang M L. Overview of ballistic missile penetration countermeasures[J]. Aerodynamic Missile Journal 2012; (10): 45-51.

[2] Mu Z, Jie P, Zhou Z, *et al*. A survey of the pursuit–evasion problem in swarm intelligence[J]. Frontiers of Information Technology & Electronic Engineering 2023; 1093-1116. https://doi.org/10.1631/FITEE.2200590

[3] Liang H, Li Z, Wu J, *et al*. Optimal guidance laws for a hypersonic multiplayer pursuit-evasion game based on a differential game strategy[J]. Aerospace 2022; 9(2): 97. https://doi.org/10.3390/aerospace9020097

[4] Hu G, Guo J, Guo Z, *et al*. ADP-Based intelligent tracking algorithm for reentry vehicles subjected to model and state uncertainties[J]. IEEE Transactions on Industrial Informatics 2023; 19(4): 6047-6055. https://doi.org/10.1109/TII.2022.3171327

[5] Yuan Y, Zhang P, Li X. Synchronous fault-tolerant near-optimal control for discrete-time nonlinear PE game[J]. IEEE Transactions on Neural Networks and Learning Systems 2021; 32(10): 4432-4444. https://doi.org/10.1109/TNNLS.2020.3017762

[6] Wang Y, Ning G, Wang X, *et al*. Maneuver penetration strategy of near space vehicle based on differential game[J]. Acta Aeronautica et Astronautica Sinica 2020; 41(S2): 724276.

[7] Cheng T, Zhou H, Dong X, *et al*. Differential game guidance law design for integration of penetration and strike of multiple flight vehicles[J]. Journal of Beijing University of Aeronautics and Astronautics 2022; 48(5): 898-909.

[8] Zhang K, Zhang K, Tan M, *et al*. Strategies for spacecraft pursuit-evasion games in asymmetric non-zero-sum conditions[J]. Journal of Astronautics 2024; 45(12): 1886-1896.

[9] Qian C, Zhou H, Wang Y, *et al*. Cooperative penetration strategy for multi - UAV swarms based on distributed game - theoretic optimization[J]. Journal of Intelligent and Robotic Systems 2024; 95(3-4): 553-568.

[10] Liu Y, Chen X, Wang Z, *et al*. Cooperative game - based penetration strategy for multi - vehicle systems in adversarial environments[J]. Journal of Systems Engineering and Electronics 2023; 34(5): 1031-1042.

[11] Dong X, Zhang H, Zhong M. Adaptive optimal control via Q-learning for multi-agent pursuit-evasion games[J]. IEEE Transactions on Circuits and Systems II: Express Briefs 2024; 3056-3060. https://doi.org/10.1109/TCSII.2024.3354120

[12] Yan T, Cai Y. General evasion guidance for air-breathing hypersonic vehicles with game theory and specified miss distance[C]. Proceedings of the 9th IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems 2019; 1125-1130. https://doi.org/10.1109/CYBER46603.2019.9066556

[13] Yu X, Wang X, Lin H. Optimal penetration guidance law with controllable missile escape distance[J]. Journal of Astronautics 2023; 44(07): 1053-1063.

[14] Feng L, Lu W, Wang F, *et al*. Optimal penetration guidance law for high-speed vehicles against an interceptor with modified proportional navigation guidance[J]. Symmetry 2023; 15(7). https://doi.org/10.3390/sym15071337

[15] Gao S, Lin D, Zheng D, *et al*. Intelligent maneuvering penetration guidance strategies for aerial vehicles considering interceptor detection capability limitations[J]. Acta Aeronautica et Astronautica Sinica 2025; 46(10): 331304.

[16] Guo Y, Jiang Z, Huang H, *et al*. Intelligent maneuver strategy for a hypersonic pursuit-evasion game based on deep reinforcement learning[J]. Aerospace, 2023. https://doi.org/10.3390/aerospace10090783

[17] Zhao S, Zhu J, Bao W. Multi-Constraints guidance and maneuvering penetration strategy via meta deep reinforcement learning[J]. Aerospace Science and Technology 2023; 137: 108531. https://doi.org/10.20944/preprints202308.1512.v1

[18] Li Z, Liu H, Wu Q, *et al*. Reinforcement learning-based intelligent penetration strategy for missiles in network-centric warfare [J]. Acta Armamentarii 2023; 44(10): 2456-2467.

[19] He X, Chen J, Guo H, *et al*. Attack and defense game of high-speed aircraft based on deep reinforcement learning. Aerospace Control 2022; 40(04): 76-83.

[20] Wang X, Gu K. A penetration strategy combining deep reinforcement learning and imitation learning[J]. Journal of Astronautics 2023; 44(06): 914-925.

[21] Ni W, Wang Y, Xu C, *et al*. Cooperative game guidance method for hypersonic vehicles based on reinforcement learning[J]. Acta Aeronautica et Astronautica Sinica 2023; 44(S2): 729400.

[22] Yan T, Liu C, Gao, M, *et al*. A deep reinforcement learning-based intelligent maneuvering strategy for the high-speed UAV pursuit-evasion game[J]. Drones 2024; 8(7): 309. https://doi.org/10.3390/drones8070309

[23] Su Z, Zheng S, *et al*. Evade unknown pursuer via pursuit strategy identification and model reference policy adaptation (MRPA) algorithm. Drones 8(11): 655. https://doi.org/10.3390/drones8110655

[24] Li B, Zhang H, He P, *et al*. Hierarchical maneuver decision method based on PG-option for UAV pursuit-evasion game. Drones, 7(7): 449. https://doi.org/10.3390/drones7070449

[25] Zhong Z, Dong Z, Duan X, *et al*. Collaboration strategies for two heterogeneous pursuers in a pursuit-evasion game using deep reinforcement learning[C], 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Abu Dhabi, United Arab Emirates 2024; 11962-11968. https://doi.org/10.1109/IROS58592.2024.10802839

[26] Zhang Z, Zong Q, Liu D, *et al*. A pursuit-evasion game on a real-city virtual simulation platform based on multi-agent reinforcement learning[C]. 2023 42nd Chinese Control Conference 2023; 6018-6023. https://doi.org/10.23919/CCC58697.2023.10240527

[27] Zhang K, Xu Y, Liu C, *et al*. Intelligent air combat maneuvering decision method of multi-UAV system based on TA-MASAC[C]. 2023 5th International Conference on Robotics, Intelligent Control and Artificial Intelligence (RICAI), Hangzhou, China 2023; 299-303. https://doi.org/10.1109/RICAI60863.2023.10489458

[28] Xu J, Zhang Z, Wang J, *et al*. Multi-AUV pursuit-evasion game in the internet of underwater things: an efficient training framework via offline reinforcement learning[J]. in IEEE Internet of Things Journal 2024; 11(19): 31273-31286. https://doi.org/10.1109/JIOT.2024.3416616

https://doi.org/10.65904/3083-3450.2026.02.02