Article

Multi-Constraints Guidance and Maneuvering Penetration Strategy via Meta Deep Reinforcement Learning

Sibo Zhao , JianWen Zhu * , Weimin Bao , Xiaoping Li , Haifeng Sun

Posted Date: 22 August 2023

doi: 10.20944/preprints202308.1512.v1

Keywords: Gliding flight; Vehicle penetration; Multi-constraints optimal guidance; Meta learning; SAC networks

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 22 August 2023 doi:10.20944/preprints202308.1512.v1

Article

Multi-Constraints Guidance and Maneuvering Penetration Strategy via Meta Deep Reinforcement Learning

SiBo Zhao [1], JianWen Zhu [1,2,*], WeiMin Bao [1,3], XiaoPing Li [1] and HaiFeng Sun [1]

[1] School of Aerospace Science and Technology, Xidian University, Xi'an, 710126, China 15161133950@163.com;
[2] College of Missile Engineering, Rocket Force University of Engineering, Xian,710025, China, zhujianwen1117@163.com;
[3] China Aerospace Science and Technology Corporation, Beijing 100048, China , baoweimin@cashq.ac.cn
* Correspondence: zhujianwen1117@163.com; Tel: (86)18392175968.

Abstract: In response to the issue of UAV escape guidance, this manuscript proposes a unified intelligent control strategy synthesizing optimal guidance and Meta Deep Reinforcement Learning (DRL). Optimal control with minor energy consumption is introduced to meet terminal latitude, longitude and altitude. Maneuvering escape is realized by adding longitudinal and lateral direction maneuver overloads. Maneuver command decision model is calculated based on Soft-Actor-Critic (SAC) networks. Meta learning is introduced to enhance autonomous escape capability, which improves generalization performance to time-varying scenarios not encountered in the training process. In order to obtain training samples at a faster speed, this manuscript uses the prediction method to solve reward values, which avoiding a large number of numerical integration. The simulation results manifest that the proposed intelligent strategy can achieve high precise guidance and effective escape.

Keywords: gliding flight; UAV penetration; multi-constraints optimal guidance; meta learning; SAC networks

1. Introduction

The hypersonic UAV mainly glides in the near space [1]. In the early phase, higher flight velocity is acquired relying on the thin atmospheric environment, which is an advantage to effectively avoid interception of the defense system. At the end of gliding flight, the velocity is mainly influenced by aerodynamic force, and suffered from restrictions of heat flow, dynamic pressure, overload [2]. The velocity advantage is hard to penetration, so orbital maneuvering is applied by UAV to achieve penetration. The main flight mission is split into avoiding defense system interception and satisfying terminal multiple constraints [3]. The core of manuscript is designing penetration guidance strategy via orbital maneuvering capability, avoiding the interception and reduce the penetration impact on guidance accuracy.

The penetration strategy is summarized as tactical penetration and technical penetration strategy [4]. The technical penetration strategy changes the flight path through maneuvering, aiming to increase the miss distance to successfully penetrate. Common maneuver manners include sine maneuver, step maneuver, square wave maneuver and spiral maneuver [5]. There are some limitations and instability for technical penetration strategy, attributed to the UAV is hard to adopt optimal penetration strategy according to the actual situation of offensive and defensive confrontation. Compared with the traditional procedural maneuver strategy, differential game guidance law has characters of real-time and intelligence as a tactical penetration strategy [6]. Penetration problems are essentially regarded as the continuous dynamic conflict problem of multi-party participants, and this strategy is an essential solution on solving multi-party optimal control problems. Applying it to the problem of attack defense confrontation can not only fully consider the relative information between UAV and interceptor, but also obtain Nash equation strategy to reduce energy consumption. Many scholars have proposed differential game models of various maneuvering control strategies based on control indexes and motion models. GARCIA [7] regarded the scenario of active target defense modeling as a zero-sum differential game, designed a complete differential game solution, and comprehensively considered the optimal strategy of closed-loop state feedback to obtain the value function. In Ref.[8], the optimal guidance problem was studied between interceptor and active defense ballistic UAV, and an optimal guidance scheme was proposed based on the linear quadratic differential game method and the numerical solution of Riccati differential equations. Liang [9] mainly analyzed the problem of pursuit and escape attack of multiple players, inducted the three body game confrontation into competition and cooperation problems, and solved the optimal solution of multiple players via differential game theory. Above methods are of great significance for analyzing and solving the confrontation process between UAV and interceptor. Nearby space UAV has characteristics of high velocity and short time in the phase of attack and defense confrontation terminal guidance [10], the differential game guidance law is difficult to show advantages in this phase. Moreover, the differential game method has a large amount of calculation, bilateral performance indicators are difficult to model [11], as a result, this theory is unable to be applied in practice.

DRL is a research hotspot in the field of artificial intelligence that has sprung up in recent years, and amazing learning results are achieved in robot control, guidance and control technologies[12]. DRL specifically refers to agents learning in the process of interaction with the environment to find the best strategy to maximize cumulative rewards [13]. With advantages of dealing with high-dimensional abstract problems and giving decisions quickly, DRL provides a new solution for maneuvering penetration of high velocity UAV. In order to solve the problem of intercepting high maneuvering target, an auxiliary DRL algorithm was proposed in Ref. [14] to optimize the frontal interception guidance control strategy based on neural network. Simulation results showed that DRL had higher hit rate and larger terminal interception angle than traditional methods and proximal policy optimization algorithms. Gong [15] proposed an Omni bearing attack guidance law of agile UAV via DRL, which effectively deal with aerodynamic uncertainty and strong nonlinearity at high attack angle. DRL was used to generate guidance law of attack angle in agile turning phase. Furfaro [16] proposed an adaptive guidance algorithm based on classical zero-efforts velocity, and limitations of this algorithm was overcome via RL. A closed-loop guidance algorithm was created, which is lightweight and flexible enough to adapt to a given constrained scene.

Compared with the differential game theory, DRL is convenient to establish the performance index function, and it is a feasible method to solve the dynamic programming problem by utilizing the powerful numerical calculation ability of computer to skillfully avoid solving the function analytical solution. However, the traditional DRL has some limitations, such as high sample complexity, low sample utilization, long training time and so on. Once the mission changing, original DRL parameters are hard to adapt to new mission and need to learn from scratch. The change of mission or environment will lead to the failure of trained model and poor generalization ability of model. In order to solve existing problems of DRL, researchers introduce Meta learning into DRL and propose Meta DRL [17]. By learning useful meta knowledge from a group of related missions, agents acquire the ability to learn to learn, and the learning efficiency on new missions is improved and the complexity of samples is reduced. When facing with new missions or environments, the network is responded quickly based on the previously accumulated knowledge, so that only a small number of samples are needed to quickly adapt new mission.

Based on above analysis, the manuscript proposes Meta DRL to solve the UAV guidance penetration strategy, and the DRL is improved, resulting in enhancing the adaptability of UAV in the complex and changeable attack and defense confrontation. Besides, the idea of Meta learning is used to make UAV learn to learn and improve the ability of autonomous flight penetration. Core contributions of this manuscript are as follows.

(1) By modeling the three-dimensional attack and defense scene between UAV and interceptor, analyzing terminal and process constraints of UAV, guidance penetration strategy based on DRL

is proposed, aiming to solve the optimal solution of maneuvering penetration under constant environment or mission.

(2) Meta learning is used to improve the UAV guidance penetration strategy, to make the UAV learn to learn and improve the autonomous flight penetration ability. Improving generalization performance to time-varying scenarios not encountered in the training process.

(3) Testing the network parameters based on Meta DRL, and analyzing the flight path and state under different attack and defense distances. Besides, analyzing the penetration strategy, and exploring penetration timing and maneuvering overload, and summarizing the penetration tactics.

2. Modeling of the Penetration Guidance Problem

2.1. Modeling of UAV Motion

The three-degree-of-freedom motion equation is adopted to describe UAV, and the dynamic equation is established in ballistic coordinating system:

$$\begin{cases} \dot{v} = -\dfrac{\rho v^2 S_m C_D}{2m} + g_r' \sin\theta + g_{\omega e}\left(\cos\sigma\cos\theta\cos\phi + \sin\theta\sin\phi\right) \\ \qquad + \omega_e^2 r\left(\cos^2\phi\sin\theta - \cos\phi\sin\phi\cos\sigma\cos\theta\right) \\ \dot{\theta} = \dfrac{\rho v^2 S_m C_L \cos\upsilon}{2mv} + \dfrac{g_r'\cos\theta}{v} - 2\omega_e\sin\sigma\cos\phi + \dfrac{v\cos\theta}{r} \\ \qquad + \dfrac{g_{\omega e}}{v}\left(\cos\theta\sin\phi - \cos\sigma\sin\theta\cos\phi\right) + \dfrac{\omega_e^2 r}{v}\left(\cos\phi\sin\phi\cos\sigma\sin\theta + \cos^2\phi\cos\theta\right) \\ \dot{\sigma} = -\dfrac{\rho v^2 S_m C_L \sin\upsilon}{2mv\cos\theta} - \dfrac{g_{\omega e}\sin\sigma\cos\phi}{v\cos\theta} + \dfrac{\omega_e^2 r\left(\cos\phi\sin\phi\sin\sigma\right)}{v\cos\theta} + \dfrac{v\tan\phi\cos\theta\sin\sigma}{r} \\ \qquad - 2\omega_e\left(\sin\phi - \cos\sigma\tan\theta\cos\phi\right) \\ \dot{\phi} = \dfrac{v\cos\theta\cos\sigma}{r} \\ \dot{\lambda} = -\dfrac{v\cos\theta\sin\sigma}{r\cos\phi} \\ \dot{r} = v\sin\theta \end{cases} \tag{1}$$

v is the velocity of UAV relative to the earth, θ is the velocity slope angle, σ is the velocity azimuth, and the positive direction is clockwise from the north. r represents the geocentric distance, and (λ,φ) is the longitude and latitude. The differential argument is flight time t. $g_{\omega e}$ is the component of the earth gravitational acceleration in direction of the earth rotational angular rate ω_e , meanwhile, g_r' is the component of the earth gravitational acceleration in direction of the geo-center. ρ is the density of atmosphere, moreover, in addition, m and S_m are the mass and reference area of UAV. C_D, C_L are the drag and lift coefficient, relating to the Mach number and attack angle, so the control variable attack angle α is implicit in it, the other control variable is the bank angle υ .

For a hypersonic UAV with large *L*/*D*, heat flow, overload, and dynamic pressure are considered as flight process constraints:

$$\begin{cases} k_h \rho^{1/2} v^3 \le Q_{s\max} \\ \dfrac{\rho v^2}{2} \le q_{\max} \\ \dfrac{\sqrt{D^2 + L^2}}{m g_0} \le n_{\max} \end{cases} \tag{2}$$

$Q_{s\max}$, $q_{\max}$ and $n_{\max}$ are the maximum heat flow, dynamic pressure and overload, respectively, parameter k_h is constant coefficients. In order to keep gliding state steady and effectively prevent trajectory jumps, the balance of lift and gravity is required by UAV [18]. Stable gliding is generally considered as quasi-equilibrium gliding (QEGs). At present, the international definition of QEG can be summarized into two types[19]: the velocity slope angle is considered as constant, expressed by $\dot{\theta} = 0$, or flight altitude variance ratio is considered as constant, expressed by $\ddot{h} = 0$. $\dot{\theta} = 0$ was adopted by the traditional QEG guidance, and $\ddot{h} = 0$ mainly appeared in the analytical prediction guidance of the Mars landing at the end of last century [20].

For the UAV with large range, the manuscript adopts $\dot{\theta} = 0$ as QEG condition. Referring to the second equation in Eq.(1), $\dot{\theta} = 0$ is transferred into Eq.(3).

$$m(g - \frac{v^2}{r})\cos\theta - L\cos\upsilon = 0 \tag{3}$$

2.2. *Description of Flight Missions*

2.2.1. Guidance Mission

The physical meaning of gliding guidance is eliminating heading errors, satisfying complex process constraints, and minimizing the energy loss. The UAV is guided to glide unpowered to the setting terminal target point ($h_f, \lambda_f, \varphi_f$), to satisfy the terminal altitude, longitude and latitude. Hence, terminal constraints are expressed by Eq.(4).

$$\begin{cases} h(L_{Rf}) = h_f ,\ \lambda(L_{Rf}) = \lambda_f \\ \varphi(L_{Rf}) = \varphi_f , \Delta\sigma_f \le \Delta\sigma_{\max} \end{cases} \tag{4}$$

where, the terminal range L_{Rf} is given, $\Delta\sigma$ represents the heading error, and $\Delta\sigma_{\max}$ is a pre-setting allowable value. The guidance problem is the process of determining α and υ .

2.2.2. Penetration Mission

The main indexes are used to judge penetration probability as follow:

(1) The miss distance D_{miss} with interceptor at the encounter moment.

(2) The overload N_m of interceptor at the last phase.

(3) The line-of-sight (LOS) angular rates $\dot{\theta}_{\text{int}\,los}$ and $\dot{\sigma}_{\text{int}\,los}$ with interceptor at the encounter moment.

where D_{miss} directly reflects the result of penetration, the larger of D_{miss}, the greater of penetration probability. N_m and δ_{MT} indirectly reflect the result of penetration. The larger of $\dot{\theta}_{\text{int}\,los}$ and $\dot{\sigma}_{\text{int}\,los}$, the more difficult for interceptor to successfully intercept, which is attributed to $\theta_{\text{int}\,los}$ and $\sigma_{\text{int}\,los}$

are hard to converge to a constant value at the encounter moment. Besides, a larger overload is required to adjust $\theta_{\text{int}los}$ and $\sigma_{\text{int}los}$ at the end of interception. The larger of N_m, the greater control cost of interceptor has to pay to complete the interception mission. Once N_m exceeds the overload limit of interceptor, indicating the control is saturated, which reflects the interceptor is hard to intercept based on the current maximum overload constraint. Conferred to the size and flight characteristics of UAV [21] , the manuscript assumes the penetration mission is completed if D_{miss} is greater than 2 m.

The maximum maneuvering overload of UAV is constrained by structure and QEG condition [19]. The lateral maneuver overload is too large, which will cause the UAV to deviate from the course, eventually, leading to the failure of guidance mission. A large longitudinal maneuver amplitude will have a significant impact on the lift-drag ratio (L/D) of UAV, affecting the safety. According to the above analysis, the maximum lateral maneuvering overload is set as 2 g, and the maximum longitudinal maneuvering overload is set as 1 g.

2.3. The Guidance Law of Interceptor

Guidance law of the interceptor relies on the inertial navigation system to obtain information such as position and velocity of the UAV. The relative motion is shown in Figure 1.

P, T and M respectively represent UAV, target and interceptor. r is the relative position between UAV and interceptor, and r_t is for target.

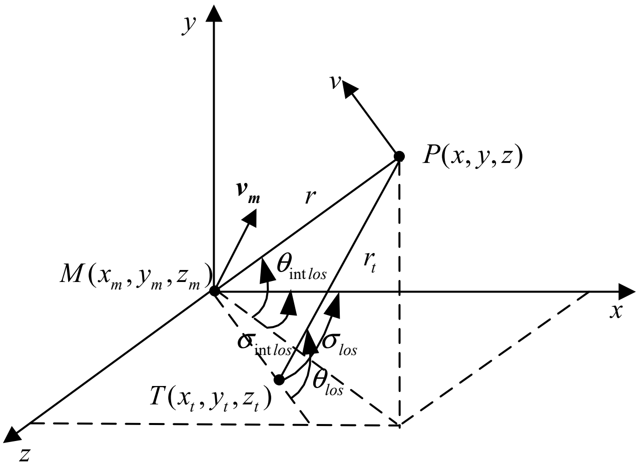

Figure 1. Attack-defense model.

The LOS angular rate and the approach velocity to UAV are obtained by the interceptor. Overload control command is derived by generalized proportional navigation guidance (GPNG).

$$\begin{cases} n_{y_2}^{*} = K_D\left|\dot{r}\right|\dot{\theta}_{\text{int}los}\big/G_0 \\ n_{z_2}^{*} = K_T\left|\dot{r}\right|\dot{\sigma}_{\text{int}los}\big/G_0 \end{cases} \tag{5}$$

As shown in Eq.(5), K_D and K_T are navigation ratios in the longitudinal and lateral direction respectively. $\dot{r}$ is the approach velocity. $\dot{\theta}_{\text{int}los}$ and $\dot{\sigma}_{\text{int}los}$ represent LOS angular rate between the longitudinal and lateral direction.

$$\begin{cases} r = \sqrt{(x-x_m)^2 + (y-y_m)^2 + (z-z_m)^2} \\ \theta_{\text{int}\,los} = \arcsin((y-y_m)/r) \\ \sigma_{\text{int}\,los} = \arctan((z-z_m)/(x-x_m)) \end{cases} \tag{6}$$

Differentiating Eq.(6) with respect to t, we have

$$\begin{cases} \dot{\theta}_{\text{int}\,los} = \dfrac{1}{\sqrt{1-(\dfrac{y-y_m}{r})^2}} \cdot \dfrac{(\dot{y}-\dot{y}_m)r-(y-y_m)\dot{r}}{r^2} \\ \dot{\sigma}_{\text{int}\,los} = \dfrac{1}{1+(\dfrac{z-z_m}{x-x_m})^2} \cdot \dfrac{(\dot{z}-\dot{z}_m)\cdot(x-x_m)-(z-z_m)\cdot(\dot{x}-\dot{x}_m)}{(x-x_m)^2} \\ \dot{r} = \dfrac{(x-x_m)\cdot(\dot{x}-\dot{x}_m)+(y-y_m)\cdot(\dot{y}-\dot{y}_m)+(z-z_m)\cdot(\dot{z}-\dot{z}_m)}{r} \end{cases} \tag{7}$$

3. Design of Penetration Strategy Considering Guidance

3.1. Guidance Penetration Strategy Analysis

Generally, the UAV achieves penetration through velocity advantage or increasing maneuvering overload. The former is used in pre-gliding flight, compared with interceptor, the velocity of UAV is relatively large, which is conductive to penetrate defense. More threats to UAV come from the defense system of intended target, resulting in intercept threat focusing on end of glide flight. However, the flight velocity gradually decreases, which is not enough to penetrate escape. Based on the above analysis, the manuscript designs penetration strategy by increasing the maneuvering overload.

Firstly, in the previous research [22], the energy-optimized gliding guidance law has been designed. Then, avoiding intercept by maneuvering becomes the key point. The real-time flight information of interceptor is difficult to be accurately obtained by UAV, contrarily, the real-time flight information of UAV is obtained by interceptor. Based on the UAV penetration mission for only knowing the initial launch position of interceptor, the manuscript simulates the attack and defense environment between the UAV and interceptor, applying the DRL method to solve the overload command of UAV maneuvering penetration. DNN parameters are trained offline, and maneuvering penetration command is constructed online. The launch position of interceptor is changeable, and stable penetration command cannot adapt to the complex penetration environment. To improve the adaptability of DNN parameters, the original DRL is optimized by adopting the idea of Meta learning, and the ontology and environmental information are fully utilized. The optimization of Meta learning enhances flight capability, fast response to complex missions and flight self-learning. Finally, the UAV guidance penetration strategy based on Meta DRL is proposed by this manuscript, as shown in Figure 2.

doi:10.20944/preprints202308.1512.v1

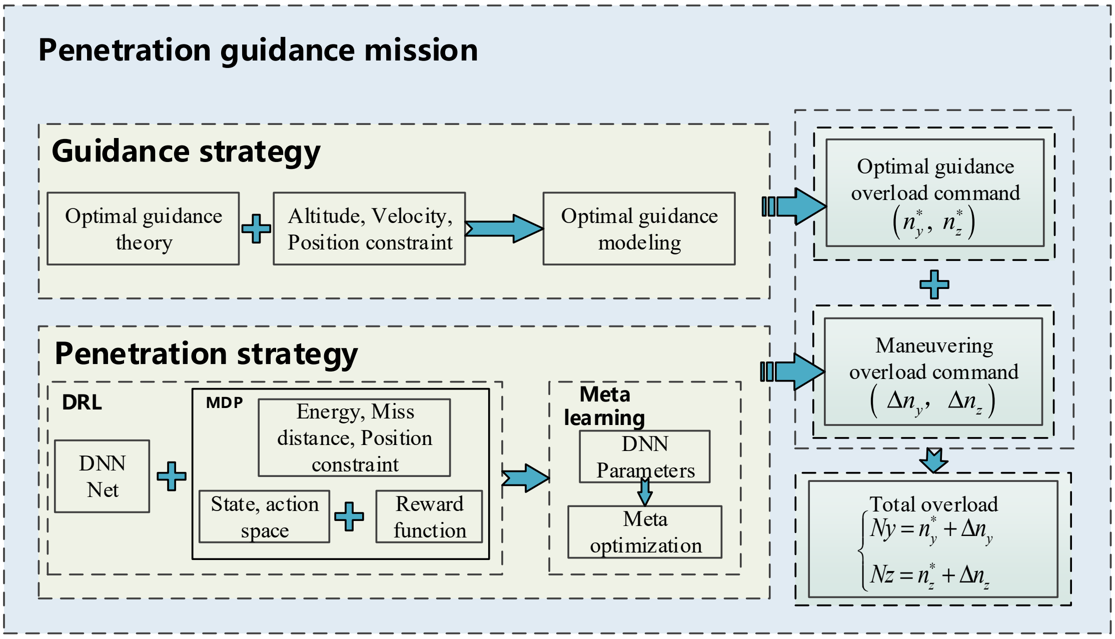
Penetration guidance mission
Guidance strategy
Optimal guidance theory
Altitude, Velocity, Position constraint
Optimal guidance modeling
Penetration strategy
DRL
DNN Net
MDP
Energy, Miss distance, Position constraint
State, action space
Reward function
Meta learning
DNN Parameters
Meta optimization
Optimal guidance overload command $(n_y^*,\ n_z^*)$
Maneuvering overload command $(\Delta n_y,\ \Delta n_z)$
Total overload $\begin{cases} Ny = n_y^* + \Delta n_y \\ Nz = n_z^* + \Delta n_z \end{cases}$

Figure 2. guidance penetration strategy.

As for gliding guidance mission, DRL and optimal control are used to achieve guidance penetration mission of UAV. Optimal guidance command was conducted in previous research [22], which is introduced to satisfy constraints of terminal position, altitude and minimal energy loss. Maneuvering overloads are added between longitudinal and lateral directions, aiming to achieve penetration mission at the end of gliding flight. Maneuvering overloads are solved by DRL. The DNN parameters are optimized by Meta learning.

Guidance command $(n_y^*,\ n_z^*)$ is generated by optimal guidance strategy, and maneuvering overload command $(\Delta n_y, \Delta n_z)$ is generated by Meta DRL. The flight total overload is shown in Eq.(8).

$$\begin{cases} Ny = n_y^* + \Delta n_y \\ Nz = n_z^* + \Delta n_z \end{cases} \tag{8}$$

Maneuvering penetration command via Meta DRL is the core section in this manuscript. The penetration considering guidance is described as Markov Decision Process (MDP), which consists of finite-dimensional continuous flight state space, longitudinal and lateral direction overload set, and reward function judging the penetration strategy. Flight data generated by numerical integration, and SAC networks are introduced to train and learn MDP. Optimizing network parameters via Meta learning, aims at adjusting network parameters with very little flight data when the UAV is faced with online mission changing, adapting to the new environment as soon as possible.

3.2. Energy Optimal Gliding Guidance Method

In the previous research [22], basing on the QEG condition and taking the required overload as the control amount, the performance index with minimum energy loss is established. The optimal longitudinal and lateral overload are designed respectively, satisfying the constraints on terminal latitude, longitude, altitude and velocity. The required overload command is shown in Eq.(9).

$$\begin{cases} u_y^* = k\left(C_h L_R - C_\theta\right) + 1 \\ u_z^* = \dfrac{\sigma_{los} - \sigma}{k\left(L_{Rf} - L_R\right)} \end{cases} \tag{9}$$

where, $u_y^* = n_y^*$ and $u_z^* = n_z^*$ are optimal overload between longitudinal and lateral. $k = \dfrac{g_0}{v^2} \approx \dfrac{g}{v^2}$, L_R is the current range, and L_{Rf} is the total range of gliding phase. C_h and C_θ are the guidance coefficients based on optimal control, represented as Eq.(10).

$$\begin{cases} C_h = \dfrac{6\left(\left(L_R - L_{Rf}\right)\left(\theta_f + \theta\right) - 2h + 2h_f\right)}{k^2\left(L_R - L_{Rf}\right)^3} \\ C_\theta = \dfrac{2\left(L_R L_{Rf}\left(\theta - \theta_f\right) - L_{Rf}^2\left(2\theta + \theta_f\right) + L_R^2\left(2\theta_f + \theta\right) + 3\left(L_{Rf} + L_R\right)\left(h_f - h\right)\right)}{k^2\left(L_R - L_{Rf}\right)^3} \end{cases} \tag{10}$$

Based on Eq.(11), the control variable α and υ are calculated as:

$$\begin{cases} \dfrac{\rho v^2 S_m C_L(Ma,\alpha)}{2g_0} = \sqrt{n_y^{*2} + n_z^{*2}} \\ \upsilon = \arctan\left(\dfrac{n_z^*}{n_y^*}\right) \end{cases} \tag{11}$$

where g_0 is the gravitational acceleration at sea level, α is obtained by calculating contrast value in Eq.(11).

4. RL Model for Penetration Guidance

The problem of maneuvering penetration is modeled as a series of stationary MDP [23] with unknown transition probabilities. The continuous flight state space, action set, and reward function for judging the command are determined in this section.

4.1. MDP of Penetration Guidance Mission

Deterministic MDP with continuous state and action, which is defined as $(\boldsymbol{S}, \boldsymbol{A}, T, R, \gamma)$ by a quintuple. $\boldsymbol{S}$ is described as continuous state space, $\boldsymbol{A}$ is described as finite action set, and T is depicted as the state transition function. $\boldsymbol{S} \times \boldsymbol{A} \to T$, reflecting deterministic state transition relationships. R is defined as immediate reward. $\gamma \in [0,1]$ is the discount factor, to balance immediate and forward reward.

The agent choices action $a_t \in \boldsymbol{A}$ at current state s_t, while state changes from s_t to $s_{t+1} \in \boldsymbol{S}$, the environment returns immediate reward $R_t = f(s_t, a_t)$ for agent. Cumulative rewards are obtained with controlled action sequence τ, as shown in Eq.(12).

$$G\left(s_0, \tau\right) = R_0 + \gamma R_1 + \gamma^2 R_2 + \ldots = \sum_{t=0}^{\infty} \gamma^t R^t \tag{12}$$

The goal of MDP is determining policy, maximizing the expected accumulated rewards:

$$\tau^* = \arg\max_{\tau}\left\{G\left(s_0, \tau\right)\right\} \tag{13}$$

4.1.1. State Space Design

The environment of attack-defense is abstracted as state space of MDP, which applies guiding for action command. UAV is hard to acquire the flight information and guidance law of interceptor. Hence, the information of UAV and target is only considered as state space.

$$\boldsymbol{S}=\left\{x_r,y_r,z_r,\theta_{los},\sigma_{los}\right\} \tag{14}$$

(x_r,y_r,z_r) represents the relative distance between UAV and target under North East Down (NED) Coordinate System. $(\theta_{los},\sigma_{los})$ represents respectively longitudinal and lateral LOS angle. In order to eliminate dimension difference and enhance compatibility among states, $\boldsymbol{S}$ is normalized by Eq.(15).

$$\boldsymbol{S}=\left(x_r=\frac{x_r}{x_{r0}},\ y_r=\frac{y_r}{y_{r0}},\ z_r=\frac{z_r}{z_{r0}},\ \theta_{los}=\frac{\theta_{los}}{2\pi},\ \sigma_{los}=\frac{\sigma_{los}}{2\pi}\right) \tag{15}$$

4.1.2. Action Space Design

Action is a decision selected by UAV based on current state, and action space $\boldsymbol{A}$ is the set of all possible decisions. Overload directly affects velocity azimuth and slope angle, indirectly affects gliding flight status, hence, the manuscript determines overload as an intermediate control variable.

On the basis of optimal guidance law and flight process constraints, longitudinal overload Δn_y and lateral overload Δn_z are added as action space of MDP. Considering the safety of UAV and heading error, the manuscript assumes longitudinal and lateral maximum maneuvering overload are 1 g and 2 g respectively. $\boldsymbol{A}$ is defined as continuous set, showing in Eq.(16).

$$\boldsymbol{A}=\begin{cases}\Delta n_y\in[-g,\ g]\\ \Delta n_z\in[-2g,\ 2g]\end{cases} \tag{16}$$

4.2. Multi-Missions Reward Function Designing

The reward function f_r is an essential section for guiding and training maneuvering penetration strategy. After execution the action command, f_r returns reward value to UAV, which reflects the fairness and scientific of action judgement. The rationality of f_r directly affects the training result, and determines the efficiency of SAC training. In this manuscript, the aim of f_r is guiding UAV to achieve guidance penetration mission, while satisfying terminal multi-constraints. Given requirements of mission, f_r consists the miss distance with the interceptor and the terminal deviate with target.

$$f_r(s,a)=c_1d_{miss}-c_2d_{error} \tag{17}$$

where d_{miss} and d_{error} are miss distance and terminal deviation after execution the action command. The sufficient and necessary condition of satisfying terminal position deviate is eliminating heading error, which is directly related to LOS angular rate. Similarly, d_{miss} can be reflected by the LOS angular rate at the encounter time between UAV and interceptor. Therefore, the normalized f_r is expressed by Eq.(18).

$$f_r=c_1\sqrt{\left(\dot{\bar{\theta}}_{\text{int}los}\right)^2+\left(\dot{\bar{\sigma}}_{\text{int}los}\right)^2}-c_2\sqrt{\left(\dot{\bar{\theta}}_{los}\right)^2+\left(\dot{\bar{\sigma}}_{los}\right)^2} \tag{18}$$

where $\dot{\bar{\theta}}_{\text{int}\,los}$ represents the normalized LOS angular rate in the longitudinal direction with interceptor, $\dot{\bar{\sigma}}_{\text{int}\,los}$ represents the normalized LOS angular rate in the lateral direction with interceptor. $\dot{\bar{\theta}}_{los}$ represents the normalized LOS angular rate in the longitudinal direction with target, $\dot{\bar{\sigma}}_{los}$ represents the normalized LOS angular rate in the lateral direction with target. In this manuscript, the LOS angular rates of at encounter time and terminal time are solved analytically by numerical calculation.

4.2.1. The Solution of LOS Angular Rate in the Lateral Direction

The attack defense confrontation model in lateral direction is shown in Figure 3.

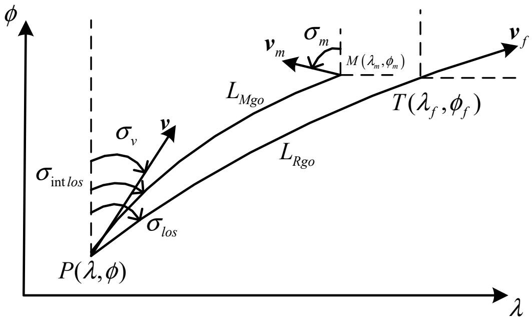

Figure 3. The attack defense confrontation model in lateral direction.

P, T and M respectively represent UAV, target and interceptor. L_{Rgo} and L_{Mgo} are represented as remaining range among UAV, target and interceptor, which are calculated in Eq.(19) and (20).

Lateral relative motion model between UAV and target is shown in Eq.(19).

$$\begin{cases} \dot{L}_{Rgo} = -v\cos\Delta\delta \\ L_{Rgo}\dot{\sigma}_{los} = v\sin\Delta\delta \end{cases} \tag{19}$$

Lateral relative motion model between UAV and interceptor is shown in Eq.(20).

$$\begin{cases} \dot{L}_{Mgo} = -v\cos\Delta\sigma - v_m\cos\Delta\sigma_m \\ L_{Mgo}\dot{\sigma}_{\text{int}\,los} = v\sin\Delta\sigma - v_m\sin\Delta\sigma_m \end{cases} \tag{20}$$

In order to simplify the calculation, the relative motion equation Eq.(20) is conducted as Eq.(21)

$$\begin{cases} \dot{L}_{Mgo} = -v_r\cos\Delta\sigma_{mn} \\ L_{Mgo}\dot{\sigma}_{\text{int}\,los} = v_r\sin\Delta\sigma_{mn} \end{cases} \tag{21}$$

where v_r and $\Delta\sigma_{mn}$ are calculated as Eq.(22).

$$\begin{cases} v_r = \sqrt{(v\cos\sigma_v - v_m\cos\sigma_m)^2 + (v\sin\sigma_v - v_m\sin\sigma_m)^2} \\ \Delta\sigma_{mn} = \text{atan}\dfrac{v\sin\sigma_v - v_m\sin\sigma_m}{v\cos\sigma_v - v_m\cos\sigma_m} \end{cases} \tag{22}$$

To facilitate the analysis and prediction, $\dot{\sigma}_{los}$ is calculated as follows. Taking the derivation of the second formula in Eq.(19):

doi:10.20944/preprints202308.1512.v1

$$\dot{L}_{Rgo}\dot{\sigma}_{LOS}+L_{Rgo}\ddot{\sigma}_{LOS}=\dot{v}\sin\Delta\sigma+v\Delta\dot{\sigma}\cos\Delta\sigma \tag{23}$$

Bring the heading error and first formula in Eq.(19) into Eq.(23), the rate of LOS angular rate is calculated by Eq.(24).

$$\begin{aligned}
&\dot{L}_{Rgo}\dot{\sigma}_{los}+L_{Rgo}\ddot{\sigma}_{los}=\dot{v}\sin\Delta\sigma+v\Delta\dot{\sigma}\cos\Delta\sigma\Rightarrow\\
&\dot{L}_{Rgo}\dot{\sigma}_{los}+L_{Rgo}\ddot{\sigma}_{los}=\frac{\dot{v}}{v}L_{Rgo}\dot{\sigma}_{los}-\dot{L}_{Rgo}\dot{\sigma}_{los}+\dot{L}_{Rgo}\dot{\sigma}\Rightarrow\\
&\ddot{\sigma}_{los}=\left(\frac{\dot{v}}{v}-\frac{2\dot{L}_{Rgo}}{L_{Rgo}}\right)\dot{\sigma}_{los}+\frac{\dot{L}_{Rgo}}{L_{Rgo}}\dot{\sigma}
\end{aligned} \tag{24}$$

Defining T_{goc} as the predicted remaining time of flight, T_{goc} is derived via the remaining flight range and variation in range, expressing in Eq.(25).

$$T_{goc}=-\frac{L_{Rgo}}{\dot{L}_{Rgo}} \tag{25}$$

Defining the value of state x=$\dot{\sigma}_{los}$ and the value of control $u=\dot{\sigma}$, the differential of LOS angular rate is obtained, which is shown in Eq.(26).

$$\dot{x}=\left(\frac{\dot{v}}{v}+\frac{2}{T_{goc}}\right)x-\frac{1}{T_{goc}}u \tag{26}$$

In the latter phase of glide flight, $\frac{\dot{v}}{v}$ is an order of magnitude smaller than $\frac{2}{T_{goc}}$. Eq.(26) is further simplified to Eq.(27).

$$\dot{x}=\frac{2}{T_{goc}}x-\frac{1}{T_{goc}}u \tag{27}$$

The current remaining flight time T_{goc}, a certain time t of future flight starting from the current time t, and the remaining flight time T_{go1} at time t satisfy the following relationship:

$$T_{goc}=T_{go1}+t \tag{28}$$

$dT_{go1}=-dt$ represents the derivation of remaining flight time. For a given control input u, the definite integral of Eq.(27) is solved, and the calculated result is shown in Eq.(29).

$$\begin{aligned} x(t) &= \mathrm{e}^{\int \frac{2}{T_{go1}}\mathrm{d}t}\left(\int -\frac{u}{T_{go1}}\mathrm{e}^{-\int \frac{2}{T_{go1}}\mathrm{d}t}\,\mathrm{d}t + C\right) \\ &= \mathrm{e}^{\int \frac{2}{T_{goc}-t}\mathrm{d}t}\left(\int -\frac{u}{T_{goc}-t}\mathrm{e}^{-\int \frac{2}{T_{goc}-t}\mathrm{d}t}\,\mathrm{d}t + C\right) \\ &= \mathrm{e}^{-2\ln\left(T_{goc}-t\right)}\left(\int \frac{u}{t-T_{goc}}\mathrm{e}^{2\ln\left(t-T_{goc}\right)}\,\mathrm{d}t + C\right) \\ &= \frac{1}{\left(T_{goc}-t\right)^2}\left(\int u\left(t-T_{goc}\right)\mathrm{d}t + C\right) \\ &= \frac{1}{\left(T_{goc}-t\right)^2}\left(u\left(\frac{1}{2}t^2 - T_{goc}t\right) + C\right) \end{aligned} \tag{29}$$

The LOS angular rate is $\dot{\sigma}_{los}$, at the current time t=0, the constant C is expressed by Eq.(30).

$$C = \dot{\sigma}_{los}T_{goc}^2 \tag{30}$$

$\dot{\sigma}_{los}$ is obtained by Eq.(31).

$$\dot{\sigma}_{los}(t) = \frac{1}{\left(T_{goc}-t\right)^2}\left(u\left(\frac{1}{2}t^2 - T_{goc}t\right) + \dot{\sigma}_{los}T_{goc}^2\right) \tag{31}$$

$\dot{\sigma}_{los}$ at the terminal time t_f is shown in Eq.(32).

$$\dot{\sigma}_{los} = \dot{\sigma}_{los}\left(t_f\right) = \frac{1}{\left(T_{goc}-t_f\right)^2}\left(u\left(\frac{1}{2}t_f^2 - T_{goc}t_f\right) + \dot{\sigma}_{los}T_{goc}^2\right) \tag{32}$$

Similarly, based on above analysis, $\dot{\sigma}_{\mathrm{int}los}$ is calculated via the analysis and prediction. The solution is shown in Eq.(33).

$$\dot{\sigma}_{\mathrm{int}\,los} = \dot{\sigma}_{\mathrm{int}\,los}\left(t_{\mathrm{int}\,f}\right) = \frac{1}{\left(T_{\mathrm{int}\,goc}-t_{\mathrm{int}\,f}\right)^2}\left(u\left(\frac{1}{2}t_{\mathrm{int}\,f}^2 - T_{\mathrm{int}\,goc}t_{\mathrm{int}\,f}\right) + \dot{\sigma}_{\mathrm{int}\,los}T_{\mathrm{int}\,goc}^2\right) \tag{33}$$

where $T_{\mathrm{int}\,goc}$ represents the total encounter time based on the current moment, $t_{\mathrm{int}\,f}$ represents encounter time with interceptor, and u is the input overload.

4.2.2. The Solution of LOS Angular Rate in the Longitudinal Direction

The attack defense confrontation model in longitudinal direction is shown in Figure 4.

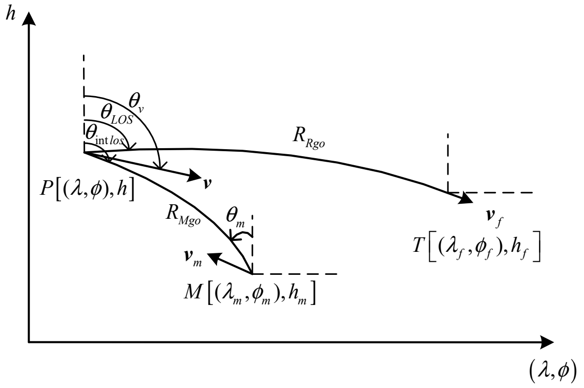

Figure 4. The attack defense confrontation model in longitudinal direction.

R_{Rgo} and R_{Mgo} are represented as remaining range among UAV, target and interceptor, which are calculated in Eq. (34)and(35).

Longitudinal relative motion model between UAV and target is shown in Eq.(19).

$$\begin{cases} \dot{R}_{Rgo} = -v\cos\Delta\theta \\ R_{Rgo}\dot{\theta}_{los} = v\sin\Delta\theta \end{cases} \tag{34}$$

Longitudinal relative motion model between UAV and interceptor is shown in Eq.(35).

$$\begin{cases} \dot{R}_{Mgo} = -v\cos\Delta\theta - v_m\cos\Delta\theta_m \\ \dot{R}_{Mgo}\dot{\theta}_{\text{int}los} = v\sin\Delta\theta - v_m\sin\Delta\theta_m \end{cases} \tag{35}$$

In order to simplify the calculation, the relative motion equation Eq.(35) is conducted as Eq.(36)

$$\begin{cases} \dot{R}_{Mgo} = -v_r\cos\Delta\theta_{mn} \\ R_{Mgo}\dot{\theta}_{\text{int}los} = v_r\sin\Delta\theta_{mn} \end{cases} \tag{36}$$

where v_r and $\Delta\theta_{mn}$ are calculated as Eq.(37).

$$\begin{cases} v_r = \sqrt{\left(v\cos\theta_v - v_m\cos\theta_m\right)^2 + \left(v\sin\theta_v - v_m\sin\theta_m\right)^2} \\ \Delta\theta_{mn} = \operatorname{atan}\dfrac{v\sin\theta_v - v_m\sin\theta_m}{v\cos\theta_v - v_m\cos\theta_m} \end{cases} \tag{37}$$

To facilitate the analysis and prediction, $\dot{\theta}_{los}$ is calculated as follows. Taking the derivation of the second formula in Eq.(34):

$$\dot{R}_{Rgo}\dot{\theta}_{LOS} + R_{Rgo}\ddot{\theta}_{LOS} = \dot{v}\sin\Delta\theta + v\Delta\dot{\theta}\cos\Delta\theta \tag{38}$$

Bring the heading error and first formula in Eq.(34) into Eq.(38), the rate of LOS angular rate is calculated by Eq.(39).

$$\begin{aligned}
&\dot{R}_{Rgo}\dot{\theta}_{los} + R_{Rgo}\ddot{\theta}_{los} = \dot{v}\sin\Delta\theta + v\Delta\dot{\theta}\cos\Delta\theta \Rightarrow \\
&\dot{R}_{Rgo}\dot{\theta}_{los} + R_{Rgo}\ddot{\theta}_{los} = \frac{\dot{v}}{v}R_{Rgo}\dot{\theta}_{los} - \dot{R}_{Rgo}\dot{\theta}_{los} + \dot{R}_{Rgo}\dot{\theta} \Rightarrow \\
&\ddot{\theta}_{los} = \left(\frac{\dot{v}}{v} - \frac{2\dot{R}_{Rgo}}{R_{Rgo}}\right)\dot{\theta}_{los} + \frac{\dot{R}_{Rgo}}{R_{Rgo}}\dot{\theta}
\end{aligned} \tag{39}$$

Based on the predicted remaining time of flight T_{goc} in lateral prediction, the LOS angular rate is $\dot{\sigma}_{los}$, at the current time *t*=0, the constant *C* is expressed by Eq.(40).

$$C{=}\dot{\theta}_{los}T_{goc}^{2} \tag{40}$$

$\dot{\theta}_{los}$ is obtained by Eq.(41).

$$\dot{\theta}_{los}(t) = \frac{1}{\left(T_{goc}-t\right)^{2}}\left(u\left(\frac{1}{2}t^{2} - T_{goc}t\right) + \dot{\theta}_{los}T_{goc}^{2}\right) \tag{41}$$

$\dot{\theta}_{los}$ at the terminal time t_f is shown in Eq.(42).

$$\dot{\theta}_{los}{=}\dot{\theta}_{los}\left(t_f\right) = \frac{1}{\left(T_{goc}-t_f\right)^{2}}\left(u\left(\frac{1}{2}t_f^{2} - T_{goc}t_f\right) + \dot{\theta}_{los}T_{goc}^{2}\right) \tag{42}$$

Similarly, based on above analysis, $\dot{\theta}_{\text{int}\,los}$ is calculated via the analysis and prediction. The solution is shown in Eq.(33).

$$\dot{\theta}_{\text{int}\,los}{=}\dot{\theta}_{\text{int}\,los}\left(t_{\text{int}\,f}\right) = \frac{1}{\left(T_{\text{int}\,goc}-t_{\text{int}\,f}\right)^{2}}\left(u\left(\frac{1}{2}t_{\text{int}\,f}^{2} - T_{\text{int}\,goc}t_{\text{int}\,f}\right) + \dot{\theta}_{\text{int}\,los}T_{\text{int}\,goc}^{2}\right) \tag{43}$$

where $T_{\text{int}\,goc}$ represents the total encounter time based on the current moment, $t_{\text{int}\,f}$ represents encounter time with interceptor, and u is the input overload.

According to the above analysis, $\left(\dot{\sigma}_{los}, \dot{\theta}_{los}\right)$ and $\left(\dot{\sigma}_{\text{int}\,los}, \dot{\theta}_{\text{int}\,los}\right)$ depend on change of LOS angle rate. For the guidance mission, $\left(\dot{\sigma}_{los}, \dot{\theta}_{los}\right)$ is related to overload of UAV, the smaller value, the closer UAV approaches target at the end of gliding flight. For the penetration mission, $\left(\dot{\sigma}_{\text{int}\,los}, \dot{\theta}_{\text{int}\,los}\right)$ is related to overloads of UAV and interceptor, the greater value, the higher cost of interceptor at the interception terminal phase, the easier of UAV will break through if the control overload of interceptor reaches saturation .

5. DRL Penetration Guidance Law

5.1. SAC Training Model

Standard DRL is maximizing the sum of expected rewards $\sum_{t}\mathbb{E}_{(s_t,a_t)\sim\rho_\pi}\left[f_r\left(s_t,a_t\right)\right]$. For the problem of multi-dimensional continuous state inputs and continuous action output, SAC networks are introduced to solve the MDP model.

Compared with other policy learning algorithm [24], SAC augments the standard RL objective with expected policy entropy by

$$J_\pi = \sum\nolimits_t \gamma^t \mathbb{E}_{(\mathrm{s}_t,\mathrm{a}_t)\sim\rho_\pi}\left[f_r(\mathrm{s}_t,\mathrm{a}_t) + \tau\mathcal{H}\left(\pi\left(\cdot\,|\mathrm{s}_t\right)\right)\right] \tag{44}$$

The entropy term $\tau\mathcal{H}\left(\pi\left(\cdot\,|\mathrm{s}_t\right)\right)$ is shown in Eq.(45), which represents the stochastic feature of strategy, balancing the exploration and learning of networks. The entropy parameter τ determines the relative importance of entropy against immediate reward.

$$\begin{aligned}\mathcal{H}\left(\pi\left(\cdot\,|\mathrm{s}_t\right)\right) &= -\int_{\mathrm{a}\in A}\pi\left(\mathrm{a}|\mathrm{s}_t\right)\log\pi\left(\mathrm{a}|\mathrm{s}_t\right)d\mathrm{a} \\ &= \mathop{\mathbb{E}}_{\mathrm{a}\sim\pi\left(\cdot\,|\mathrm{s}_t\right)}\left[-\log\pi\left(\mathrm{a}|\mathrm{s}_t\right)\right]\end{aligned} \tag{45}$$

The optimal strategy of SAC is shown in the Eq.(46), aiming of maximizing the cumulative reward and policy entropy.

$$\pi^*_{MaxEnt} = \arg\max_{\pi} \sum\nolimits_t \gamma^t \mathbb{E}_{(\mathrm{s}_t,\mathrm{a}_t)\sim\rho_\pi}\left[f_r(\mathrm{s}_t,\mathrm{a}_t) + \tau\mathcal{H}\left(\pi\left(\cdot\,|\mathrm{s}_t\right)\right)\right] \tag{46}$$

The frame of sac networks is shown in the Figure 5, consists of Actor net and Critic net. Actor net is generating action, and environment returns the reward and next state. All of ballistics data is stored in experience pool, including state, action, reward, and next state.

Critic net is used to judge the found strategies, which guides impartially the strategy of Actor network. At the beginning, Actor net and Critic net are given random parameters. Actor net is difficult generating the optimal strategy, and Critic net is difficult judging scientifically towards the strategy of Actor net. The parameters of networks are need to update based on continuously generating and sampling ballistics data.

For updating Critic net, Critic net outputs the expected reward $\mathop{\mathbb{E}}_{\mathrm{a}\sim\pi\left(\cdot\,|\mathrm{s}_t\right)}$ based on samples, and Actor net outputs the action probability, which is depicted by entropy term $\mathcal{H}\left(\pi\left(\cdot\,|\mathrm{s}_t\right)\right)$. Combining $\mathop{\mathbb{E}}_{\mathrm{a}\sim\pi\left(\cdot\,|\mathrm{s}_t\right)}$ with $\mathcal{H}\left(\pi\left(\cdot\,|\mathrm{s}_t\right)\right)$, the value function is conducted and shown in Eq.(47)

$$Q_{soft}(\mathbf{s}_t,\mathbf{a}_t) = \mathop{\mathbb{E}}_{(\mathbf{s}_t,\mathbf{a}_t)\sim\rho_\pi}\left[\sum_{t=0}^{\infty}\gamma^t r(\mathbf{s}_t,\mathbf{a}_t) + \tau\sum_{t=1}^{\infty}\gamma^t\mathcal{H}(\pi(\cdot\,|\,\mathbf{s}_t))\right] \tag{47}$$

Further obtain the Bellman equation:

$$Q_{soft}(\mathbf{s}_t,\mathbf{a}_t) = \mathop{\mathbb{E}}_{\substack{\mathbf{s}_{t+1}\sim\rho(\mathbf{s}_{t+1}|\mathbf{s}_t,\mathbf{a}_t)\\ \mathbf{a}_{t+1}\sim\pi}}\left[r(\mathbf{s}_t,\mathbf{a}_t) + \gamma\left(Q_{soft}(\mathbf{s}_{t+1},\mathbf{a}_{t+1}) + \tau H(\pi(\cdot\,|\,\mathbf{s}_{t+1}))\right)\right] \tag{48}$$

Given by Eq.(49), the loss function of Critic net is acquired:

$$J_Q(\psi) = \mathop{\mathbb{E}}_{\substack{(\mathbf{s}_t,\mathbf{a}_t,\mathbf{s}_{t+1})\sim D\\ \mathbf{a}_{t+1}\sim\pi}}\left[\frac{1}{2}\left(Q_{soft}(s_t,a_t) - \left(r(\mathbf{s}_t,\mathbf{a}_t) + \gamma\left(Q_{soft}(\mathbf{s}_{t+1},\mathbf{a}_{t+1}) - \tau\log\left(\pi\left(\mathbf{a}_{t+1}\,|\,\mathbf{s}_{t+1}\right)\right)\right)\right)\right)^2\right] \tag{49}$$

For updating Actor net, the updating strategy is shown in Eq.(50)

$$\pi_{new} = \arg\min_{\pi\in\Pi} D_{KL}\left(\pi(\cdot\,|\,\mathbf{s}_t)\middle\|\frac{\exp\left(\frac{1}{\tau}Q^{\pi_{old}}_{soft}(\mathbf{s}_t,\cdot)\right)}{Z^{\pi_{old}}_{soft}(\mathbf{s}_t)}\right) \tag{50}$$

where, Π represents the set of strategy, $Z$is partition function, used to normalized distribution. D_{KL} is Kullback-Leibler (KL) divergence [25].

Combining re-parameterization technique with Eq.(51), the loss function of Actor net is obtained:

$$J_{\pi}(\phi) = \mathop{\mathbb{E}}_{\mathbf{s}_t \sim D, \varepsilon_t \sim N}[\tau \log \pi(f(\varepsilon_t;\mathbf{s}_t) \,|\, \mathbf{s}_t) - Q_{soft}(\mathbf{s}_t, f(\varepsilon_t;\mathbf{s}_t))] \tag{51}$$

In which $\mathbf{a}_t = f(\varepsilon_t;\mathbf{s}_t)$, ε_t is the input noise, obeying the distribution N.

The method of stochastic gradient descent is introduced to minimize the loss function of networks. The optimal parameters of Actor-Critic networks are obtained with repeating the updating process, passing respectively the parameters to target networks via soft-updating.

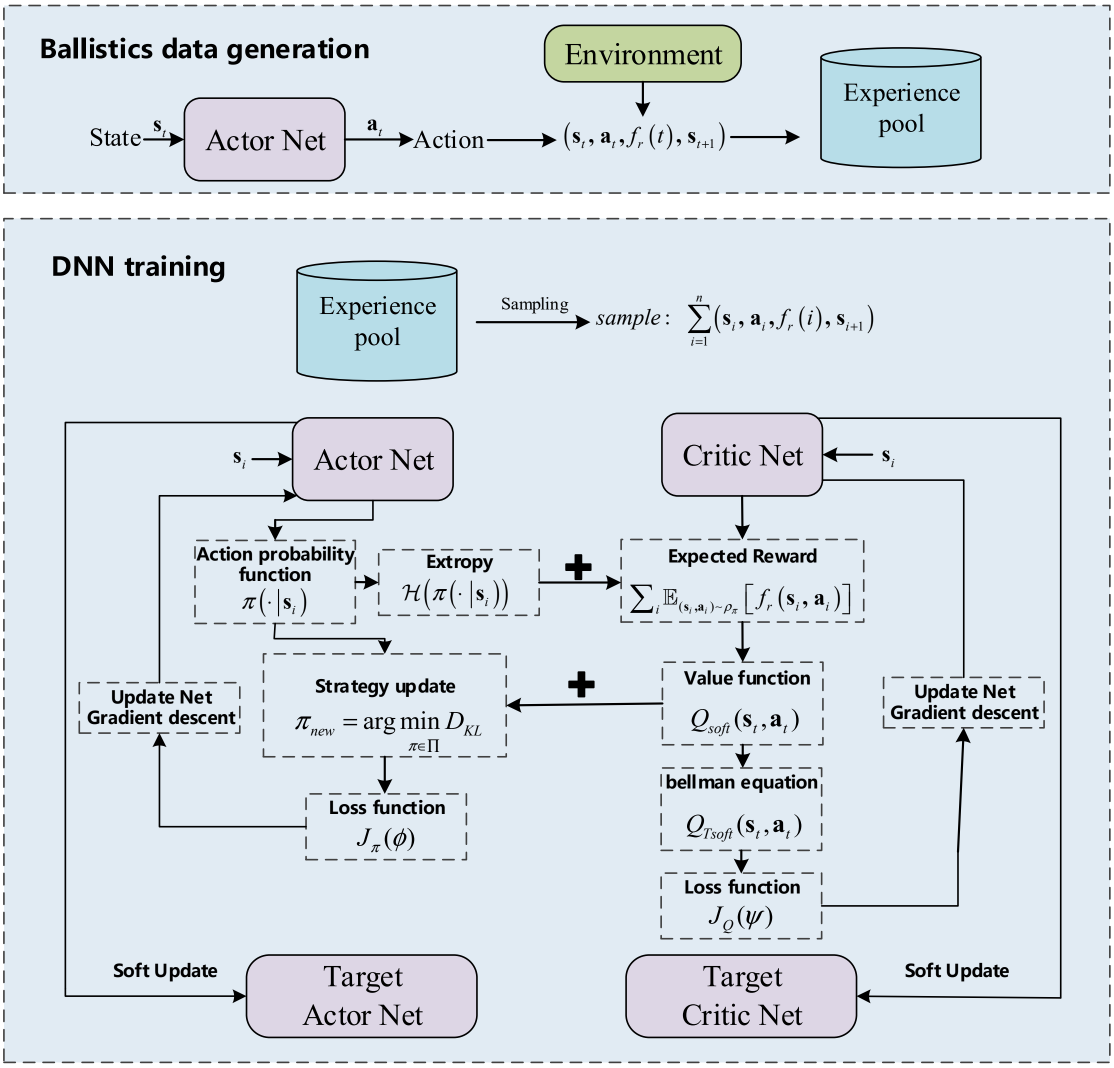
Ballistics data generation
Environment
Experience pool
State
$\mathbf{s}_t$
Actor Net
$\mathbf{a}_t$
Action
$(\mathbf{s}_t, \mathbf{a}_t, f_r(t), \mathbf{s}_{t+1})$
DNN training
Experience pool
Sampling
$sample: \sum_{i=1}^{n}(\mathbf{s}_i, \mathbf{a}_i, f_r(i), \mathbf{s}_{i+1})$
$\mathbf{s}_i$
Actor Net
Critic Net
$\mathbf{s}_i$
Action probability function
$\pi(\cdot|\mathbf{s}_i)$
Extropy
$\mathcal{H}(\pi(\cdot|\mathbf{s}_i))$
Expected Reward
$\sum_i \mathbb{E}_{(\mathbf{s}_i,\mathbf{a}_i)\sim\rho_\pi}[f_r(\mathbf{s}_i, \mathbf{a}_i)]$
Update Net Gradient descent
Strategy update
$\pi_{new} = \arg\min_{\pi\in\Pi} D_{KL}$
Value function
$Q_{soft}(\mathbf{s}_t, \mathbf{a}_t)$
Update Net Gradient descent
bellman equation
$Q_{Tsoft}(\mathbf{s}_t, \mathbf{a}_t)$
Loss function
$J_\pi(\phi)$
Loss function
$J_Q(\psi)$
Soft Update
Target Actor Net
Target Critic Net
Soft Update

Figure 5. updating principle of SAC.

5.2. Meta SAC Optimization Algorithm

The learning algorithm in DRL relies on a lot of interaction between agent and environment, and high training costs. Once the environment changed, the original strategy is no longer applicable, and need to learn from scratch. The penetration guidance problem under the stable flight environment can be solved by SAC networks. For the changeable flight environment, such as the initial position of interceptor changes greatly, or the interceptor guidance law deviates greatly from preset value, the strategy solved by traditional SAC is hard to adapt, which is need to restudy and redesign. The manuscript introduces Meta learning to optimize and improve SAC performance. The training goal of Meta SAC is to obtain initial SAC model parameters. When the UAV penetration mission is changed, through a few scenes of learning, the UAV can adapt to new environment and complete the corresponding guidance penetration mission, without relearning model parameters. Meta SAC can achieve “learn while flying” for UAV, and strengthen the adaptability of UAV.

The Meta SAC algorithm is shown in Algorithm 1, which is divided into meta training and meta testing phase. Meta training phase is to solve the optimal meta learning parameters based on multi-experience missions. In the meta testing phase, the trained meta parameters are interactively learned with the new mission environment to fine-tune the meta parameters.

Algorithm 1 Meta SAC

```
Initialize the experience pool $\Omega$ , Storage space $N$
Meta training:
Inner loop
for iteration $k$ do
    sample mission($k$) from $\mathcal{T} \sim p(\mathcal{T})$
    update actor policy $\Theta$ to $\Theta'$ using SAC based on mission($k$):
    $\Theta' \leftarrow SAC(\Theta, mission(k))$ .
Outer loop
$\Theta = mmse\left(\sum_{i=1}^{k} \Theta_i'\right)$
Generate $\mathcal{D}_1$ from $\Theta$ and estimate the reward of $\Theta$.
Add a hidden layer feature as a random noise.
$\Theta_i' = \Theta + \alpha_\Theta \nabla_\Theta E_{\alpha_t \sim \pi(\alpha_t|s_t;\Theta,z_i),z_i \sim \mathcal{N}(\mu_i,\sigma_i)}\left[\sum_t R_t(s_t)\right]$
The meta learning process of different missions is carried out through SGD.
for iteration $mission(k)$ do
    $\min_\Theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\Theta_i'}\right) = \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\Theta-\alpha} \nabla\Theta \mathcal{L}_{\mathcal{T}_i}(f_\Theta)\right)$
    $\Theta = \Theta - \beta \nabla \Theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\Theta_i'}\right)$
Meta testing
Initialize the experience pool $\Omega$ , Storage space $N$.
Load meta training network parameters $\Theta$.
Set training parameters.
for iteration $i$ do
    sample mission from $\mathcal{T} \sim p(\mathcal{T})$
    $\Theta' = SAC(\Theta, mission(k))$
End for
```

The basic assumption of Meta SAC is that the experience mission for meta training and the new mission for meta testing obey the same mission distribution $p(\mathcal{T})$. Therefore, there are some common characteristics between different missions. In the DRL scenario, our goal is to learn a function f_θ with parameter θ, which can minimize the loss function $\mathcal{L}_\mathcal{T}$ of specific mission $\mathcal{T}$. In the Meta DRL scenario, our goal is to learn a learning process $\theta' = \mu_\psi\left(\mathcal{D}_\mathcal{T}^{tr}, \theta\right)$, which can quickly adapt to the new mission $\mathcal{T}$ with a very small dataset $\mathcal{D}_\mathcal{T}^{tr}$. Meta SAC can be summarized as optimizing the parameters θ, ψ in the learning process:

$$\min_{\theta,\psi} E_{\mathcal{T}\sim P(\mathcal{T})}\left[\mathcal{L}\left(\mathcal{D}_\mathcal{T}^{test}, \theta'\right)\right] \quad \text{s.t.} \;\; \theta' = \mu_\psi\left(\mathcal{D}_\mathcal{T}^{tr}, \theta\right) \tag{52}$$

where $\mathcal{D}_\mathcal{T}^{test}$ and $\mathcal{D}_\mathcal{T}^{tr}$ respectively represent training and testing missions sampled from $p(\mathcal{T})$, $\mathcal{L}\left(\mathcal{D}_\mathcal{T}^{test}, \theta'\right)$ represents the testing loss function. In the meta training phase, parameters are optimized by inner loop and outer loop.

In the inner loop, Meta SAC updates model parameters with a small amount of randomly selected data of specific mission $\mathcal{T}$ as the training data, reducing the loss of model on mission $\mathcal{T}$. In this part, updating of model parameters is the same as the original SAC algorithm, and the agent learns several scenes on randomly selected missions.

The minimum mean square error of strategy parameters θ corresponding to different missions in the inner loop phase is solved to obtain the initial strategy parameters θ_{ini} of the outer loop. In this manuscript, a hidden layer feature is added to the input part of strategy θ_{ini} as a random noise. The random noise is sampled again in each episode, in order to provide a more continuous random exploration in time, which is helpful for agent to adjust their overall strategy exploration according to the current mission MDP. The goal of Meta learning is to let agent learn how to quickly adapt to new missions by simultaneously updating a small amount of gradient of strategy parameters and hidden layer features. Therefore, the θ of $\theta' = \mu_\psi\left(\mathcal{D}_\mathcal{T}^{tr}, \theta\right)$ includes not only parameters of the neural network, but also the distribution parameters of hidden variables of each mission, namely the mean and variance of the Gaussian distribution, as shown in Eq.(53).

$$\begin{aligned}
\mu_i' &= \mu_i + \alpha_\mu \nabla_{\mu_i} E_{\alpha_t\sim\pi(\alpha_t|s_t;\theta,z_i),z_i\sim\mathcal{N}(\mu_i,\sigma_i)}\left[\sum_t R_t(s_t)\right] \\
\sigma_i' &= \sigma_i + \alpha_\sigma \nabla_{\sigma_i} E_{\alpha_t\sim\pi(\alpha_t|s_t;\theta,z_i),z_i\sim\mathcal{N}(\mu_i,\sigma_i)}\left[\sum_t R_t(s_t)\right] \\
\theta_i' &= \theta + \alpha_\theta \nabla_\theta E_{\alpha_t\sim\pi(\alpha_t|s_t;\theta,z_i),z_i\sim\mathcal{N}(\mu_i,\sigma_i)}\left[\sum_t R_t(s_t)\right]
\end{aligned} \tag{53}$$

The model is represented by a parameterized function f_θ with parameter θ, and when it is transferred to a new mission $\mathcal{T}$, model parameter θ is updated to θ' through gradient rise, as shown in Eq.(54).

$$\theta_i' = \theta - \alpha\nabla_\theta \mathcal{L}_{\mathcal{T}_i}\left(f_\theta\right) \tag{54}$$

Update step α is a fixed super parameter. Model parameter θ is updated to maximize the performance $f_{\theta_i'}$ of different missions, as shown in Eq.(55).

$$\min_\theta \sum_{\mathcal{T}_i\sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\theta_i'}\right) = \sum_{\mathcal{T}_i\sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\theta-\alpha}\nabla\theta\mathcal{L}_{\mathcal{T}_i}\left(f_\theta\right)\right) \tag{55}$$

The meta learning process of different missions is carried out through SGD, and the principle of θ is:

doi:10.20944/preprints202308.1512.v1

$$\theta = \theta - \beta \nabla \theta \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}\left(f_{\theta_i'}\right) \qquad (56)$$

where β is meta update step.

In the meta testing phase, a small amount of experience in new missions is used to quickly learn strategies for solving new missions. A new mission may involve completing a new mission goal or achieving the same mission goal in a new environment. The updating process of model in this phase is the same as the cycle part in the meta training phase, by calculating the loss function with the data collected in the new mission, adjusting the model through back propagation, at last, new mission is adapted by agent.

6. Simulation Analysis

In this section, the manuscript analyzes and verifies escape guidance strategy based on Meta SAC. SAC is used to solve specific escape guidance mission. We conduct comprehensive experiments to verify whether the UAV can complete the guidance escape mission under satisfying terminal constraints and process constraints. Once the UAV guidance escape mission changes, the original strategy based on SAC is difficult to adapt to the changed mission, which is need to be relearned and retrained. The manuscript proposes an optimization method via Meta learning, which improves the learning ability of UAV during the training process. This section focuses on verifying the validation of Meta SAC, demonstrating the performance in various new missions. Besides, the maneuvering overload commands under different pursuit evading distances are analyzed, which is used to explore the influence of different maneuvering timings and distances on the escape results. Taking CAV-H to verify the escape guidance performance. The initial conditions, terminal altitude and Meta SAC training parameters are given in Table 1.

Table 1. Simulation and Meta SAC training conditions.

Simulation Conditions		Meta SAC Training Parameters	
UAV initial velocity	4000m/s	Learning episodes	1000
Initial velocity Inclination	0°	Guidance period	0.1s
Initial velocity azimuth	0°	Data sampling interval	30Km
Initial position	(3°E, 1°N)	Discount factor	$\gamma = 0.99$
Initial altitude	45km	Soft update tau	0.001
Terminal altitude	40km	Learing rate	0.005
Target position	(0°E, 0°N)	Sampling size for each train	128
Interceptor Initial velocity	1500m/s	Net layers	2
Initial velocity Inclination	longitudinal LOS angle	Net nodes	256
Initial velocity azimuth	lateral LOS angle	Capacity of experience pool	20000

6.1. Validity verification on SAC

In order to verify the effectiveness of SAC, three different pursuit evading scenarios are constructed, and the terminal reward value, miss distance and terminal position deviate are respectively analyzed. As shown in Figure 8(a), the terminal reward value is poor in the initial phase of training, which manifests the optimal strategy is not found. After 500 episodes, the terminal reward value increases gradually, indicating better strategy is explored and converged. At the last 100 episodes, the optimal strategy is trained and learned, meanwhile, the network parameters have been adjusted to the optimal. As can be seen from Figure 8(b), the miss distance is relatively divergent in the first 150 episodes of training, indicating the action network in SAC is constantly exploring new strategies, and the critic network is also learning scientific evaluation criteria. After 500 training episodes, the network is gradually learning and training in the direction of optimal solution. The miss distance at the encounter moment converges to about 20m. As shown in Figure 8(c), the terminal position of UAV has a large deviation in the early training phase, which is attributed to the exploration of escape strategy by network. In the later training phase, the position deviation is less affected by exploration. These pursuit evading scenarios tested in the manuscript can achieve convergence, and the final convergence values are all within 1m.

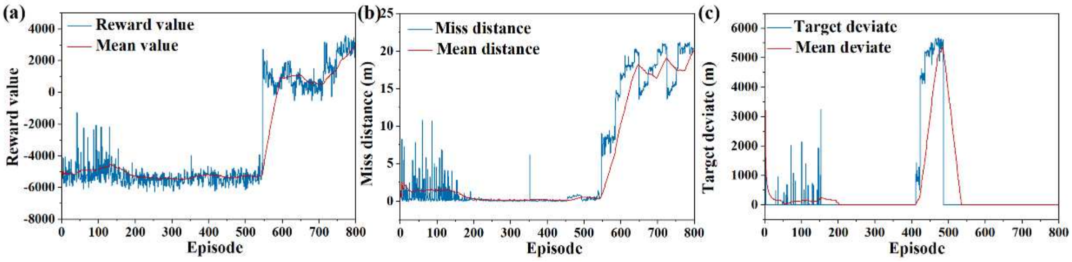

Figure 6. Train results of SAC. (a) reward value, (b) miss distance, (c) target deviate.

In order to verify whether the SAC algorithm can solve the escape guidance strategy that meets the mission requirements in different pursuit and evasion scenarios, the pursuing and evading distance is changed, and the training results are shown in the Figure 7. In medium range scenario, the miss distance converges to about 2m, and the terminal deviation converges to about 1m.

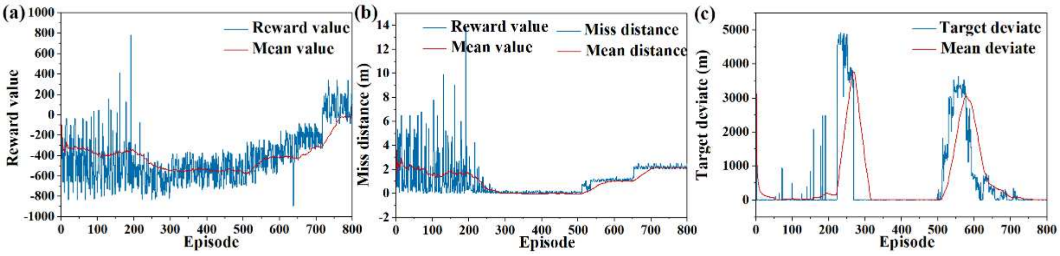

Figure 7. Train results of SAC. (a) reward value, (b) miss distance, (c) target deviate.

As shown in Figure 8, In long range attack and defense scenarios, the miss distance converges to about 5m, and the terminal deviation converges to about 1m.

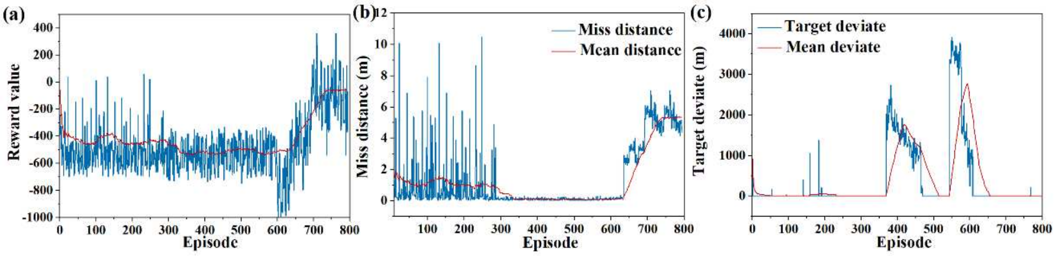

Figure 8. Train results of SAC. (a) reward value, (b) miss distance, (c) target deviate.

Based on above simulation analysis, SAC is a feasible method to solve the UAV guidance escape strategy. After limited episodes of learning and training, network parameters are converged, which is used to test on flight mission.

6.2. Validity Verification on Meta SAC

When the mission of UAV changes, the original SAC parameters can not meet acquirements of new mission,which needs to be re-trained and learned. The SAC proposed in the manuscript is improved via Meta learning. Strong adaptive network parameters are found by learning and training, when the pursuit evading environment changes, the network parameters is fine-tuned to adapt to the new environment immediately.

Meta SAC is divided into meta training phase and meta testing phase, Initialization parameters for SAC network are trained in meta training phase, which is fine-tuned by interacting with the new environment in meta testing phase. By changing the initial interceptor position, three different pursuit evading scenarios are constructed, which respectively represents short distance, medium and long distance.

Training results of Meta SAC and SAC are compared, terminal reward values are represented as shown in Figure 9(a). Meta SAC is an effective method to speed up the training process, after 100 episodes, better strategy is learned by the network and converged gradually, contraryly, the SAC network needs 500 episodes to find the optimal solution. Miss distance is shown in Figure 9(b). The better strategy is quickly learned by Meta SAC, which is more effective than the SAC method. Figure 9(c) show the terminal deviate between UAV and target.

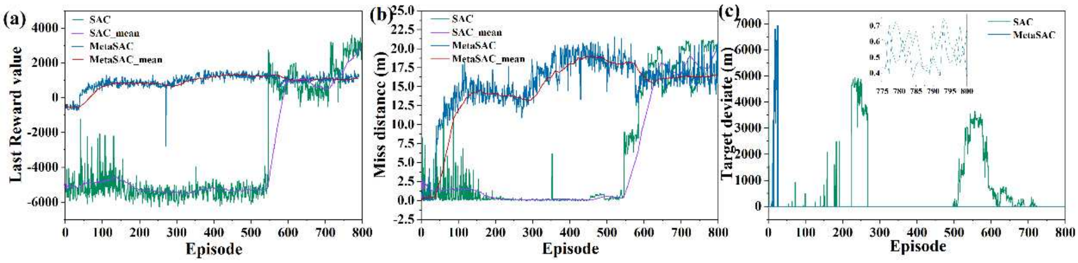

Figure 9. Meta SAC training results. (a) Reward value, (b) miss distance, (c) target deviate.

To explore the optimal solution as much as possible, some strategies with large terminal position deviation appear in the training process. As shown in Figure 10(b-c), in medium range attack and defense scenarios, the miss distance converges to about 8m based on Meta SAC, and the terminal deviation converges to about 1m.

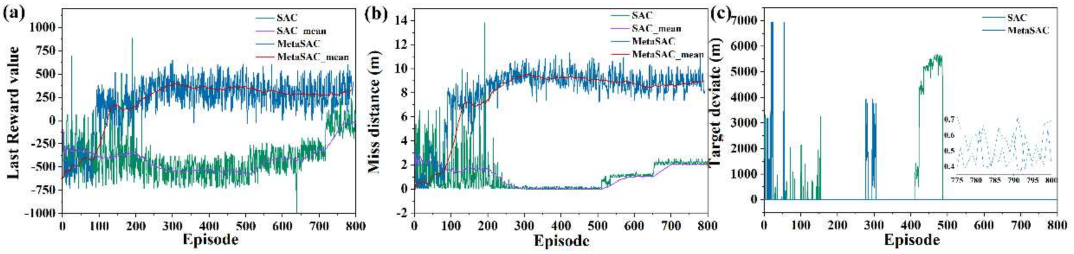

Figure 10. Meta SAC training results. (a) Reward value, (b) miss distance, (c) target deviate.

As shown in Figure 11(b-c), in long range attack and defense scenarios, the miss distance converges to about 10m based on Meta SAC, and the terminal deviation converges to about 1m.

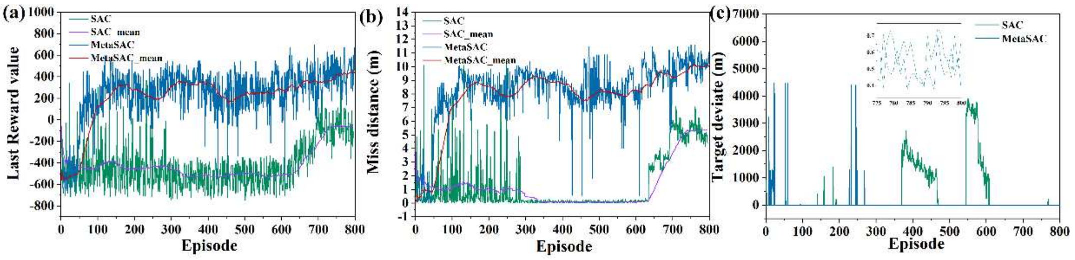

Figure 11. Meta SAC training results. (a) Reward value, (b) miss distance, (c) target deviate.

According to the theoretical analysis, in the training process, new missions corresponding to the same distribution are used to execute micro-testing by Meta SAC, resulting in more gradient descending directions of optimal solution are learned by network. Combined with the theory analysis and training results, the manuscript manifests Meta learning is a feasible method to accelerate convergence and improve the efficiency of training.

In the previous analysis, when the pursuit evading secenerio is changed, network parameters obtained in the meta training phase are fine-tuned through few interactions. The manuscript verifies meta testing performance by changing the initial interceptor position, and results compared with SAC method are shown in the Table 2. Based on the network parameters of meta training phase, the strategic solutions meeting escape guidance missions are found through training within 10 episodes. On the contrary, network parameters based on SAC need more interaction to find solutions, and the the episode of interactions is basically more than 50 episodes. According to above simulation, the adaptability of Meta SAC is much greater than SAC, once the escape mission changing, through very few episodes of learning, the new mission is completed by UAV without re-learning and designing strategy. The method provides possibility for realizing UAV learning while flying.

Table 2. Results compared with SAC method.

Interceptor Initial position (Km)	**Interaction episodes**		**Miss distance (m)**		**Terminal deviate (m)**	
	SAC	Meta SAC	SAC	Meta SAC	SAC	Meta SAC
(0, 30, 0)	74	1	3.78	3.29	0.56	0.61
(2, 30, 6)	75	4	2.80	2.72	0.68	0.72
(4, 30, 12)	59	8	6.93	3.75	0.69	0.58
(6, 30, 18)	59	1	2.71	6.82	0.68	0.72
(8, 30, 24)	26	2	3.16	3.70	0.47	0.50
(10, 30, 30)	58	3	3.50	2.37	0.61	0.64
(12, 30, 36)	67	1	2.86	2.21	0.68	0.45
(14, 30, 42)	56	8	2.18	2.89	0.55	0.61
(16, 30, 48)	69	1	2.73	2.23	0.61	0.72
(18, 30, 54)	106	1	2.45	3.71	0.56	0.63
(20, 30, 60)	94	1	2.7	2.35	0.49	0.54
(22, 30, 66)	59	1	2.23	2.51	0.73	0.71
(24, 30, 72)	62	1	2.11	3.47	0.48	0.67
(26, 30, 78)	63	1	2.04	4.5	0.48	0.57
(28, 30, 84)	63	4	2.64	5.12	0.47	0.40
(30, 30, 90)	63	9	2.95	6.05	0.68	0.47

6.3. Strategy Analysis Based on Meta SAC

This section tests the network parameters based on Meta SAC, and analyzes the escape strategy and flight state under different pursuit evading distances. As shown in Figure 12(a), for the pursuit evading scene of short distance, the longitudinal maneuvering overload is larger in the first half phase of escape, resulting in velocity slope angle decreases gradually. In the second half phase of escape, if strategy is executed under the original maneuvering overload, the terminal altitude constraint can not be satisfied, therefore, the overload gradually decreases, the velocity slope angle is slowly reduced. As shown from Figure 12(b), at the beginning of escape, the lateral maneuvering overload is positive, and the velocity azimuth angle is constantly increasing. With the distance between UAV and interceptor reducing, the overload increases gradually in the opposite direction, and the velocity azimuth angle decrease. On the one hand, it can confuse the strategy of interceptor, on the other hand, the guidance course is corrected.

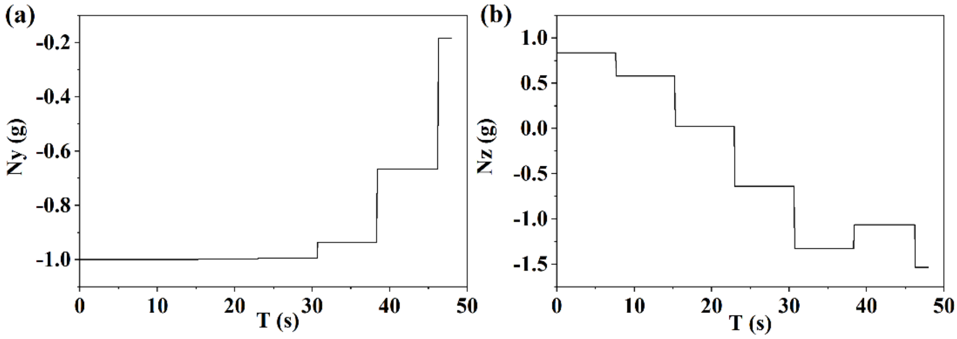

Figure 12. the maneuvering overload. (a) longitudinal direction under short distance, (b) lateral direction under short distance.

As shown in Figure 13(a), compared with the pursuit evading scene of short distance, the medium escape process takes longer, the pursuing time left to interceptor is longer, and the UAV flies under the direction of increasing the velocity slope angle. The timing of maximum escape overload corresponding to the medium distance is also different. As shown in Figure 13(b), in the first half phase of escape, lateral maneuvering overload corresponding to medimum distance is larger than that in the short distance, and in the second half phase of the escape, the corresponding reverse maneuvering overload is smaller, resulting in UAV can use the longer escape time to slowly correct the course.

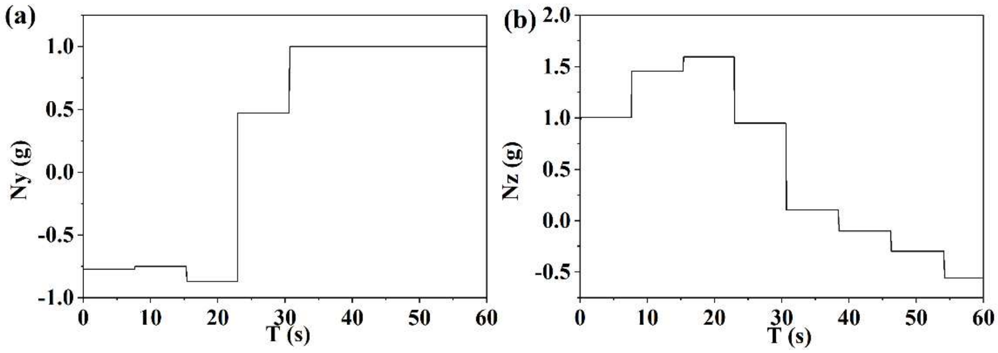

Figure 13. the maneuvering overload. (a) longitudinal direction under medimum distance, (b) lateral direction under medium distance.

As shown in Figure 14, under the long pursuit distance, the overload change of UAV maneuver is similar to that of medium range, and the escape timing is basically the same as the escape strategy.

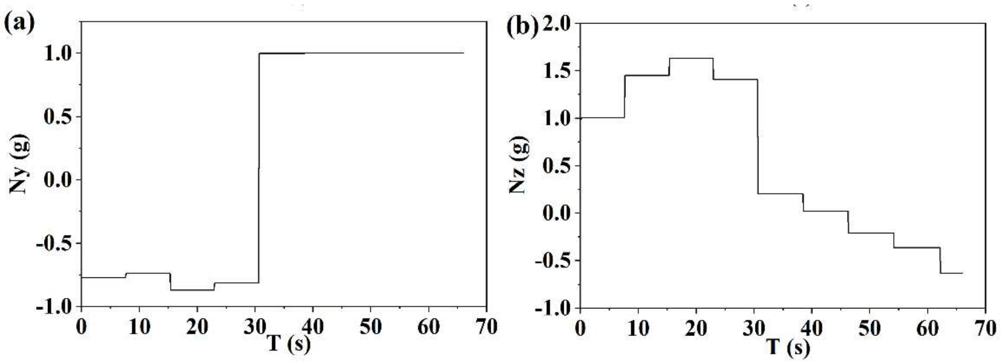

Figure 14. the maneuvering overload. (a) longitudinal direction under long distance, (b) lateral direction under long distance.

According to the above analysis, the escape guidance strategy via Meta SAC can be used as a tactical escape strategy, and the timing of escape and maneuvering overload are adjusted timely under different pursuit evading distances. On the one hand, the overload corresponding to this strategy can confuse the interceptor and cause some interference, on the other hand, it can take into account the guidance mission, correcting the course deviation caused by escape.

Figure 15(a) shows the flight trajectory of interceptor against UAV at the North East Down (NED) coordinate (10 km,30 km,30 km), the trajectory point at the encountering moment is shown in Figure 15(b), and the miss distance is 19 m in this pursuit evading scene. To verify the scientific and applicability of Meta SAC, the initial position of interceptor is changed. Flight trajectories are respectively represented shown in Figure 15(c,e), and trajectory points at the encountering moment are shown in Figure 15(d,f). The miss distances in these two pursuit evading scenarios are 3 m and 6 m respectively. Based on the CAV-H structure, the miss distance between UAV and interceptor is greater than 2 m at the encountering moment, which means the escape mission is achieved.

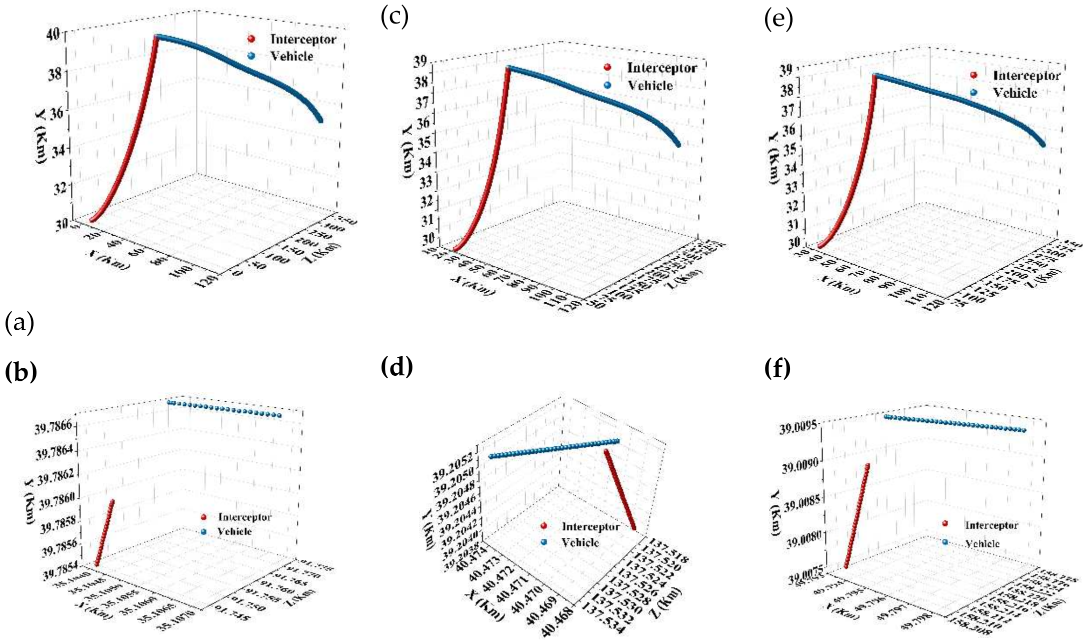

Figure 15. The ballistic flight diagrams of whole course under different pursuit evading distances, (b) and (d) and (f) represent trajectories at the encountering time under different pursuit evading distances.

doi:10.20944/preprints202308.1512.v1

Based on the principle of Meta SAC and optimal guidance, flight states are shown in the Figure 16. Longitude, latitude and altitude during flight of UAV are shown in Figure 16(a)-(b), under different pursuit evading scenarios, terminal position and altitude constraints are meet. There is larger amplitude modification in the velocity slope and azimuth angle, which is attributed to escape strategy via lateral and longitudinal maneuvering, as shown in Figure 16(c)-(d). The total change of velocity slope and azimuth angle is within two degrees, which meets flight process constraints. Through the analysis of flight states, this escape strategy is an effective measure for guidance escape with high accuracy.

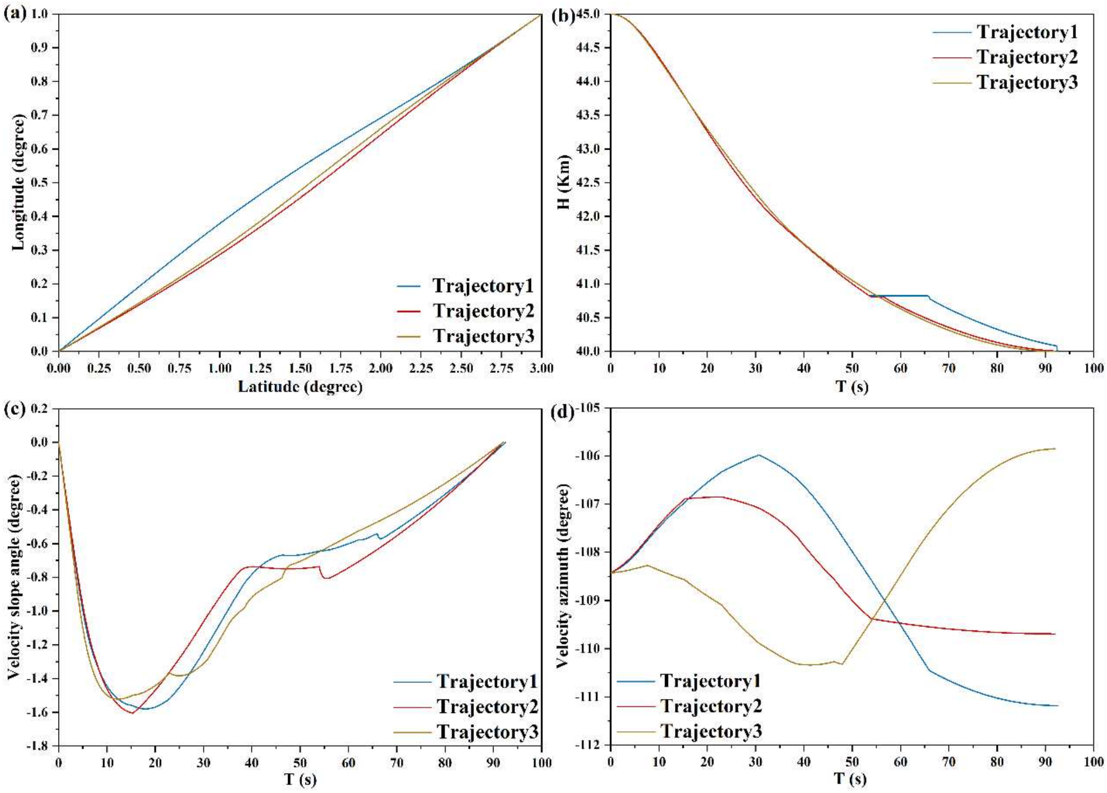

Figure 16. Flight states of UAV. (a) the longitude and latitude, (b) the height, (c) the velocity slope angle, (d) the velocity azimuth angle.

Flight process deviation mainly includes aerodynamic calculation deviation and output overload deviation. For the aerodynamic deviation, the manuscript uses interpolation method to calculate based on the flight Mach number and angle of attack, which may have some deviation. Therefore, when calculating the aerodynamic coefficient, random noise with an amplitude of 0.1 is added to verify whether the UAV can complete the guidance mission. As shown in Figure 17 (a), aerodynamic deviation noise causes certain disturbances to the angle of attack during flight. At the 10th second and end of flight, the maximum deviation of the angle of attack is 2 °. However, overall, the impact of aerodynamic deviation on the entire flight is relatively small, and the change in angle of attack is still within the safe range of the UAV. As shown in Figure 17 (b), due to the constraints of UAV game confrontation and guidance missions, the bank angle during the entire flight process changes significantly, and aerodynamic deviation noise has a small impact on the bank angle. After increasing the aerodynamic deviation noise, the miss distance between the UAV and the interceptor at the time of encounter is 8.908m, and the terminal position deviation is 0.52m. Therefore, under the influence of aerodynamic deviation, the UAV can still complete the escape guidance mission.

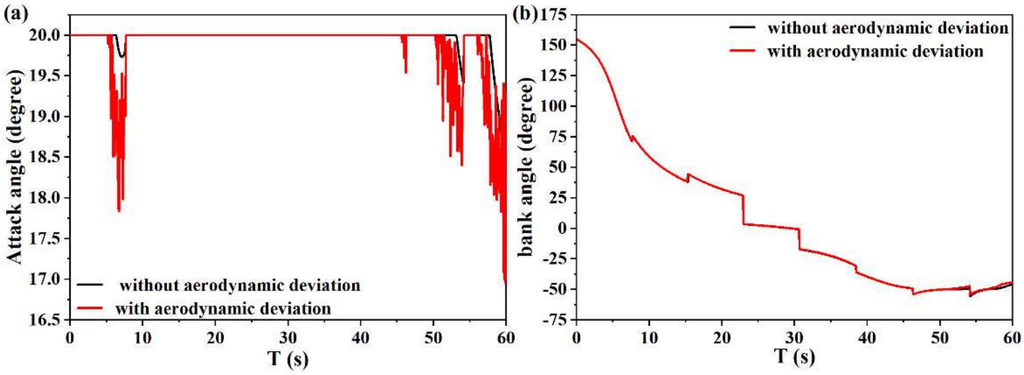

Figure 17. a) the attack angle, (b) the bank angle.

For the output overload deviation, the total overload is composed of the guiding overload derived from the optimal guidance law and the maneuvering overload output from the neural network. random maneuvering overload with an amplitude of 0.1 is added to verify whether the UAV can complete the maneuver guidance mission. As shown in Figure 18, random overloads are added in the longitudinal and lateral directions respectively. Through simulation testing, the miss distance between the UAV and the interceptor at the encounter point is 10.51m, and the terminal deviation of the UAV is 0.6m. Under this deviation, the UAV can still achieve high-precision guidance and efficient penetration.

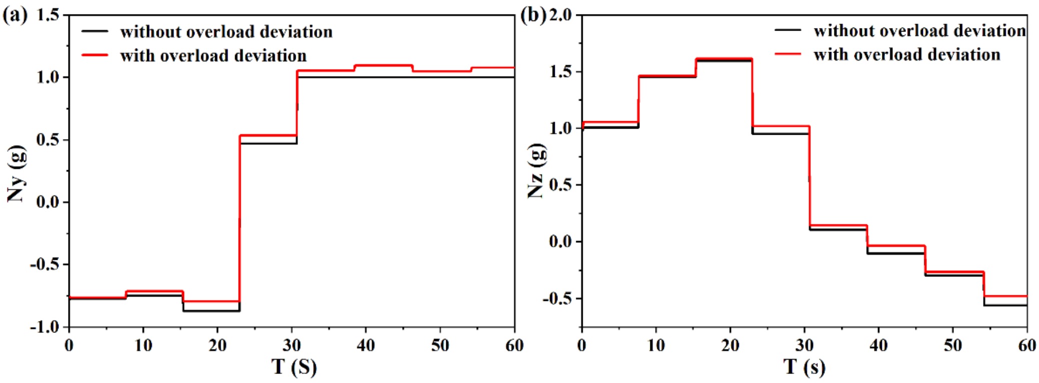

Figure 18. (a) the maneuvering overload in the longitudinal direction, (b) the maneuvering overload in the lateral direction.

7. Conclusion

The manuscript proposes escape guidance strategy satisfying terminal multiple constraints via SAC. The action space is designed under the UAV process constraint. Considering the fact that the real-time interceptor information is hard to obtain in the actual escape process, the state space considering the target and the heading angle is designed. The reward function is an important index function, which is used to guide and evaluate the training results. Based on the pursuit evading model, terminal LOS angle rates of lateral and longitudinal are derived to describe the deviation and miss distance. In order to improve the adaptability of escape guidance strategy, the manuscript improves SAC via meta learning, and compares meta SAC with SAC. The strong adaptive escape strategy based on meta SAC is analyzed. In view of the above theoretical numerical analysis, we have obtained the following conclusions.

(1) The escape guidance strategy based on SAC is a feasible tactical escape strategy, which can achieve high precision guidance under meeting the escape requirements.

(2) Meta SAC can significantly improve the adaptability of escape strategy. When the escape mission changes, it can fine-tune network parameters to adapt to mission through a small

number of training. This method provides a possibility for the UAV to learn while flying.

(3) The strong adaptive escape strategy based on meta SAC can adjust the escape timing and maneuvering overload in real time according to the pursuit evading distance. On the one hand, the overload corresponding to this strategy can confuse the interceptor and cause some interference, on the other hand, it can take into account the guidance mission, correcting the course deviation caused by escape.

Author Contributions: Conceptualization, SiBo Zhao. and JianWen Zhu.; methodology, SiBo Zhao.; software, SiBo Zhao; validation, SiBo Zhao. and JianWen. Zhu; formal analysis, SiBo Zhao; investigation, SiBo Zhao.; resources, XiaoPing Li.; data curation, HaiFeng Sun.; writing—original draft preparation, HaiFeng Sun.; writing—review and editing, XiaoPing Li.; visualization, SiBo Zhao.; supervision, WeiMin Bao. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] H. W. LI ZB, Z. J. S. ZHANG, and T. n. Review, "Summary of the Hot Spots of Near Space Vehicles in 2018," vol. 37, no. 1, p. 44, 2019.

[2] G. Li, H. Zhang, G. J. A. S. Tang, and Technology, "Maneuver characteristics analysis for hypersonic glide vehicles," vol. 43, pp. 321-328, 2015.

[3] L. Wang *et al.*, "Applications and prospects of agricultural unmanned aerial vehicle obstacle avoidance technology in China," vol. 19, no. 3, p. 642, 2019.

[4] Y. WANG, T. ZHOU, W. CHEN, T. J. J. o. B. U. o. A. HE, and Astronautics, "Optimal maneuver penetration strategy based on power series solution of miss distance," vol. 46, no. 1, p. 159.

[5] J.-W. Rim, I.-S. J. I. T. o. A. Koh, and E. Systems, "Survivability simulation of airborne platform with expendable active decoy countering RF missile," vol. 56, no. 1, pp. 196-207, 2019.

[6] F. Liu, X. Dong, Q. Li, Z. J. A. s. Ren, and technology, "Robust multi-agent differential games with application to cooperative guidance," vol. 111, p. 106568, 2021.

[7] E. Garcia, D. W. Casbeer, and M. J. I. T. o. A. C. Pachter, "Design and analysis of state-feedback optimal strategies for the differential game of active defense," vol. 64, no. 2, pp. 553-568, 2018.

[8] H. Liang, W. Jianying, W. Yonghai, W. Linlin, and L. J. C. J. o. A. Peng, "Optimal guidance against active defense ballistic missiles via differential game strategies," vol. 33, no. 3, pp. 978-989, 2020.

[9] L. Liang, F. Deng, Z. Peng, X. Li, and W. J. A. Zha, "A differential game for cooperative target defense," vol. 102, pp. 58-71, 2019.

[10] S. Liu, Y. Wang, Y. Li, B. Yan, and T. J. T. A. J. Zhang, "Cooperative guidance for active defence based on line-of-sight constraint under a low-speed ratio," pp. 1-19, 2022.

[11] D. Zhang, T. Zhang, Y. Lu, Z. Zhu, and B. J. A. i. N. I. P. S. Dong, "You only propagate once: Accelerating adversarial training via maximal principle," vol. 32, 2019.

[12] L. Ruthotto, S. J. Osher, W. Li, L. Nurbekyan, and S. W. J. P. o. t. N. A. o. S. Fung, "A machine learning framework for solving high-dimensional mean field game and mean field control

problems," vol. 117, no. 17, pp. 9183-9193, 2020.

[13] Z. Ullah, F. Al-Turjman, L. Mostarda, and R. J. C. C. Gagliardi, "Applications of artificial intelligence and machine learning in smart cities," vol. 154, pp. 313-323, 2020.

[14] H. Song, J. Bai, Y. Yi, J. Wu, and L. J. I. C. I. M. Liu, "Artificial intelligence enabled Internet of Things: Network architecture and spectrum access," vol. 15, no. 1, pp. 44-51, 2020.

[15] X. Gong, W. Chen, Z. J. A. S. Chen, and Technology, "All-aspect attack guidance law for agile missiles based on deep reinforcement learning," p. 107677, 2022.

[16] R. Furfaro, A. Scorsoglio, R. Linares, and M. J. A. A. Massari, "Adaptive generalized ZEM-ZEV feedback guidance for planetary landing via a deep reinforcement learning approach," vol. 171, pp. 156-171, 2020.

[17] Y. Yuan, G. Zheng, K.-K. Wong, and K. B. J. I. T. o. V. T. Letaief, "Meta-reinforcement learning based resource allocation for dynamic V2X communications," vol. 70, no. 9, pp. 8964-8977, 2021.

[18] L.-B. Zhao, W. Xu, C. Dong, G.-S. Zhu, and L. Zhuang, "Evasion guidance of re-entry vehicle satisfying no-fly zone constraints based on virtual goals," *Scientia Sinica-Physica Mechanica & Astronomica,* vol. 51, no. 10, 2021 2021, Art. no. 104706.

[19] Y. Guo, X. Li, H. Zhang, L. Wang, and M. Cai, "Entry Guidance With Terminal Time Control Based on Quasi-Equilibrium Glide Condition," *Ieee Transactions on Aerospace and Electronic Systems,* vol. 56, no. 2, pp. 887-896, Apr 2020.

[20] S. M. Krasner *et al.*, "Reconstruction of Entry, Descent, and Landing Communications for the InSight Mars Lander," *Journal of Spacecraft and Rockets,* vol. 58, no. 6, pp. 1569-1581, Nov-Dec 2021.

[21] Z. Huang, Y. Zhang, and Y. Liu, "Research on state estimation of hypersonic glide vehicle," in *Journal of Physics: Conference Series*, 2018, vol. 1060, no. 1, p. 012088: IOP Publishing.

[22] J. Zhu, D. Su, Y. Xie, and H. Sun, "Impact time and angle control guidance independent of time-to-go prediction," *Aerospace Science and Technology,* vol. 86, pp. 818-825, Mar 2019.

[23] C. Ni, A. R. Zhang, Y. Duan, M. Wang, and Ieee, "Learning Good State and Action Representations via Tensor Decomposition," in *IEEE International Symposium on Information Theory (ISIT)*, Electr Network, 2021, pp. 1682-1687, 2021.

[24] Y. Ma *et al.*, "Reinforcement Learning-Based Fed-Batch Optimization with Reaction Surrogate Model," in *American Control Conference (ACC)*, Electr Network, 2021, pp. 2581-2586, 2021.

[25] X. Yang *et al.*, "Learning high-precision bounding box for rotated object detection via kullback-leibler divergence," vol. 34, pp. 18381-18394, 2021.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.