sensors

MDPI

Article

Multi-UAV Cooperative Hunting in Obstructed Environments via a Multi-Agent Proximal Policy Optimization with Curriculum Learning

Longjie Zheng [1], Junlin Zhou [2], Haijun Peng [1], Bai Li [3] and Xinwei Wang [1,*]

[1] Department of Engineering Mechanics, State Key Laboratory of Structural Analysis, Optimization and CAE Software for Industrial Equipment, Dalian University of Technology, Dalian 116024, China; 1563162495@mail.dlut.edu.cn (L.Z.); hjpeng@dlut.edu.cn (H.P.)
[2] School of Mathematical Science, Dalian University of Technology, Dalian 116024, China; zhoujunlin@mail.dlut.com
[3] School of Information & Electronic Engineering, East China Normal University, Shanghai 200241, China; libai@zju.edu.cn
* Correspondence: wangxinwei@dlut.edu.cn

Abstract

With the increasing complexity of unmanned aerial vehicle (UAV) missions in complex obstacle environments, cooperative hunting of maneuvering ground targets by UAV swarms has become an important problem for multi-agent autonomous decision-making. This paper focuses on a simulated three-UAV hunting scenario in a two-dimensional obstructed environment, where UAVs must search for, approach, encircle, and continuously track a target while avoiding static obstacles under local observation. To address the problem of multi-UAV cooperative hunting of dynamic targets in complex obstacle environments, this paper proposes a curriculum learning (CL)-based Multi-Agent Proximal Policy Optimization algorithm, termed CL-MAPPO. Specifically, a three-stage progressive training curriculum is designed to overcome the challenges of low exploration efficiency, slow environmental adaptation, and difficult convergence of cooperative hunting policies faced by multi-agent deep reinforcement learning in hunting tasks, thereby gradually enhancing the cooperative hunting capability of UAVs in complex environments. Curriculum I employs fixed obstacles and a stationary target position to train the UAVs' basic obstacle avoidance and target search abilities. Curriculum II introduces randomly generated obstacles and target positions to improve the UAVs' adaptability to varying environments. Curriculum III further incorporates a dynamic target, prompting the UAVs to learn effective hunting strategies against maneuvering targets. The simulation experiment includes ablation experiments against MAPPO without curriculum learning and comparative simulations against MADDPG and MADQN, using reward convergence curves and trajectory visualizations to evaluate the training results. The results show that, under the same training episodes in the ablation experiment, CL-MAPPO reaches a higher and more stable reward level than vanilla MAPPO, indicating improved learning efficiency without increasing model complexity. In the comparative experiment, the CL-MAPPO algorithm achieved a higher success rate in cooperative hunting. These simulation experiments verify the effectiveness and superiority of the CL-MAPPO algorithm in multi-agent cooperative hunting tasks.

Keywords: UAV; cooperative hunting; noncooperative target; CL-MAPPO; obstructed environment

Check for updates

Academic Editor: Andrey V. Savkin

Received: 26 April 2026
Revised: 7 June 2026
Accepted: 17 June 2026
Published: 19 June 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

1.1. Background and Significance

With the rapid development of artificial intelligence and UAV technologies, the application of UAVs has become increasingly widespread in various fields [1–4]. For multi-agent cooperative hunting within complex environments, UAV swarms play a pivotal role by collaboratively searching, encircling, and capturing dynamic targets, thereby significantly improving mission execution efficiency. Nevertheless, the high speed and agility of UAVs, coupled with the dynamic and obstacle-rich environment, pose severe challenges for human operators, who often struggle with timely and accurate decision-making under intense cognitive load. Meanwhile, traditional decision-making approaches, such as rule-based strategies, artificial potential field (APF) methods, and optimization-based algorithms, often suffer from local minima and high computational complexity [5,6]. Similarly, finite state machines (FSM) [7] suffer from the state explosion problem and lack the reactivity required for dynamic environments, while heuristic search algorithms [8,9] struggle with real-time performance and adaptability as they heavily rely on global map information.

Consequently, there is a pressing need to investigate real-time intelligent decision-making methods for multi-agent cooperative hunting in complex environments. Deep reinforcement learning (DRL), with its capability to learn sophisticated strategies directly from high-dimensional inputs, offers a promising solution [10–12]. Integrating CL further enables agents to progressively master increasingly challenging tasks, improving both learning efficiency and policy robustness. Research on multi-agent cooperative hunting not only advances the application of UAV technology but also establishes a foundation for future intelligent and networked unmanned systems. To this end, this paper proposes a CL-MAPPO to address the cooperative hunting of noncooperative maneuvering targets in complex obstacle environments.

1.2. Related Works

1.2.1. Deep Reinforcement Learning for Multi-Agent Decision-Making

Multi-agent deep reinforcement learning (MADRL) provides a powerful framework for addressing sequential decision-making problems in cooperative hunting tasks, and existing methods can be broadly categorized into value-based and policy-based approaches.

Value-based MADRL methods extend single-agent value-based reinforcement learning to multi-agent scenarios, with the core idea being to learn the optimal joint action-value function and then derive each agent's optimal policy indirectly. To mitigate the combinatorial explosion of the joint action space, these methods decompose the joint Q-function into individual local Q-functions while leveraging centralized value networks to enhance cooperative performance. Representative algorithms include Value-Decomposition Networks (VDN) and QMIX [13,14]. VDN decomposes the joint action-value function additively into the sum of per-agent local action-value functions, i.e., $Q_{jont}(\tau, a) = \sum_i Q_i(\tau_i, a_i)$, where τ_i denotes the historical observation trajectory of agent i. This simple linear structure facilitates efficient training, but the additive assumption is overly restrictive and cannot capture complex cooperative relationships among agents, limiting its performance in challenging coordination tasks. QMIX extends VDN by introducing a mixing network that nonlinearly combines local Q-values into the joint Q-value while enforcing monotonicity constraints. This design ensures that improvements in local Q-values lead to improvements in the joint Q-value, enabling QMIX to better accommodate complex multi-agent cooperative requirements.

Policy-based MADRL methods directly model and optimize the joint policy of multiple agents, making them particularly well-suited for continuous action spaces and large-scale cooperative scenarios. Representative algorithms include MAPPO and Multi-Agent Deep

Deterministic Policy Gradient (MADDPG) [10]. MAPPO adopts centralized training, decentralized Execution (CTDE) architecture [10]. Each agent maintains an independent policy network that generates actions using only local observations, while a centralized critic network utilizes global environmental information to evaluate the value of the joint policy, computing advantage functions to guide the policy updates of all agents. This architecture is characterized by simple implementation, training stability, and strong scalability, effectively mitigating the non-stationarity problem in multi-agent environments. In contrast, MADDPG equips each agent with both an independent policy network and a centralized critic network. The policy network generates actions based solely on local observations, whereas the critic network can access the observations and actions of all agents to evaluate the value of joint actions, providing each agent with more accurate gradient information for policy improvement. This architecture has demonstrated strong performance in continuous control, cooperative navigation, and multi-agent dynamic interaction scenarios.

1.2.2. Development Status of Multi-Agent Cooperative Hunting

The problem of multi-agent cooperative hunting has been extensively investigated as a fundamental paradigm of distributed collaborative decision-making and motion control in multi-agent systems. The core objective is to achieve target encirclement, tracking, and stable hunting in dynamic and complex environments through information interaction and behavioral coordination among multiple autonomous agents. With the rapid advancement of unmanned systems and intelligent decision-making technologies, cooperative hunting methodologies are progressively evolving from traditional control strategies toward intelligent, distributed learning frameworks, forming a technological landscape centered on reinforcement learning, attention mechanisms, artificial potential fields, and distributed control [15].

The integration of attention mechanisms with reinforcement learning constitutes a pivotal pathway for enhancing the intelligent decision-making capabilities of cooperative hunting systems, and various improved algorithms have been effectively deployed across different robotic platforms. Yu et al. [11] proposed a hybrid attention-oriented experience replay method integrated with MADDPG, which enables agents to focus on critical information during the interaction process by designing a hybrid attention module that separately encodes states and actions. This approach significantly improves experience utilization efficiency and learning convergence performance. Xue et al. [16] combined multi-head self-attention mechanisms with multi-agent reinforcement learning to further strengthen the system's perception of target and environmental features, effectively accelerating the convergence speed of hunting policies and improving the overall task success rate in multi-unmanned surface vehicles (USV) scenarios. Wang et al. [17] introduced a knowledge-guided reinforcement learning method incorporating APF-based demonstrations, which enhances learning stability and decision quality through expert trajectory pre-training and real-time potential field guidance during the fine-tuning phase. This method has been successfully applied to multi-autonomous underwater vehicles (AUVs) cooperative hunting tasks. Wang et al. [18] developed a MAPPD approach with multi-head attention, effectively mitigating the challenges posed by sparse rewards and improving algorithmic adaptability in multi-evader environments. Furthermore, distributed algorithms based on decentralized Partially Observable Markov Decision Processes have achieved centralized training with decentralized execution, thereby enhancing the cooperative hunting capability of large-scale agent swarms [19]. Zhong et al. [20] proposed a cooperative hunting method based on advantage decomposition and sequential decision-making, enabling each agent to independently make decisions according to preceding individual behaviors

and current environmental information, which further improves the stability and execution efficiency of the hunting process.

The robustness and efficiency of cooperative hunting systems can be optimized through diversified control strategies and coordination mechanisms tailored to the demands of different complex scenarios. Luo et al. [21] proposed a hybrid control strategy combining artificial rules with self-learning, which employs prior rules to realize basic hunting behaviors and subsequently utilizes the Twin Delayed Deep Deterministic Policy Gradient (TD3) algorithm for adaptive optimization. This hybrid paradigm enhances the system's capability for tracking and encircling dynamic targets with stronger robustness. He et al. [22] developed a dynamic multi-target self-organization hunting control method for multi-agent systems based on fuzzy inference, particle swarm optimization, and potential field theory. This approach decomposes the multi-target hunting problem into multiple dynamic single-target sub-tasks, achieving efficient matching and cooperative hunting between agents and targets. Xu et al. [23] proposed a two-stage distributed control algorithm to address the moving target surrounding control problem under input saturation constraints, driving multi-agent systems to stably encircle dynamic targets in uniform formations. To reduce unnecessary information transmission while maintaining hunting performance, Xu et al. [24] further developed an event-triggered surrounding a formation control method for multiple dynamic targets, effectively reducing inter-agent communication overhead. Li et al. [25] investigated consensus tracking control over asynchronous cooperation-competition networks, providing theoretical support for the complex individual interaction mechanisms inherent in the cooperative hunting process.

Nevertheless, despite the extensive research efforts devoted to multi-agent cooperative hunting in recent years, MADRL-based approaches still face significant challenges, including low exploration efficiency, slow environmental adaptation, difficult convergence of cooperative policies, and high sample complexity when dealing with continuous control in high-dimensional state-action spaces [26,27].

Classical and hybrid approaches for pursuit-evasion, formation control, and obstacle avoidance are highly relevant to the cooperative hunting problem. Pursuit-evasion games have been analyzed using differential game theory and geometric methods, but these often assume perfect information and simple environments. Formation control strategies, such as leader-follower, virtual structure, and behavior-based methods, provide effective ways to maintain desired inter-UAV geometries. However, they typically rely on accurate communication and predefined patterns. Obstacle avoidance is commonly addressed via APF [5], rapidly exploring random trees (RRT), or model predictive control (MPC). Despite their effectiveness in structured settings, these traditional methods struggle with dynamic, partially observable, and obstacle-rich environments. Hybrid approaches, e.g., combining APF with reinforcement learning [17,21], have shown promise by leveraging prior knowledge to guide exploration.

To overcome the remaining challenges, including low exploration efficiency, slow environmental adaptation, and difficult convergence in complex multi-agent tasks, CL has been introduced as a progressive training strategy. CL decomposes a complex task into a sequence of sub-tasks with increasing difficulty, guiding agents to gradually acquire sophisticated behavioral patterns [28,29]. Originally developed for supervised learning, CL has been extended to deep reinforcement learning to address sparse reward and hard exploration problems. In multi-agent settings, CL helps alleviate non-stationarity by allowing agents to master simple cooperative patterns before facing challenging interactions. Several works have applied CL to multi-agent tasks. QU et al. [28] proposed an inheritance training method based on CL for multi-agent soft actor-critic (MASAC) in USV collaboration. Li et al. [29] developed an RNN-enhanced curriculum-driven algorithm for POMDPs with

limited experience. Despite these advances, the application of CL to multi-UAV cooperative hunting in obstructed environments with dynamic targets remains underexplored.

1.2.3. Research Gap and Motivation

Existing multi-agent cooperative hunting methods face several critical limitations when applied to obstructed environments with maneuvering targets: (1) Most MADRL algorithms (e.g., MADDPG, QMIX) suffer from low sample efficiency and poor convergence in high-dimensional state-action spaces with sparse rewards. (2) Traditional control-based methods (APF, MPC, RRT) lack adaptability to unpredictable obstacle configurations and dynamic target behaviors. (3) Although CL has been introduced into reinforcement learning to improve exploration efficiency and training stability, most existing studies focus on single-agent tasks or relatively simple multi-agent scenarios. For multi-UAV cooperative hunting, especially in environments containing obstacles and a dynamic target, existing curriculum-learning-based MAPPO methods still lack a systematic progressive training mechanism that transfers agents from basic obstacle avoidance to coordinated target encirclement.

To address this gap, this paper proposes a CL-enhanced MAPPO framework, named CL-MAPPO, for multi-UAV cooperative hunting in obstructed environments. The proposed method designs a three-stage curriculum according to the increasing complexity of the hunting task. Curriculum I trains the UAVs to approach a stationary target while avoiding fixed obstacles. Curriculum II introduces randomly generated obstacle layouts and target positions to improve environmental adaptability. Curriculum III further introduces a moving target, requiring the UAVs to learn coordinated pursuit and hunting under dynamic conditions. Through this progressive task decomposition, the UAVs can gradually acquire obstacle avoidance, target searching, formation maintenance, and cooperative hunting capabilities.

Therefore, the main contribution of this study lies in constructing a task-oriented CL strategy for MAPPO-based multi-UAV cooperative hunting, which improves training stability, accelerates convergence, and enhances policy generalization in complex obstacle environments. From an application perspective, the proposed framework is most beneficial for UAV swarm missions that require rapid adaptation in obstacle-rich and uncertain environments, such as cooperative surveillance, border patrol, search-and-rescue target tracking, disaster-area monitoring, and autonomous interception of mobile ground targets. Compared with purely rule-based or direct end-to-end training approaches, CL-MAPPO provides a structured way to acquire reusable skills in a controllable order: first safe obstacle avoidance and target approach, then adaptation to unseen obstacle layouts, and finally dynamic cooperative hunting. These capabilities are important for reducing manual rule design, improving training stability, and enabling UAV teams to form coordinated hunting behaviors under local observations.

1.3. Organization

The rest of this paper is organized as follows. Section 2 formulates the multi-UAV cooperative hunting problem in obstructed environments, including the kinematic modeling of UAVs and the construction of the cooperative hunting situation model. Section 3 presents the design of the CL-MAPPO algorithm, encompassing the MAPPO algorithm architecture with the CTDE framework and loss functions, the design of the state space, action space, and composite reward function, and the three-stage CL framework. Section 4 provides extensive simulation experiments, including the configuration of the training environment and hyperparameter settings, ablation experiments to analyze the impact of CL, comparative experiments against MADDPG and MADQN, and qualitative evaluation with

trajectory visualization for each curriculum stage and each compared algorithm. Finally, Section 5 concludes the paper and discusses potential directions for future research.

2. Problem Formulation and Modeling

2.1. Kinematic Modeling for UAV Cooperative Hunting

In practical UAV applications, UAVs naturally operate in three-dimensional space. However, in the cooperative hunting task considered in this paper, the target is a ground target, and all UAVs are assumed to maintain the same flight altitude during the hunting process. Under this assumption, the vertical motion of the UAVs does not directly affect the hunting process. Therefore, the original three-dimensional pursuit problem can be reasonably simplified into a two-dimensional planar motion control problem. This simplification allows the study to focus on the key challenges of cooperative hunting, including target approach, obstacle avoidance, formation coordination, and hunting stability, while reducing the dimensionality of the state and action spaces. In this task, UAVs are required to achieve effective target hunting by maintaining a specific formation configuration. Consequently, it is assumed that each UAV possesses a consistent initial flight heading and that the initial configuration adheres to a standard formation geometry. This paper takes the cooperative hunting of three UAVs as the research object, selects an equilateral triangle as the standard formation configuration, and defines its side length as the target hunting radius.

A two-dimensional cooperative hunting kinematic model for UAVs is established, in which the state of a UAV comprises its position, velocity, and heading angle, as illustrated in Figure 1.

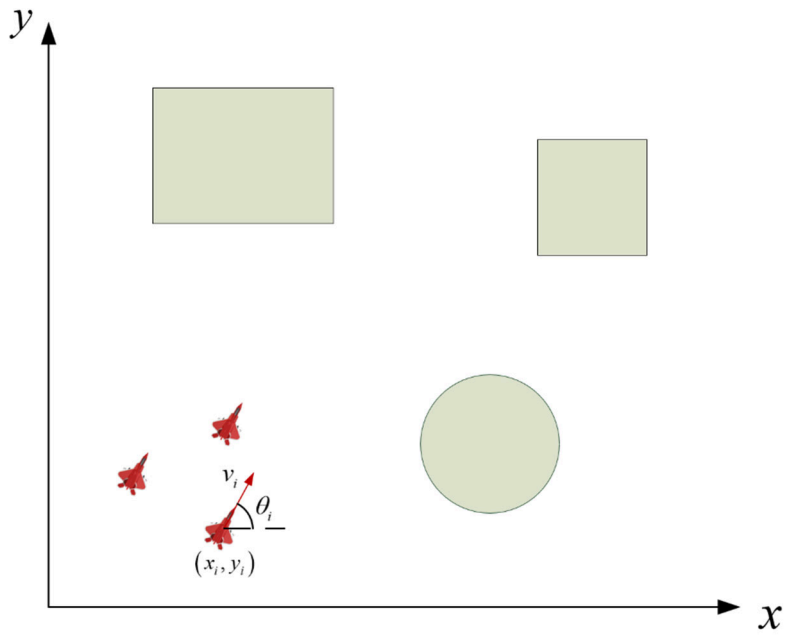


Figure 1. The UAV motion description is in a two-dimensional plane.

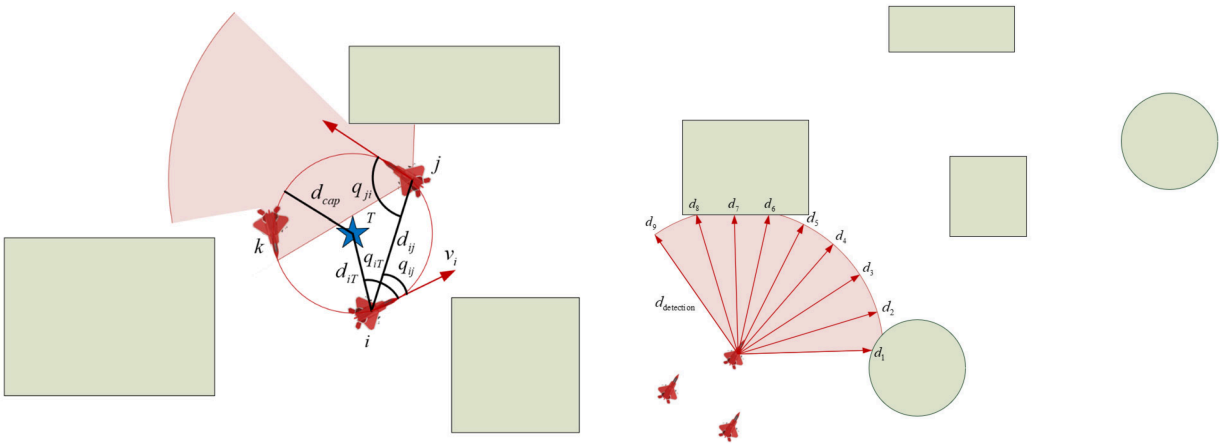
Let the position of UAV i in the ground inertial coordinate system be denoted as $P_i = [x_i, y_i]^T$, the heading angle ψ_i be defined as the angle between the velocity vector and the positive x-axis, and the velocity magnitude be v_i. The kinematic equations of the UAV can be expressed as follows:

$$\begin{cases} \dot{x}_i = v_i \cos\theta_i \\ \dot{y}_i = v_i \sin\theta_i \\ \dot{v}_i = a_{v,i} \\ \dot{\theta}_i = \omega_i \end{cases} \tag{1}$$

where $a_{v,i}$ denotes the tangential acceleration of the UAV, adjusted through engine thrust to achieve acceleration or deceleration, and ω_i represents the heading angular velocity, achieved through coordinated aileron and rudder maneuvers. UAVs must maintain a certain forward velocity to sustain lift. Hence, a minimum speed constraint $v_{\min}$ exists, and hovering or reverse flight maneuvers are not feasible. Due to aerodynamic constraints, the maximum heading angular velocity is limited by the maximum overload, expressed as $|\omega_i| \leq \omega_{\max}$. Limited by engine thrust, the tangential acceleration satisfies $a_{\min} \leq a_i \leq a_{\max}$, where $a_{\max}$ represents the maximum acceleration capability and $a_{\min}$ denotes the maximum deceleration capability.

2.2. Cooperative Hunting Situation Model Construction

The cooperative hunting situation model for UAVs in an obstructed environment is illustrated in Figure 2.



(**a**) (**b**)

Figure 2. Cooperative hunting situational model in an obstructed environment. (**a**) Hunting model of the UAV. (**b**) Detection model of the UAV (nine radar rays).

The flight direction vector of the UAV is defined as $\boldsymbol{v}_i = [\cos\psi_i \quad \sin\psi_i]$. The relative position vectors among UAVs are defined as $\boldsymbol{d}_{ij} = P_j - P_i$. The relative position vector between UAV i and the target is defined as $\boldsymbol{d}_{iT} = P_T - P_i$.

The transformation relationship between relative situation information and the ground inertial coordinate system is described as follows:

$$\begin{cases} d_{ij} = |\boldsymbol{d}_{ij}| \\ d_{iT} = |\boldsymbol{d}_{iT}| \\ \varphi_{ij} = \arccos(\boldsymbol{v}_i \cdot \boldsymbol{d}_{ij}/d_{ij}) \\ \varphi_{iT} = \arccos(\boldsymbol{v}_i \cdot \boldsymbol{d}_{iT}/d_{iT}) \end{cases} \tag{2}$$

where d_{iT} denotes the relative distance between UAV i and target, d_{ij} denotes the relative distance between UAV i and UAV j. φ_{ij} represents the antenna train angles (ATA) between UAV i and UAV j, defined as the angle between the velocity vector $\boldsymbol{v}_i$ and the relative position vector $\boldsymbol{d}_{ij}$. And φ_{iT} represents the ATA between UAV i and the target, defined as the angle between the velocity vector $\boldsymbol{v}_i$ and the relative position vector $\boldsymbol{d}_{iT}$. The hunting range of the UAV is illustrated in Figure 2a, assuming a maximum hunting distance of d_{cap}.

During the cooperative hunting process, each UAV detects its surroundings and acquires distance information relative to obstacles. As illustrated in Figure 2b, the UAV emits nine radar rays within its detection range, spaced at 15° intervals in a plane parallel to the ground. The central ray is aligned with the UAV's velocity vector $\boldsymbol{v}_i$. When no

obstacle is present along a given ray, the maximum detection distance $d_{\text{detection}}$ is returned. Let $d_1^i,\ d_2^i, \cdots,\ d_9^i$ denote the measured distances along each of the nine ray directions, where $d_x^i \in [0, d_{\text{detection}}]$. The normalized obstacle distance information is then defined as $D_1^i,\ D_2^i, \cdots,\ D_9^i$, with each component given by:

$$D_x^i = \frac{d_x^i}{d_{\text{detection}}}, (x = 1, 2 \cdots, 9) \tag{3}$$

the value of D_x^i close to 1 indicates that the UAV maintains a safe separation from obstacles in the corresponding direction, whereas $D_x^i = 0$ signifies an imminent collision.

3. CL-MAPPO Algorithm Design

3.1. Partially Observable Markov Decision Process

The problem of multi- UAV cooperative hunting can be modeled as a POMDP, represented by the tuple $(S, O, A, P, Z, R, \gamma)$. Where S denotes the global state space, encompassing information such as positions and velocities of all UAVs, ground targets, and obstacles. $O = [o^1, \ldots, o^n]$ denotes the joint observation space, where o^i is the local observation of UAV i. $A = [a^1, \ldots, a^n]$ denotes the joint action space. $P : S \times A \times S \to [0, 1]$ is the state transition probability function. Z is the observation function, indicating the probability of receiving observation O after executing action A in state S'. $R = [r^1, \ldots, r^n]$ represents the reward function for each agent, and $\gamma \in [0, 1]$ is the discount factor.

At time step t, each UAV selects an action a_t^i based on its local observation o_t^i and policy $\pi_{\theta i}(a_t^i | o_t^i)$, forming a joint action $A_t = \left[a_t^i, \ldots, a_t^n\right]$. The environment updates to the next state s_{t+1} according to the state transition probability P, and returns an immediate reward r_t^i. The cumulative discounted return for UAV i is $R_i = \sum_{t=0}^{\infty} \gamma^t r_t^i$. The optimization objective is to find the optimal policy π_θ^* that maximizes the expected cumulative return.

3.2. MAPPO Algorithm

In the task of multi-UAV cooperative hunting of a ground target, UAVs must make decisions based on local observations in dynamic environments while cooperating with other UAVs to form a hunting posture. To address this problem, this section adopts the MAPPO algorithm as the core decision-making framework. The MAPPO algorithm is based on a CTDE architecture, utilizing global information to guide policy optimization during the training phase and relying solely on local observations for decision-making during the execution phase. This effectively addresses the non-stationarity issues prevalent in multi-agent environments.

The MAPPO algorithm is an extension of the Proximal Policy Optimization (PPO) algorithm to the multi-agent domain and adopts an Actor-Critic architecture. In this setup, the Actor network is responsible for generating action policies, while the Critic network evaluates state values. Unlike independent learning PPO algorithms, MAPPO inputs global state information into the Critic network during training, enabling a more accurate assessment of joint action values. During execution, each agent makes decisions independently through its Actor network based solely on local observations.

Figure 3 illustrates the MAPPO-based training framework for cooperative hunting by multiple UAVs. During training, agents output actions based on local observations, the environment provides rewards and updates states, and trajectory data is stored in an experience replay buffer for subsequent network parameter optimization.

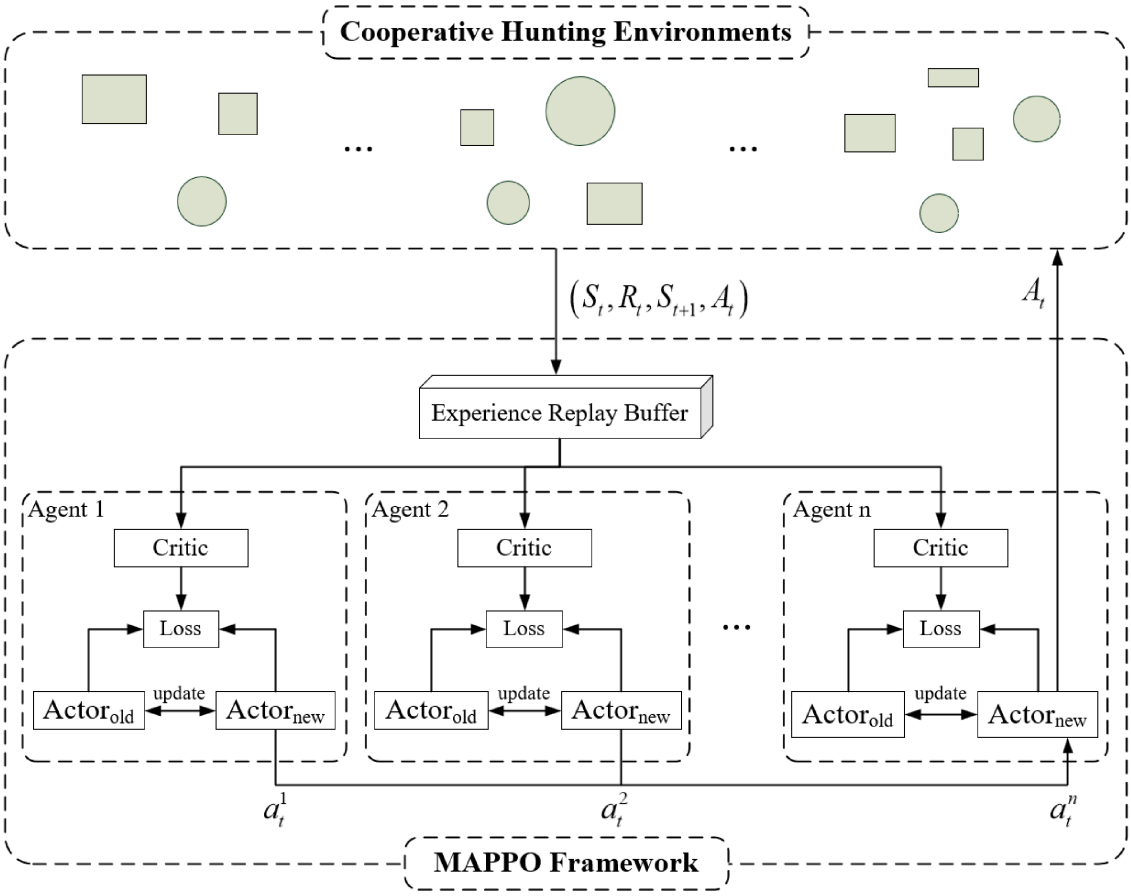


Figure 3. MAPPO-based training framework for cooperative hunting of multiple UAVs.

The optimization objective of the MAPPO algorithm comprises two components: the policy network loss and the value network loss. The policy network employs a Truncated importance sampling method to ensure stable policy updates while improving sample efficiency.

For agents within the UAVs, the loss function of the policy network is defined as:

$$L^{CLIP}(\theta) = \frac{1}{BN}\sum_{i=1}^{B}\sum_{k=1}^{N} min\left(r_{\theta,i}^{(k)} A_i^{(k)}, clip\left(r_{\theta,i}^{(k)}, 1-\varepsilon, 1+\varepsilon\right) A_i^{(k)}\right) \tag{4}$$

where B is the batch size, $r_{\theta,i}^{(k)} = \frac{\pi_\theta(a_i^{(k)}|o_i^{(k)})}{\pi_{\theta_{\text{old}}}(a_i^{(k)}|o_i^{(k)})}$ is the importance sampling weight, $A_i^{(k)}$ is the estimated advantage function, and ε is the clipping hyperparameter, controlling the maximum step size of policy updates.

The advantage function is computed using Generalized Advantage Estimation (GAE):

$$A_i^{(k)} = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \delta_{t+l}^{(k)} \tag{5}$$

where $\delta_t^{(k)} = r_t^{(k)} + \gamma V_\phi(s_{t+1}) - V_\phi(s_t)$ is the temporal-difference error, and λ is the GAE parameter balancing bias and variance.

The loss function of the value network also adopts a clipping approach to prevent excessive value estimation updates:

$$L^{CLIP}(\phi) = \frac{1}{BN}\sum_{i=1}^{B}\sum_{k=1}^{N} max\left(\begin{array}{l}\left(V_\phi\left(s_i^{(k)}\right) - \hat{R}_i^{(k)}\right)^2, \\ \left(clip\left(V_\phi\left(s_i^{(k)}\right), V_{\phi old}\left(s_i^{(k)}\right) - \epsilon, V_{\phi old}\left(s_i^{(k)}\right) + \epsilon\right) - \hat{R}_i^{(k)}\right)^2\end{array}\right) \tag{6}$$

where $\hat{R}_i^{(k)} = A_i^{(k)} + V_\phi(s_i^{(k)})$ is the estimated discounted return.

The total loss function is a weighted sum of the policy loss and value loss, augmented with a policy entropy regularization term to encourage exploration:

$$L(\theta, \phi) = L^{CLIP}(\theta) - c_1 L^{CLIP}(\phi) + c_2 H(\pi_\theta) \tag{7}$$

where $H(\pi_\theta)$ is the policy entropy, and c_1, c_2 are weight coefficients.

The pseudocode for the MAPPO algorithm employed in this study is as follows (Algorithm 1):

Algorithm 1. The framework of MAPPO

```
Initialize network parameters;
Initialize experience replay buffer R;
For episode = 1 to max episode do
|  Initialize environment, obtain initial global state S_0, local observations O_0;
|  For step = 1 to T do
|  |  For each UAV i do
|  |  |  Select action a_t^i according to current policy π_θ(·|o_t^i);
|  |  End For
|  |  Execute joint action a_t, reward r_t, next state s_{t+1} observations o_{t+1}^i;
|  |  Store trajectory data {s_t, a_t, r_t, s_{t+1}} experience replay buffer R;
|  |  Update state s ← s_{t+1};
|  End For
|  Sample a batch of data from R for training;
|  Compute advantage estimates Â_t^(k) and discounted returns R̂_t^(k) using GAE
|  Compute policy gradient and update network parameters;
|  Clear replay buffer R;
End For
Save policy network
```

3.3. State Space

The state feature o^i of UAV i comprises local features o^i_{loc}, target features o^i_{tar} and ally features o^i_{ally}.

Local features $o^i_{loc} = \{v_i, \theta_i, D^i_1, D^i_2, \cdots, D^i_9\}$ describe the motion state of UAV i itself and the obstacle detection information. It should be noted that a minimum speed constraint $v_{\min}$ applies to UAVs to prevent stalling. Target features $o^i_{tar} = \{\varphi_{iT}, d_{iT}\}$ describe the observation information of UAV i regarding the target. Ally features $o^i_{ally} = \{q_{ij}, q_{ki}, d_{ij}, d_{ki}\}, (i \neq j \neq k)$ describe the situation information between UAV i and allied UAVs.

Consequently, the complete state feature of UAV i is represented as $o^i = \left\{o^i_{loc}, o^i_{tar}, o^i_{ally}\right\}$.

3.4. Action Space

In the cooperative hunting task, the UAV acts as a continuous motion rigid body system, whose complete dynamic behavior is typically governed by the coupling of multiple aerodynamic control surfaces, such as thrust, elevator, aileron, and rudder. However, to

reduce the action dimension and improve policy search efficiency within the reinforcement learning framework, this study simplifies the UAV control input to a two-dimensional continuous action vector $[a_{v,i}, \omega_i]$. Where $a_{v,i}$ represents the tangential acceleration command along the longitudinal axis of UAV i, and ω_i denotes the yaw rate command about the vertical axis.

3.5. Reward Function

The reward function is designed to guide the UAVs to complete the cooperative hunting task in a progressive and stable manner. Specifically, the distance reward encourages each UAV to approach the target. The angle reward guides the UAVs to form an approximately equilateral hunting structure around the target. The obstacle avoidance reward penalizes unsafe proximity to obstacles. The formation consistency reward maintains reasonable distances among UAVs. The success reward provides a large positive signal when the target is successfully captured, and the step penalty encourages the UAVs to complete the task efficiently.

The composite reward is therefore not only used to maximize the hunting success rate, but also to balance multiple task requirements, including safety, cooperation, formation stability, and efficiency. This reward design is consistent with the progressive curriculum structure, allowing the UAVs to first learn basic obstacle avoidance and target approach behaviors, and then gradually acquire more complex cooperative hunting strategies.

1. Distance reward

The distance reward is designed to encourage UAVs to continuously approach the target. Let the distance between UAV i and the target at time t be $d_{iT,t}$, and the hunting radius be d_{cap}. The distance reward $r_{dist,i}$ is defined as:

$$r_{dist,i} = \begin{cases} d_{iT,t-1} - d_{iT,t}, & d_{iT,t} > d_{\text{cap}} \\ 1, & d_{iT,t} \leq d_{\text{cap}} \end{cases} \tag{8}$$

this design encourages UAVs to prioritize approach speed at long distances while focusing on maintaining position once within the hunting radius.

2. Angle and formation reward

In order to achieve a uniform hunting of the target, the three UAVs are expected to maintain an angular separation of approximately 120° relative to the target. Accordingly, an angle reward r_{ang} is introduced, which is derived from the interior angles of the triangle formed by the current positions P_1, P_2, P_3 of the three UAVs:

$$r_{\text{ang,i}} = 1 - \frac{\left|\angle P_1 - 60^\circ\right| + \left|\angle P_2 - 60^\circ\right| + \left|\angle P_3 - 60^\circ\right|}{180^\circ} \tag{9}$$

where the reward function measures the deviation of each interior angle from 60°, thereby encouraging the UAVs to form a near-equilateral triangular formation around the target, which corresponds to an ideal symmetric encirclement.

Furthermore, to prevent formation failure due to excessive or insufficient spacing between UAVs, a formation reward $r_{\text{form},i}$ is introduced:

$$r_{\text{form},i} = -\mathbb{I}\{d_{near} > 1.25 d_{\text{cap}}\} - \mathbb{I}\{d_{near} < 0.25 d_{\text{cap}}\} \tag{10}$$

where d_{near} is the distance between UAV i and its adjacent UAVs, $\mathbb{I}\cdot$ is the indicator function. This reward encourages UAVs to maintain reasonable separation distances.

3. Obstacle avoidance reward

UAVs must avoid obstacles and remain within boundaries. The environment detects surrounding obstacles via 9 rays, returning a normalized distance vector $D_x^i \in [0,1](x = 1,\ldots,9)$. The obstacle avoidance reward $r_{obs,i}$ consists of detection distance, reward and collision penalty.

The detection distance reward encourages UAVs to stay away from obstacles, with obstacles located in front of the UAV incurring a greater penalty than those to the sides:

$$r_{detect,i} = -\exp\left(-\left(\sum_{x=1}^{9} D_x^i \cos(\alpha_x) / \sum_{x=1}^{9} \cos(\alpha_x)\right)\right) \tag{11}$$

where α_x denotes the angle between the x-th radar ray and the primary heading direction of the UAV.

A collision penalty is applied upon collision with an obstacle:

$$r_{collision,i} = \begin{cases} -100, & if \ \ collision\ occurs \\ 0 & else \end{cases} \tag{12}$$

The composite obstacle avoidance reward is $r_{obs,i} = r_{detect,i} + r_{collision,i}$.

4. Success reward and step penalty

Hunting is considered successful when the following conditions are simultaneously met for all three UAVs: the distance between each UAV and the target is less than the hunting radius d_{cap}, the target is located inside the triangle formed by the three UAVs, and the three UAVs have circled the target for three complete orbits. Upon successful triggering, a substantial one-time reward $r_{success} = 300$ is granted, and the current episode terminates. Additionally, to encourage rapid task completion, a minor time penalty is applied at each step:

$$r_{\text{step}} = -0.05 \cdot \Delta t \tag{13}$$

where Δt is the simulation step size.

5. Comprehensive reward

The comprehensive reward $r_{i,t}$ for the UAV i is defined as the following linear combination:

$$r_{i,t} = \left(0.25 \cdot r_{dist,i} + 0.25 \cdot r_{\text{ang},i} + 0.25 \cdot r_{\text{form},i} + 0.25 \cdot r_{obs,i}\right) \cdot \Delta t + r_{success} + r_{\text{step}} \tag{14}$$

where weights balance objectives, including target approach, obstacle avoidance, formation keeping, and successful task completion. This reward function provides dense and effective learning signals for the multi-agent reinforcement learning algorithm, guiding the UAV swarm to collaboratively accomplish the hunting task in complex, obstructed environments.

3.6. CL Framework

In complex cooperative hunting tasks, the high dimensionality of the state-action space and the highly dynamic nature of the environment often lead to prolonged training cycles, convergence difficulties, and suboptimal policy performance during direct training. To address this, a CL strategy based on progressive environmental difficulty is designed in this section, comprising three stages, as shown in Figure 4.

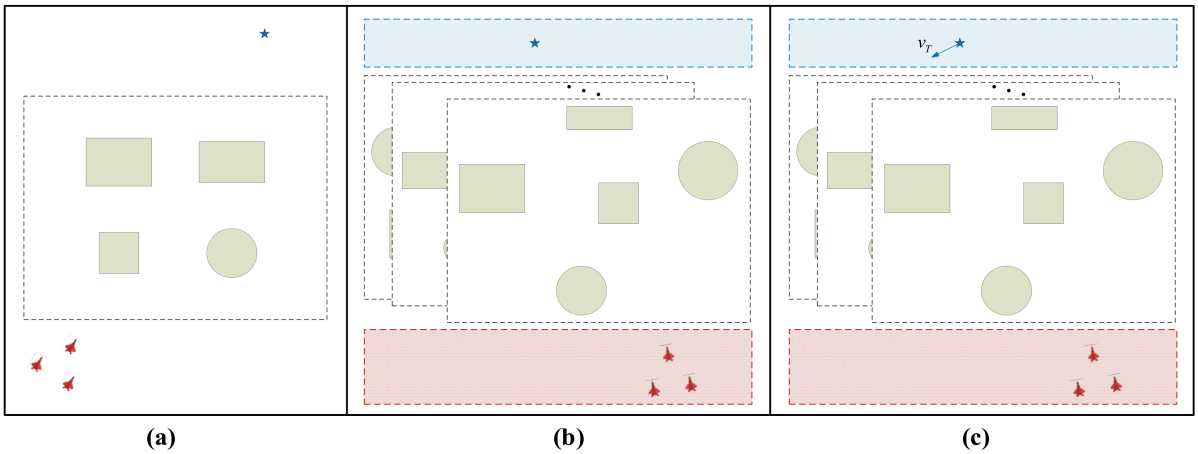


Figure 4. Three-stage CL environment settings. (**a**) Curriculum I: fixed obstacles and a stationary target. (**b**) Curriculum II: random obstacle positions and counts, random target position. (**c**) Curriculum III: adds random target movement.

Figure 4a illustrates the environment setup for Curriculum I. This stage employs fixed spatial distributions of obstacles to simulate simplified mountainous or canyon terrain. The target position is fixed, and the hunting UAVs are initialized randomly within a small range. The training objective is for UAVs to master fundamental obstacle avoidance, target search, and hunting capabilities, laying the groundwork for subsequent complex tasks.

Figure 4b illustrates the environment setup for Curriculum II. In this stage, the positions and quantities of obstacles are randomly generated to simulate more variable mountainous or canyon terrains. The target position is randomized, and the hunting UAVs are initialized randomly over a larger area. This setup aims to enhance the UAVs' adaptability and decision-making capabilities for hunting in complex, obstructed environments.

Figure 4c illustrates the environment setup for Curriculum III. Building upon Curriculum II, this stage further increases task difficulty by endowing the target with simple random mobility. The target velocity is assumed to be half the maximum flight speed of the UAVs. Obstacle distribution and UAV initial positions remain randomized. This environment compels the agents to learn effective hunting strategies against dynamic targets while simultaneously managing complex static obstacles, thereby enhancing their task execution capabilities in realistic scenarios.

This study integrates CL with MAPPO to propose the CL-MAPPO algorithm. Figure 5 illustrates the UAV cooperative hunting decision-making model in obstructed environments based on the CL-MAPPO algorithm. The operational flow of the model is as follows: The Actor network generates actions for the UAVs, which interact with the environment based on these actions. The resulting transition $\{S_t, A_t, R, S_{t+1}\}$ is stored in an experience replay buffer for subsequent network training. During training, the maneuvering strategy acquired from a given curriculum serves as the initial policy for the subsequent curriculum. Upon completing the training across all curricula, a high-quality cooperative hunting policy is ultimately obtained. The overall training workflow of the CL-MAPPO algorithm for multi-agent cooperative hunting decision-making is depicted in Figure 5.

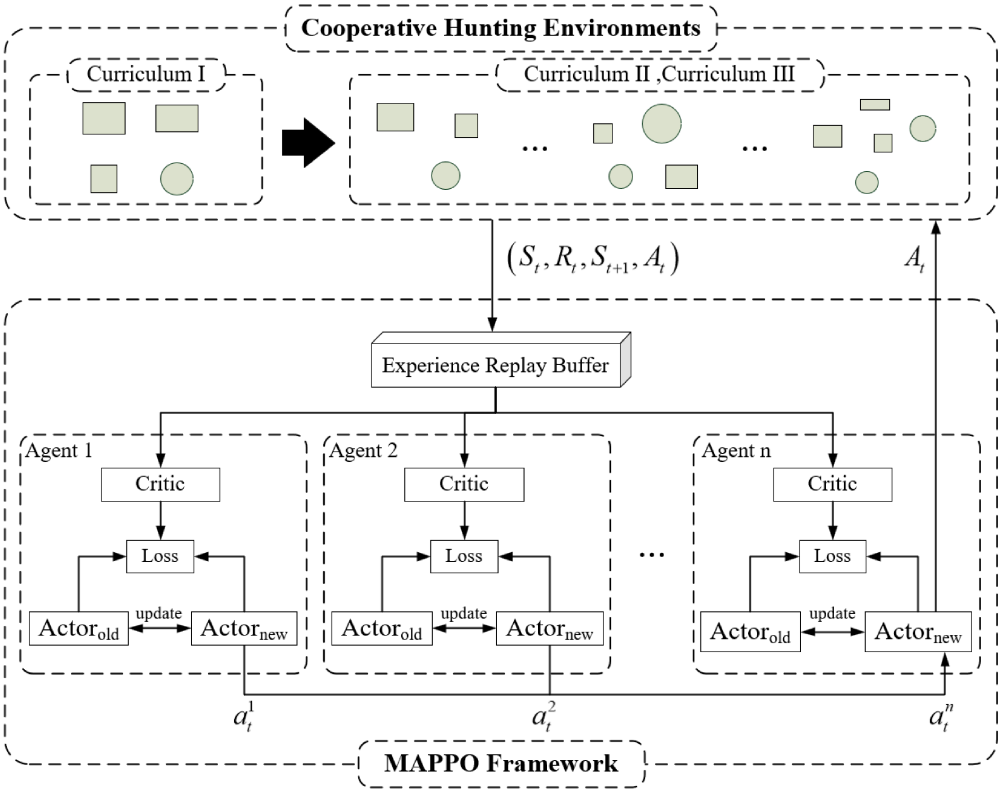


Figure 5. CL-MAPPO-based decision model for cooperative hunting in an obstructed environment.

4. Experimental Design and Results Analysis

4.1. Simulation Environment and Parameter Settings

The environmental settings for Curriculum I, Curriculum II, and Curriculum III for the cooperative hunting task are presented in Table 1, where $(x_1, y_1, v_1, \theta_1)$ denotes the coordinates, velocity, and heading angle of the UAV initially positioned closest to the target. Throughout the training process, the UAV's minimum speed $v_{\min}$ is set to 50, the maximum speed $v_{\max}$ is set to 150, the hunting radius d_{cap} is 300, and the maximum detection distance $d_{\text{detection}}$ is 800.

Table 1. Cooperative hunting environment settings.

Symbol	Curriculum I	Curriculum II	Curriculum III
x_1	500	$1000 \le x_1 \le 4000$	
y_1	900	$500 \le y_1 \le 1000$	
x_T	4000	$1000 \le x_T \le 4000$	
y_T	4000	$4000 \le y_T \le 4500$	
θ_1	$\pi/4$	$[\pi/4, 3\pi/4]$	
v_T	0		$v_{\max}/2$
v_1	100		

The number of training episodes for each curriculum was set to 2000. This choice was based on empirical observations from preliminary experiments: in Curriculum I, the UAVs began to obtain successful hunting experiences at approximately 1000 episodes, after which the cumulative reward steadily increased and converged around 2000 episodes. Therefore, 2000 episodes per stage ensures sufficient convergence while maintaining reasonable training cost. Adaptive or performance-based curriculum transitions (e.g., triggering the

next curriculum after a sustained reward threshold is reached) are promising directions for future work. In this study, a fixed-episode transition strategy is adopted to ensure experimental consistency and comparability.

The algorithm was implemented in Python 3.9, and all training and simulation experiments were conducted on a computer equipped with an Intel(R) Core(TM) i9-14900KF CPU and an NVIDIA GeForce RTX 4070 GPU. The hyperparameter settings for the algorithm are listed in Table 2. To ensure a fair comparison, CL-MAPPO and the baseline algorithms are trained under the same network architecture and basic training settings whenever applicable. The discount factor, learning rates, optimizer and hidden layer size are kept consistent across different experimental groups.

Table 2. Hyperparameter settings.

Hyperparameter	Symbol	Value
Discount factor	γ	0.99
Number of hidden layers	/	2
Hidden layer size	/	128
Optimizer	/	Adam
Policy network learning rate	/	1×10^{-4}
Value network learning rate	/	1×10^{-3}
Batch size	Bs	20
Rollout episodes	/	15
Clipping parameter	ε	0.2

4.2. Ablation Experiment

To investigate the effectiveness of CL, an ablation experiment was conducted by comparing the proposed CL-MAPPO algorithm with the baseline MAPPO without CL. As shown in Figure 6, both CL-MAPPO and MAPPO were trained for 6000 episodes. Since CL only changes the task difficulty progression without modifying any training hyperparameters, the training time is approximately the same for both algorithms. The faster convergence of CL-MAPPO compared to vanilla MAPPO can be attributed to the curriculum-based task decomposition. In Curriculum I, the task difficulty (fixed obstacles, stationary target) is substantially lower than the original full task. This allows the agents to quickly learn fundamental skills such as obstacle avoidance and target approach, without being overwhelmed by simultaneous challenges. The policy acquired in this easy stage provides a warm start for the next curriculum, effectively reducing the initial exploration burden. Consequently, CL-MAPPO reaches higher reward levels earlier than MAPPO, which must solve the full complex task from scratch. Notably, a temporary reward drop occurs at each curriculum transition (e.g., from Curriculum I to II). This phenomenon is expected and instructive. The policy learned in Curriculum I is tailored to a simpler environment (fixed obstacles, stationary target). When the same policy is applied to Curriculum II (random obstacles/positions, dynamic target), its performance degrades because the new task demands greater environmental adaptability. The subsequent recovery of reward indicates that the agents are able to adapt the previously learned skills to the more challenging setting. This demonstrates that while the policy is not directly transferable without loss, it serves as a valuable initial point that accelerates re-convergence compared to learning from a random policy. Thus, within the same training time, incorporating CL enables CL-MAPPO to achieve faster convergence, higher cumulative rewards, and a more robust final hunting policy than MAPPO without CL.

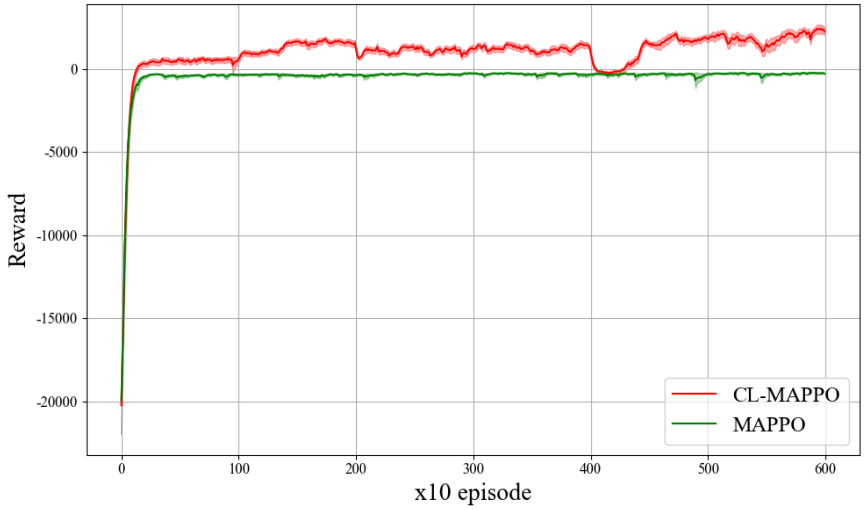


Figure 6. Reward convergence curves for CL-MAPPO and MAPPO algorithms.

4.3. Comparative Experiments

To further evaluate the effectiveness of the proposed algorithm, comparative experiments were conducted among CL-MAPPO, MADDPG, and MADQN. The cumulative reward per episode during training for each algorithm is shown in Figure 7.

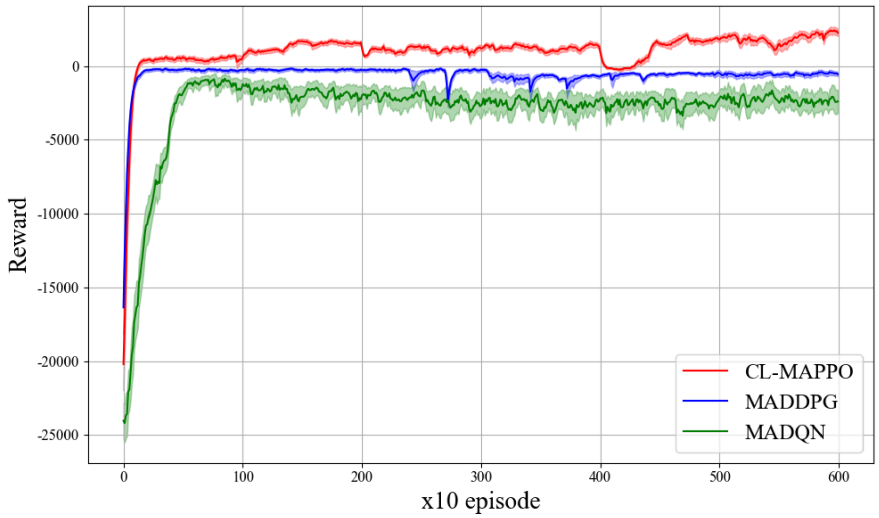


Figure 7. Reward convergence curves for different algorithms (CL-MAPPO, MADDPG, MADQN).

As illustrated by the reward convergence curves in Figure 7, the CL-MAPPO algorithm maintains the highest cumulative reward level throughout the training phase. Moreover, the reward value steadily increases and ultimately converges to a significantly positive range as training progresses. This result indicates that the CL-MAPPO algorithm can learn an effective and stable cooperative hunting strategy, successfully achieving the task target. In contrast, under the same task environment, the cumulative rewards for the MADDPG and MADQN algorithms consistently remain below zero until convergence, suggesting that the strategies learned by these algorithms in complex environments are virtually incapable of successfully capturing the target. Specifically, the CL-MAPPO algorithm benefits from a phased progressive training mechanism based on CL. Although each curriculum transition is accompanied by a temporary drop in reward, CL-MAPPO can rapidly re-converge based on the prior guidance from CL, ultimately acquiring a generalizable policy for handling complex dynamic scenarios. Conversely, due to the lack of a phased guidance mechanism, the MADDPG and MADQN algorithms directly confront highly complex task environments

from the initial stage, making their policies prone to becoming trapped in local optima or stagnation, with post-convergence rewards remaining negative. This stark contrast underscores the significant advantage of the CL strategy in enhancing the convergence efficiency and final decision-making performance of multi-agent reinforcement learning for cooperative hunting tasks.

To further validate the actual hunting performance of the trained policies, the network models saved upon completion of each curriculum stage were selected for qualitative evaluation and trajectory visualization in the test environment, with results shown in Figure 8.

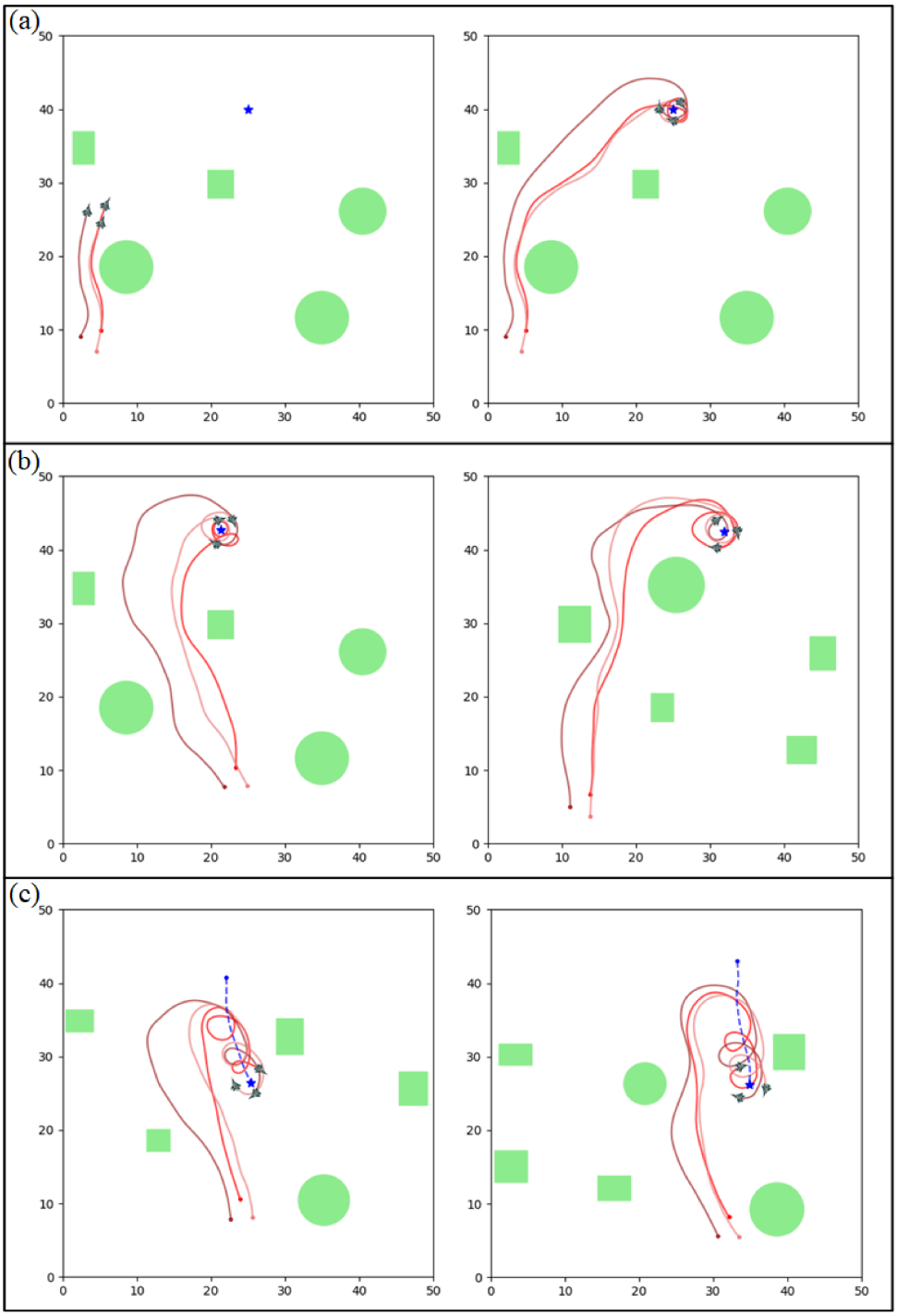


Figure 8. Trajectory visualization of UAV hunting performance after each curriculum stage. (**a**) Curriculum I. (**b**) Curriculum II. (**c**) Curriculum III.

Figure 8a presents the hunting performance after training in Curriculum I. At this stage, the UAV formation can stably maintain its configuration in a fixed obstacle environment, effectively avoid obstacles, and perform search and hunting tasks against a stationary target. The policy acquired in this stage possesses fundamental behavioral capabilities such as obstacle avoidance flight, target search, and formation encirclement.

During the Curriculum II training phase, the environment features complex and variable obstacle distributions, with target positions initialized randomly. The trajectories shown in Figure 8b demonstrate that the UAV formation can flexibly adapt to diverse terrains and obstacle layouts, safely and accurately completing the hunting task for random fixed targets while maintaining a relatively intact formation structure. Training at this stage significantly enhances the decision-making adaptability of the UAV policy in complex, obstructed environments.

Curriculum III training further introduces random maneuvering capabilities of the target within the complex obstacle environment. As shown in Figure 8c, the UAV formation can maintain its configuration, continuously tracking and encircling the moving target, and safely completing the hunting task across various complex, obstructed environments. Observing the specific morphology of the hunting trajectories, the UAV formation tends to approach the hunting circle from the target's flank, subsequently forming a loitering posture around the target to complete the hunting. This phenomenon correlates with the maneuvering characteristics of the UAVs employed. In summary, through the three-stage progressive curriculum training, the resulting UAV hunting policy exhibits the capability to handle complex obstacle scenarios and efficiently hunt dynamic maneuvering targets, thereby validating the effectiveness and practicality of the CL framework in complex tasks.

The obstacle configurations in curriculum III are randomly generated at the beginning of each episode. Specifically, both the positions and the number of obstacles vary across episodes. The UAVs have no prior knowledge of the obstacle layout and can only perceive nearby obstacles through their onboard radar observations. Therefore, each episode presents a previously unseen environment to the UAVs.

This randomization mechanism continuously exposes the agents to different obstacle layouts and obstacle densities during training and evaluation. As a result, the learned policy is not limited to a fixed obstacle map, but is trained to adapt to diverse and unseen obstacle configurations. The experimental results show that CL-MAPPO can maintain stable hunting behavior under such randomized environments, demonstrating its generalization ability in obstacle-rich scenarios.

The network models saved upon completion of training for MAPPO, MADQN, and MADDPG were selected for qualitative evaluation and trajectory visualization, with results shown in Figure 9.

As shown in Figure 9a, the policy learned by MAPPO enables the UAV formation to maintain a relatively stable configuration and safely approach the target while performing hunting maneuvers. The three UAVs exhibit a coordinated converging trend toward the target region, indicating that the formation possesses basic cooperative hunting awareness. However, the hunting effectiveness remains insufficient, as the target can easily escape from the encircling circle before a tight hunting is established. The overall hunting intention is relatively weak. Specifically, the UAVs exhibit overly conservative obstacle avoidance behavior, abandoning the hunting attempt even when a considerable distance remains from the obstacles. This excessive caution can easily prevent the formation from closing in on the target effectively, resulting in hunting failure.

Figure 9b shows that the MADDPG policy allows the UAV formation to accomplish obstacle avoidance and attempt target encirclement. The UAVs demonstrate a general tendency to move toward the target while navigating around obstacles. However, the

algorithm's overall cooperative hunting awareness is relatively weak, and it completely fails to maintain the formation configuration at the initial hunting stage, with each UAV approaching the target independently, causing the desired triangular formation structure to collapse. Although basic target approach and obstacle avoidance capabilities are present, it is difficult to achieve effective cooperative hunting.

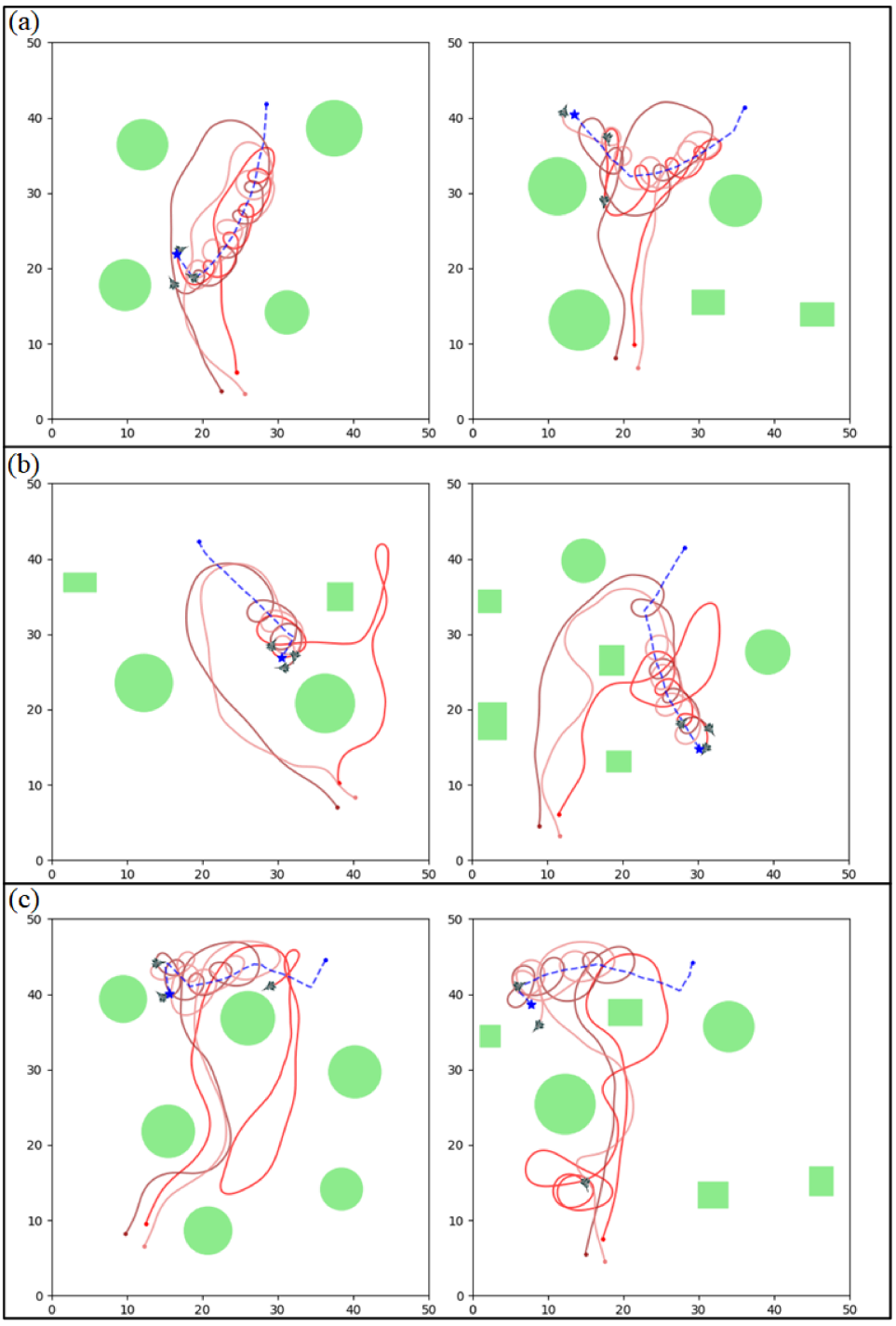


Figure 9. Trajectory visualization of baseline algorithms. (**a**) MAPPO. (**b**) MADDPG. (**c**) MADQN.

As illustrated in Figure 9c, the MADQN algorithm exhibits the worst performance among the three compared algorithms. The learned policy only enables basic obstacle avoidance with only a weak formation maintenance capability at the initial hunting stage. Moreover, only two agents approach the target and attempt encirclement, while the third

agent easily abandons its intention to encircle, moves independently, and plays a weak role throughout the engagement, showing no discernible cooperative pattern. This indicates a relatively poor policy learning outcome for cooperative hunting.

Figure 10 presents a quantitative comparison of four algorithms, namely CL-MAPPO, MAPPO, MADDPG, and MADQN, in terms of success rate and collision rate over 100 hunting episodes. CL-MAPPO achieves the highest success rate (89%) and the lowest collision rate (0%). Through a three-stage curriculum learning, the UAV formation acquires robust capabilities in formation keeping, obstacle avoidance, target search, and cooperative hunting. MAPPO yields a success rate of 10% and a collision rate of 3%. Its policy is overly cautious, abandoning hunting attempts too early and allowing the target to escape easily. MADDPG shows a success rate of 9% and a collision rate of 7%. Cooperative hunting awareness is weak, and the triangular formation collapses at the initial stage, impairing effective encirclement. MADQN performs worst, with a success rate of 2% and the highest collision rate of 10%. The learned policy enables only primitive obstacle avoidance and minimal cooperation, with one agent often leaving the hunting attempt.

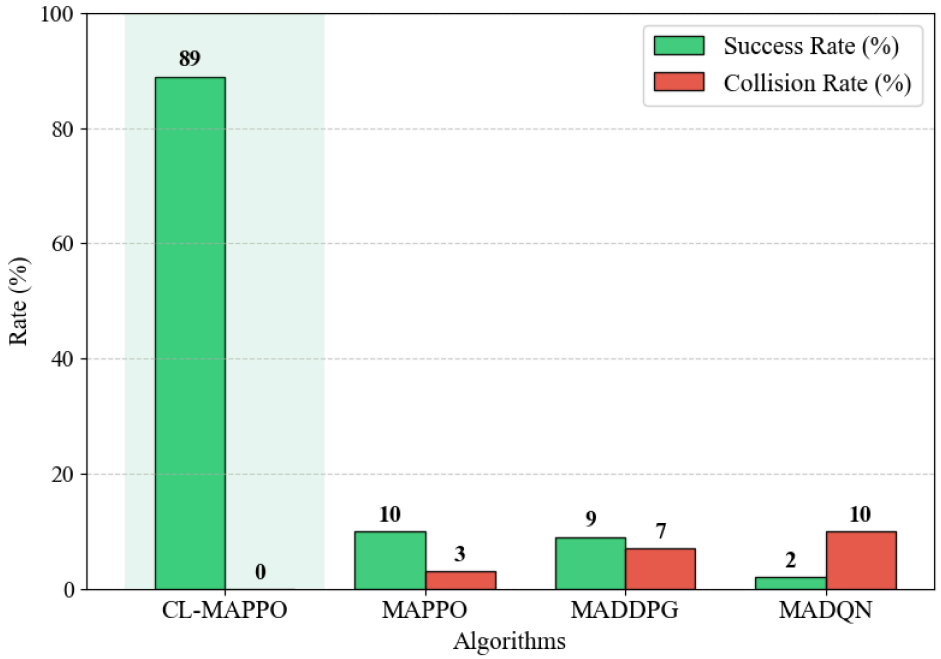


Figure 10. Algorithm performance comparison in terms of success rate and collision rate for CL-MAPPO, MAPPO, MADDPG, and MADQN.

In summary, CL-MAPPO significantly outperforms the other three algorithms. The curriculum learning framework effectively balances safety and hunting aggressiveness, overcoming the limitations of over-conservatism (MAPPO), formation collapse (MADDPG), and poor cooperation (MADQN) in complex multi-UAV hunting tasks. These results highlight the practical advantage of the proposed three-stage curriculum. The UAVs do not need to learn obstacle avoidance, formation keeping, and moving-target hunting simultaneously from the beginning. Instead, each stage provides a more learnable subproblem and transfers the acquired policy to the next stage. As a result, the final policy demonstrates capabilities that are difficult for the baseline algorithms to obtain directly in the complete task, including safer navigation in cluttered spaces, more stable triangular formation maintenance, and persistent pursuit of a maneuvering target. Such capabilities are particularly valuable for time-critical autonomous swarm missions in which manual rule tuning is difficult, and the environment cannot be fully predefined.

5. Conclusions

This paper proposed a curriculum learning-based multi-agent proximal policy optimization algorithm, termed CL-MAPPO, for multi-UAV cooperative hunting in obstructed environments. A three-stage progressive curriculum was designed to decompose the complex hunting task. Curriculum I uses fixed obstacles with a stationary target. Curriculum II introduces random obstacles and target positions. Curriculum III further incorporates a dynamic maneuvering target. Experimental results demonstrate the effectiveness of the proposed approach, which effectively addresses the key challenges of low exploration efficiency, slow environmental adaptation, and difficult convergence in multi-agent cooperative hunting tasks. The proposed CL-MAPPO framework provides a practical and scalable solution for UAV swarm operations in obstacle-rich and uncertain environments.

Beyond target intelligence, two additional limitations should be acknowledged [30]. First, the current study considers only three UAVs in a two-dimensional planar environment. While the 2D simplification is valid for ground-target hunting at constant altitude, scaling the approach to larger swarms (e.g., 5 or more UAVs) may introduce new challenges. Nevertheless, the CTDE architecture of MAPPO is known to scale reasonably well, and we expect that increasing the number of agents would be feasible with moderate modifications to the reward structure and network capacity. Second, full three-dimensional flight dynamics (e.g., altitude changes, vertical obstacle avoidance) are not considered. Extending the current framework to 3D would require reformulating the kinematic model, updating the obstacle detection rays to a spherical grid, and redesigning the formation reward. These extensions are left as important directions for future research.

Author Contributions: Conceptualization, X.W. and H.P.; methodology, L.Z. and J.Z.; software, L.Z. and J.Z.; validation, L.Z., J.Z. and B.L.; formal analysis, L.Z.; investigation, L.Z., J.Z. and B.L.; writing—original draft preparation, L.Z.; writing—review and editing, X.W. and B.L.; visualization, L.Z.; project administration, H.P.; funding acquisition, X.W. All authors have read and agreed to the published version of the manuscript.

Funding: The authors are thankful for the financial support of the Fundamental Research Funds for the Central Universities (DUT25GF207).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

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

Abbreviations

The following abbreviations are used in this manuscript:

PPO	Proximal policy optimization
MAPPO	Multi-agent proximal policy optimization
CL	Curriculum learning
UAV	Unmanned aerial vehicle
DRL	Deep reinforcement learning
APF	Artificial potential field
MADRL	Multi-agent deep reinforcement learning
VDN	Value-decomposition networks
MADDPG	Multi-agent deep deterministic policy gradient
TD3	Twin delayed deep deterministic policy gradient
CTDE	Centralized training decentralized execution

References

1. Yu, X.; Li, X.; Wang, L.; Jin, J.; Zhang, G.; Su, X.; Tao, L.; Lu, C.; Wang, X. Carrier platform-enhanced multiple-UAV cooperative task assignment with dual heterogeneities. *Artif. Intell. Rev.* **2025**, *58*, 8. [CrossRef]
2. Meng, Y.; Xu, S.; Liang, X.; Zeng, S.; Yao, Z.; Bian, C.; Wang, L.; Zhong, C.; Wang, X. Spatiotemporal synchronization-aware cross-domain mission planning for air-sea-underwater heterogeneous unmanned swarms. *Ocean. Eng.* **2026**, *353*, 124694.
3. Cai, W.; Chen, H.; Zhang, M. A survey on collaborative hunting with robotic swarm: Key technologies and application scenarios. *Neurocomputing* **2024**, *598*, 128008. [CrossRef]
4. Zhao, X.; Tan, J.; Meng, W.; Yu, Z.; Yan, Y.; Zhang, Z. Cooperative Encirclement and Obstacle Avoidance of Fixed-Wing UAVs via MADDPG with Curriculum Learning. *Drones* **2025**, *9*, 727. [CrossRef]
5. Yu, X.; Wang, L.; Gao, X.; Wang, X.; Lu, C. Mission Planning for Heterogeneous UAVs in Obstacle-Dense Environment. In *Proceedings of the 2022 International Conference on Autonomous Unmanned Systems (ICAUS 2022)*; Springer: Berlin/Heidelberg, Germany, 2023; Volume 1010, pp. 823–836.
6. Sousa, A.; Oliveira, A.; Teixeira, S. Multi-Agent Robot Systems: Analysis, Classification, Applications, Challenges and Directions. In Proceedings of the 2024 IEEE International Conference on Industrial Technology (ICIT), Bristol, UK, 25–27 March 2024.
7. Mallapragada, G.; Chattopadhyay, I.; Ray, A. Autonomous navigation of mobile robots using optimal control of finite state automata. In Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA, 13–15 December 2006; pp. 13–26.
8. Khan, M.; Ibrahim, M.; Mahmud, A.; Samat, F.A.; Jasni, F.; Mardzuki, M.I. Advancing mobile robot navigation with DRL and heuristic rewards: A comprehensive review. *Neurocomputing* **2025**, *652*, 131036. [CrossRef]
9. Waga, A.; Benhlima, S.; Bekri, A.; Abdouni, J.; Saber, F.Z. A survey on autonomous navigation for mobile robots: From traditional techniques to deep learning and large language models. *J. King Saud Univ. Comput. Inf. Sci.* **2025**, *37*, 198. [CrossRef]
10. Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; Mordatch, I. Multi-agent actor critic for mixed cooperative-competitive environments. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017*; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6379–6390.
11. Yu, L.; Huo, S.; Wang, Z.; Li, K. Hybrid attention-oriented experience replay for deep reinforcement learning and its application to a multi-robot cooperative hunting problem. *Neurocomputing* **2023**, *523*, 44–57. [CrossRef]
12. Lillicrap, T.; Hunt, J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2016**, arXiv:1509.02971.
13. Sunehag, P.; Lever, G.; Gruslys, A.; Czarnecki, W.M.; Zambaldi, V.; Jaderberg, M.; Lanctot, M.; Sonnerat, N.; Leibo, J.Z.; Tuyls, K.; et al. Value-Decomposition Networks For Cooperative Multi-Agent Learning. *arXiv* **2017**, arXiv:1706.05296v1.
14. Rashid, T.; Samvelyan, M.; de Witt, C.S.; Farquhar, G.; Foerster, J.; Whiteson, S. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, 10–15 July 2018; pp. 4295–4304.
15. He, J.; Meng, Z.; Zhang, J.; Wang, Z.; Xie, Y. Autonomous manoeuvre decision-making algorithm for air combat based on hybrid temporal difference error-reward prioritised experience replay with twin delayed deep deterministic policy gradient. *Aeronaut. J.* **2025**, *129*, 3484–3505. [CrossRef]
16. Xue, S.; Zhao, N.; Wang, L.; Zhang, W.; Zhang, J.; Zhu, F. Multi-agent self-attention reinforcement learning for multi-USV hunting target. *Neural Netw.* **2025**, *189*, 107574. [CrossRef] [PubMed]
17. Wang, Y.; Wang, W.; Chen, D. Knowledge-Guided Reinforcement Learning with Artificial Potential Field-Based Demonstrations for Multi-Autonomous Underwater Vehicle Cooperative Hunting. *J. Mar. Sci. Eng.* **2025**, *13*, 423. [CrossRef]
18. Wang, Y.; Hu, H.; Yan, H.; Wen, G. Cooperative Hunting Strategy for USVs Based on Reinforcement Learning. *IEEE Trans. Autom. Sci. Eng.* **2026**, *23*, 3536–3546. [CrossRef]
19. Xia, J.; Luo, Y.; Liu, Z.; Zhang, Y.; Shi, H.; Liu, Z. Cooperative multi-target hunting by unmanned surface vehicles based on multi-agent reinforcement learning. *Def. Technol.* **2023**, *29*, 80–94. [CrossRef]
20. Zhong, W.; Tan, Q.; Yan, S.; Meng, Y.; Ye, N. USVs cooperative hunting method based on advantage decomposition and sequential decision. *Ocean. Eng.* **2025**, *318*, 120123. [CrossRef]
21. Luo, J.; Xu, B.; Li, X.; Yao, Z. Autonomous Cooperative Hunting with Rule-Based and Self-Learning Control for Multiagent Systems. *J. Intell. Robot. Syst.* **2024**, *110*, 160. [CrossRef]
22. He, S.; Wang, L.; Liu, M.; Wu, Z. Dynamic Multi-Target Self-Organization Hunting Control of Multi-Agent Systems. *Appl. Sci.* **2024**, *14*, 3875. [CrossRef]
23. Xu, B.; Zhang, H.; Meng, H.; Hu, B.; Chen, D.; Chen, G. Moving Target Surrounding Control of Linear Multiagent Systems With Input Saturation. *IEEE Trans. Syst. Man Cybern. Syst.* **2022**, *52*, 1705–1715. [CrossRef]
24. Xu, B.; Zhang, H.; Ding, Y.; Ren, W. Event-Triggered Surrounding Formation Control of Multiagent Systems for Multiple Dynamic Targets. *IEEE Trans. Control. Netw. Syst.* **2023**, *10*, 752–764. [CrossRef]

25. Li, W.; Yan, S.; Shi, L.; Yue, J.; Shi, M.; Lin, B.; Qin, K. Multiagent Consensus Tracking Control Over Asynchronous Cooperation-Competition Networks. *IEEE Trans. Cybern.* **2025**, *55*, 4347–4360. [CrossRef] [PubMed]
26. Zheng, L.; Li, X.; Su, X.; Li, B.; Wang, L.; Zhou, J.; Peng, H.; Tian, W.; Wang, X. Within-visual-range air combat maneuver decision-making in obstructed environments via a curriculum self-play soft actor-critic with an attention mechanism. *Def. Technol.* **2026**, *57*, 122–137. [CrossRef]
27. Wang, X.; Wang, Y.; Su, X.; Wang, L.; Lu, C.; Peng, H.; Liu, J. Deep reinforcement learning-based air combat maneuver decision-making: Literature review, implementation tutorial and future direction. *Artif. Intell. Rev.* **2023**, *57*, 1. [CrossRef]
28. Qu, X.; Jiang, Y.; Long, F.; Zhang, R.; Gao, Y. Stage-induced learning-based cooperative target hunting strategy for multiple un manned surface vehicles. *Chin. J. Ship Res.* **2025**, *20*, 162–171.
29. Li, K.; Zang, K.; Wei, Z.; Piao, H.; Yuan, B.; Wang, B.; Cheng, J. An RNN-Enhanced Diverse Curriculum-Driven Learning Algorithm Based on Deep Reinforcement Learning for POMDPs with Limited Experience. *Drones* **2026**, *10*, 142. [CrossRef]
30. Ni, J.; Zhao, Y.; Zhang, Z.; Ke, C.; Yang, S.X. A survey on theories and applications for multi-robot cooperative hunting. *Robot. Auton. Syst.* **2026**, *197*, 105296. [CrossRef]

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.