SCOPE: Safety-Constrained Online Preview Enforcement for Efficient Encirclement in Multi-UAV Pursuit–Evasion

Cheng Chen[1], **Weiwei Yuan**[1*], **Xiaozhen Lu**[1,2*], **Yicong Li**[1,3] and **Jiale Zhang**[1]
[1]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics
[2]National Mobile Communications Research Laboratory, Southeast University
[3]State Key Laboratory of Novel Software Technology, Nanjing University
{98177qdn, yuanweiwei, luxiaozhen, liyicong, jiale-zhang}@nuaa.edu.cn

Abstract

Unmanned aerial vehicle swarms in pursuit–evasion requires encirclement efficiency while maintaining safety constraints, facing a critical safety–efficiency trade-off. Existing safe multi-agent reinforcement learning (MARL) methods often yield either **unsafe task policies** or **conservative policies**. This is challenging since forward feasible safety under dense interactions requires online safety preview. To address this issue, we propose **S**afety-**C**onstrained **O**nline **P**review **E**nforcement (SCOPE), a MARL-based algorithm that balances encirclement efficiency and safety by short-horizon preview and safety enforcement. SCOPE learns an online safety-preview dynamics model that rolls out future trajectories to inform encirclement decisions checking. By proposing preview-fused actor–critic, SCOPE uses short-horizon previews for efficient encirclement and less unsafe behavior. Hierarchical safety enforcement performs safety look-ahead and online action correction to maintain forward feasible safety. Experiments show that SCOPE better balances encirclement efficiency and safety than safe MARL baselines, maintaining average encirclement time while reducing the average agent cost by 65.6%. Code could be found at https://github.com/98177qdn/SCOPE.

1 Introduction

In safety-critical pursuit–evasion scenarios, safety constraints often require that unmanned aerial vehicle (UAV) swarms intercept a rogue UAV in a non-destructive manner. To achieve non-destructive interception, a common solution is cooperative encirclement, especially in cluttered low-altitude airspace [Yu *et al.*, 2024; Valianti *et al.*, 2024; Ma *et al.*, 2020]. Compared with direct capture, encirclement requires multiple UAVs to close in simultaneously and maintain a coherent formation under dense interactions among UAVs, the evader, and moving obstacles [Dong *et al.*, 2020]. To address this problem, research on reinforcement learning for safety-critical has attracted considerable attention.

*Corresponding authors.

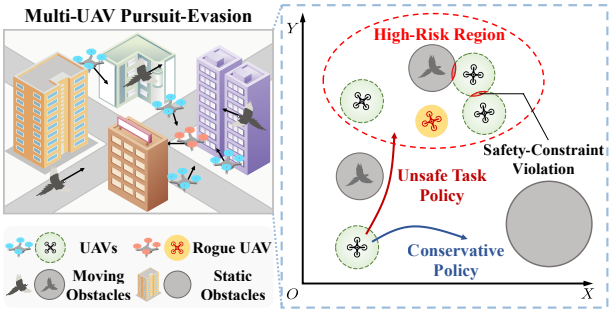

Figure 1: Multi-UAV encirclement in pursuit-evasion exposes the safety-efficiency trade-off: unsafe task policies may achieve fast closure but violate safety constraints, whereas conservative policies stay safe but are too inefficient to complete the task.

Multi-agent reinforcement learning (MARL) has achieved encouraging success in safety-critical autonomy, including robotic control [Duan *et al.*, 2024], autonomous driving [Lee and Kwon, 2024], and airspace security [Du *et al.*, 2021]. However, safety constraints are frequently incorporated only as a negative reward, which may not explicitly prevent instantaneous violations during execution [Qu *et al.*, 2024; Zhang *et al.*, 2023b; Zhang *et al.*, 2024]. In practice, soft penalty allows the policy to trade safety for return, so it may incur instantaneous violations for higher efficiency, which is not permitted in safety-critical encirclement. As illustrated in Figure 1, **multi-UAV pursuit–evasion requires efficient encirclement under safety constraints**. Since the UAV swarm must satisfy strict multi-constraint safety, even instantaneous violations can cause collisions or mission failure [Lv *et al.*, 2024]. In such settings, traditional MARL can lead to safety-efficiency trade-off: an unsafe task policy that violates safety constraints, or a conservative policy that prevents successful encirclement [Valianti *et al.*, 2024].

Recently, more research has focused on safe MARL to handle safety constraints. Nevertheless, for multi-UAV pursuit–evasion, it is still hard for existing safe MARL methods to achieve safe and efficient encirclement [Chen *et al.*, 2025; Li *et al.*, 2023]. In general, safe MARL approaches can be divided into objective-level and execution-level methods. Objective-level methods, such as primal-dual formulations, penalty-based schemes [Liu *et al.*, 2020; Zhang *et al.*, 2022], and risk-sensitive objectives [Chow *et al.*, 2015],

build constraints into the return or the optimization objective [Achiam *et al.*, 2017], but they often provide only soft guarantees and may still allow brief safety violations during execution. Execution-level methods adjust actions before execution to enforce safety online [Alshiekh *et al.*, 2018; Zhang *et al.*, 2023a; Zhou *et al.*, 2025], but they usually do not account for whether safety remains feasible over predicted trajectories, and they are often designed mainly for obstacle avoidance while ignoring inter-agent safety constraints [Zhao *et al.*, 2025; Wabersich and Zeilinger, 2023].

To achieve safe and efficient encirclement, three challenges need to be addressed. **(1) Safety constraints are often violated when there's no preview.** With moving obstacles, aggressive closing rapidly shrinks safety margins, making near-future violations hard to predict and easy to miss until they occur. **(2) Safety constraints often induces conservative policies that undermine encirclement efficiency.** Conservative policies can fragment the formation and delay or prevent encirclement, making it difficult to enforce strict safety constraints without sacrificing efficiency. **(3) Ensuring forward feasible under dense interaction is challenging.** Safety constraints depend on other moving entities and evolve over time, making forward feasible safety difficult to maintain and constraint satisfaction hard to guarantee in future states.

To address these challenges, we propose Safety-Constrained Online Preview Enforcement (SCOPE) for safe multi-UAV pursuit–evasion. SCOPE combines an online Safety-Preview Dynamics Model (SPDM) to produce short-horizon safety previews and predict future trajectory under dense interactions. Then Preview-Fused Actor–Critic uses these previews to learn efficient encirclement policies without excessive conservatism. Hierarchical Safety Enforcement applies safety look-ahead and strict margin tightening to correct joint actions and maintain forward feasible safety. In addition, the safety enforcement provides a safety loss to networks update, encouraging safe and effective coordination during training. Together, these components address the safety–efficiency trade-off by improving encirclement performance while maintaining safety constraints.

The main contributions are summarized as follows:

- We design an online SPDM with preview-fused actor–critic that uses short-horizon safety previews to better mitigate the safety–efficiency trade-off in encirclement.
- We propose hierarchical safety enforcement, which enforces safety constraints via online action correction and safety loss during policy learning, while safety look-ahead maintains forward feasible safety.
- We build a challenging multi-UAV pursuit–evasion task and validate SCOPE on it, showing advantages over state-of-the-art baselines via comparative and ablation studies. Results show that SCOPE completes tasks efficiently while maintaining forward feasible safety.

2 Related Works

2.1 Model-Based Methods for Encirclement

Encirclement and pursuit–evasion have long been studied from the perspective of model-based control and planning. Cyclic-pursuit coordination is a typical choice for forming enclosing patterns [Kim *et al.*, 2010]. Under limited sensing, range-only and coordinate-free guidance has been explored [Dong *et al.*, 2020]. Model predictive control is often used when safety constraints are needed in dynamic encirclement [Hafez *et al.*, 2014]. These methods can work well under their assumptions, but they are difficult to extend to dense agent interactions and moving obstacles.

2.2 Safe MARL for Encirclement

Safe MARL introduces safety costs into MARL and optimizes return under a violation budget [Hazra *et al.*, 2025]. Dense constraints and non-stationary learning, however, make safe coordination difficult. Existing methods are often discussed at two levels: objective-level and execution-level. For objective-level methods, costs are folded into the optimization objective through primal–dual, penalty-based, or trust-region style updates [Liu *et al.*, 2020; Zhang *et al.*, 2022; Achiam *et al.*, 2017]. MAP3O is a representative penalty-based method [Zhang *et al.*, 2022]. Lyapunov-based methods enforce long-horizon safety via local constraints [Chow *et al.*, 2018], whereas PID-Lagrangian stabilizes multiplier updates [Stooke *et al.*, 2020]. Objective-level methods can be sensitive to hyperparameters, and their constraint satisfaction is usually soft or asymptotic [Xiao and Feroskhan, 2024; Li and Azizan, 2024]. In dense interactions, this can result in conservative policies or occasional instantaneous violations.

Execution-level methods, often referred to as shielding, modify actions at runtime to prevent violations [Alshiekh *et al.*, 2018; Brorholt *et al.*, 2024]. Control barrier function based filters support online checking and correction [Cai *et al.*, 2021], and recovery-based methods invoke recovery behaviors near unsafe regions [Zhao *et al.*, 2025]. Frequent interventions may alter coordination strategies and degrade task efficiency in dense interactions [Sun *et al.*, 2023].

3 Preliminaries

We model the multi-UAV pursuit–evasion problem as a Constrained Markov Decision Process (CMDP), defined by the tuple $\langle \mathcal{S}, \mathcal{O}, \mathcal{A}, \mathcal{R}, \mathcal{C}, \mathcal{P}, \gamma \rangle$, where $\mathcal{S}$ denotes the state space, $\mathcal{O}$ denotes the observation space, $\mathcal{A}$ denotes the joint action space, $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is reward function, $\mathcal{C} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the cost function used to quantify safety constraints, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is transition probability function,$\gamma \in (0, 1)$ is the discount factor for calculating the return. At time step t, the environment is in state $\boldsymbol{s}^{(t)} \in \mathcal{S}$ and each agent i receives a observation $\boldsymbol{o}_i^{(t)}$ drawn from an observation kernel $\Omega : \mathcal{S} \to \Delta(\mathcal{O})$. The objective of CMDP is to find a joint policy $\boldsymbol{\pi} : \mathcal{O} \to \Delta(\mathcal{A})$, that maximizes the expected discounted return:

$$\arg\max_{\boldsymbol{\pi} \in \Pi_{\mathcal{C}}} \mathcal{J}(\boldsymbol{\pi}) = \mathbb{E}_{\boldsymbol{\tau} \sim \boldsymbol{\pi}} \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(\boldsymbol{s}^{(t)}, \tilde{\boldsymbol{a}}^{(t)}) \right], \quad (1)$$

where $\Pi_{\mathcal{C}} = \{\boldsymbol{\pi} \in \Pi : \mathcal{J}_{\mathcal{C}}(\boldsymbol{\pi}) \leq d\}$ is allowable policy set, $\boldsymbol{\tau}$ denotes a joint trajectory, $\tilde{\boldsymbol{a}}^{(t)}$ denotes joint actions. $\mathcal{J}_{\mathcal{C}}(\pi)$ is the cumulative constraints similar to reward and d is the constraint threshold. Thus, the task reduces to learning a joint policy $\boldsymbol{\pi}$ that maximizes $\mathcal{J}(\boldsymbol{\pi})$ while ensuring $\mathcal{J}_{\mathcal{C}}(\boldsymbol{\pi}) \leq d$.

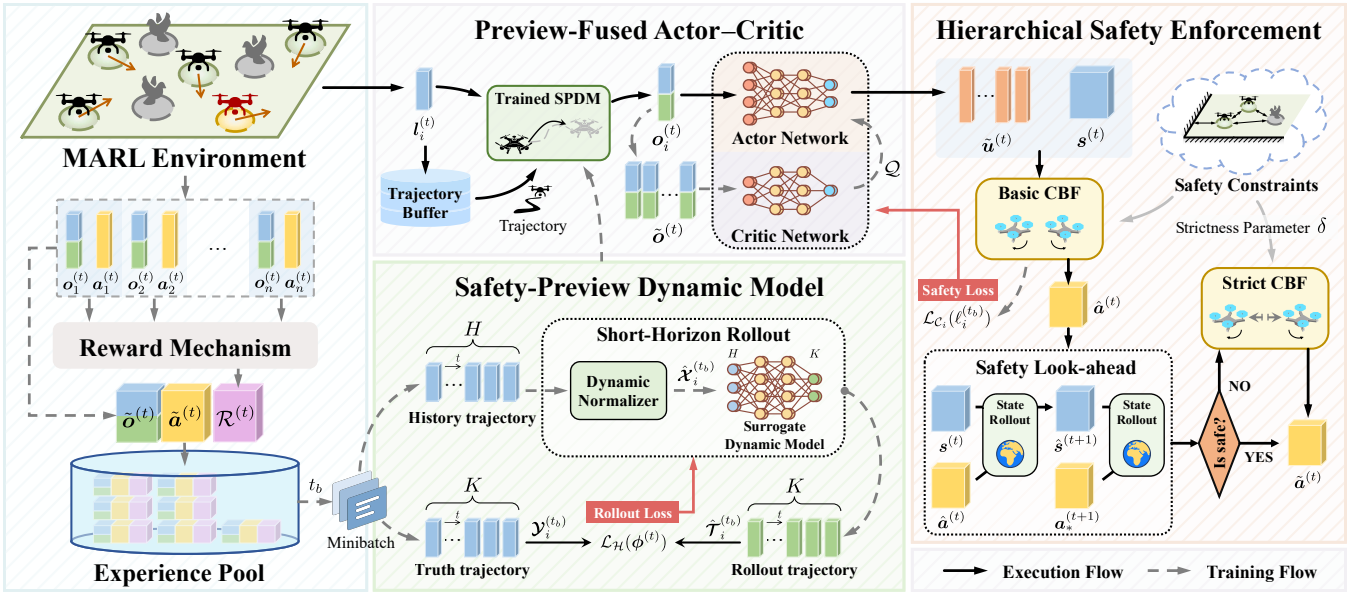

Figure 2: Pipeline of SCOPE. The Safety-Preview Dynamics Model learns short-horizon dynamics from the replay buffer and generates safety previews. The Preview-Fused Actor–Critic uses these previews for policy learning. Hierarchical Safety Enforcement then corrects raw actions using a CBF-based look-ahead and strict tightening, and provides safety loss during training.

4 Methodology

Pipeline of SCOPE is shown in Figure 2. SCOPE combines short-horizon safety preview with online constraint enforcement for multi-UAV pursuit–evasion. Safety-Preview Dynamics Model (SPDM) predicts future trajectory for Preview-Fused Actor–Critic. Hierarchical Safety Enforcement corrects actions and injects safety loss to guide learning, enabling forward feasible safety in dense environments.

4.1 Safety-Preview Dynamics Model

The safety margin between agents and obstacles changes rapidly as the formation closes in during pursuit–evasion. To address this, we construct a SPDM to predict short-horizon future observations and provide previews for policy learning under safety constraints.

For stability and low online overhead, we employ a parameter-shared SPDM assign each agent a dynamic normalizer to handle distribution shift from evolving policies and heterogeneous observations. The surrogate dynamics model $\mathcal{H}$, parameterized by $\phi^{(t)}$, takes as input a length-H history of normalized observations $\mathcal{Z}_i^{(t)} = [\, \boldsymbol{z}_i^{(t-H+1)}, \ldots, \boldsymbol{z}_i^{(t)} \,]$ and generates a K-step rollout for all N agents:

$$\hat{\mathcal{T}}_i^{(t)} = \mathcal{H}\left(\mathcal{Z}_i^{(t)} | \phi^{(t)}\right), \tag{2}$$

where $\hat{\mathcal{T}}_i^{(t)} = [\hat{\boldsymbol{l}}_i^{(t+1)}, \ldots, \hat{\boldsymbol{l}}_i^{(t+K)}]$ denotes the trajectory within horizon K for agent i. For initial timesteps with insufficient history, we pad missing observations with a mask so as to preserve input length and temporal alignment.

Unlike conventional dynamics models trained offline, the SPDM is updated alongside the RL process using experience sampled from the replay buffer $\mathcal{D}$. At each update step, we sample a mini-batch of B transitions and define a mapping $f : \{1, \ldots, B\} \to \mathbb{N}$ that associates batch indices with global timesteps. For the b-th sample in the mini-batch, we denote its corresponding global timestep by $t_b = f(b)$. The per-agent dynamic normalizers are updated incrementally with each batch, in parallel with the SPDM parameter updates. For the SPDM, each sampled trajectory $\mathcal{T}_i^{(t_b)} = [\boldsymbol{l}_i^{(t_b-H+1)}, \ldots, \boldsymbol{l}_i^{(t_b)}, \ldots, \boldsymbol{l}_i^{(t_b+K)}]$ is transformed into a normalized input $\mathcal{Z}_i^{(t_b)}$ and ground-truth targets $\mathcal{Y}_i^{(t_b)}$. Feeding the history $\mathcal{Z}_i^{(t_b)}$ into the dynamics model yields the trajectory rollout $\hat{\mathcal{T}}_i^{(t_b)}$, which is kept accurate for predicting imminent safety violations by online training. We train SPDM by minimizing the discrepancy between rolled-out and ground-truth trajectories, with the rollout loss defined as:

$$\mathcal{L}_{\mathcal{H}}\left(\phi^{(t)}\right) = \frac{1}{BNK} \sum_{b=1}^{B} \sum_{i=1}^{N} \sum_{k=1}^{K} \left\| \hat{\boldsymbol{l}}_i^{(t_b+k)} - \boldsymbol{l}_i^{(t_b+k)} \right\|^2. \tag{3}$$

The learned SPDM is then used online to generate the K-step preview $\hat{\mathcal{T}}_i^{(t)}$, which is fused with the current observation to form the inputs to the actor–critic described in Sec. 4.2.

4.2 Preview-Fused Actor–Critic

Early training can be particularly challenging due to high-dimensional inputs and tightly coupled multi-agent learning dynamics. General actor–critic often struggles here: the actor may overfit noisy or irrelevant channels, and the Critic may fail to capture inter-agent dependencies while policies co-adapt. We introduce two lightweight enhancements as illustrated in Figure 3: a gating module in the actor for data-dependent feature selection, and a multi-head attention mod-

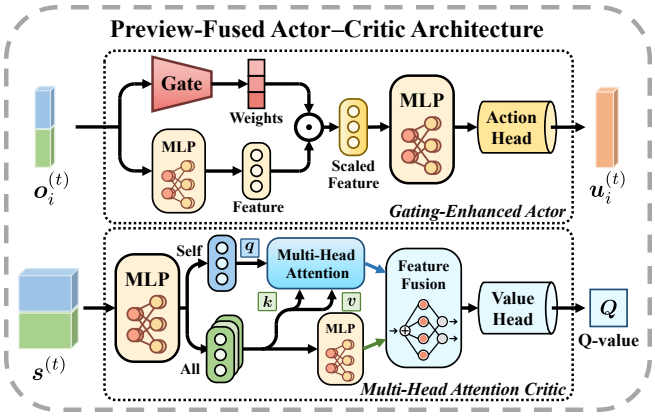

Figure 3: Design of the Preview-Fused Actor–Critic networks.

ule in the Critic to model interactions between agents and downweight unreliable near-future previews.

Gating-Enhanced Actor

To improve robustness under high-dimensional inputs, we introduce a gating module related to observation that performs feature selection. For each agent i, the actor μ_i, parameterized by $\boldsymbol{\theta}_i^{(t)}$, takes the local observation $\boldsymbol{o}_i^{(t)}$ and produces a control decision $\boldsymbol{u}_i^{(t)}$ before the hierarchical safety enforcement. We first transform the local observation $\boldsymbol{o}_i^{(t)}$ into a feature vector $\boldsymbol{x}_i^{(t)}$. Then, the gating module generates a set of weight from $\boldsymbol{o}_i^{(t)}$ and uses it to scale the features:

$$\hat{\boldsymbol{x}}_i^{(t)} = \boldsymbol{x}_i^{(t)} \odot \sigma\left(\boldsymbol{W}^{(t)}\boldsymbol{o}_i^{(t)} + \boldsymbol{b}^{(t)}\right), \tag{4}$$

where $\hat{\boldsymbol{x}}_i^{(t)}$ is scaled feature, $\boldsymbol{W}^{(t)}$, $\boldsymbol{b}^{(t)}$ are trainable parameters of the gating unit, $\sigma(\cdot)$ is the sigmoid activation function and $\odot$ is the Hadamard product. The gate vector has the same dimensionality as the feature. This design improves robustness when SPDM rollouts are noisy or partially missing.

Multi-Head Attention Critic

We introduce a multi-head attention critic to robustly aggregate multi-agent information. The critic $\mathcal{Q}_i$, parameterized by $\boldsymbol{\psi}_i^{(t)}$, is responsible for estimating the expected return of agent i given the joint state–action pair $[\boldsymbol{s}^{(t)}, \boldsymbol{a}^{(t)}]$. Finishing partitioning, a shared state–action network encodes each agent's feature into an embedding. For attention head h:

$$\alpha_{i,h} = \frac{\exp\left(\boldsymbol{q}_{i,h}\boldsymbol{K}_h^\top\right)}{\sum_{j=1}^{N}\exp\left(\boldsymbol{q}_{j,h}\boldsymbol{K}_h^\top\right)}, \tag{5}$$

where $\boldsymbol{q}_{i,h}$ is a learnable query from $\boldsymbol{o}_i^{(t)}$ and $\boldsymbol{K}_h, \boldsymbol{V}_h$ are the key and value derived from $\boldsymbol{s}^{(t)}$. The weighted feature embeddings are aggregated into global representation $\boldsymbol{c}_h = \sum_{i=1}^{N}\alpha_{i,h}\boldsymbol{V}_h$ and fused with the original features.

4.3 Hierarchical Safety Enforcement

To further provide runtime safety protection and prevent instantaneous violations under coupled constraints, especially in high-risk regions. Therefore, we introduce hierarchical safety enforcement, an online mechanism based on control barrier functions. Given the actor's decision $\tilde{\boldsymbol{u}}^{(t)}$, this module computes minimally modified joint actions $\tilde{\boldsymbol{a}}^{(t)}$ to satisfy coupled safety constraints. Meanwhile it provides safety loss through a CBF-based penalty integrated into the actor loss.

Basic CBF Enforcement

At each timestep t, the actor network outputs joint decision $\tilde{\boldsymbol{u}}^{(t)}$, and the basic CBF performs a preliminary enforcement step. We cast action correction as a joint quadratic program that satisfies the safety constraints while minimally deviating from the policy intent. Specifically, the basic CBF enforcement finds the minimally modified action closest to $\tilde{\boldsymbol{u}}^{(t)}$ subject to the CBF constraints. The objective of the basic CBF is to minimize the magnitude of action corrections while enforcing the joint safety constraints $\xi(\cdot)$:

$$\begin{aligned}\hat{\boldsymbol{a}}^{(t)} &= \arg\min_{\boldsymbol{a}\in\mathcal{A}} \left\|\boldsymbol{a} - \tilde{\boldsymbol{u}}^{(t)}\right\|^2, \\ \text{s.t. } \ddot{\xi}\left(\boldsymbol{s}^{(t)}, \boldsymbol{a}\right) &+ \lambda_1\dot{\xi}\left(\boldsymbol{s}^{(t)}\right) + \lambda_2\xi\left(\boldsymbol{s}^{(t)}\right) \geq 0,\end{aligned} \tag{6}$$

where $\hat{\boldsymbol{a}}$ denotes the corrected actions, $\lambda_1, \lambda_2 > 0$ are damping factors that regulate the responsiveness of the barrier function. The safety constraints are detailed in Appendix C.

Safety Look-Ahead

While the basic CBF enforcement guarantees instantaneous safety, it does not ensure forward feasible safety. This may lead to deadlock or infeasibility in highly dynamic pursuit–evasion encirclement. To address this issue, we add a one-step safety look-ahead feasibility check and invoke a strict CBF enforcement when necessary.

Specifically, after obtaining $\hat{\boldsymbol{a}}^{(t)}$, we use the SPDM to produce the one-step preview $\hat{\boldsymbol{s}}^{(t+1)}$ from the observation history, which is then used to maintain forward-feasible safety:

$$\ddot{\xi}\left(\hat{\boldsymbol{s}}^{(t+1)}, \boldsymbol{a}_*^{(t+1)}\right) + \lambda_1\dot{\xi}\left(\hat{\boldsymbol{s}}^{(t+1)}\right) + \lambda_2\xi\left(\hat{\boldsymbol{s}}^{(t+1)}\right) \geq 0, \tag{7}$$

Feasibility at $\hat{\boldsymbol{s}}^{(t+1)}$ holds if there exists a conservative action $\boldsymbol{a}_*^{(t+1)}$ satisfying the safety constraints.

Strict CBF Enforcement

When the conservative action $\boldsymbol{a}_*^{(t)}$ does not exist, the basic CBF enforcement may fail to maintain continuous feasibility. Therefore, we introduce a strict CBF enforcement that tightens the safety margin with strictness parameter $\delta > 0$ to recover feasibility. The optimization objective remains the same as Equation. (6), while we adopt a strict CBF constraint:

$$\ddot{\xi}\left(\boldsymbol{s}^{(t)}, \boldsymbol{a}\right) + \lambda_1\dot{\xi}\left(\boldsymbol{s}^{(t)}\right) + \lambda_2\xi\left(\boldsymbol{s}^{(t)}\right) \geq \delta. \tag{8}$$

We integrate a smooth CBF-based penalty into the actor loss, jointly optimizing safety and task performance. The loss due to CBF constraint is derived as: $\mathcal{L}_{\mathcal{C}_i}(\ell_i^{(t_b)}) = \log(1 + \exp(-\ell_i^{(t_b)}))$. The combined loss is stated as follows:

$$\mathcal{L}_\mu\left(\boldsymbol{\theta}_i^{(t)}\right) = -\hat{\mathbb{E}}_b\left[\mathcal{Q}_i(\boldsymbol{s}^{(t_b)}, \tilde{\boldsymbol{a}}^{(t_b)}|\boldsymbol{\psi}_i^{(t)}) - \eta_i\mathcal{L}_{\mathcal{C}_i}(\ell_i^{(t_b)})\right], \tag{9}$$

Algorithm 1 Overall pipeline of SCOPE

Input: Initial parameters $\tilde{\boldsymbol{\theta}}^{(0)}, \tilde{\boldsymbol{\psi}}^{(0)}, \boldsymbol{\phi}^{(0)}$, Initial state $\boldsymbol{s}^{(0)}$
Output: Trained parameters $\tilde{\boldsymbol{\theta}}^{(T)}, \tilde{\boldsymbol{\psi}}^{(T)}, \boldsymbol{\phi}^{(T)}$

for $t = 1$ to T or mission complete **do**
  # Hierarchical safety enforcement
  For each agent i, select $\boldsymbol{u}_i^{(t)} = \mu_i(\boldsymbol{o}_i^{(t)}|\boldsymbol{\theta}_i^{(t)}) + \mathcal{N}^{(t)}$
  Correct preliminary actions $\tilde{\boldsymbol{u}}^{(t)} = (\boldsymbol{u}_1^{(t)}, \ldots, \boldsymbol{u}_N^{(t)})$ to $\tilde{\boldsymbol{a}}^{(t)} = (\boldsymbol{a}_1^{(t)}, \ldots, \boldsymbol{a}_N^{(t)})$ using Equation. (5)
  Execute actions $\tilde{\boldsymbol{a}}^{(t)}$ and receive reward $\boldsymbol{r}^{(t)}$, new state $\boldsymbol{s}^{(t+1)}$ and raw observation $\boldsymbol{l}^{(t+1)}$
  # SPDM rollout and update
  Roll out $\{\hat{\boldsymbol{\mathcal{T}}}_i^{(t+1)}\}_{i=1}^N$ using Equation. (2) and get fused observation $\tilde{\boldsymbol{o}}^{(t+1)} = (\boldsymbol{o}_1^{(t+1)}, \ldots, \boldsymbol{o}_N^{(t+1)})$
  Sample a random minibatch of B trajectories from $\mathcal{D}$
  For each batch b, split $\{\boldsymbol{\mathcal{T}}_i^{(t_b)}\}_{i=1}^N$ into normalized input $\{\boldsymbol{\mathcal{Z}}_i^{(t_b)}\}_{i=1}^N$ and ground truth $\{\boldsymbol{\mathcal{Y}}_i^{(t_b)}\}_{i=1}^N$
  Update parameters $\boldsymbol{\phi}^{(t)}$ using Equation. (3)
  # Policy update
  for agent i to N **do**
    $y_i^{(t_b)} = r_i^{(t_b)} + \gamma \mathcal{Q}_i\left(\boldsymbol{s}^{(t_b+1)}, \tilde{\boldsymbol{a}}^{(t_b+1)}|\hat{\boldsymbol{\psi}}_i^{(t)}\right)$
    Update critic $\boldsymbol{\psi}_i^{(t)}$ by minimizing the loss:
    $\mathcal{L}_{\mathcal{Q}}\left(\boldsymbol{\psi}_i^{(t)}\right) = \hat{\mathbb{E}}_b\left[\left(y_i^{(t_b)} - \mathcal{Q}_i(\boldsymbol{s}^{(t_b)}, \tilde{\boldsymbol{a}}^{(t_b)}|\boldsymbol{\psi}_i^{(t)})\right)^2\right]$
    Update actor $\boldsymbol{\theta}_i^{(t)}$ minimizing Equation. (9)
  end for
  Update target network parameters for each agent i: $\hat{\boldsymbol{\omega}}_i^{(t+1)} \leftarrow (1-\tau)\,\hat{\boldsymbol{\omega}}_i^{(t)} + \tau\,\boldsymbol{\omega}_i^{(t)}, \boldsymbol{\omega}_i \in \{\boldsymbol{\theta}_i, \boldsymbol{\psi}_i\}$.
end for

where $\eta_i > 0$ is the penalty factor, $\ell_i^{(t_b)} = \ddot{\xi}_i(\boldsymbol{s}^{(t_b)}, \tilde{\boldsymbol{a}}^{(t_b)}) + \lambda_1\dot{\xi}_i(\boldsymbol{s}^{(t_b)}) + \lambda_2\xi_i(\boldsymbol{s}^{(t_b)})$ denotes the CBF constraint of agent i. The overall pipeline of SCOPE as Algorithm 1.

The output of the algorithm is multi-agent policies and surrogate dynamic model. Here $\boldsymbol{\omega}_i$ is the target network parameters of agent i.

5 Experiment

In this section, we present the experimental evaluation of SCOPE on multi-UAV pursuit–evasion environments and compare its performance with safe MARL baselines. For benchmarking, we consider the following methods.

- Lagrangian method: MADDPG [Lowe *et al.*, 2017] with a Lagrangian multiplier and MACPO [Gu *et al.*, 2022].
- Penalty-based method: MAP3O [Zhang *et al.*, 2022].
- Shield-based method MADDPG with a control. barrier functions [Cai *et al.*, 2021]
- Recovery RL method: MSMAR [Zhao *et al.*, 2025].

All the above algorithms use either on-policy or off-policy replay buffers, following their respective original implementations. Additionally, we include a comparison with the MADDPG algorithm [Lowe *et al.*, 2017] as a baseline. Baseline implementations partially rely on the OmniSafe repository [Ji *et al.*, 2024] for standard safe MARL components. We evaluate all methods in our multi-UAV pursuit–evasion environments, where multiple interceptor UAVs must collaboratively encircle a rogue intruder while satisfying safety constraints with respect to other agents, obstacles, and workspace boundaries. The environments include both static and moving obstacles that test the agents' ability to maximize cumulative rewards while adhering to the safety constraints formalized in Equation. (1). We additionally evaluate SCOPE in MuJoCo [Todorov *et al.*, 2012] to verify its effectiveness under physical dynamics, with details reported in Appendix B.

5.1 Multi-UAV Pursuit–Evasion Setup

Our multi-UAV pursuit–evasion environment simulates a team of $N = 4$ interceptor UAVs attempting to safely encircle a rogue UAV in a cluttered airspace. The environment is modeled as a continuous workspace with boundaries and several obstacles, where each interceptor is equipped with a short-range jamming device.The objective is to encircle the fleeing rogue UAV for on-board jamming while maintaining minimum separations among UAVs and from obstacles. For these experiments, we use a fixed safety radius r_{safe} and jamming range r_{jam}, task duration $T = 400$, and evaluate each trained policy on $N_{\text{E}} = 200$ test episodes. Detailed environment descriptions are provided in the supplementary material.

5.2 Evaluation Metrics

We evaluate all methods from both task performance and safety perspectives. Task performance is measured by:

1) *Success rate* SR: the fraction of episodes in which the interceptors successfully encircle the rogue UAV within the horizon T;
2) *Average time* AT: the average number of steps to success, where unsuccessful episodes are assigned T.

Safety performance is measured by:

3) *Average agent cost* AC: the average number of safety-constraint violations per agent during an episode:

$$\text{AC} = \frac{1}{N_{\text{E}}}\sum_{e=1}^{N_{\text{E}}}\frac{C_e}{N}, \tag{10}$$

where N_{E} is the number of evaluation episodes, N is the number of interceptors, and C_e denotes the total safety cost accumulated in episode e. Specifically, C_e consists of the magnitude of constraint violations against obstacles and other agents, and an additional cost for shield activation when a shielding or safety enforcement intervenes.

5.3 Performance Comparison

We evaluate SCOPE against the safe MARL baselines listed above under identical interaction budgets and environment configurations. All methods share the same reward design, termination conditions, observation space and action spaces. We report the mean performance over multiple random seeds, and evaluate each trained policy on test episodes using the metrics in Sec. 5.2.

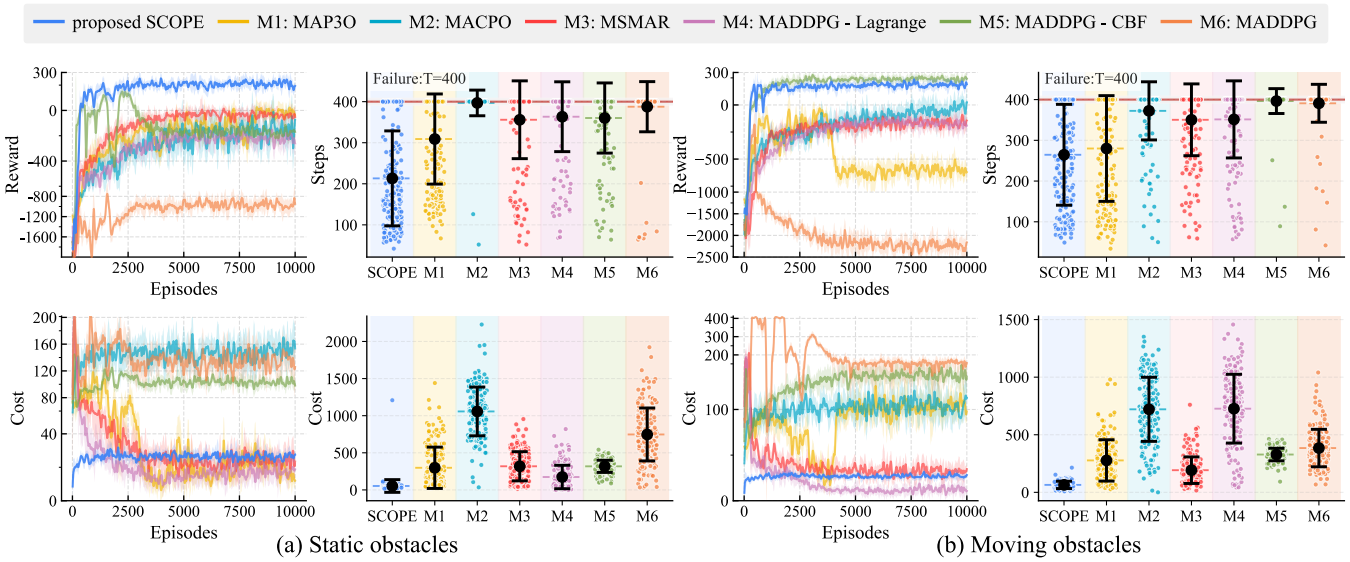

Figure 4: Comparison with baselines in multi-UAV pursuit–evasion under (a) static obstacles and (b) moving obstacles. Curves show training reward/cost, while scatter plots report evaluation Steps and accumulated safety cost; black markers denote means with error bars, and $T = 400$ indicates time-out failures. Our method is marked as SCOPE (blue).

Scenarios	Metrics	MAP3O	MACPO	MSMAR	MADDPG-Lagrange	MADDPG-CBF	MADDPG	SCOPE
Static Obstacles	SR(%) ↑	47.10 ±3.76	1.20 ±0.45	20.90 ±3.58	18.30 ±5.02	27.20 ±3.91	2.50 ±1.00	**83.10 ±1.52**
	AT(steps) ↓	312.26 ±7.01	396.88 ±1.13	355.36 ±7.61	366.40 ±8.68	354.62 ±4.96	393.66 ±3.47	**218.77** ±5.67
	AC ↓	77.01 ±4.95	258.983 ±4.312	75.59 ±11.94	45.02 ±2.05	80.19 ±1.32	187.26 ±4.02	**12.667** ±0.95
Moving Obstacles	SR(%) ↑	60.20 ±4.56	17.50 ±1.41	28.70 ±3.83	26.30 ±1.75	5.90 ±2.86	8.20 ±2.20	**61.40 ±5.61**
	AT(steps) ↓	**274.00 ±7.37**	371.71 ±1.80	353.54 ±4.31	357.63 ±3.71	388.67 ±5.63	383.49 ±4.47	279.38 ±13.59
	AC ↓	67.93 ±3.20	185.205 ±4.749	46.66 ±1.94	183.41 ±4.27	77.55 ±3.02	95.96 ±2.98	**16.07 ±0.55**

Table 1: Performance comparison between SCOPE and competing algorithms on the multi-UAV pursuit–evasion environment, evaluated in terms of Success rate (SR), Average time (AT), and Average agent cost (AC). The average values and standard deviations over 200 tests are reported. Bold denotes the best, and underline denotes the second best.

Figure 4 shows that SCOPE achieves higher episodic returns while maintaining low safety costs more consistently throughout training in both static- and moving-obstacle settings. Vanilla MADDPG exhibits large safety-cost spikes and unstable learning, and many runs time out at $T = 400$, indicating failures to complete encirclement under unconstrained exploration in cluttered scenes. MAP3O reduces safety costs relative to MADDPG, but converges to lower returns with longer episode lengths (Steps), while MACPO shows low SR and high AC, suggesting conservative policies that hinder efficient pursuit–evasion. MSMAR similarly suppresses safety costs, yet also converges to lower returns and longer Steps, indicating conservative recovery behaviors that slow down encirclement. MADDPG-Lagrange shows degraded performance in the moving-obstacle setting with increased costs and more failed episodes, highlighting the difficulty of balancing coupled constraints under non-stationary dynamics. MADDPG-CBF improves returns compared with penalty-based baselines, but still incurs substantially higher safety costs than SCOPE, consistent with frequent online interventions in dense interactions.

Table 1 further confirms that SCOPE achieves the strongest overall performance in both scenarios. With moving obstacles, SCOPE reaches the highest success rate of 61.40% and the lowest safety cost of 16.07, while keeping the completion time competitive at 279.38 steps. With static obstacles, SCOPE attains 83.10% success, the fastest completion time of 218.77 steps, and the lowest safety cost of 12.67. Other methods typically sacrifice either success and speed for safety, or safety for task completion, whereas SCOPE remains strong on both.

5.4 Ablation Studies

To examine the contribution of each component in SCOPE, we perform ablation studies by removing one module at a time while keeping the training protocol and hyperparameters fixed. Figure 5 plots the learning curves of episodic returns and safety costs. We conduct ablations in the moving-obstacle setting, as it is the most challenging scenario with non-stationary interactions and thus better differentiates the effects of each module.

Overall, SCOPE achieves the best safety-performance trade-off, converging to the highest return while main-

Method	SR(%) ↑	AT(steps) ↓	AC ↓
SCOPE	**61.12 ± 6.43**	**279.07 ± 15.67**	21.523 ± 0.814
SCOPE^{-GA}	60.62 ± 4.53	286.60 ± 8.99	26.048 ± 0.545
SCOPE^{-HSE}	1.88 ± 0.63	395.19 ± 1.61	276.909 ± 4.158
SCOPE^{-SPDM}	54.12 ± 3.35	290.88 ± 13.11	**19.908 ± 1.029**
SCOPE^{-All}	3.12 ± 3.28	393.58 ± 7.32	139.526 ± 8.608

Table 2: Ablation results of SCOPE on the multi-UAV pursuit–evasion task, evaluated by SR, AT, and AC.

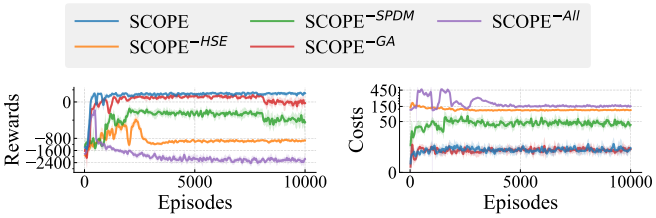

Figure 5: Training curves for SCOPE and its ablated variants.

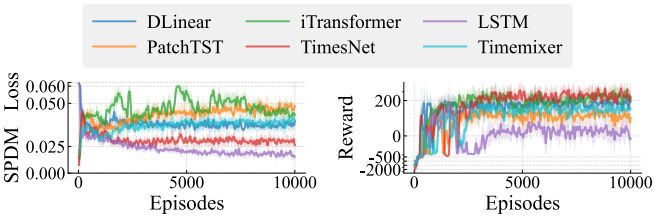

Figure 6: SPDM loss and reward with different SPDM backbones.

taining consistently low cost. Removing the SPDM (SCOPE^{-SPDM}) slows convergence and reduces the final return, indicating that short-horizon previews provide anticipatory information for coordinated pursuit–evasion. Removing the hierarchical safety enforcement (SCOPE^{-HSE}) incurs persistently higher costs and leads to a markedly lower return plateau, suggesting that execution-time safety correction is critical and that frequent constraint violations can hinder policy learning. Removing the preview-fused actor–critic network enhancements (SCOPE^{-GA}) results in only a minor drop in return with comparable costs, implying that the gating/attention modules mainly improve coordination and credit assignment. Finally, removing both preview and safety enforcement (SCOPE^{-All}) yields the worst performance, characterized by pronounced early cost spikes and a low-return regime throughout training.

5.5 Component Study on SPDM

We study the impact of the predictive backbone used in SPDM. To ensure a fair comparison, we keep the environment, RL algorithm, reward design, and hierarchical safety enforcement unchanged, and replace the SPDM with six models: DLinear, PatchTST, iTransformer, TimesNet, LSTM, and TimeMixer.

As shown in Figure 6 (right), TimesNet and iTransformer yield the highest and most stable returns, while DLinear achieves competitive performance with fast early improvement, suggesting that different backbones trade off early learning speed and asymptotic performance. PatchTST and TimeMixer converge to lower returns, and LSTM performs worst, indicating that the predictor choice can affect cooperative pursuit–evasion and training robustness. Notably, Figure 6 (left) shows that lower prediction loss does not necessarily imply better control performance: although LSTM attains a low SPDM loss, it leads to poor returns, suggesting that minimizing the regression loss does not necessarily lead to optimal control performance.

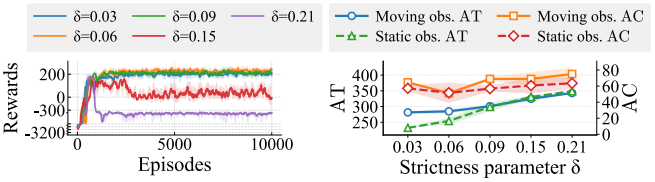

Figure 7: Sensitivity of SCOPE to the strictness parameter δ.

5.6 Parameter Sensitivity Analysis

We analyze SCOPE's sensitivity to the strictness parameter δ by varying δ while keeping all other hyperparameters fixed. Since δ directly sets the tightening level of the enforced CBF constraint, it affects the size of the feasible action set and thus influences both task execution speed and constraint satisfaction. We evaluate each setting using the metrics in Sec. 5.2.

As shown in Figure 7, SCOPE is stable for $\delta \leq 0.09$ with similar convergence returns, while larger margins degrade learning. $\delta = 0.15$ leads to a clear return drop and $\delta = 0.21$ collapses to a low-return regime, suggesting that overly strict tightening restricts the feasible action set and hinders coordinated pursuit. Meanwhile, increasing δ monotonically increases the time-related metric R_t, indicating slower encirclement, whereas the safety cost R_c is minimized at a moderate margin and rises again for larger δ, reflecting a clear safety and efficiency trade-off. We therefore use $\delta = 0.06$ in the remaining experiments.

6 Conclusion

In this paper, we proposed Safety-Constrained Online Preview Enforcement (SCOPE), a MARL-based algorithm for multi-UAV encirclement in pursuit–evasion. SCOPE integrates a Safety-Preview Dynamics Model (SPDM) for short-horizon anticipation with hierarchical safety enforcement and look-ahead feasibility checking to maintain forward feasible safety. Extensive experiments in challenging scenarios, together with additional evaluations in MuJoCo, demonstrate that SCOPE consistently improves task performance while substantially reducing safety cost compared to state-of-the-art safe MARL baselines.

Future work will focus on scaling SCOPE to larger swarms and more diverse team sizes, where coordination and safety constraints become increasingly complex. We also plan to incorporate richer physical constraints and more realistic onboard limitations, and to study adaptive safety margins that adjust online to the environment dynamics and risk level.

Ethical Statement

This work targets civilian public-safety applications, such as interception of unauthorized drones in restricted airspace. The proposed method is not intended for military, offensive, or harmful use, and should be deployed only under legal regulations and proper human oversight.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No.62472220, No.62571240, No.U22B2062, No.62502204), the Natural Science Fund of Jiangsu Province (Grant No.BK20251381), the Open Project Program of State Key Laboratory of Novel Software Technology, Nanjing University (KFKT2025B14), the open research fund of National Mobile Communications Research Laboratory, Southeast University(Grant 2026D06), the Fundamental Research Funds for the Central Universities (NS2025041), Collaborative Innovation Center of Novel Software Technology and Industrialization. This work was partially supported by High Performance Computing Platform of Nanjing University of Aeronautics and Astronautics.

References

[Achiam *et al.*, 2017] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 22–31. PMLR, 06–11 Aug 2017.

[Alshiekh *et al.*, 2018] Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018.

[Brorholt *et al.*, 2024] Asger Horn Brorholt, Kim Guldstrand Larsen, and Christian Schilling. Compositional shielding and reinforcement learning for multi-agent systems. *arXiv preprint arXiv:2410.10460*, 2024.

[Cai *et al.*, 2021] Zhiyuan Cai, Huanhui Cao, Wenjie Lu, Lin Zhang, and Hao Xiong. Safe multi-agent reinforcement learning through decentralized multiple control barrier functions. *CoRR*, abs/2103.12553, 2021.

[Chen *et al.*, 2025] Jiayu Chen, Chao Yu, Guosheng Li, Wenhao Tang, Shilong Ji, Xinyi Yang, Botian Xu, Huazhong Yang, and Yu Wang. Online planning for multi-uav pursuit-evasion in unknown environments using deep reinforcement learning. *IEEE Robotics and Automation Letters*, 10(8):8196–8203, 2025.

[Chow *et al.*, 2015] Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a cvar optimization approach. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[Chow *et al.*, 2018] Yinlam Chow, Mohammad Ghavamzadeh, Lucas Janson, and Marco Pavone. Risk-constrained reinforcement learning with percentile risk criteria. *Journal of Machine Learning Research*, 18(167):1–51, 2018.

[Dong *et al.*, 2020] Fei Dong, Keyou You, and Shiji Song. Target encirclement with any smooth pattern using range-based measurements. *Automatica*, 116:108932, 2020.

[Du *et al.*, 2021] Wenbo Du, Tong Guo, Jun Chen, Biyue Li, Guangxiang Zhu, and Xianbin Cao. Cooperative pursuit of unauthorized uavs in urban airspace via multi-agent reinforcement learning. *Transportation Research Part C: Emerging Technologies*, 128:103122, 2021.

[Duan *et al.*, 2024] Wei Duan, Jie Lu, and Junyu Xuan. Group-aware coordination graph for multi-agent reinforcement learning. In Kate Larson, editor, *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 3926–3934. International Joint Conferences on Artificial Intelligence Organization, 8 2024. Main Track.

[Gu *et al.*, 2022] Shangding Gu, Jakub Grudzien Kuba, Munning Wen, Ruiqing Chen, Ziyan Wang, Zheng Tian, Jun Wang, Alois Knoll, and Yaodong Yang. Multi-agent constrained policy optimisation, 2022.

[Hafez *et al.*, 2014] A.T. Hafez, M. Iskandarani, S.N. Givigi, S. Yousefi, and A. Beaulieu. Uavs in formation and dynamic encirclement via model predictive control. *IFAC Proceedings Volumes*, 47(3):1241–1246, 2014. 19th IFAC World Congress.

[Hazra *et al.*, 2025] Somnath Hazra, Pallab Dasgupta, and Soumyajit Dey. Incentivizing safer actions in policy optimization for constrained reinforcement learning. In James Kwok, editor, *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 5318–5326. International Joint Conferences on Artificial Intelligence Organization, 8 2025. Main Track.

[Ji *et al.*, 2024] Jiaming Ji, Jiayi Zhou, Borong Zhang, Juntao Dai, Xuehai Pan, Ruiyang Sun, Weidong Huang, Yiran Geng, Mickel Liu, and Yaodong Yang. Omnisafe: An infrastructure for accelerating safe reinforcement learning research. *Journal of Machine Learning Research*, 25(285):1–6, 2024.

[Kim *et al.*, 2010] Tae-Hyoung Kim, Shinji Hara, and Yutaka Hori. Cooperative control of multi-agent dynamical systems in target-enclosing operations using cyclic pursuit strategy. *International Journal of Control*, 83(10):2040–2052, 2010.

[Lee and Kwon, 2024] Dongsu Lee and Minhae Kwon. Episodic future thinking mechanism for multi-agent reinforcement learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 11570–11601. Curran Associates, Inc., 2024.

[Li and Azizan, 2024] Zeyang Li and Navid Azizan. Safe multi-agent reinforcement learning with convergence to generalized nash equilibrium, 2024.

[Li *et al.*, 2023] Yan Li, Xuejun Zhang, Yuanjun Zhu, and Ziang Gao. A uav path planning method in three-dimensional urban airspace based on safe reinforcement learning. In *2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC)*, pages 1–7, 2023.

[Liu *et al.*, 2020] Yongshuai Liu, Jiaxin Ding, and Xin Liu. Ipo: Interior-point policy optimization under constraints. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):4940–4947, Apr. 2020.

[Lowe *et al.*, 2017] Ryan Lowe, YI WU, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[Lv *et al.*, 2024] Xinyuan Lv, Chi Peng, and Jianjun Ma. Control barrier function-based collision avoidance guidance strategy for multi-fixed-wing uav pursuit-evasion environment. *Drones*, 8(8), 2024.

[Ma *et al.*, 2020] Junchong Ma, Huimin Lu, Junhao Xiao, Zhiwen Zeng, and Zhiqiang Zheng. Multi-robot target encirclement control with collision avoidance via deep reinforcement learning. *Journal of Intelligent & Robotic Systems*, 99(2):371–386, Aug 2020.

[Qu *et al.*, 2024] Yang Qu, Jinming Ma, and Feng Wu. Safety constrained multi-agent reinforcement learning for active voltage control. In Kate Larson, editor, *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 184–192. International Joint Conferences on Artificial Intelligence Organization, 8 2024. Main Track.

[Stooke *et al.*, 2020] Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive safety in reinforcement learning by PID lagrangian methods. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9133–9143. PMLR, 13–18 Jul 2020.

[Sun *et al.*, 2023] Lijun Sun, Yu-Cheng Chang, Chao Lyu, Chin-Teng Lin, and Yuhui Shi. Matrixworld: A pursuit-evasion platform for safe multi-agent coordination and autocurricula. *arXiv preprint arXiv:2307.14854*, 2023.

[Todorov *et al.*, 2012] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

[Valianti *et al.*, 2024] Panayiota Valianti, Kleanthis Malialis, Panayiotis Kolios, and Georgios Ellinas. Cooperative multi-agent jamming of multiple rogue drones using reinforcement learning. *IEEE Transactions on Mobile Computing*, 23(12):12345–12359, 2024.

[Wabersich and Zeilinger, 2023] Kim P. Wabersich and Melanie N. Zeilinger. Predictive control barrier functions: Enhanced safety mechanisms for learning-based control. *IEEE Transactions on Automatic Control*, 68(5):2638–2651, 2023.

[Xiao and Feroskhan, 2024] Jiaping Xiao and Mir Feroskhan. Learning multipursuit evasion for safe targeted navigation of drones. *IEEE Transactions on Artificial Intelligence*, 5(12):6210–6224, 2024.

[Yu *et al.*, 2024] Hao Yu, Xiuxia Yang, Yi Zhang, and Zijie Jiang. Cooperative target fencing control for unmanned aerial vehicle swarm with collision, obstacle avoidance, and connectivity maintenance. *Drones*, 8(7), 2024.

[Zhang *et al.*, 2022] Linrui Zhang, Li Shen, Long Yang, Shixiang Chen, Xueqian Wang, Bo Yuan, and Dacheng Tao. Penalized proximal policy optimization for safe reinforcement learning. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 3744–3750. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.

[Zhang *et al.*, 2023a] Linrui Zhang, Qin Zhang, Li Shen, Bo Yuan, Xueqian Wang, and Dacheng Tao. Evaluating model-free reinforcement learning toward safety-critical tasks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(12):15313–15321, Jun. 2023.

[Zhang *et al.*, 2023b] Ruilong Zhang, Qun Zong, Xiuyun Zhang, Liqian Dou, and Bailing Tian. Game of drones: Multi-uav pursuit-evasion game with online motion planning by deep reinforcement learning. *IEEE Transactions on Neural Networks and Learning Systems*, 34(10):7900–7909, 2023.

[Zhang *et al.*, 2024] Lijun Zhang, Lin Li, Wei Wei, Huizhong Song, Yaodong Yang, and Jiye Liang. Scalable constrained policy optimization for safe multi-agent reinforcement learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 138698–138730. Curran Associates, Inc., 2024.

[Zhao *et al.*, 2025] Yang Zhao, Wenzhe Zhao, and Xuelong Li. Msmar-rl: Multi-step masked-attention recovery reinforcement learning for safe maneuver decision in high-speed pursuit-evasion game. In James Kwok, editor, *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 311–319. International Joint Conferences on Artificial Intelligence Organization, 8 2025. Main Track.

[Zhou *et al.*, 2025] Minnan Zhou, Mustafa Shaikh, Vatsalya Chaubey, Patrick Haggerty, Shumon Koga, Dimitra Panagou, and Nikolay Atanasov. Control strategies for pursuit-evasion under occlusion using visibility and safety barrier functions. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12863–12869, 2025.