

Resolving Conflicting Constraints in Multi-Agent Reinforcement Learning with Layered Safety

Jason J. Choi^{*1}, Jasmine Jerry Aloor^{*2}, Jingqi Li^{*1}, Maria G. Mendoza¹,
Hamsa Balakrishnan^{†2}, Claire J. Tomlin^{†1}

¹University of California, Berkeley Email: {jason.choi, jingqili, maria_mendoza, tomlin}@berkeley.edu

²Massachusetts Institute of Technology Email: {jjaloor, hamsa}@mit.edu

^{*}Equal contributions [†]Equal advising

Abstract—Preventing collisions in multi-robot navigation is crucial for deployment. This requirement hinders the use of learning-based approaches, such as multi-agent reinforcement learning (MARL), on their own due to their lack of safety guarantees. Traditional control methods, such as reachability and control barrier functions, can provide rigorous safety guarantees when interactions are limited only to a small number of robots. However, conflicts between the constraints faced by different agents pose a challenge to safe multi-agent coordination.

To overcome this challenge, we propose a method that integrates multiple layers of safety by combining MARL with safety filters. First, MARL is used to learn strategies that minimize multiple agent interactions, where multiple indicates more than two. Particularly, we focus on interactions likely to result in conflicting constraints within the engagement distance. Next, for agents that enter the engagement distance, we prioritize pairs requiring the most urgent corrective actions. Finally, a dedicated safety filter provides tactical corrective actions to resolve these conflicts. Crucially, the design decisions for all layers of this framework are grounded in reachability analysis and a control barrier-value function-based filtering mechanism.

We validate our Layered Safe MARL framework in 1) hardware experiments using Crazyflie drones and 2) high-density advanced aerial mobility (AAM) operation scenarios, where agents navigate to designated waypoints while avoiding collisions. The results show that our method significantly reduces conflict while maintaining safety without sacrificing much efficiency (i.e., shorter travel time and distance) compared to baselines that do not incorporate layered safety. [Project Webpage]¹

I. INTRODUCTION

A. Motivation

Collision-free operation is a fundamental requirement for multi-robot coordination tasks, such as formation control [1], multi-robot payload transport [2], and autonomous navigation [3]. When only two agents interact, there is a single collision-avoidance constraint, which can be easily managed using a safety filter. However, with multiple nearby agents, the resolution of a constraint between two agents can conflict with a constraint involving a third agent. These conflicts may result in suboptimal task performance, such as creating a severe gridlock that prevents agents from taking actions to achieve their tasks. More crucially, the inability to simultaneously satisfy all constraints can result in an agent taking an action that makes collision inevitable. In particular, this issue poses

a significant safety risk in high-density scenarios like air taxi operations for Advanced Air Mobility (AAM) [4].

Prior works have addressed safe multi-robot coordination problems by using model-based control methods like control barrier functions (CBFs) [5] and reachability analysis [6]. Although CBFs and reachability provide a framework for safety assurance, they generally offer rigorous guarantees only when a single safety constraint is considered. The fundamental challenge in extending these methods to the multi-agent case is that the intersection of the safe sets corresponding to individual constraints (each derived from a pair of agents) does not necessarily represent the true safe set when all constraints are considered together (see Figure 2 for an example). In the Hamilton-Jacobi (HJ) reachability literature, the gap between these two regions is referred to as the “leaky corner” [7]. Agents that enter a leaky corner can no longer satisfy all safety constraints simultaneously and are inevitably forced to violate at least one. Unfortunately, identifying leaky corners without recomputing the reachability analysis from scratch while incorporating all constraints remains an open problem [8, 9]. Performing reachability analysis or designing CBFs for all possible combinatorial interaction scenarios is computationally intractable. In summary, the fundamental challenge in achieving scalability with such control-theoretic methods in multi-agent settings lies in handling conflicting constraints.

In this work, we combine the control barrier-value function (CBVF) [10], which is a CBF design method based on Hamilton-Jacobi reachability, with multi-agent reinforcement learning (MARL) into a layered safety architecture. This integration is driven by the essential role MARL can play in learning to strategically optimize task performance in multi-agent scenarios while proactively navigating potential conflicting constraints, which helps achieve safer and more effective behaviors. As a result, our approach enhances both safety and performance to a level that neither safe control methods nor MARL alone could achieve.

B. Contributions

- 1) *Architecture*: We propose a layered architecture that combines safety-informed MARL-based policy and CBVF-based safety filtering mechanism (Figure 1), which can significantly mitigate the issues arising from

¹Project Webpage: <https://dinamo-mit.github.io/Layered-Safe-MARL/>

conflicting constraints, such as inefficiency due to grid-lock and the leaky corner problem.

- 2) *Training method*: We propose a method to incorporate a CBVF-based safety filter into the training of MARL, considering two key aspects. First, a main challenge in this safety-constrained training is that the conservativeness introduced by safety filtering can hinder the exploration necessary for MARL to learn an effective policy. To address this, we introduce curriculum learning into the application of the safety filter, carefully balancing safety and exploration. Second, based on reachability analysis, we derive a conservative estimate of the safe region that is free from the issue of conflicting constraint (represented by the range r_{conflict} in Figure 1). Based on this estimate, the MARL policy is informed to minimize entry into this region, thereby avoiding potential conflicting constraints. Crucially, unlike many existing methods [11, 12], our proposed training approach does not impose safety through penalty terms directly penalizing the safety violation. Instead, MARL learns to enhance safety by making strategic decisions that mitigate conflicting constraints. This indirect approach significantly reduces unnecessary conservativeness, a common side effect of safe reinforcement learning-based methods.

- 3) *Experimental validation*: We conduct hardware experiments using Crazyflie drones and perform simulations of high-density AAM scenarios to validate our hypothesis.

The remainder of this paper is organized as follows. Section II provides background on safety for multi-agent coordination. Section III describes the system, environment, and problem statement. Section IV presents the safety analysis of multi-agent problems under collision avoidance constraints. Sections V and VI present our proposed Layered Safe MARL approach, the experiments performed, and the results obtained. We discuss some limitations of our approach in Section VII. Finally, we conclude and propose future work in Section VIII.

II. RELATED WORK

A. Core related works—safety for multi-agent problems

Classic Control barrier function (CBF)-based approaches. CBFs [13] are used to design safe controllers via the principle of set invariance, and their application to safe multi-agent coordination has been explored in [5, 14, 15]. A primary challenge in employing CBFs lies in constructing a valid CBF, which often requires system-specific, handcrafted designs [5]. In [15, 14], a more generic design principle based on exponential CBFs [16] is employed; however, this approach does not address control input bounds. Another common limitation of existing methods is the treatment of multiple, potentially conflicting CBF constraints, which can lead to infeasibility. To address this, we adopt the CBVF-based framework to construct valid CBFs and handle multiple safety constraints in multi-agent coordination via a layered safety architecture.

Neural CBF-based approaches. While learning-based methods [17, 12, 18, 19] are proposed to design approximate CBFs,

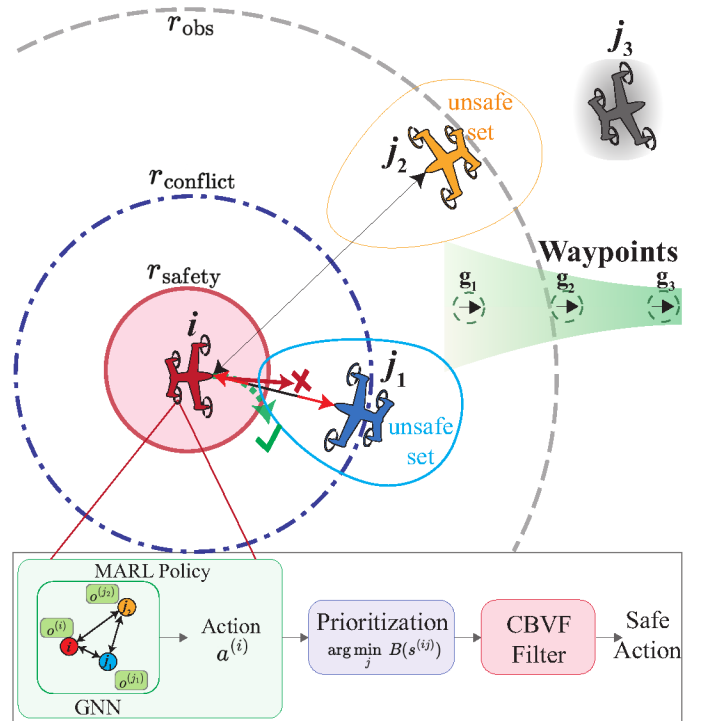


Fig. 1: The figure shows our approach using an example scenario of four agents. Agent i must reach the waypoints shown on the right. Our Layered Safe MARL framework consists of three key components, and we describe it as applied through agent i : 1) The MARL policy generates an action based on the observation within the range r_{obs} while aiming to reduce the likelihood of entering other agents' potential conflict range r_{conflict} . 2) The prioritization module identifies the most critical neighboring agent in a potential collision scenario by evaluating the CBVF. In this example, agent j_1 is within the potential conflict region and forms a *potential collision pair*. 3) The CBVF safety filter adjusts the action to ensure safe navigation.

they lack deterministic safety guarantees due to the nonconvexity of the learning problems. Graphical CBF (GCBF) in [12, 18] offers a CBF based on local observations under multi-agent interaction, but how it learns to handle multiple constraints is not explicitly examined. Discrete Graphical Proximal Policy Optimization (DG-PPO) [19] proposes a model-free approach to learning decentralized CBFs and a safe control policy optimizing the task objective. Unlike DG-PPO, our approach leverages model-based information to compute Control Barrier Value Functions (CBVFs) [10] for pairwise collision avoidance, thereby ensuring deterministic safety guarantees. Finally, the aforementioned methods focus on learning safety certificates and policies for uncertain dynamical systems, often with high-dimensional system states. In contrast, our work specifically addresses the challenge of conflicting constraints in multi-agent interactions—a critical issue that persists even when each agent's dynamics can be effectively represented by simple, low-dimensional models.

Reachability for multi-agent interaction. Classical HJ reachability analysis computes the set of states that are guaranteed to be safe by computing the optimal control value function with dynamic programming. Prior works have investigated the reachability analysis for the special case of three-agent interaction [20] and using the value function to guide the planner, and its optimal control law is used in the tracking controller for safe multi-agent interaction [6]. Due to the curse of dimensionality in dynamic programming [21], the applicability of HJ reachability to high-dimensional systems is inherently limited. Recent work leveraged deep learning techniques to learn high-dimensional reachability [22, 23, 24, 25, 26], demonstrating their use in multi-agent collision avoidance scenarios. However, the learned solution does not generalize to new scenarios involving different agents. Additionally, the question of how to certify the safety of the learned safe set is still an open research topic [27, 28, 26]. Finally, alternative methods for solving reachable sets with over-approximation have been used in the context of multi-agent problems and air traffic management [29, 30].

Safe multi-agent reinforcement learning (MARL). A common approach to safe MARL is through constrained Markov decision processes (CMDPs) [31]. In theory, CMDPs have no duality gap under certain assumptions [32], but in practice, training with PPO-Lagrangian [33], and its multi-agent variant [34] often suffers from instability due to suboptimal policies and inaccurate Lagrange multipliers. Another approach is shielded MARL, which uses safety filters [35, 11] to enforce safety during training and deployment. Originally introduced for single-agent RL [36], this method has been extended to multi-agent settings [37]. However, designing a shielding policy remains challenging due to the curse of dimensionality.

B. Other related works

Multi-agent reinforcement learning. Multi-agent extensions of single-agent RL algorithms, such as PPO [38] and DDPG [39], include MA-PPO [40] and MA-DDPG [41], both of which assume full observability. However, in many real-world applications, such as autonomous driving, agents have to make decisions based on their local information and coordinate effectively with other agents. A key challenge in MARL is the decentralized decision-making under partial observation. InforMARL [42] leverages graphical neural networks for information sharing to develop an efficient, coordinated learning framework for acquiring a high-performance MARL policy. Our approach builds on InforMARL to allow agents to make decentralized decisions based on their local observations.

Control and game-theoretic methods. In collaborative multi-agent settings, Model Predictive Control (MPC) has been used to ensure safe control [43, 44, 45, 46, 15] and its integration with MARL is explored in [47]. However, the complexity of the constrained optimization involved in MPC often limits its real-time execution in complex systems. When agents pursue distinct objectives, the problem becomes a non-cooperative game, with various solution approaches proposed in prior work [48, 49, 50, 51, 52, 53]. However, treating the safety

constraints in the game-theoretic solutions remains an open research challenge [54, 55].

Collision avoidance & conflict resolution for air traffic control. With the growing interest in AAM applications such as drone deliveries and air taxi services, developing a scalable low-altitude air traffic management system that is automated or semi-automated has become an urgent need. Compared to current aviation, AAM operations are expected to be large-scale, ad hoc, on-demand, and dynamic. These characteristics motivate the development of a new air traffic management (ATM) framework that can achieve scalable, efficient, and collision-free operations [4, 56].

Existing work on collision avoidance and conflict resolution for ATM is categorized into *strategic deconfliction*, which focuses on preemptive deconfliction, and *tactical deconfliction*, which focuses on imminent proactive collision avoidance. A substantial body of work leverages control theory to design methods for strategic deconfliction. An early work proposed a flight mode switching framework derived from a hybrid automaton and reachability-based analysis [57]. As this method suffers from the computational complexity of HJ reachability, [20] uses a mixed integer program to assign avoidance responsibilities and resolve conflicts cooperatively. The work in [58] alternatively organizes vehicles into platoons on structured air highways, treating each platoon as a coordinated entity. While these methods provide strong safety guarantees, they rely on a predefined set of coordination rules for those guarantees to hold. Additionally, the approach in [59] uses preemptive strategic speed adjustments to prevent perceived conflicts without requiring controller intervention. Finally, a negotiation-based framework is introduced in [60] for collision-free strategic planning.

In parallel, the aviation community employs tactical collision avoidance modules as the final layer for safety. The Traffic Alert and Collision Avoidance System (TCAS) is an onboard system developed in the 1980s for conventional airliners, designed to detect and prevent collisions through vertical separation [61]. A method for tactical collision avoidance through horizontal resolution is also proposed in [62]. The successor of TCAS, the Airborne Collision Avoidance System (ACAS) X, integrates predictive modeling with real-time sensor inputs [63] using a partially observable Markov decision process framework. These existing methods crucially rely on the assumption that no more than two vehicles are involved in a single conflict resolution. This assumption is typically upheld by the upstream strategic deconfliction decisions.

Various methods in both strategic and tactical deconfliction are integrated further into layered, hierarchical decision-making architectures, enhancing the safety of ATM [64, 65, 66]. Our work is inspired by these layered approaches in the aviation community; however, the separations between layers underlying the existing approaches do not directly apply to high-volume AAM scenarios. As such, we have to consider how to achieve safe collision avoidance in instances of simultaneous multi-vehicle engagement.

Finally, MARL-based methods have also been explored in

air traffic control to ensure tactical deconfliction through pre-conditioned strategic planning [67], demonstrating improved safety and efficiency over rule-based methods. However, the available actions of each agent in this work are limited to the adjustment of speed or position.

III. PROBLEM FORMULATION

In this section, we define the system, environment, each agent's dynamics and their safety requirement, and the problem statement.

1) *Preliminaries*: We formulate our multi-agent system as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) defined by the tuple $\langle N, S, O, \mathcal{A}, P, R, \gamma \rangle$, where:

- N is the number of agents
- $s^{(i)} \in \mathbb{R}^D$ is the state of each agent with D as the state dimension, including their position variables,
- $s \in S = \mathbb{R}^{N \times D}$ is the environment state, which is the concatenation of each agent's states and the state space of the environment, respectively,
- $o^{(i)} = O(s^{(i)}) \in \mathbb{R}^d$ is the observation of agent i ,
- $a^{(i)} \in \mathcal{A}$ is the action space for agent i . $\mathbf{a}^{(i)}$ denotes the sequence of actions for timesteps $k = 0, 1, \dots$,
- $P(s'|s, a)$ is the transition probability from s to s' given the joint action a , the concatenation of each agent's actions,
- $R(o^{(i)}, a^{(i)})$ is the common reward function of all agents,
- $\gamma \in [0, 1)$ is the discount factor.

The objective is to find a policy $\Pi = (\pi^{(1)}, \dots, \pi^{(N)})$, where $\pi^{(i)}(a^{(i)}|o^{(i)})$ is agent i 's policy that selects an action based on its observation.

2) *Agent's dynamics & safety constraint*: We consider each agent's dynamics as a sampled data system, meaning that their underlying physical dynamics evolve continuously in time, but their actions are updated at discrete timesteps. Their continuous dynamics are given as

$$\dot{s}^{(i)}(t) = f^{(i)}(s^{(i)}(t), a^{(i)}(t)), \quad s^{(i)}(0) = s_0^{(i)}, \quad (1)$$

and their action is updated at every sampling time Δt —i.e. the action sequence $\mathbf{a}^{(i)}$ maps to the signal in time given as $a^{(i)}(t) \equiv a_k^{(i)}$ for $t \in [k\Delta t, (k+1)\Delta t)$. Their discrete-time state is given as $s_k^{(i)} = s^{(i)}(k\Delta t)$.

The primary safety constraint we consider in this work is the collision avoidance between agents. For all time $t \geq 0$, agents must satisfy

$$\text{dist}(s^{(i)}(t), s^{(j)}(t)) \geq r_{\text{safety}}, \quad \text{for } \forall i \neq j, \quad (2)$$

where r_{safety} is the safety distance.

In the subsequent safety analysis, we consider the relative dynamics between a pair of agents, (i, j) . We define the relative state between the agents, which can be given as $s^{(ij)} := \text{rel}(s^{(i)}, s^{(j)})$, where rel is a mapping from two agents' states to the relative state. We assume that relative position variables are part of $s^{(ij)}$; thus, dist can be defined

based on $s^{(ij)}$. The dynamics of the relative state are described by

$$\dot{s}^{(ij)}(t) = f^{(ij)}(s^{(ij)}(t), a^{(i)}(t), a^{(j)}(t)), \quad (3)$$

which is derived from (1).

3) *Observations*: For each agent to learn an effective policy for performance and safety, the observations $o^{(i)}$ need to contain adequate information. We make the following assumptions that are generic for many multi-agent robot tasks.

- Each agent i 's observation $o^{(i)}$ consists of its local observations of other agents and entities relevant to their task goals (e.g., goal location) within their observation range defined as r_{obs} and any additional information needed for its task. Thus, the reward given to agent i at each timestep, $R(o^{(i)}, a^{(i)})$, is defined based on its local observation and action.
- We define $I(i) : \mathbb{N} \rightarrow 2^{\mathbb{N}}$ as the index set of the agents within the observation range of agent i . We assume that $o^{(i)}$ contains information that can be used to reconstruct $s^{(ij)}$ from $o^{(i)}$ for all $j \in I(i)$. Thus, for agent i , with its observation, it is feasible to execute a feedback policy on $s^{(ij)}$ if agent j is within its observation range. This assumption will be used in the design of our safety framework.

4) *Problem statement*: To sum up, the decentralized multi-agent coordination problem, subjected to the collision avoidance constraint we consider in this work, can be described as

$$\begin{aligned} \max_{\pi^{(i)}} \mathbb{E} \left[\sum_{k=0}^{\infty} \gamma^k R(o_k^{(i)}, a_k^{(i)}) \right] \\ \text{s.t. } s_{k+1} \sim P(s \mid s_k, a_k) \\ a_k^{(i)} \sim \pi^{(i)}(a^{(i)} \mid o_k^{(i)}) \end{aligned} \quad (4)$$

$$\text{dist}(s^{(i)}(t), s^{(j)}(t)) \geq r_{\text{safety}}, \quad \text{for } \forall i \neq j, \forall t \geq 0,$$

where each agent's action $a_k^{(i)}$ is determined by their policy $\pi^{(i)}$, based on their local observations. The agent learns to maximize its objective subject to its collision avoidance constraint.

IV. SAFETY ANALYSIS

In this section, we present the safety analysis of the multi-agent problem under collision avoidance constraints. Specifically, we first derive the safety analysis for a pair of agents and then investigate how it applies to the multi-agent scenario.

A. Collision avoidance for a pair of agents

To ensure $\text{dist}(s^{(ij)}(t)) \geq r_{\text{safety}}$ for all $t \geq 0$, we consider the following cost function, which captures the closest relative distance along the trajectory:

$$J(s_0^{(ij)}, \mathbf{a}^{(i)}, \mathbf{a}^{(j)}) := \min_{t \in [0, \infty)} \text{dist}(s^{(ij)}(t)). \quad (5)$$

If $J(s_0^{(ij)}, \mathbf{a}^{(i)}, \mathbf{a}^{(j)}) \geq r_{\text{safety}}$, the agents i and j are rendered safe (collision-free) by their actions.

1) *Reachability analysis for computing the maximal safe set*: The agent pair prioritizing safety would want to maximize (5) to move away from each other. From this intuition, we can consider the following optimal control problem

$$V(s_0^{(ij)}) := \max_{\mathbf{a}^{(i)}, \mathbf{a}^{(j)}} J(s_0^{(ij)}, \mathbf{a}^{(i)}, \mathbf{a}^{(j)}). \quad (6)$$

Solving V is a specific type of reachability problem called the minimal Backward Reachable Tube (BRT) problem [68]. To see this, consider $\mathcal{L}^{(ij)} = \{s^{(ij)} \mid \text{dist}(s^{(ij)}) < r_{\text{safety}}\}$, the near-collision region, as the target set. The minimal BRT of $\mathcal{L}^{(ij)}$ is defined as

$$\text{BRT}(\mathcal{L}^{(ij)}) := \{s_0^{(ij)} \mid \forall \mathbf{a}^{(i)}, \mathbf{a}^{(j)}, \exists t \geq 0 \text{ s.t. } s^{(ij)}(t) \in \mathcal{L}^{(ij)}\}, \quad (7)$$

which encapsulates a region from which no action sequence can prevent the relative state from entering the near-collision region $\mathcal{L}^{(ij)}$. Using the definition in (6), we can express $\text{BRT}(\mathcal{L}^{(ij)})$ as $\{s_0^{(ij)} \mid V(s_0^{(ij)}) < r_{\text{safety}}\}$.

The complement of $\text{BRT}(\mathcal{L}^{(ij)})$ becomes the *maximal safe set* from which the agent pair can avoid collisions since it encompasses all the states from which there exist action sequences $\mathbf{a}^{(i)}$ and $\mathbf{a}^{(j)}$ that can avoid collision. This maximal safe set is denoted as

$$\mathcal{S}^{(ij)} := \{s_0^{(ij)} \mid \exists \mathbf{a}^{(i)}, \mathbf{a}^{(j)}, \text{ s.t. } \forall t \geq 0, s^{(ij)}(t) \notin \mathcal{L}^{(ij)}\}, \quad (8)$$

and satisfies

$$\mathcal{S}^{(ij)} = (\text{BRT}(\mathcal{L}^{(ij)}))^c = \{s_0^{(ij)} \mid V(s_0^{(ij)}) \geq r_{\text{safety}}\}. \quad (9)$$

We use the open-source library in [69] to compute (6) and $\mathcal{S}^{(ij)}$, which computes the Hamilton-Jacobi (HJ) partial differential equation (PDE) associated with the BRT problem [70].

Running example. We consider the reduced-order dynamics of an autonomous air taxi given as

$$\dot{x} = v \cos \theta, \quad \dot{y} = v \sin \theta, \quad \dot{\theta} = \omega, \quad \dot{v} = a, \quad (10)$$

where the robot state consists of $s^{(i)} = [x; y; \theta; v]$, representing the positions, heading, and speed. The allowable actions are $\mathbf{a}^{(i)} = [\omega, a]$, corresponding to the angular rate and the longitudinal acceleration, respectively. The speed is limited to the range of $[v_{\min}, v_{\max}]$. The action space is defined as $\mathcal{A} = [-\omega_{\max}, \omega_{\max}] \times [a_{\min}, a_{\max}]$. The parameters we use are defined in Table I and are explained in more detail in Section VI-D.

The relative state $s^{(ij)} = [x^{(ij)}; y^{(ij)}; \theta^{(ij)}; v^{(i)}; v^{(j)}]$ includes the relative position and heading of agent j from agent i 's perspective, where x -axis is in the direction of the agent i 's heading. The relationship between $s^{(ij)}$ and $(s^{(i)}, s^{(j)})$ and the relative state dynamics are given in Appendix B.

The computation of V was completed within an hour using an Nvidia RTX A4500 GPU.² The computed maximal safe set $\mathcal{S}^{(ij)}$, defined in the relative state space of $s^{(ij)}$, can be projected to the position space of the ego agent (agent

i), which incorporates all safe positions of the agent that can ensure collision avoidance, given its heading, speed, and the opponent agent (agent j)'s state. Examples of $\mathcal{S}^{(ij)}$ are visualized in Fig. 2 (a) with respect to two different opponent states.

2) *Control barrier-value function-based safety filtering*: Next, we investigate how the computed value function V can be used to constrain the relative state $s^{(ij)}$ to stay within the safe set $\mathcal{S}^{(ij)}$. Since each agent makes their primary decision based on their MARL policy in our framework, we consider how to filter the MARL action if it is potentially unsafe.

To achieve this safety filtering mechanism, we consider the barrier constraint-based mechanism of the CBFs. For a generic state variable s and its dynamics $\dot{s} = f(s, a)$, if a function $B(s)$ satisfies the barrier constraint given by

$$\nabla B(s) \cdot f(s, a) + \gamma B(s) \geq 0, \quad (11)$$

for every state inside the zero-superlevel set of B , i.e. $\forall s \in \{s \mid B(s) \geq 0\}$, and for some constant $\gamma > 0$, we can guarantee that $B(s(t)) \geq 0$, for all $t \geq 0$ [13]. Thus, the state can be maintained to stay within $\{s \mid B(s) \geq 0\}$.

If the computed reachability value function V in (6) is almost-everywhere differentiable, we can construct a CBF by taking $B(s^{(ij)}) = V(s^{(ij)}) - r_{\text{safety}}$. This choice of B satisfies the barrier constraint (11) almost everywhere, and results in the maximal safe set to be represented as the CBF zero-superlevel set, $\mathcal{S}^{(ij)} = \{s^{(ij)} \mid B(s^{(ij)}) \geq 0\}$. Such usage of the reachability value function as the CBF is referred to as the Control Barrier-Value Function (CBVF) [10].

Remark 1. In [19], V is denoted as a constraint-value function and is used to learn Graphical CBF (GCBF) for uncertain dynamics. In this work, we consider its exact computation for a hard safety guarantee. However, our approach can be combined with the learning methods proposed in [19] or other learning-enabled approaches [71, 72] to be extended to agents subjected to uncertain dynamics.

Remark 2. For certain types of dynamics, the value function can be discontinuous without introducing a discount factor in time to the cost function (5) [73].

Finally, the CBVF-based safety filtering can be implemented in a decentralized manner, with each agent executing its own safety filter. Here, we assume that the agents are *cooperative for safety*, meaning that although their unfiltered actions can be selfish, their final filtered actions are coordinated to avoid collision with each other. To achieve this coordination, agent i and agent j can individually solve the identical optimization program defined as

CBVF Safety Filter (cooperative case):

$$\begin{aligned} (a_{\text{safe}}^{(i)}, a_{\text{safe}}^{(j)}) = \arg \min_{(a^{(i)}, a^{(j)}) \in \mathcal{A}} & \|a^{(i)} - a_{\text{marl}}^{(i)}\|^2 + \|a^{(j)} - a_{\text{marl}}^{(j)}\|^2 \\ \text{s.t. } & \nabla B(s^{(ij)}) \cdot f^{(ij)}(s^{(ij)}, a^{(i)}, a^{(j)}) + \gamma B(s^{(ij)}) \geq 0, \end{aligned} \quad (12)$$

²The computation time is not a critical concern in our setting, as the value function is computed offline rather than during real-time deployment.

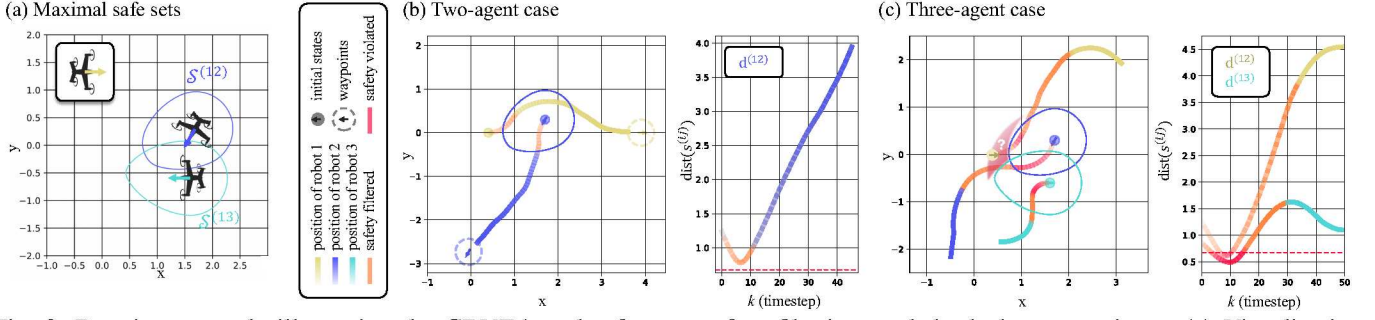


Fig. 2: Running example illustrating the CBVF-based safe sets, safety filtering, and the leaky corner issue. (a) Visualization of the ego agent ($s^{(1)} = [0.4\text{km}, 0\text{km}, 0^\circ, 110\text{kt}]$)’s maximal safe sets (exterior of the level sets) against two agents, $s^{(2)} = [1.7\text{km}, 0.3\text{km}, -120^\circ, 110\text{kt}]$ and $s^{(3)} = [1.7\text{km}, -0.6\text{km}, -180^\circ, 60\text{kt}]$. (b) In the two-agent case, each agent executing their CBVF safety filters (12) successfully prevents collision. (c) In the three-agent case, although agent 1 started inside the intersection of $\mathcal{S}^{(12)}$ and $\mathcal{S}^{(13)}$, it is not able to prevent safety violation. This is because the initial state of robot 1 is in the leaky corner.

and then execute their own action. If the dynamics $f^{(ij)}$ are affine in actions, the optimization becomes a quadratic program [13, 10].

If the opponent agent is non-cooperative for safety, agent i can solve for its own safe action considering the worst-case possible action of the opponent:

CBVF Safety Filter (non-cooperative case):

$$\begin{aligned} a_{\text{safe}}^{(i)} &= \arg \min_{a^{(i)} \in \mathcal{A}} \|a^{(i)} - a_{\text{marl}}^{(i)}\|^2 \\ \text{s.t. } \min_{a^{(j)} \in \mathcal{A}} \nabla B(s^{(ij)}) \cdot f^{(ij)}(s^{(ij)}, a^{(i)}, a^{(j)}) + \gamma B(s^{(ij)}) &\geq 0, \end{aligned}$$

where now B has to be constructed based on a value function for a differential game, which considers the opponent’s worst-case actions [74], given as

$$V_{\text{worst}}(s_0^{(ij)}) := [\min_{\{a_k^{(j)}, a_k^{(i)}\}_{k \geq 0}} \max_{\{a_k^{(j)}, a_k^{(i)}\}_{k \geq 0}}] J(s_0^{(ij)}, a^{(i)}, a^{(j)}). \quad (13)$$

Here, $[\min \max]$ denotes the alternating operation \min (over $a_k^{(j)}$) and \max (over $a_k^{(i)}$). The computation of this worst-case value function can be done similarly to the computation of V by solving the min-max HJ PDE [70].

Running example. In the two-agent case, in Figure 2 (b), the initial relative state between agents 1 and 2 is set near the boundary of the maximal safe set $\mathcal{S}^{(12)}$. By each agent applying the CBVF safety filter, both agents reach their goals safely under the safety-filtered MARL actions.

B. Analysis of the multi-agent case

We begin the analysis of this section by continuing with the running example of the multi-agent case:

Running example. In Figure 2 (c), we now consider the case where a third agent is introduced. The relative states still remain within the pairwise maximal safe sets $\mathcal{S}^{(12)}$, $\mathcal{S}^{(13)}$, and $\mathcal{S}^{(23)}$. Despite all agents actively attempting to avoid collisions, their relative distances fall below r_{safety} . As mentioned in the introduction, this demonstrates the issue of conflicting constraints, implying that although agent 1’s

initial state did not cross the boundaries of the individual safe sets, it may already be outside the true safe set when considering all interactions simultaneously. Computing this true safe set requires defining the relative dynamics of the three agents, which increases the system’s dimensionality. While approximations of this set have been computed, such as in [22], the computation of this multiple-agent safe set is challenging.

As can be seen in the above running example, it is crucial to prevent the agents from falling into the region in which one safety constraint can potentially conflict with the other, i.e., the leaky corners. Although their exact computation is hard, we define the region that can tightly overapproximate this potential conflict region.

Proposition 1. Define

$$\begin{aligned} \hat{\mathcal{S}}^{(i)} &:= \left\{ \{s^{(j)}\}_{j \in I(i)} \mid \forall j \in I(i), V(s^{(ij)}) \geq r_{\text{safety}}, \& \right. \\ &\quad \left. \nexists j_1, j_2 \in I(i) \text{ s.t. } V_{\text{worst}}(s^{(ij_1)}) < r_{\text{safety}} \& V_{\text{worst}}(s^{(ij_2)}) < r_{\text{safety}} \right\} \end{aligned} \quad (14)$$

where V is defined in (6), and V_{worst} is defined in (13). Note the difference between V and V_{worst} . Then for any opponent agent states $\{s^{(j)}\}_{j \in I(i)} \in \hat{\mathcal{S}}^{(i)}$, there exists $a^{(i)}$ and $a^{(j)}$ for all $j \in I(i)$, such that $\forall t \geq 0$, $s^{(ij)}(t) \notin \mathcal{L}^{(ij)}$ for all $j \in I(i)$. In other words, set $\hat{\mathcal{S}}^{(i)}$ can be maintained forward invariant.

Proof: The second condition in (14) allows at most one opponent agent to enter the area in which $V_{\text{worst}}(s^{(ij)}) < r_{\text{safety}}$. We denote this agent as j_{near} . For (i, j_{near}) , since $V(s^{(ij_{\text{near}})}) \geq r_{\text{safety}}$ based on the first condition in (14), agent i and agent j_{near} are within their CBVF safe set $\mathcal{S}^{(ij_{\text{near}})}$ and can select their action sequences $a^{(i)}$ and $a^{(j_{\text{near}})}$, such that $s^{(ij_{\text{near}})}(t) \notin \mathcal{L}^{(ij_{\text{near}})}$ for all $t \geq 0$.

Next, we consider all other agents $j \in I(i) \setminus \{j_{\text{near}}\}$. Based on [75, Proposition 4], for any Lipschitz continuous V_{worst} , its level set is a robust control invariant set. Thus, for all $j \in I(i) \setminus \{j_{\text{near}}\}$, there exists $a^{(j)}$ that results in $V_{\text{worst}}(s^{(ij)}(t)) \geq r_{\text{safety}}$ for all $t \geq 0$, regardless of $a^{(i)}$, ensuring $s^{(ij)}(t) \in \hat{\mathcal{S}}^{(i)}$. ■

Intuitively, the set $\hat{\mathcal{S}}^{(i)}$ prevents conflict of constraints by

allowing only one agent to coordinate for collision avoidance with the ego agent and by prohibiting the other agents from entering the worst-case safe set. These other agents are able to stay away from the pair (i, j_{near}) due to the robust invariance property of the level set of V_{worst} .

Practical implementation: In practice, enforcing each agent to stay within $\hat{\mathcal{S}}^{(i)}$ can be computationally expensive since we have to evaluate V and V_{worst} for all pairs of interaction. In the next section, we use this analysis to inform MARL to implicitly learn not to enter this region. For this, we define the *potential conflict range* as below:

$$r_{\text{conflict}} := \min_r \quad (15)$$

$$\text{s.t. } V_{\text{worst}}(s^{(ij)}) \geq r_{\text{safety}} \quad \forall s^{(ij)} \text{ s.t. } \text{dist}(s^{(ij)}) \geq r.$$

Then, the set of opponent agent states is defined as

$$\hat{\mathcal{S}}^{(i)} := \left\{ \{s^{(j)}\}_{j \in I(i)} \mid \forall j \in I(i), V(s^{(ij)}) \geq r_{\text{safety}}, \& \right. \\ \left. \nexists j_1, j_2 \in I(i) \text{ s.t. } \text{dist}(s^{(ij_1)}) < r_{\text{conflict}} \& \text{dist}(s^{(ij_2)}) < r_{\text{conflict}} \right\} \quad (16)$$

This is an underapproximation of the true conflict-free set $\hat{\mathcal{S}}^{(i)}$ by definition (15). To ensure safety, we want to restrict the number of opponent agents entering this region to be at most one.

Our analysis requires that the observation range be larger than the potential conflict range, $r_{\text{obs}} > r_{\text{conflict}}$. Rather than a restriction, this serves as a design guideline for the observation stack of the robot for safe multi-robot coordination. As shown in Figure 1, the concept of the potential conflict range divides a robot’s proximity into three layers: (1) the range $\text{dist}(s^{(ij)}) < r_{\text{safety}}$, where collision is imminent; (2) the range $r_{\text{safety}} < \text{dist}(s^{(ij)}) < r_{\text{conflict}}$, where engaging with multiple vehicles may introduce safety risks; and (3) the region $r_{\text{conflict}} < \text{dist}(s^{(ij)})$, where the maneuvers of other agents pose minimal safety concerns. A similar three-layer structure was proposed and manually designed in [76]. However, our approach provides a theoretical foundation for defining these boundaries based on reachability analysis.

Remark 3. (Limitation) The set $\hat{\mathcal{S}}^{(i)}$ is a conflict-free safe set only from agent i ’s perspective. In other words, it does not guarantee that the collision-avoidance maneuvers of other agents $j \in I(i)$ will not interfere with one another. Addressing this issue requires analyzing the combinatorial number of possible interactions, which remains an open problem. In our work, we address this challenge by training the MARL policy to learn strategies that mitigate these conflicts.

V. MULTI-AGENT REINFORCEMENT LEARNING WITH LAYERED SAFETY

A. Extending InforMARL for improved decentralized decisions

Our work builds upon the InforMARL architecture [42], a MARL algorithm that solves the multi-agent navigation problem by using a graph representation of the environment, enabling local information-sharing across the edges of the graph. InforMARL uses graph neural networks (GNNs) to process

neighborhood entity observations, allowing the framework to operate with any number of agents and provide scalability without changing the model architecture. Each agent has a set of neighboring agents within its observation range, r_{obs} , and shares its relative position, speed, and goal information with these neighbors. Agents are tasked to navigate to their respective goal positions. When agents reach their respective goals, they get a goal reward $\mathcal{R}_{\text{goal}}(o_k^{(i)}, a_k^{(i)})$.

The extensions we make to the baseline InforMARL to incorporate the layered safety framework and to make it more practical for multi-robot navigation tasks are as follows:

1) *Sequential goal point tracking:* In the updated framework, the agents navigate to a sequence of waypoints, each specified by its position and the desired direction and speed, leading to the final goal (as shown by the green circles in Figure 1). At each time step, an agent gets the following additional rewards, $\mathcal{R}_{\text{tracking}}(o_k^{(i)}, a_k^{(i)})$ which are computed based on the heading and speed of the agent relative to the current target waypoint. The details of these terms are presented in Appendix A.

2) *Model architecture enhancements:* To improve the algorithm and generalize it over diverse scenarios, we update the observations to incorporate rotation-invariant relative distances of the ego agent to goals and neighbors. Once an agent crosses a waypoint, we no longer consider the waypoint in its observation, and the agent moves to the subsequent waypoint. We introduce dynamics-aware action spaces that are updated based on the dynamics model, angular rate, and longitudinal acceleration, as in the running example in Section IV, ensuring agents respect motion constraints specific to their dynamics.

3) *Curriculum training:* The training framework also incorporates curriculum learning where we progressively make the training environment harder [77] for improving agents’ performance, refining safety rewards, and updating the safety distance r_{safety} used in the safety filter. This is detailed in the subsequent sections.

B. Safety filter design for multiple agents

For multiple agents, the CBVF $B(s^{(ij)})$ is evaluated for each agent i and any neighboring agent j within the observation range r_{obs} . A smaller value of B indicates that the near-collision is more imminent and safety is at greater risk. The neighbor agent with the minimum pairwise $B(s^{(ij)})$ is selected as the agent whose actions will be curtailed. We term the module that selects this prioritized constraint as the *prioritization module*. If an agent pair (i, j) has each other as the minimum pairwise $B(s^{(ij)})$, then we call them a *potential collision pair*.

C. Safety-informed training

1) *Curriculum update:* We start the training routine without any safety filter or penalty applied for the first half of the training steps. This is done to optimize the task performance of MARL unconstrained by any safety parameters. Once training reaches half the number of total training steps, we activate the safety filter. Additionally, we introduce the following safety parameters, which are updated using the curriculum learning

framework during training. First, the safety distance r_{safety} is initialized to zero during the start of model training, allowing agents to approach each other at close ranges. As the training progresses, we increase r_{safety} to the desired value. Similarly, we scale the conflict radius r_{conflict} computed using Eq. (15) based on the value of the r_{safety} . This setup allows agents to explore the environment early on in the training and prevents them from converging to overly conservative behavior.

2) *Safety-informed reward*: In addition to the heading, speed, and goal rewards, we introduce some additional penalties. When more than two agents are within the conflict radius r_{conflict} , we apply a potential conflict penalty

$$\mathcal{C}_{\text{conflict}} := \sum_{j \in \{j | \text{dist}(s^{(ij)}) < r_{\text{conflict}}\}} \max\{0, r_{\text{conflict}} - \text{dist}(s^{(ij)})\} \times \underbrace{\max\left\{0, -\begin{bmatrix} x^{(ij)} & y^{(ij)} \end{bmatrix} \begin{bmatrix} v_x^{(ij)} \\ v_y^{(ij)} \end{bmatrix}\right\}}_{\text{relative distance change}}, \quad (17)$$

which evaluates whether the agent j is within the potential conflict range and is approaching towards agent i . Based on Proposition 1, we do not apply the penalty when there is just one agent within r_{conflict} .

The penalty $\mathcal{C}_{\text{conflict}}$ is carefully designed to mitigate the risks associated with potential conflicting constraints when multiple agents enter the range, while simultaneously minimizing the conservatism it may introduce. This penalty is an **indirect** safety penalty, as it is not incurred based on explicit safety violations but rather indirectly through the proximity of multiple agents.

The final reward structure is

$$\mathcal{R}_{\text{total}}(o_k^{(i)}, a_k^{(i)}) = \mathcal{R}_{\text{tracking}}(o_k^{(i)}, a_k^{(i)}) + \rho_{\text{goal}} \mathcal{R}_{\text{goal}}(o_k^{(i)}, a_k^{(i)}) - \rho_{\text{conflict}} \mathcal{C}_{\text{conflict}} \quad (18)$$

where ρ_{goal} is a binary indicator when the agent is at the goal, and ρ_{conflict} is a binary indicator when the number of other agents within the potential conflict region is more than one.

VI. RESULTS

The main robotic application we focus on is the safe autonomous navigation of aerial vehicles. We apply our framework to Crazyflie drones navigating through waypoints in both simulation and hardware experiments, as well as to the simulation of air taxi operations in realistic settings.

A. Experiment Setup

Considered dynamics. We consider two types of dynamics, one for the quadrotors and the other for the air taxi vehicle in a wing-borne flight. The parameters for both dynamics are summarized in Table I and are set to match the industry standard [78, 79, 80]. For instance, we use an angular rate bound of 0.1 rad/s for the air taxi dynamics, as it results in the lateral acceleration $0.5g$ under the nominal speed, which amounts to the maximum tolerable lateral acceleration for passenger comfort in NASA market studies [81].

TABLE I: Parameter Summary for Different Vehicle Dynamics

Parameter	Air taxi (Sim)	Crazyflie
Groundspeed		
v_{\min}	60 knot (30 m/s)	-1.0 m/s
v_{\max}	175 knot (90 m/s)	1.0 m/s
v_{nominal}	110 knot (57 m/s)	0.5 m/s
Acceleration		
a_{\min}	-3.3 ft/s ² (-1.0 m/s ²)	-0.5 m/s ²
a_{\max}	6.6 ft/s ² (2.0 m/s ²)	0.5 m/s ²
Angular Rate (ω_{\max}) (rad/s)	0.1	-
Sampling Rate (s)	1.0	0.1
Waypoint Thresholds (\pm)		
Distance to Goal	0.186 miles (0.3 km)	0.2 m
Heading	45°	45°
Speed	38.9 knot (20 m/s)	0.1 m/s
Observation Range (r_{obs})	3.1 mi. (5.0 km)	4.0 m
Safety Distance (r_{safety})	500 - 2200 ft (0.152 - 0.671 km)	0.5 m
Potential Conflict Range (r_{conflict})	4600 ft (for $r_{\text{safety}}=2200$ ft)	1.0 m

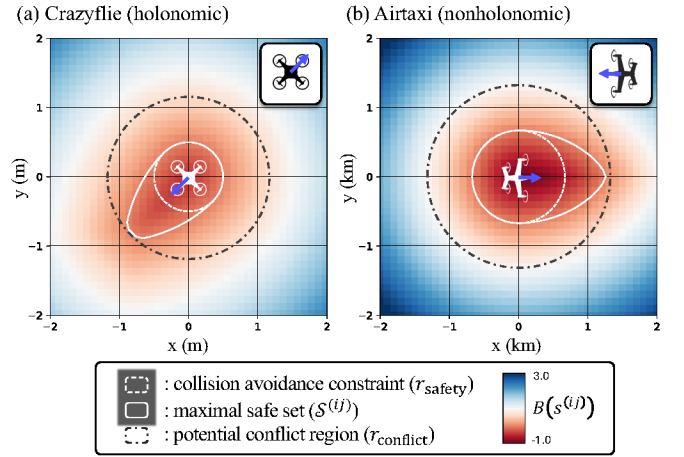


Fig. 3: Maximum safe sets (exterior of the white level sets), potential conflict region, and CBVF (colormap) for each vehicle dynamics, displayed in the relative position space when (a) relative velocity is $(v_x, v_y) = (1, 1)$ [m/s], (b) relative speed and heading is 220 knots and 180°, respectively.

The quadrotor dynamics in the horizontal plane are represented as simple double integrators with

$$\dot{x} = v_x, \quad \dot{y} = v_y, \quad \dot{v}_x = a_x, \quad \dot{v}_y = a_y, \quad (19)$$

where $s^{(i)} = [x, y, v_x, v_y]$ and $a^{(i)} = [a_x, a_y]$. The quadrotor runs the low-level onboard flight controller to track the commanded actions.

The air taxi dynamics in the horizontal plane are represented using the kinematic vehicle model in (10) of the running example in Section IV. Three features of the air taxi dynamics considered in this work make its safety assurance more challenging and interesting. First, the vehicle cannot stop as it has to maintain the wing-borne flight ($v_{\min} > 0$). Next, the dynamics are nonholonomic, meaning that its control towards the lateral direction can be achieved only by changing its direction. Finally, due to small acceleration or deceleration authority, the vehicle often has to employ turning maneuvers for

deconfliction. This is common for fixed-wing and hybrid mode vehicles like vertical takeoff-and-landing vehicles (VTOLs), those envisioned for AAM operations [82, 83, 84].

Due to these challenges, the advantages of our method for enhancing safety are particularly evident for the air taxi dynamics (Section VI-D). In contrast, for the quadrotors (Section VI-C), our safety filter design consistently ensures safety across all evaluated methods; thus, we focus more on how our approach achieves performance enhancement. The safe sets, CBVFs, and the potential conflict range computed using HJ reachability are visualized in Figure 3.

Task & Training environment. We modify Multi Particle Environments (MPE) [41] to incorporate agents to follow the dynamics as specified before and the safety filter. In our navigation task setup, the drone must pass through a waypoint with its state satisfying the threshold conditions specified in Table I to proceed to the next waypoint. The main values that define the training environments are the number of agents N , the number of waypoints per agent M , and the size of the environment L . At every episode, the initial positions of the agents, the waypoints' locations, and the headings are set randomly. The episode is terminated if all agents reach their goal, the last waypoint. After training the MARL policies, we test them in various evaluation scenarios with values of N, M, L different from those of the training environment. For the quadrotor, we use $N=4, M=2, L=4$, and for the air taxi dynamics, we use $N=4, M=2, L=6$ for the training.

B. Comparison Studies

We first conduct two sets of simulation experiments to compare our method against: 1) approaches that do not employ a safety filter or curriculum during training, and instead rely on alternative reward designs for safety, and 2) methods from prior works based on model-based CBF design and model-free safe MARL for multi-agent coordination. Both studies are conducted in the quadrotor simulation environment.

1) *Ablation study for safety-informed training:* First, we designed our experiments to evaluate the value of (1) introducing the safety filter during training, (2) using the curriculum, and (3) the effectiveness of a potential conflict penalty term for safety, as described in Section V-C. To evaluate the effect of employing the safety filter during training, we compare the results of those trained with and without the filter. To evaluate the effect of the curriculum, we compare our method against a policy trained without the curriculum update in Section V-C. Finally, to evaluate the effectiveness of the potential conflict penalty term, we compare it against three alternative penalty terms for safety suggested in the literature:

- Hinge loss for constraint violation:

$$\mathcal{C}_{\text{plain}} := \max\{0, r_{\text{safety}} - \text{dist}(s^{(ij)})\},$$

This is the most typical penalty term, introduced in the safe RL literature [85].

- CBVF-based hinge loss:

$$\mathcal{C}_{\text{cbvf}} := \max\{0, -B(s^{(ij)})\},$$

This penalizes the agent for entering the zero-sublevel set of the CBVF, the unsafe set. The use of reachability value functions as a safety penalty in RL has been explored in previous works such as [86].

- Penalty occurring when safety filter intervenes:

$$\mathcal{C}_{\text{norm.diff}} := \|a_{\text{safe}}^{(i)} - a_{\text{marl}}^{(i)}\|,$$

based on (12). This is the main penalty term used in the method of [11] to inform MARL with safety.

When we introduce each penalty term, its weight is carefully tuned to maximize task and safety metrics. In total, we test nine variants of the training methods based on the activation of the safety filter, curriculum, and choice of the reward term, which are detailed in Appendix C.

We evaluate each method in three scenarios. In addition to the random scenario same as the training environment, in the second scenario, we test how the MARL policy interacts with a larger number of agents and a more challenging waypoint configuration by setting $N=6, M=3$, and $L=6$ while also placing the first two waypoints at the same positions, representing the air corridor. The third scenario reconstructs our hardware experiment environment, which will be detailed in Section VI-C, in simulation.

Below is the summary of the key aspects of the result, while more details and the table of the comparison for the performance metrics are presented in Appendix C:

- *Effect of using the safety filter in training:* Methods that incorporate the safety filter during training consistently outperform their counterparts trained without the filter across all metrics.
- *Effect of curriculum learning:* The curriculum learning can significantly enhance performance by reducing the conservativeness of the trained policy.
- *Effect of potential conflict penalty $\mathcal{C}_{\text{conflict}}$ compared to other penalty candidates:* Our method achieves the best performance in most cases. Importantly, our method outperforms other methods especially when there is a larger number of agents (the second scenario).

2) *Comparison to other methods:* Next, we compare our method to (1) DG-PPO [19] and (2) a safety filter designed based on the exponential CBF (ECBF) [16], used for multi-agent collision avoidance in [15]. We use $N=4, M=1, L=3$ for the training of all three methods, which enables fair comparison, especially with results we can obtain from the DG-PPO source code. We run the training of both DG-PPO and our MARL policy with the same number of environment steps ($1e7$) and gradient steps ($\text{epoch_ppo}=1$), where the initial and goal positions are randomly generated.

We evaluate the trained policies in two environments for 25 episodes each: (1) same random environment with world size increased to $L=6$. (2) environment with an increased number of agents $N=8$ and world size $L=6$, where initial and goal positions are arranged in two parallel lines in random order. The results are reported in Tables II and III. While all methods perform well when the number of agents remains the same as

TABLE II: Simulation results for Crazyflie dynamics with $N=4$, with time horizon 51.2s. We evaluate goal reach rate (%) for performance and the percentage of near-collision events ($\text{dist}(s^{(ij)}) < r_{\text{safety}}$) in the timestamped trajectory data (Near collision %) for safety.

Methods	Goal reach(%)	Near collision(%)
DG-PPO	96 ± 11.8	0.04 ± 0.16
Exponential CBF	100 ± 0	0.0 ± 0.0
Our Method	100 ± 0	0.0 ± 0.0

TABLE III: Simulation results with $N=8$, and initial & goal positions arranged in lines under random order. Videos are available in the supplementary material.

Methods	Goal reach(%)	Near collision(%)
DG-PPO	100 ± 0	9.1 ± 2.7
Exponential CBF	93 ± 8.9	8.8 ± 10.7
Our Method	100 ± 0	0.0 ± 0.0

in the training environment, our method is the only method that guarantees 100% safety when N increases, whereas the percentage of near-collision events increases significantly for DG-PPO and ECBF. It must be noted that DG-PPO is a model-free method that learns a neural CBF during its training, whereas our method uses the CBVF computed based on the system dynamics model. As observed in [18], such model-free methods are vulnerable to generalization in scenarios with a large number of agents.

C. Hardware experiments with quadrotors

Next, as illustrated in Figure 4, we conduct hardware experiments with three Crazyflie 2.0 drones using a Vicon system for localization. Each drone is controlled by a hierarchical framework: its high-level system is modeled with double integrator dynamics, and the resulting high-level state is passed to onboard PID tracking controllers. We define high-level feedback control (acceleration in the x and y axes) based on real-time (100 Hz) Vicon system data. To approximate decentralized control, we run distinct decentralized policies for each drone on a single ThinkPad laptop, transmitting high-level control commands every 0.1 seconds.

In the experimental scenario, each drone is required to pass through two shared waypoints—representing an air corridor—before reaching its designated landing location ($N=3$, $M=3$, $L=3\text{m}$). We compare the policy learned under our method to the baseline, which is trained without a safety filter in hardware experiments, with the recorded trajectories shown in Figure 5. Under our approach, three drones smoothly avoid conflicts and safely navigate their individual waypoints, completing the task in 12.95 seconds. In contrast, the baseline policy requires one drone to perform a second pass, after missing its first waypoint due to the safety filter preventing it from approaching other agents passing the waypoint, thus extending the total completion time to 25.29 seconds. These results demonstrate how our Layered Safe MARL framework enhances task performance through efficient deconfliction.

D. Simulation of decentralized air taxi operations

Finally, we evaluate the application of our framework to decentralized air taxi operations. Although there is no single

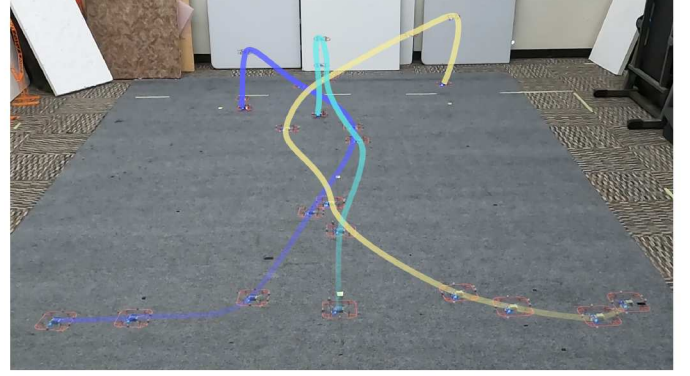


Fig. 4: Crazyflie hardware experiment with the MARL policy learned by our method. The three drones have to pass through two common waypoints to get to their landing location. The trajectories corresponding to the video footage are visualized in Fig. 5 (b).

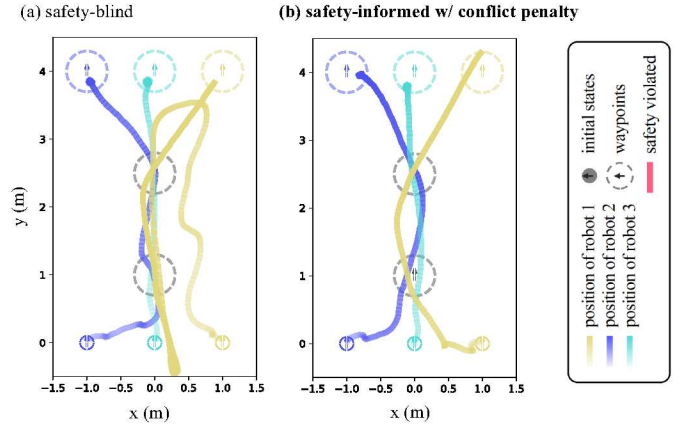


Fig. 5: We compare the recorded Crazyflie hardware experiment trajectories under our method and the baseline policy trained without the safety filter. With our approach, the drones smoothly deconflict and efficiently complete the task. In contrast, under the baseline policy, the yellow Crazyflie misses a waypoint and must make a second pass. These results demonstrate that incorporating layered safety information during training improves the performance of the MARL policy.

consensus on how the air traffic management (ATM) system will function for advanced air mobility (AAM) operations, each vehicle will likely be required to have fallback autonomy systems in place, for instance, to ensure safety in case the centralized system fails.

To conduct the study with a realistic traffic volume, we use the results of the Urban Air Mobility (UAM) demand analysis from [87, 88], which estimates how much ground traffic could be replaced by AAM considering various factors including different demographics of riders, socioeconomic factors, and historical commuting patterns. By combining these insights, our study derives a reasonable estimate of how traffic density will evolve once UAM operations reach full implementation. From their results, we consider a peak-density scenario in which each vertiport serves 500 passengers per hour during



Fig. 6: Bay Area case scenarios. The left panel illustrates routes where multiple air taxi vehicles would travel from the North and East Bay toward San Francisco, merging into a single air corridor. The right panel shows intersecting air corridors: one where the vehicles would travel from Fremont (southeast) to San Francisco and another from Oakland (north-east) to Redwood City. The blue dots represent the waypoints that UAVs follow, while the yellow dots indicate the departure or an incoming waypoint of the corridor.

peak hours, equating to two operations (takeoffs and landings) per minute, with each operation accommodating a 4-passenger aircraft. This corresponds to about 125 trips per hour. We chose vertiports from multiple locations in the Bay Area with high travel demand and designed the air corridors with a lateral separation of 1500 ft based on a preliminary analysis of the separation standards for UAM [88]. Waypoints are created to connect the trails of these corridors spaced 3-4 km apart. For simplicity and clearer visualization, our simulations focus on aerial vehicles traveling westward (from the East Bay to San Francisco and to the South Bay); we assume outbound trips use a separate fixed altitude, thus our study addresses only horizontal deconfliction.

An important modeling assumption is the required separation distance between vehicles. We base these restrictions on industry standards that define the minimum safe distance between the aircraft and potential hazards to maintain an acceptable collision risk [81]. Although there is no single global standard for the separation distance for AAM vehicles yet, we adopt a set of parameters from a variety of literature: the separation distance from dynamic obstacles, maximum accelerations, and bank angle to ensure safety and passenger comfort, [89, 81, 88], which guides the minimum horizontal distances between UAVs to range from 500 to 2200 feet. In our study, a horizontal separation of 1500 ft was used, as it aligns with NASA’s UAM corridor design and provides a good balance between collision risk and operational efficiency.

We evaluate three methods: MARL trained without the safety filter and no safety penalty (**Safety blind**), MARL trained with the safety filter under the proposed curriculum but with no safety penalty (**No penalty**), and the proposed safety-informed method employing the safety filter, curriculum, and the potential conflict penalty (**Proposed**) in two high-density air traffic scenarios. The two scenarios, illustrated in Figure 6, represent different commuting patterns in the Bay Area. In

TABLE IV: Simulation results of air taxi operations emulating potential peak traffic around the Bay Area—a scenario in which all vehicles merge into the city-inbound corridor. For performance, we evaluate the mean travel time (s). For safety, we evaluate the percentage of near-collision events ($\text{dist}(s^{(ij)}) < r_{\text{safety}}$) in the timestamped trajectory data (Near collision %), and the percentage of instances having multiple agents encountered within the potential conflict range ($\text{dist}(s^{(ij)}) < r_{\text{conflict}}$) (Conflict %).

Methods	Merging Scenario ($N=8, M=5$)		
	Travel t(s)(↓)	Near collision(%) (↓)	Conflict(%) (↓)
Safety-blind	675.6	0.055	2.4
No penalty	617.9	0.042	5.5
Proposed	450.5	0.021	3.2

TABLE V: Simulation results of air taxi operations—a scenario in which two air corridors intersect with each other.

Methods	Intersection Scenario ($N=16, M=6$)		
	Travel t(s)(↓)	Near collision(%) (↓)	Conflict(%) (↓)
Safety blind	987.4	0.058	2.1
No penalty	780.5	0.129	3.8
Proposed	660.8	0.056	1.6

the left scenario (**Merge Scenario**), multiple air routes (eight in total) from the northern Bay merge into a single corridor leading to San Francisco. The departure time of each vehicle varies randomly within a 60-second range. In the scenario shown to the right (**Intersection Scenario**), two westbound air corridors intersect. Here, we set the UAVs to leave the origin every 90 ± 15 seconds to intentionally induce congestion at the intersection.

We evaluate 25 random episodes for each method and report the results in Table IV and Table V for each scenario, respectively. The results show that the proposed method achieves both the highest performance, measured by the shortest mean travel time, and the lowest percentage of near-collision events.

Examples of vehicle trajectories for each scenario and method are visualized in Figure 7. In the merging scenario, our method demonstrates the most efficient deconfliction of trajectories when multiple vehicles merge into the air corridor. In the intersection scenario, near the intersection, the region occupied by vehicles as they maneuver to avoid collisions is noticeably larger in our method compared to the second method, trained with no penalty. This indicates that vehicles using our approach proactively maintain greater separation to mitigate the conflicting constraints.

While we expect that a fully operational ATM system for AAM will be significantly more efficient, seamless, and safer than our simulation study suggests, we present this study as an initial guideline for resolving hypothetical emergency scenarios. For instance, the situations we simulated can emerge when the airspace congestion coincides with the loss of centralized traffic control, requiring each agent to make independent, safe decisions.

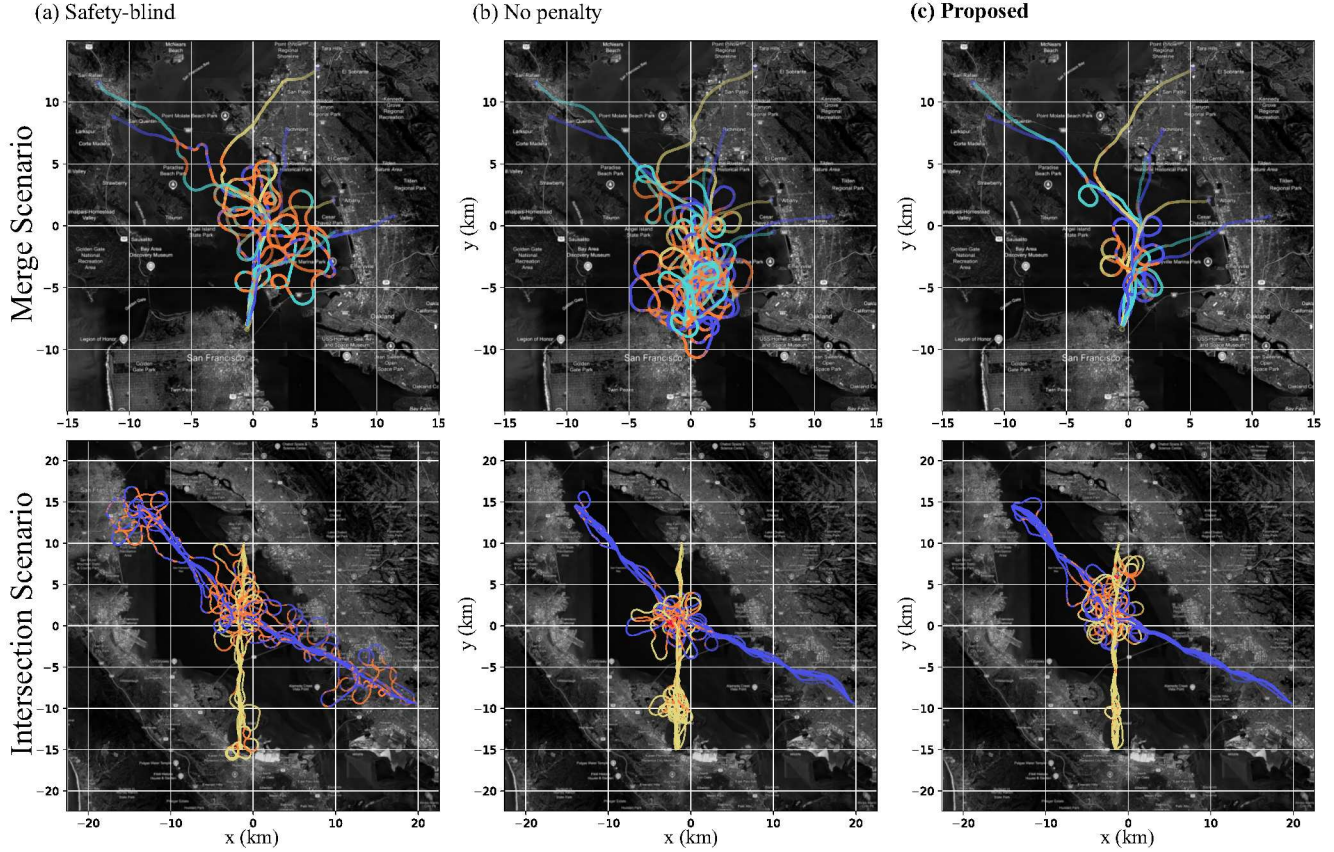


Fig. 7: Comparison of air taxi trajectories in merging and crossing scenarios: The top row illustrates the single-lane merging scenario, where UAVs converge into a shared inbound air corridor, while the bottom row depicts intersecting air corridors. In the merging scenario, our method achieves the most efficient deconfliction of trajectories, minimizing congestion near the corridor. In the crossing scenario, our method demonstrates a wider safety buffer around intersections, as UAVs actively maintain greater separation to mitigate conflicts. Videos are available in the supplementary material.

VII. LIMITATIONS

While our framework shows significant improvements in achieving safety and performance, there are limitations, which we list below.

- 1) *Scalability to higher dimensions*: Our current framework is designed for 2D scenarios, and needs extending it to 3D environments.
- 2) *Guarantees for multiple engagements*: Our safety guarantees are currently limited to pairwise interactions. While our method is designed to significantly mitigate collision risks in multi-agent interactions based on theoretical analysis, it does not provide formal guarantees for scenarios involving multi-agent engagements. For further details, see Remark 3.
- 3) *Hardware experimentation constraints*: In our hardware experiments, local observations were emulated using a motion capture (mocap) system rather than being obtained through onboard sensing. This simplification may not fully reflect real-world operational constraints and should be addressed in future implementations. Additionally, fixed-wing vehicles were not included in

our experiments.

- 4) *Communication range constraints*: The impact of communication range limitations in our hardware experiments was not analyzed.

VIII. CONCLUSIONS

In this work, we presented a layered architecture combining a CBVF-based safety filtering mechanism with a MARL policy, demonstrating its effectiveness in ensuring both safety and efficiency. Our approach enables MARL to navigate conflicts proactively while benefiting from safety-informed reward signals. Along with the safety filter introduced during training using a curriculum learning approach, the Layered Safe MARL framework achieved shorter travel times and reached more waypoints with fewer conflicts. The key components of our approach, curriculum learning, and cost terms that inform potential conflict zones are agnostic to the choice of the MARL algorithm. We validated our method by applying it to two distinct dynamics—quadrotor and fixed-wing AAM flight dynamics—and evaluated it in progressively complex scenarios. We also conducted hardware experiments on three

TABLE VI: Results of policies trained under various methods for Crazyflie dynamics: We evaluate mean travel time (s) and number of reached waypoints (Waypoint #) for performance, and the percentage of the events involving multiple agents encountered within the potential conflict range in the trajectory data (Conflict %) for safety risk. Note that in these simulations, the agent never violated safety for all methods due to our safety filter, except in the training scenario when the agent is initialized at the safety-violating states. (N =number of agents, M =number of waypoints, L =world size)

Methods	Scenario 1 (Training) ($N=4, M=2, L=4$)			Scenario 2 ($N=6, M=3, L=6$)			Scenario 3 ($N=3, M=3, L=3$)		
	Travel time(s)(↓)	Waypoint#(↑)	Conflict%(↓)	Travel t	Waypoint #	Conflict	Travel t	Waypoint #	Conflict
1 (safety-blind)	18.33	1.62 ± 0.25	8.7	29.18	2.11 ± 0.24	21.9	19.08	2.14 ± 0.53	16.2
2	18.21	1.67 ± 0.19	7.1	29.77	2.06 ± 0.24	19.7	18.92	2.40 ± 0.47	16.0
3	17.73	1.76 ± 0.16	6.6	28.59	2.26 ± 0.21	19.7	18.44	2.41 ± 0.46	14.9
4	18.73	1.58 ± 0.23	7.2	29.11	2.20 ± 0.21	17.1	18.79	2.17 ± 0.47	13.4
5 (no penalty)	17.56	1.75 ± 0.17	5.3	28.46	2.33 ± 0.20	16.9	16.09	2.78 ± 0.28	11.8
6	18.31	1.67 ± 0.17	5.9	28.92	2.27 ± 0.21	17.0	17.69	2.42 ± 0.42	15.0
7	17.90	1.71 ± 0.18	5.3	28.98	2.32 ± 0.18	15.8	17.59	2.54 ± 0.34	11.2
8	20.52	1.26 ± 0.21	2.7	31.25	1.74 ± 0.27	6.9	18.36	2.20 ± 0.36	8.6
9 (proposed)	17.81	1.78 ± 0.18	5.4	28.59	2.42 ± 0.20	15.1	16.91	2.71 ± 0.24	10.8

Crazyflie drones, highlighting the applicability of our method in real-world systems.

Our method integrates model-based safety tools from control theory (CBVFs) with learning-based methods (MARL), together forming a framework that addresses two major challenges in multi-agent problems—safety and efficient coordination. While deep reinforcement learning has faced skepticism in safety-critical applications such as air traffic management, recent advances—including our work—demonstrate the viability of hybrid approaches that combine learning and control, and illustrate how RL can be responsibly applied in safety-critical settings.

Future research could investigate decomposition techniques and learning-based reachability analysis (e.g., DeepReach [22]) to extend safety verification to higher-dimensional settings. Adapting to other methods, including other MARL algorithms (e.g., MAPPO or even further refining DG-PPO), MPC, or game-theoretic solutions, by treating the potential conflict zone as a soft constraint, is an exciting future work direction. Further investigation is needed to assess how communication constraints affect coordination and safety in decentralized multi-agent systems. Finally, an important future direction is testing the proposed approach’s applicability in various robotics domains, ranging from higher-order dynamics to complex environments and sensing constraints, such as ground robots, underwater autonomous vehicles, and space robots.

ACKNOWLEDGMENTS

We thank Dr. Mir Abbas Jalali (Joby), George Gorospe (NASA), Dr. Anthony Evans (Airbus), Inkyu Jang (SNU) and Kanghyun Ryu (UC Berkeley) for the helpful discussions. Jasmine Aloor and Hamsa Balakrishnan were supported in part by NASA under Grant No. 80NSSC23M0220. Jason J. Choi, Jingqi Li and Claire J. Tomlin were supported in part by DARPA Assured Autonomy under Grant FA8750-18-C-0101, DARPA ANSR under Grant FA8750-23-C-0080, the NASA ULI Program in Safe Aviation Autonomy under Grant 62508787-176172, and ONR Basic Research Challenge in Multibody Control Systems under Grant N00014-18-1-2214.

Jason J. Choi received the support of a fellowship from Kwanjeong Educational Foundation, Korea. Jasmine J. Aloor was supported in part by a Mathworks Fellowship. Maria G. Mendoza acknowledges support from NASA under the Clean Sheet Airspace Operating Design project MFRA2018-S-0471. The authors would like to thank the MIT SuperCloud [90] and the Lincoln Laboratory Supercomputing Center for providing high performance computing resources that have contributed to the research results reported within this paper.

APPENDIX

A. Reward function design for goal reaching

We assume that, from the agent’s local observation, it can evaluate its state, including the position and heading relative to the current target waypoint, denoted as $s_{\text{ref}}^{(i)}$. The additional reward $\mathcal{R}_{\text{tracking}}$ is designed to guide agents to efficiently navigate to the target waypoint. It uses a distance-like measure relative to the waypoint position, and can incorporate additional information like the errors from the desired heading angle and speed. The vehicle dynamics also inform the reward design. We design a specific reward for each quadrotor and air taxi dynamics. For the quadrotor, although the vehicle dynamics are holonomic, we want the vehicle to approach the waypoint from a specific target heading direction. To achieve this, we design a reference velocity field, $v_{\text{ref}}(s_{\text{ref}}^{(i)})$, around the waypoint, shaping it like the magnetic field around a solenoid. Then, the reward for the waypoint tracking is given as $\mathcal{R}_{\text{tracking}}(o_k^{(i)}, a_k^{(i)}) = \|v^{(i)} - v_{\text{ref}}\|$. For the air taxi dynamics, shaping the reference velocity field is not straightforward, as the vehicle is nonholonomic and is mainly limited by its maximum turning radius, determined by its speed and yaw rate. Thus, we compute the time-to-reach function [91], which is the minimum time required to reach the target waypoint (satisfying the threshold conditions) subject to the vehicle dynamics. This time-to-reach reward is also used in [92] for RL-based navigation of mobile robots. The use of the time-to-reach reward guides the vehicle to learn how to perform a 360-degree turn when it misses its waypoint.

B. Air taxi dynamics: additional information

The relationship between $s^{(ij)}$ and $(s^{(i)}, s^{(j)})$ for the air taxi dynamics in (10) is given by

$$\begin{aligned} x^{(ij)} &= \cos \theta^{(ij)} (x^{(j)} - x^{(i)}) + \sin \theta^{(ij)} (y^{(j)} - y^{(i)}), \\ y^{(ij)} &= -\sin \theta^{(ij)} (x^{(j)} - x^{(i)}) + \cos \theta^{(ij)} (y^{(j)} - y^{(i)}), \\ \theta^{(ij)} &= \theta^{(i)} - \theta^{(j)}. \end{aligned} \quad (20)$$

The relative dynamics (3) can be derived from (10) and (20), and are express as

$$\begin{aligned} \dot{x}^{(ij)} &= -v^{(i)} + v^{(j)} \cos \theta^{(ij)} + y^{(ij)} \omega^{(i)} \\ \dot{y}^{(ij)} &= v^{(j)} \sin \theta^{(ij)} - x^{(ij)} \omega^{(i)} \\ \dot{\theta}^{(ij)} &= \omega^{(j)} - \omega^{(i)}. \end{aligned} \quad (21)$$

C. Ablation Study: Details

We conduct comparison studies among the following nine methods:

- 1) Policy trained without the safety filter and no safety penalty (**Safety blind**)
- 2) Policy trained without safety filter and with $\mathcal{C}_{\text{plain}}$
- 3) Policy trained without safety filter and with $\mathcal{C}_{\text{conflict}}$
- 4) Policy trained with the safety filter and without curriculum learning (with no penalty)
- 5) Policy trained with the safety filter and no safety penalty (**No penalty**)
- 6) Policy trained with the safety filter and with $\mathcal{C}_{\text{plain}}$
- 7) Policy trained with the safety filter and with $\mathcal{C}_{\text{cbvf}}$
- 8) Policy trained with the safety filter and with $\mathcal{C}_{\text{norm.diff}}$
- 9) **Policy trained with the safety filter and with $\mathcal{C}_{\text{conflict}}$ (Proposed)**

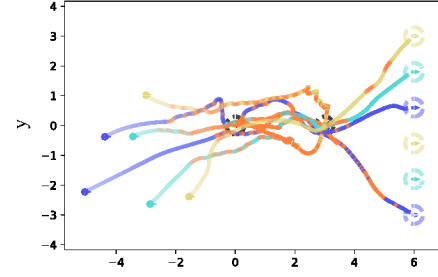
Note that methods 5-9 are trained with curriculum learning on r_{safety} , as described in Section V-C. Every policy we compared has been trained for the same number of environment steps. Methods 1, 5, and 9 correspond to the methods we also evaluate in the air taxi operation simulation.

Table VI summarizes the results of the simulation study, and Figure 8 visualizes example trajectories in Scenario 2 under the policies of methods 1, 5, and 9. Each method is evaluated using four random seeds, with 25 episodes per seed, totaling 100 random episodes. Note that in these simulations, the agent never violated safety for all methods due to our safety filter, except in the training scenario when the agent is initialized at the safety-violating states. Thus, the percentage of near-collision events (safety violation) is not reported, and only the rate of potential conflict (for instance, when more than two agents enter the potential conflict range r_{conflict}) is calculated.

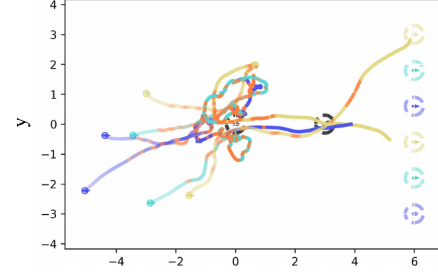
The key aspects of the result in Table VI are:

- *Effect of using the safety filter in training (1-3 vs 5-6, 9):* Methods that incorporate the safety filter during training consistently outperform their counterparts trained without the filter across all metrics.
- *Effect of curriculum learning (4 vs 5):* The curriculum learning can significantly enhance performance by reducing the conservativeness of the policy.

(a) safety-blind



(b) safety-informed, no penalty



(c) safety-informed w/ conflict penalty

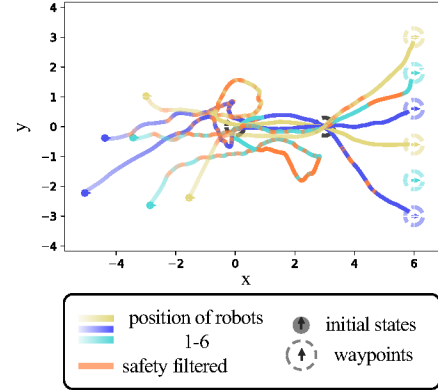


Fig. 8: Simulation results of (a) safety-blind (method 1), (b) safety-informed with no penalty (method 5), and (c) safety-informed with potential conflict penalty (method 9) under Scenario 2 in Table VI, trained for double integrator dynamics. Agents are initialized at random positions and have to merge into a line formed by two waypoints before reaching their final waypoints. While our safety filter ensures safety for all cases, the MARL method trained with a potential conflict penalty shows the most efficient behavior for reaching waypoints. Videos are available in the supplementary material.

- *Effect of potential conflict penalty $\mathcal{C}_{\text{conflict}}$ compared to other penalty candidates (6, 7, 8 vs 9):* Although method 8 that uses $\mathcal{C}_{\text{norm.diff}}$ consistently shows the lowest rate of potential conflict, and its performance is significantly impaired by the penalty. Our method achieves the best performance in most cases. Importantly, our method outperforms other methods, especially when there is a larger number of agents (Scenario 2).

REFERENCES

- [1] Hasan A Poonawala, Aykut C Satıcı, Hazen Eckert, and Mark W Spong. Collision-free formation control with decentralized connectivity preservation for nonholonomic-wheeled mobile robots. *IEEE Transactions on Control of Network Systems*, 2014. 1
- [2] Minghua Liu, Hang Ma, Jiaoyang Li, and Sven Koenig. Task and path planning for multi-agent pickup and delivery. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2019. 1
- [3] Keonyup Chu, Minchae Lee, and Myoungcho Sunwoo. Local path planning for off-road autonomous driving with avoidance of static obstacles. *IEEE Transactions on Intelligent Transportation Systems*, 2012. 1
- [4] Federal Aviation Administration. Urban Air Mobility Concept of Operations 2.0. Technical report, Federal Aviation Administration, 2023. 1, 3
- [5] Li Wang, Aaron D. Ames, and Magnus Egerstedt. Safety Barrier Certificates for Collisions-Free Multirobot Systems. *IEEE Transactions on Robotics*, 2017. 1, 2
- [6] Xinrui Wang, Karen Leung, and Marco Pavone. Infusing Reachability-Based Safety into Planning and Control for Multi-agent Interactions. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. 1, 3
- [7] Ian M Mitchell. Scalable calculation of reach sets and tubes for nonlinear systems with terminal integrators: a mixed implicit explicit formulation. In *Proceedings of the International Conference on Hybrid Systems: Computation and Control*, 2011. 1
- [8] Donggun Lee, Mo Chen, and Claire J. Tomlin. Removing Leaking Corners to Reduce Dimensionality in Hamilton-Jacobi Reachability. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019. 1
- [9] Frank J Jiang, Kaj Munhoz Arfvidsson, Chong He, Mo Chen, and Karl H Johansson. Guaranteed Completion of Complex Tasks via Temporal Logic Trees and Hamilton-Jacobi Reachability. *arXiv preprint arXiv:2404.08334*, 2024. 1
- [10] Jason J Choi, Donggun Lee, Koushil Sreenath, Claire J Tomlin, and Sylvia L Herbert. Robust control barrier-value functions for safety-critical control. In *IEEE Conference on Decision and Control (CDC)*, 2021. 1, 2, 5, 6
- [11] Federico Pizarro Bejarano, Lukas Brunke, and Angela P Schoellig. Safety Filtering While Training: Improving the Performance and Sample Efficiency of Reinforcement Learning Agents. *IEEE Robotics and Automation Letters*, 2024. 2, 3, 9
- [12] Songyuan Zhang, Kunal Garg, and Chuchu Fan. Neural graph control barrier functions guided distributed collision-avoidance multi-agent control. In *Conference on Robot Learning*. PMLR, 2023. 2
- [13] Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 2016. 2, 5, 6
- [14] Mrdjan Jankovic, Mario Santillo, and Yan Wang. Multiagent systems with CBF-based controllers: Collision avoidance and liveness from instability. *IEEE Transactions on Control Systems Technology*, 2023. 2
- [15] Manohari Goarin, Guanrui Li, Alessandro Saviolo, and Giuseppe Loianno. Decentralized nonlinear model predictive control for safe collision avoidance in quadrotor teams with limited detection range. *arXiv preprint arXiv:2409.17379*, 2024. 2, 3, 9
- [16] Quan Nguyen and Koushil Sreenath. Exponential control barrier functions for enforcing high relative-degree safety-critical constraints. In *American Control Conference (ACC)*, 2016. 2, 9
- [17] Zengyi Qin, Kaiqing Zhang, Yuxiao Chen, Jingkai Chen, and Chuchu Fan. Learning Safe Multi-agent Control with Decentralized Neural Barrier Certificates. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021. 2
- [18] Songyuan Zhang, Oswin So, Kunal Garg, and Chuchu Fan. Gcbf+: A neural graph control barrier function framework for distributed safe multi-agent control. *IEEE Transactions on Robotics*, 2025. 2, 10
- [19] Songyuan Zhang, Oswin So, Mitchell Black, and Chuchu Fan. Discrete GCBF Proximal Policy Optimization for Multi-agent Safe Optimal Control. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025. 2, 5, 9
- [20] Mo Chen, Jennifer C. Shih, and Claire J. Tomlin. Multi-vehicle collision avoidance via hamilton-jacobi reachability and mixed integer programming. In *IEEE Conference on Decision and Control (CDC)*, 2016. 3
- [21] Richard Bellman. *Dynamic Programming*. Princeton University Press, 1957. 3
- [22] Somil Bansal and Claire J Tomlin. Deepreach: A deep learning approach to high-dimensional reachability. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2021. 3, 6, 13
- [23] Kai-Chieh Hsu, Vicenç Rubies-Royo, Claire Tomlin, and Jaime F Fisac. Safety and Liveness Guarantees through Reach-Avoid Reinforcement Learning. In *Proceedings of Robotics: Science and Systems*, 2021. 3
- [24] Kai-Chieh Hsu, Duy Phuong Nguyen, and Jaime Fernandez Fisac. Isaacs: Iterative soft adversarial actor-critic for safety. In *Learning for Dynamics and Control Conference*. PMLR, 2023. 3
- [25] Kai Zhu, Fengbo Lan, Wenbo Zhao, and Tao Zhang. Safe Multi-Agent Reinforcement Learning via Approximate Hamilton-Jacobi Reachability. *Journal of Intelligent & Robotic Systems*, 2024. 3
- [26] Jingqi Li, Donggun Lee, Jaewon Lee, Kris Shengjun Dong, Somayeh Sojoudi, and Claire Tomlin. Certifiable Reachability Learning Using a New Lipschitz Continuous Value Function. *IEEE Robotics and Automation*

Letters, 2025. 3

- [27] Yujie Yang, Hanjiang Hu, Tianhao Wei, Shengbo Eben Li, and Changliu Liu. Scalable synthesis of formally verified neural value function for hamilton-jacobi reachability analysis. *arXiv preprint arXiv:2407.20532*, 2024. 3
- [28] Albert Lin and Somil Bansal. Verification of neural reachable tubes via scenario optimization and conformal prediction. In *Learning for Dynamics and Control Conference*. PMLR, 2024. 3
- [29] Abenezer G Taye, Josh Bertram, Chuchu Fan, and Peng Wei. Reachability based online safety verification for high-density urban air mobility trajectory planning. In *AIAA Aviation Forum*, 2022. 3
- [30] Josh Bertram and Peng Wei. Distributed computational guidance for high-density urban air mobility with cooperative and non-cooperative collision avoidance. In *AIAA Scitech Forum*, 2020. 3
- [31] Eitan Altman. *Constrained Markov decision processes*. Routledge, 2021. 3
- [32] Santiago Paternain, Luiz Chamon, Miguel Calvo-Fullana, and Alejandro Ribeiro. Constrained reinforcement learning has zero duality gap. *Advances in Neural Information Processing Systems*, 2019. 3
- [33] Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning. *arXiv preprint arXiv:1910.01708*, 2019. 3
- [34] Shangding Gu, Jakub Grudzien Kuba, Muning Wen, Ruiqing Chen, Ziyang Wang, Zheng Tian, Jun Wang, Alois Knoll, and Yaodong Yang. Multi-agent constrained policy optimisation. *arXiv preprint arXiv:2110.02793*, 2021. 3
- [35] Kai-Chieh Hsu, Haimin Hu, and Jaime F Fisac. The safety filter: A unified view of safety-critical control in autonomous systems. *Annual Review of Control, Robotics, and Autonomous Systems*, 2023. 3
- [36] Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 3
- [37] Ingy ElSayed-Aly, Suda Bharadwaj, Christopher Amato, Rüdiger Ehlers, Ufuk Topcu, and Lu Feng. Safe Multi-Agent Reinforcement Learning via Shielding. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2021. 3
- [38] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. 2017. 3
- [39] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015. 3
- [40] Chao Yu, Akash Velu, Eugene Vinitsky, Jiaxuan Gao, Yu Wang, Alexandre Bayen, and Yi Wu. The surprising effectiveness of ppo in cooperative multi-agent games. *Advances in Neural Information Processing Systems*, 2022. 3
- [41] Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in Neural Information Processing Systems*, 2017. 3, 9
- [42] Siddharth Nayak, Kenneth Choi, Wenqi Ding, Sydney Dolan, Karthik Gopalakrishnan, and Hamsa Balakrishnan. Scalable Multi-Agent Reinforcement Learning through Intelligent Information Aggregation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning*, PMLR, 2023. 3, 7
- [43] Nathan Slegers, Jason Kyle, and Mark Costello. Nonlinear model predictive control technique for unmanned air vehicles. *Journal of Guidance, Control, and Dynamics*, 2006. 3
- [44] Sami El-Ferik, Bilal A Siddiqui, and Frank L Lewis. Distributed nonlinear MPC of multi-agent systems with data compression and random delays. *IEEE Transactions on Automatic Control*, 2015. 3
- [45] Utku Eren, Anna Prach, Başaran Bahadır Koçer, Saša V Raković, Erdal Kayacan, and Behçet Açıkmeşe. Model predictive control in aerospace systems: Current state and opportunities. *Journal of Guidance, Control, and Dynamics*, 2017. 3
- [46] Dingjiang Zhou, Zijian Wang, Saptarshi Bandyopadhyay, and Mac Schwager. Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells. *IEEE Robotics and Automation Letters*, 2017. 3
- [47] Murad Dawood, Sicong Pan, Nils Dengler, Siqi Zhou, Angela P Schoellig, and Maren Bennewitz. Safe Multi-Agent Reinforcement Learning for Behavior-Based Cooperative Navigation. *arXiv preprint arXiv:2312.12861*, 2023. 3
- [48] Tamer Başar and Geert Jan Olsder. *Dynamic noncooperative game theory*. SIAM, 1998. 3
- [49] Antony Evans, Vikrant Vaze, and Cynthia Barnhart. Airline-driven performance-based air traffic management: Game theoretic models and multicriteria evaluation. *Transportation Science*, 2016. 3
- [50] Thulasi Mylvaganam, Mario Sassano, and Alessandro Astolfi. A differential game approach to multi-agent collision avoidance. *IEEE Transactions on Automatic Control*, 2017. 3
- [51] David Fridovich-Keil, Ellis Ratner, Lasse Peters, Anca D Dragan, and Claire J Tomlin. Efficient iterative linear-quadratic approximations for nonlinear multi-player general-sum differential games. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2020. 3
- [52] Maulik Bhatt, Yixuan Jia, and Negar Mehr. Efficient constrained multi-agent trajectory optimization using dy-

- dynamic potential games. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023. 3
- [53] Haimin Hu, Gabriele Dragotto, Zixu Zhang, Kaiqu Liang, Bartolomeo Stellato, and Jaime F. Fisac. Who Plays First? Optimizing the Order of Play in Stackelberg Games with Many Robots. *arXiv preprint arXiv:2402.09246*, 2024. 3
- [54] Forrest Laine, David Fridovich-Keil, Chih-Yuan Chiu, and Claire Tomlin. The computation of approximate generalized feedback nash equilibria. *SIAM Journal on Optimization*, 2023. 3
- [55] Jingqi Li, Somayeh Sojoudi, Claire J Tomlin, and David Fridovich-Keil. The Computation of Approximate Feedback Stackelberg Equilibria in Multiplayer Nonlinear Constrained Dynamic Games. *SIAM Journal on Optimization*, 2024. 3
- [56] Parimal Kopardekar, Joseph L. Rios, Thomas Prevot, Marcus Johnson, Jaewoo Jung, and John E. Robinson. Unmanned Aircraft System Traffic Management (UTM) Concept of Operations. 2016. 3
- [57] C. Tomlin, G.J. Pappas, and S. Sastry. Conflict resolution for air traffic management: a study in multi-agent hybrid systems. *IEEE Transactions on Automatic Control*, 1998. 3
- [58] Mo Chen, Qie Hu, Jaime F Fisac, Kene Akametalu, Casey Mackin, and Claire J Tomlin. Reachability-based safety and goal satisfaction of unmanned aerial platoons on air highways. *Journal of Guidance, Control, and Dynamics*, 2017. 3
- [59] Eva Cruck and John Lygeros. Subliminal air traffic control: Human friendly control of a multi-agent system. In *American Control Conference*, 2007. 3
- [60] Min Xue, Abraham K. Ishihara, and Paul U. Lee. Negotiation Model for Cooperative Operations in Upper Class E Airspace. In *IEEE/AIAA Digital Avionics Systems Conference (DASC)*, 2022. 3
- [61] James Kuchar, John Andrews, Ann Drumm, Tim Hall, Val Heinz, Steven Thompson, and Jerry Welch. A safety analysis process for the traffic alert and collision avoidance system (TCAS) and see-and-avoid systems on remotely piloted vehicles. In *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*, 2004. 3
- [62] Heinz Erzberger and Karen Heere. Algorithm and operational concept for resolving short-range conflicts. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 2010. 3
- [63] Jessica E Holland, Mykel J Kochenderfer, and Wesley A Olson. Optimizing the next generation collision avoidance system for safe, suitable, and acceptable operational performance. *Air Traffic Control Quarterly*, 2013. 3
- [64] Claire Tomlin, G Pappas, J Lygeros, D Godbole, S Sastry, and G Meyer. Hybrid Control in Air Traffic Management Systems. *IFAC Proceedings Volumes*, 1996. 3
- [65] Huabin Tang, Dallas Denery, Heinz Erzberger, and Russell Paielli. Tactical Separation Algorithms and Their Interaction with Conflict Avoidance Systems. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008. 3
- [66] Heinz Erzberger, Todd A Lauderdale, and Yung-Cheng Chu. Automated conflict resolution, arrival management, and weather avoidance for air traffic management. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of aerospace engineering*, 2012. 3
- [67] Shulu Chen, Antony D. Evans, Marc Brittain, and Peng Wei. Integrated Conflict Management for UAM With Strategic Demand Capacity Balancing and Learning-Based Tactical Deconfliction. *IEEE Transactions on Intelligent Transportation Systems*, 2024. 4
- [68] Kim P. Wabersich, Andrew J. Taylor, Jason J. Choi, Koushil Sreenath, Claire J. Tomlin, Aaron D. Ames, and Melanie N. Zeilinger. Data-Driven Safety Filters: Hamilton-Jacobi Reachability, Control Barrier Functions, and Predictive Methods for Uncertain Systems. *IEEE Control Systems Magazine*, 2023. 5
- [69] Stanford Autonomous Systems Lab. *hj_reachability: Hamilton-Jacobi reachability analysis in JAX*, 2025. Accessed: 2025-01-28. 5
- [70] Somil Bansal, Mo Chen, Sylvia Herbert, and Claire J. Tomlin. Hamilton-Jacobi reachability: A brief overview and recent advances. In *IEEE Conference on Decision and Control (CDC)*, 2017. 5, 6
- [71] Jaime F Fisac, Neil F Lugovoy, Vicenç Rubies-Royo, Shromona Ghosh, and Claire J Tomlin. Bridging hamilton-jacobi safety analysis and reinforcement learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019. 5
- [72] Vamsi Krishna Chilakamarri, Zeyuan Feng, and Somil Bansal. Reachability analysis for black-box dynamical systems. *arXiv preprint arXiv:2410.07796*, 2024. 5
- [73] Jason J Choi, Donggun Lee, Boyang Li, Jonathan P How, Koushil Sreenath, Sylvia L Herbert, and Claire J Tomlin. A forward reachability perspective on robust control invariance and discount factors in reachability analysis. *arXiv preprint arXiv:2310.17180*, 2023. 5
- [74] I.M. Mitchell, A.M. Bayen, and C.J. Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 2005. 6
- [75] Jaime F. Fisac, Anayo K. Akametalu, Melanie N. Zeilinger, Shahab Kaynama, Jeremy Gillula, and Claire J. Tomlin. A General Safety Framework for Learning-Based Control in Uncertain Robotic Systems. *IEEE Transactions on Automatic Control*, 2019. 6
- [76] Ronojoy Ghosh and Claire Tomlin. Maneuver design for multiple aircraft conflict resolution. In *American Control Conference*, 2000. 7
- [77] Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research*,

- [78] Joby Aviation. Joby aviation - official website, 2025. URL <https://www.jobyaviation.com/>. Accessed: 2025-01-31. 8
- [79] Archer Aviation. Archer aviation - our aircraft, 2025. URL <https://archer.com/aircraft>. Accessed: 2025-01-31. 8
- [80] Wisk Aero. Wisk aero - our aircraft, 2025. URL <https://wisk.aero/aircraft/>. Accessed: 2025-01-31. 8
- [81] Kimberlee H. Shish, Nick B. Cramer, George Gorospe, Thomas Lombaerts, Vahram Stepanyan, and Keerthana Kannan. *Survey of Capabilities and Gaps in External Perception Sensors for Autonomous Urban Air Mobility Applications*. 2021. 8, 11
- [82] Luis E Alvarez, Ian Jessen, Michael P Owen, Joshua Silberman, and Paul Wood. ACAS sXu: Robust decentralized detect and avoid for small unmanned aircraft systems. In *IEEE/AIAA Digital Avionics Systems Conference (DASC)*, 2019. 9
- [83] John D. Anderson. *Introduction to Flight*. McGraw-Hill Education, New York, NY, 8th edition, 2016. 9
- [84] Federal Aviation Administration. *Pilot's Handbook of Aeronautical Knowledge*. U.S. Department of Transportation, Federal Aviation Administration, 2023. 9
- [85] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International Conference on Machine Learning*. PMLR, 2017. 9
- [86] Salar Asayesh, Mo Chen, Mehran Mehrandehz, and Kamal Gupta. Least-restrictive multi-agent collision avoidance via deep meta reinforcement learning and optimal control. In *International Conference on Robot Intelligence Technology and Applications*. Springer, 2022. 9
- [87] Vishwanath Bulusu, Emin Burak Onat, Raja Sengupta, Pavan Yedavalli, and Jane Macfarlane. A Traffic Demand Analysis Method for Urban Air Mobility. *IEEE Transactions on Intelligent Transportation Systems*, 2021. 10
- [88] Seungman Lee, Michael Abramson, James D. Phillips, and Huabin Tang. Preliminary Analysis of Separation Standards for Urban Air Mobility using Unmitigated Fast-Time Simulation. In *IEEE/AIAA Digital Avionics Systems Conference (DASC)*, 2022. 10, 11
- [89] Federal Aviation Administration. Aeronautical Information Manual (AIM), Chapter 7: Safety of Flight, Section 7: Near Midair Collision Reporting. https://www.faa.gov/air_traffic/publications/atpubs/aim_html/chap7_section_7.html. Accessed: 26 January 2025. 11
- [90] Albert Reuther, Jeremy Kepner, Chansup Byun, Siddharth Samsi, William Arcand, David Bestor, Bill Bergeron, Vijay Gadepally, Michael Houle, Matthew Hubbell, Michael Jones, Anna Klein, Lauren Milechin, Julia Mullen, Andrew Prout, Antonio Rosa, Charles Yee, and Peter Michaleas. Interactive supercomputing on 40,000 cores for machine learning and data analysis. In *IEEE High Performance extreme Computing Conference (HPEC)*, 2018. 13
- [91] Insoon Yang, Sabine Becker-Weimann, Mina J Bissell, and Claire J Tomlin. One-shot computation of reachable sets for differential games. In *Proceedings of the International Conference on Hybrid Systems: Computation and Control*, 2013. 13
- [92] Xubo Lyu and Mo Chen. Ttr-based reward for reinforcement learning with implicit model priors. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020. 13