

Safe and Scalable Multi-Drone Payload Transport via CBF-based Reinforcement Learning with Zero-Shot Sim-to-Real Transfer

Jaeyoun Choi¹, Oswin So¹, Songyuan Zhang¹, Cooper Taylor², and Chuchu Fan¹

Abstract—Multi-drone payload transportation has emerged as a promising research paradigm with potential applications in construction, logistics, and disaster response. However, the complex coupled dynamics among drones, cables, and payloads pose significant challenges, and existing approaches remain limited in safety and scalability, particularly in dynamic and unstructured environments. In this work, we propose a learning-based framework for safe and scalable multi-drone cooperative payload transport. We introduce a minimal 2D abstraction that preserves the task-relevant drone–payload coupling required for coordination and safety, while remaining computationally efficient for large-scale learning. Using domain randomization over team size and physical parameters, we train a fully distributed policy via Discrete Graph Control Barrier Function Proximal Policy Optimization (DGPPPO), enabling robust zero-shot sim-to-real transfer without fine-tuning. Extensive real-world evaluations demonstrate that a single learned policy generalizes across varying team sizes and task scenarios. Furthermore, multi-group hardware experiments show that the same policy can safely operate in dynamic environments, where other drone teams act as moving obstacles. These results indicate that the proposed framework enables efficient, safe, and scalable multi-drone payload transportation with strong generalization to complex real-world conditions.

Index Terms—Multi-Robot Systems, Robot Safety, Aerial Systems: Applications

I. INTRODUCTION

COOPERATIVE payload transportation is a central problem in multi-robot systems, where multiple robots physically coordinate to move a shared payload [1]–[6]. One prominent example is cable-suspended multi-drone systems, which offer structural simplicity, modularity, and adaptability [7]. These systems are attractive for practical applications, including last-mile delivery, disaster relief, automated construction, and logistics.

Scalability is essential because larger teams can improve payload capacity, load distribution, and deployment flexibility [7]–[9]. However, scaling beyond a few drones remains challenging in real-world settings due to coupled cable-payload interactions, inter-agent interactions, sensing noise,

and hardware heterogeneity [7]–[9]. Consequently, most real-world demonstrations remain limited to two or three drones.

Safety further compounds this challenge. Prior work has demonstrated cooperative aerial transport using centralized optimization, distributed trajectory generation, or decentralized sensing [5], [8], [10]–[12]. However, real-world validation has largely remained limited to small teams and structured environments, while safety is often enforced through *centralized* planners that rely on explicit environment representations [7], [9], [13], [14]. Achieving scalable payload transport with reactive environment-level safety under fully *distributed* execution, therefore, remains largely open.

To address these challenges, we present a learning-based framework for scalable, safe cable-suspended multi-drone transportation. To our knowledge, no prior work has demonstrated real-world cable-suspended multi-drone transportation with a safe, *fully distributed* policy in dynamic environments with moving obstacles. Our main **contributions** are:

- **Efficient simulation and domain randomization for scalable and safe policy learning:** We develop an efficient simulation environment based on a minimal 2D transport abstraction. We learn a shared-parameter distributed policy by using domain randomization that generalizes across team sizes and task instances while enforcing safety constraints.
- **Bridging learned discrete-time safety to continuous-time robotic execution:** We provide a safety analysis that connects discrete-time safety guarantees obtained from learned control barrier functions to continuous-time execution under hierarchical tracking control, enabling their practical deployment on real robotic systems.
- **Scalable and safe real-world multi-drone transportation:** We demonstrate reliable payload transport on Crazyflie teams of up to six drones, exceeding the training range, including multi-group hardware experiments in which other drone teams act as dynamic obstacles.

II. RELATED WORK

A. Control of Multi-Drone Payload Transportation

A wide range of control and planning methods has been applied to cable-suspended multi-drone transportation systems. Centralized control frameworks have demonstrated precise object transport in simulation or experimental settings [9]–[11], [13], [15]–[21]. While effective for small teams, these methods often rely on global state information and centralized computation, which limits their scalability in real-world hardware demonstrations.

¹Jaeyoun Choi, Oswin So, Songyuan Zhang, and Chuchu Fan are with the Reliable Autonomous Systems Lab, Massachusetts Institute of Technology, Cambridge, MA, USA. Emails: {cjy0051, oswinso, szhang21, chuchu}@mit.edu.

²Cooper Taylor contributed to this work while participating in the Research Science Institute program at the Reliable Autonomous Systems Lab, Massachusetts Institute of Technology, Cambridge, MA, USA. Email: cooperetaylor@gmail.com.

The authors were partly funded by the Ministry of Trade, Industry and Energy (MOTIE), Korea, through the Global Industrial Technology Cooperation Center (GITCC) Program, supervised by the Korea Institute for Advancement of Technology (KIAT), Task No. P24680172.

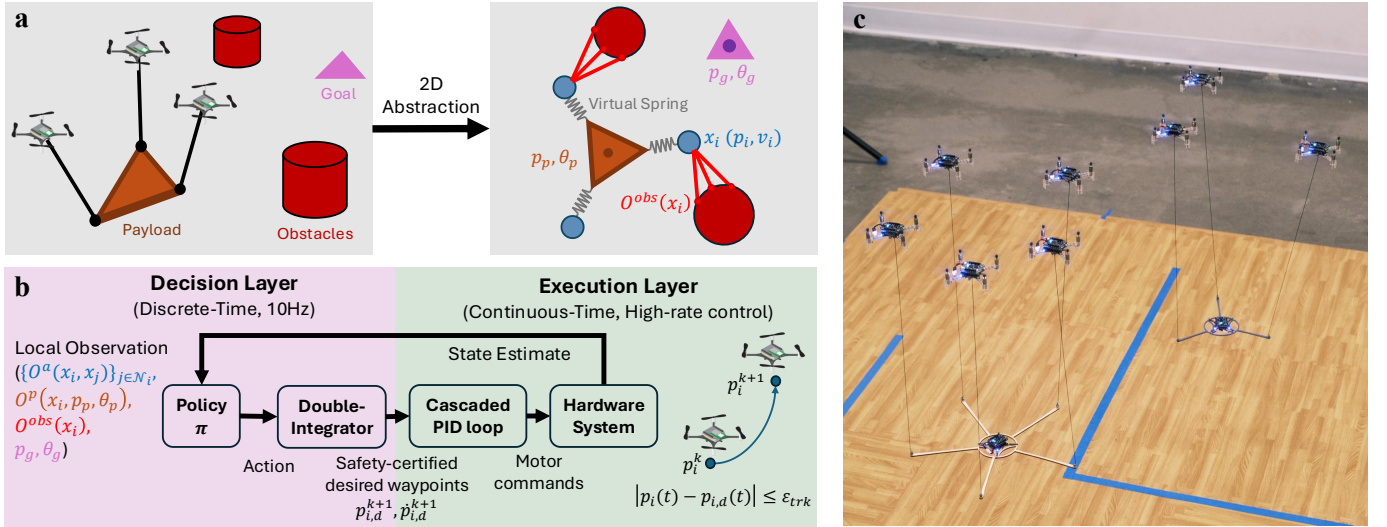


Fig. 1: **Distributed learning framework for safe and scalable multi-drone transportation.** (a) Real-world cable-suspended transport and corresponding 2D abstraction. Drones are modeled as point masses connected to the payload via virtual springs, and obstacles are detected with LiDAR. (b) Control architecture and discrete-to-continuous safety bridge. A distributed policy π generates DGCBF-certified waypoints from local observations, tracked by an onboard PID controller. With bounded tracking error ϵ_{trk} , tightened DGCBF constraints ensure continuous-time safety. (c) Zero-shot real-world deployment. The learned policy transfers directly to physical quadrotors without fine-tuning, demonstrating robust coordination in dynamic environments.

In parallel, decentralized [22]–[24] and distributed [8], [11], [12], [25]–[29] control strategies have been explored. These approaches leverage local sensing and limited communication, and several have demonstrated real-world feasibility with two or three drones. However, most existing methods either rely on leader-based architectures with a single point of failure or are validated mainly in simulation.

Learning-based approaches are another attractive control approach due to their ability to handle nonlinear dynamics and partial observability. However, most prior works focus on simulation-only results, and successful sim-to-real transfer remains limited [30]–[35]. Zeng et al. [36] recently demonstrated hardware experiments based on decentralized reinforcement learning, but their approach assumes collision-free environments and does not explicitly address safety.

B. Safety in Multi-Robot Transportation Systems

Safety is critical for real-world multi-drone transportation, especially in cluttered and dynamic environments. Recent safety-aware approaches enforce safety constraints through centralized, model-based optimization [8], [21] or exploit internal system redundancy for human-aware safety under hierarchical control [14]. However, these methods typically rely on global state information and centralized computation, which can hinder scalability and complicate deployment in distributed robotic systems.

III. PROBLEM FORMULATION

Our goal is to learn distributed control policies for N drones to collaboratively transport a suspended payload to a target position p_g and orientation θ_g , while avoiding unsafe configurations such as collisions or boundary violations. We formulate the task as a multi-agent constrained optimal control

problem under partial observability. Let the state of drone i at time step k be $x_i^k = (p_i^k, v_i^k)$, where $p_i^k \in \mathbb{R}^2$ denotes the planar position and $v_i^k \in \mathbb{R}^2$ the planar velocity. The control input is the commanded horizontal acceleration $u_i^k \in \mathcal{U}$ with $\mathcal{U} := \{u \in \mathbb{R}^2 \mid u_{\min} \leq u \leq u_{\max}\}$.

The full system state is

$$\mathbf{x}^k := [x_1^k, \dots, x_N^k, p_p^k, \theta_p^k, v_p^k, \omega_p^k, p_g, \theta_g] \in \mathcal{X},$$

which includes all drone states as well as the payload's position p_p^k , orientation θ_p^k , velocity v_p^k , and angular velocity ω_p^k . Denote the joint action by $\mathbf{u}^k := [u_1^k, \dots, u_N^k]$. The system evolves according to discrete-time dynamics

$$\mathbf{x}^{k+1} = f(\mathbf{x}^k, \mathbf{u}^k),$$

where f captures the coupled drone-payload interactions induced by the cable dynamics that will be described in the following section.

Each agent operates under partial observability. Specifically, drone i receives a local observation

$$o_i^k = \left(\{o_{ij}^k\}_{j \in \mathcal{N}_i}, o_i^{\text{payload},k}, o_i^{\text{obs},k}, p_g, \theta_g \right), \quad (1)$$

where $\mathcal{N}_i = \{j \mid \|p_j^k - p_i^k\| \leq R\}$ denotes the set of nearby agents within sensing radius R . The observation components are given by

$$\begin{cases} o_{ij}^k = \mathcal{O}^a(x_i^k, x_j^k), & (\text{inter-agent features}), \\ o_i^{\text{payload},k} = \mathcal{O}^p(x_i^k, p_p^k, \theta_p^k), & (\text{payload features}), \\ o_i^{\text{obs},k} = \mathcal{O}^{\text{obs}}(p_i^k), & (\text{obstacle features}), \end{cases} \quad (2)$$

where $\mathcal{O}^{\text{obs}}$ represents LiDAR-based obstacle observations, consisting of points on obstacle surfaces detected by each agent within a sensing radius R_{LiDAR} .

Safety requirements are encoded as M local constraint functions $\{h_i^{(m)}\}_{m=1}^M$ for each agent i , where $h_i^{(m)} : \mathcal{O} \rightarrow \mathbb{R}$

is the m -th observation-based constraint. The corresponding unsafe/avoid set for agent i is

$$\mathcal{A}_i := \left\{ o_i \in \mathcal{O} \mid \exists m \in \{1, \dots, M\} : h_i^{(m)}(o_i) > 0 \right\},$$

which can represent inter-agent and agent-obstacle collisions, workspace boundary violations, or other task-specific safety limits.

The objective is to learn a distributed policy $\pi : \mathcal{O} \rightarrow \mathcal{U}$ such that given a cost function l describing a task, the joint policy $\pi(\mathbf{x}^k) := [\pi(o_1^k), \dots, \pi(o_N^k)]$ minimizes the cumulative task cost while maintaining safety:

$$\min_{\pi} \sum_{k=0}^{\infty} \ell(\mathbf{x}^k, \pi(\mathbf{x}^k)) \quad (3a)$$

$$\text{s.t. } \mathbf{x}^{k+1} = f(\mathbf{x}^k, \pi(\mathbf{x}^k)), \quad (3b)$$

$$o_i^k = \mathcal{O}_i(\mathbf{x}^k), \quad \forall i, k, \quad (3c)$$

$$h_i^{(m)}(o_i^k) \leq 0, \quad \forall i, m, k. \quad (3d)$$

IV. FRAMEWORK FOR SCALABLE AND SAFE MULTI-DRONE TRANSPORT

A. 2D Abstraction for Multi-Drone Transport

Modeling cable-suspended multi-drone transportation in full three-dimensional space is challenging due to strongly coupled, hybrid dynamics, including cable slack-taut transitions and unmodeled aerodynamic effects. While high-fidelity models can partially capture these phenomena, they significantly increase computational complexity. Moreover, a recent study demonstrated that modeling fidelity alone does not eliminate the sim-to-real gap in practice [36].

In this work, we therefore adopt a minimal 2D abstraction (Fig. 1a) that preserves the task-relevant coupling between drone configurations and the resulting net force and torque applied to the payload, while abstracting away out-of-plane dynamics that are regulated by a low-level controller. The goal of this abstraction is to retain the decision-relevant interactions necessary for coordination and safety while enabling computationally efficient policy learning.

The abstraction is based on the following assumptions, which hold for tasks in which altitude is regulated and payload motion remains close to planar alignment:

- 1) Drones maintain a regulated altitude, such that vertical motion introduces only bounded disturbances to the planar dynamics.
- 2) The tension of each cable is proportional to its 2D projected length, capturing the dominant planar force contribution.
- 3) The payload remains approximately planar, with roll and pitch motions negligible compared to the translational and yaw dynamics relevant to the task.
- 4) Cable attachment points are fixed and evenly distributed around the payload's center of mass.

Under these assumptions, the payload is modeled as a planar rigid body governed by

$$M_p \ddot{p}_p = \sum_{i=1}^N T_i, \quad I_p \ddot{\theta}_p = \sum_{i=1}^N r_i \times T_i, \quad (4)$$

where $p_p \in \mathbb{R}^2$ denotes the planar position of the payload, θ_p its yaw angle, M_p its mass, I_p its moment of inertia about vertical axis, N the number of agents, and $T_i \in \mathbb{R}^2$ the planar tension applied at attachment point r_i , expressed in the payload frame. Under the above assumptions, each cable shares the payload weight through its vertical component, while the planar component of the tension increases with the cable angle relative to the vertical. Specifically, we model the tension vector T_i for drone i as a virtual spring pulling toward its designated target location on the payload perimeter:

$$T_i = \mathcal{K} \left[p_i - \left(p_p + r \cdot \begin{bmatrix} \cos\left(\frac{2\pi i}{N}\right) \\ \sin\left(\frac{2\pi i}{N}\right) \end{bmatrix} \right) \right], \quad (5)$$

where $\mathcal{K}$ is a spring coefficient, p_i is the position vector of the i th drone, and r is the radial distance from the center of the payload to the cable attachment point. Payload orientation is governed by the planar cable forces, which depend primarily on drone positions rather than drone orientation.

We discretize these dynamics using a double-integrator update with timestep Δt , which serves as the decision-level model for policy learning and safety certification. Any discrepancy between this abstraction and the continuous-time three-dimensional execution is handled by closed-loop tracking control and explicitly accounted for in the safety analysis in Section V.

B. Learning Safe Distributed Policies with DGPPO

1) Discrete Graph Control Barrier Function Proximal Policy Optimization (DGPPO)

We apply a sim-to-real transfer approach, where the policy is learned in simulation and then transferred to real-world hardware. DGPPO [37] is adopted as a distributed policy optimization backbone for training multi-drone transport policies under partial observability and safety constraints. Each agent executes a shared-parameter policy π using only local graph-based observations, where a gated recurrent unit (GRU) maintains an internal memory to mitigate partial observability.

DGPPO jointly learns a distributed control policy and a constraint value function that serves as a Discrete Graph Control Barrier Function (DGCBF) [38]. Control Barrier Functions (CBFs) enforce safety by maintaining control-invariant safe sets [39], [40]; however, classical formulations typically assume centralized access to the full system state. Such assumptions are impractical in our setting due to complex coupled dynamics and partial observability. Instead, DGCBFs certify safety directly in the observation space using only local information, enabling distributed safety assessment and filtering policy updates that are predicted to violate constraints without requiring global state access. We refer readers to [37] for full algorithmic details and guarantees, and formalize the continuous-time execution implications in section V.

2) Domain Randomization

To promote robustness and generalization, we apply domain randomization during training. Specifically, during each training episode, the number of agents is randomly selected from $\{3, 4, 5\}$. For each case, the agents are symmetrically arranged around the payload with equal radial distance from its center.

In addition, we randomize the virtual spring stiffness K , which governs the tension dynamics between each drone and the payload. The stiffness coefficient is uniformly sampled from the range $[0.05, 0.30] \text{N m}^{-1}$. This range is inspired by the linearized spring approximation of a pendulum, where the effective stiffness can be estimated as $k \approx \frac{mg}{Nl}$ (e.g., for $m \approx 0.027 \text{kg}$, $N = 3$, $l = 0.5 \text{m}$, and $k \approx 0.18 \text{N m}^{-1}$). By sampling across a broad range around this nominal value, we ensure that the policy remains effective under different tether dynamics and supports smooth sim-to-real transfer.

V. CONTINUOUS-TIME SAFETY UNDER HIERARCHICAL CONTROL

Zhang et al. [37] have shown that DGCBFs can guarantee safety for multi-agent systems under discrete-time dynamics and decentralized observations. In practice, however, policies learned and certified in discrete time are executed on physical systems that evolve continuously, typically through a hierarchical control architecture in which discrete-time decisions are tracked by a high-bandwidth low-level controller (Fig. 1b). This section bridges this gap by formalizing conditions under which DGCBF-based safety guarantees extend from discrete-time decision making to continuous-time execution. We begin by recalling the following corollary from prior work [37].

Corollary 1 (DGCBF). *Define the constrained value function $\tilde{V}_{h^{(m)}, \pi} : \mathcal{O} \rightarrow \mathbb{R}$, which depends only on local observations $o_i = O_i(x)$, as*

$$\tilde{V}_{h^{(m)}, \pi}(o_i^0) := \max_{k \geq 0} h^{(m)}(o_i^k), \quad (6)$$

Then, the function $\tilde{V}_{h^{(m)}, \pi}$ is DGCBF.

Assumption 1 (Tracking error bound). *At discrete decision steps k , the policy outputs desired accelerations u_i^k , which are integrated to generate constant position references $p_{i,d}(t)$ over each decision interval $t \in [k\Delta t, (k+1)\Delta t)$. A low-level feedback controller operating at a higher update rate tracks these references, yielding a continuous-time closed-loop trajectory. There exists a known constant $\varepsilon_{\text{trk}} > 0$ such that, for all agents i and all $t \geq 0$,*

$$\|p_i(t) - p_{i,d}(t)\| \leq \varepsilon_{\text{trk}}. \quad (7)$$

Assumption 2 (Lipschitz safety constraints). *Each local safety constraint function $h_i^{(m)}$ is Lipschitz in o_i with constant $L_{i,m} \geq 0$, i.e.,*

$$|h_i^{(m)}(o_i) - h_i^{(m)}(\bar{o}_i)| \leq L_{i,m} \|o_i - \bar{o}_i\|, \quad \forall o_i, \bar{o}_i \in \mathcal{O}. \quad (8)$$

Assumption 3 (Observation sensitivity to tracking error). *There exists a constant $L_{O,i} \geq 0$ such that for all $t \geq 0$,*

$$\|o_i(t) - o_{i,d}(t)\| \leq L_{O,i} \|p_i(t) - p_{i,d}(t)\|. \quad (9)$$

Definition 1 (Tracking-error-tightened constraints). *Define the tightened constraint functions*

$$h_{i,\text{tight}}^{(m)}(o_i) := h_i^{(m)}(o_i) + L_{i,m} L_{O,i} \varepsilon_{\text{trk}}. \quad (10)$$

We now show that, under bounded tracking error and mild regularity assumptions, DGCBF-certified policies ensure continuous-time safety.

Theorem 1 (Continuous-time safety under hierarchical tracking). *Suppose Assumptions 1, 2, and 3 hold. Let $t_k := k\Delta t$ and let $o_{i,d}^k := o_{i,d}(t)$ for $t \in [t_k, t_{k+1})$, where $o_{i,d}(t)$ is formed by replacing $p_i(t)$ with $p_{i,d}(t)$ in $o_i(t)$. Assume the distributed policy π renders the tightened constraints (10) forward invariant at decision steps on the reference observations, i.e., for all k ,*

$$h_{i,\text{tight}}^{(m)}(o_{i,d}^k) \leq 0 \Rightarrow h_{i,\text{tight}}^{(m)}(o_{i,d}^{k+1}) \leq 0, \quad \forall i, m. \quad (11)$$

Then, if $h_{i,\text{tight}}^{(m)}(o_{i,d}^0) \leq 0$ for all i, m , the original continuous-time constraints hold for all time:

$$h_i^{(m)}(o_i(t)) \leq 0, \quad \forall t \geq 0, \forall i, \forall m. \quad (12)$$

Proof. By (11) and the initial condition $h_{i,\text{tight}}^{(m)}(o_{i,d}^0) \leq 0$, we have

$$h_{i,\text{tight}}^{(m)}(o_{i,d}^k) \leq 0, \quad \forall k \in \mathbb{N}, \forall i, m. \quad (13)$$

Fix any $t \geq 0$, and let k be such that $t \in [t_k, t_{k+1})$. Under Assumption 1, the reference is constant on each decision interval, i.e., $p_{i,d}(t) = p_{i,d}^k$ for $t \in [t_k, t_{k+1})$, and thus $o_{i,d}(t) = o_{i,d}^k$ on the same interval. Therefore,

$$h_{i,\text{tight}}^{(m)}(o_{i,d}(t)) = h_{i,\text{tight}}^{(m)}(o_{i,d}^k) \leq 0. \quad (14)$$

It follows from Definition 1 and (14) that

$$h_i^{(m)}(o_{i,d}(t)) \leq -L_{i,m} L_{O,i} \varepsilon_{\text{trk}}. \quad (15)$$

By Assumption 2,

$$h_i^{(m)}(o_i(t)) \leq h_i^{(m)}(o_{i,d}(t)) + L_{i,m} \|o_i(t) - o_{i,d}(t)\|. \quad (16)$$

Using Assumption 3 and Assumption 1,

$$\|o_i(t) - o_{i,d}(t)\| \leq L_{O,i} \|p_i(t) - p_{i,d}(t)\| \leq L_{O,i} \varepsilon_{\text{trk}}.$$

Substituting this bound and (15) into (16) gives

$$h_i^{(m)}(o_i(t)) \leq (-L_{i,m} L_{O,i} \varepsilon_{\text{trk}}) + L_{i,m} L_{O,i} \varepsilon_{\text{trk}} = 0.$$

Therefore $h_i^{(m)}(o_i(t)) \leq 0$ for all $t \geq 0$, proving (12). $\square$

VI. EXPERIMENTS

A. Simulation Experiments

We evaluate our approach in a simulation environment based on the proposed 2D abstraction of the multi-agent cooperative transport task. In each episode, $N \in \{3, 4, 5\}$ agents (radius $r = 0.09 \text{m}$, mass 0.027kg) collaboratively translate and rotate a rigid N -gon payload (mass 0.045kg) toward a target pose, while circular obstacles are randomly sampled in the environment. The payload mass is fixed, while cable stiffness is randomized to capture variations in the dynamics. System dynamics are integrated using a JAX-based physics engine with timestep $\Delta t = 0.1 \text{s}$ and five substeps per decision step.

Each agent is equipped with a 32-beam planar LiDAR sensor with a maximum range of 0.5m and communicates with neighboring agents within a radius of 0.25m . Observations include the agent's own kinematic state, the payload pose and velocity, the relative goal, and the shortest-8 LiDAR returns. To model sensing uncertainty, we inject uniform noise of $\pm 5\%$ on positions and orientations and $\pm 30\%$ on velocities.

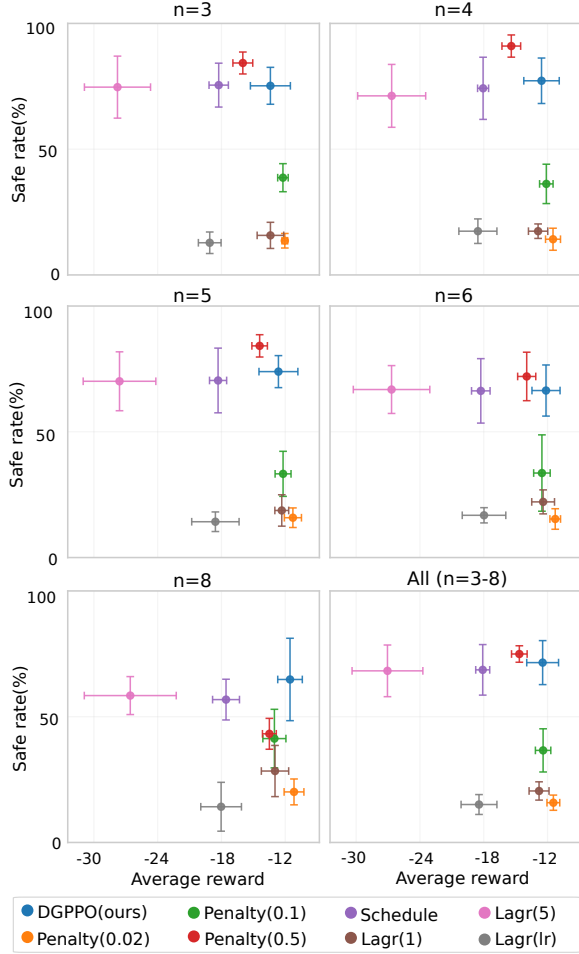


Fig. 2: **Simulation results across different team sizes.** For each number of agents ($n \in \{3, 4, 5, 6, 8\}$), we evaluate policies on 25 randomly generated scenarios. Each marker reports the mean reward and safety rate across five evaluation seeds (125 scenarios total), with error bars showing variability across seeds. We compare several distributed policy optimization backbones under the same setup.

Policies were trained for 10^5 steps using 128 parallel environments in a vectorized JAX-based simulator. Training required approximately 12 hours on a single workstation equipped with an Intel Core Ultra 9 285K CPU, 62 GiB of RAM, and an NVIDIA GeForce RTX 5090 GPU.

1) Reward and Safety Constraints Design

Reward. The reward function encourages the payload to reach the target pose while promoting smooth and low-effort control actions. At each timestep t , the reward is defined as

$$r_t = -w_d \|p_p - p_g\|_2^2 - w_\theta \|\theta_p - \theta_g\|_2^2 - w_{thr} \mathbb{1}[\|p_p - p_g\|_2 > d_{thr}] - w_u \frac{1}{N} \sum_{i=1}^N \|u_i\|_2^2 - w_s \frac{1}{N} \sum_{i=1}^N \|u_i - u_i^{\text{prev}}\|_2^2, \quad (17)$$

where p_p and p_g represent the payload and goal positions, θ_p and θ_g their corresponding orientations, u_i the planar control action of agent i , and u_i^{prev} the control action at the previous time step. The weights are set to $w_d = 0.06$, $w_\theta = 0.06$, $w_{thr} = 0.001$, $w_u = 0.1$, and $w_s = 0.1$. We optimize policies

by maximizing the cumulative reward, which is equivalent to minimizing a task cost defined as the negative reward, consistent with the formulation in (3).

Safety Constraints. In addition to the reward, we compute per-agent constraints that quantify safety. These constraints are used for reporting and for learning distributed safety certificates. The per-agent constraint function for agent i is defined as

$$h_i = \begin{bmatrix} 4(2r - \min_{j \neq i} \|p_i - p_j\|_2) \\ 2(r - \min_k \|p_i - o_k\|_2) \\ 2d_{po} \\ 10(\|p_i - p_{v,i}\|_2 - d_{tol}) \end{bmatrix}, \quad (18)$$

where p_i denotes the position of agent i , o_k denotes the position of the k -th obstacle, and $p_{v,i}$ is the position of the attachment vertex assigned to agent i . The scalar d_{po} denotes the payload-obstacle distance, and d_{tol} specifies the allowable deviation between an agent and its assigned attachment vertex. Each constraint component is margin-shifted by $\varepsilon = 0.5$ and clipped to the range $[-1, 1]$.

2) Evaluation Metrics

We evaluate policy performance using two metrics: average reward and safety rate. For evaluation, we generate multiple randomized scenarios in which the initial and goal payload positions and orientations, as well as obstacle sizes and locations, are randomly sampled. For each team size $n \in \{3, 4, 5, 6, 8\}$, we evaluate the policy on 25 distinct scenarios. The average reward is computed over the full episode, while the safety rate is defined as the fraction of agents that satisfy all safety constraints throughout the execution.

3) Policy Optimization Backbones

We compare representative policy optimization backbones for our framework and adopt DGPPPO based on the comparison. Specifically, we considered the MARL algorithm InforMARL [41] and the safe MARL algorithm MAPPO-Lagrangian [42], [43], which were trained and tested under the same environmental settings. For InforMARL, constraint violations are aggregated as $\text{Penalty}(\beta) = \beta \sum_{i=1}^N \sum_{m=1}^M \max(0, h_i^{(m)})$ and added to the cost function. We additionally evaluated a weight-scheduling scheme (*Schedule*), where β was initialized to 0.01 and increased at 50% and 75% of the total training steps. For MAPPO-Lagrangian, we employed a GNN-based policy backbone to ensure architectural comparability across backbones. We evaluated two different initializations of the Lagrange multiplier with a learning rate of 10^{-7} , denoted as *Lagr*(λ_0), and additionally increased the learning rate to 10^{-4} (*Lagr*(lr)) with $\lambda_0 = 1$. All methods were trained for the same number of update steps, selected to be sufficiently large to ensure convergence across all approaches.

4) Backbone Comparison

Fig. 2 compares the backbones across team sizes. DGPPPO achieves a favorable reward-safety trade-off: it maintains high safety rates while preserving competitive task reward compared with the Lagrangian and penalty-based variants. This trend remains stable as the number of agents increases from the training range of 3–5 to larger teams such as 6 and 8, indicating scalability beyond the training team sizes.

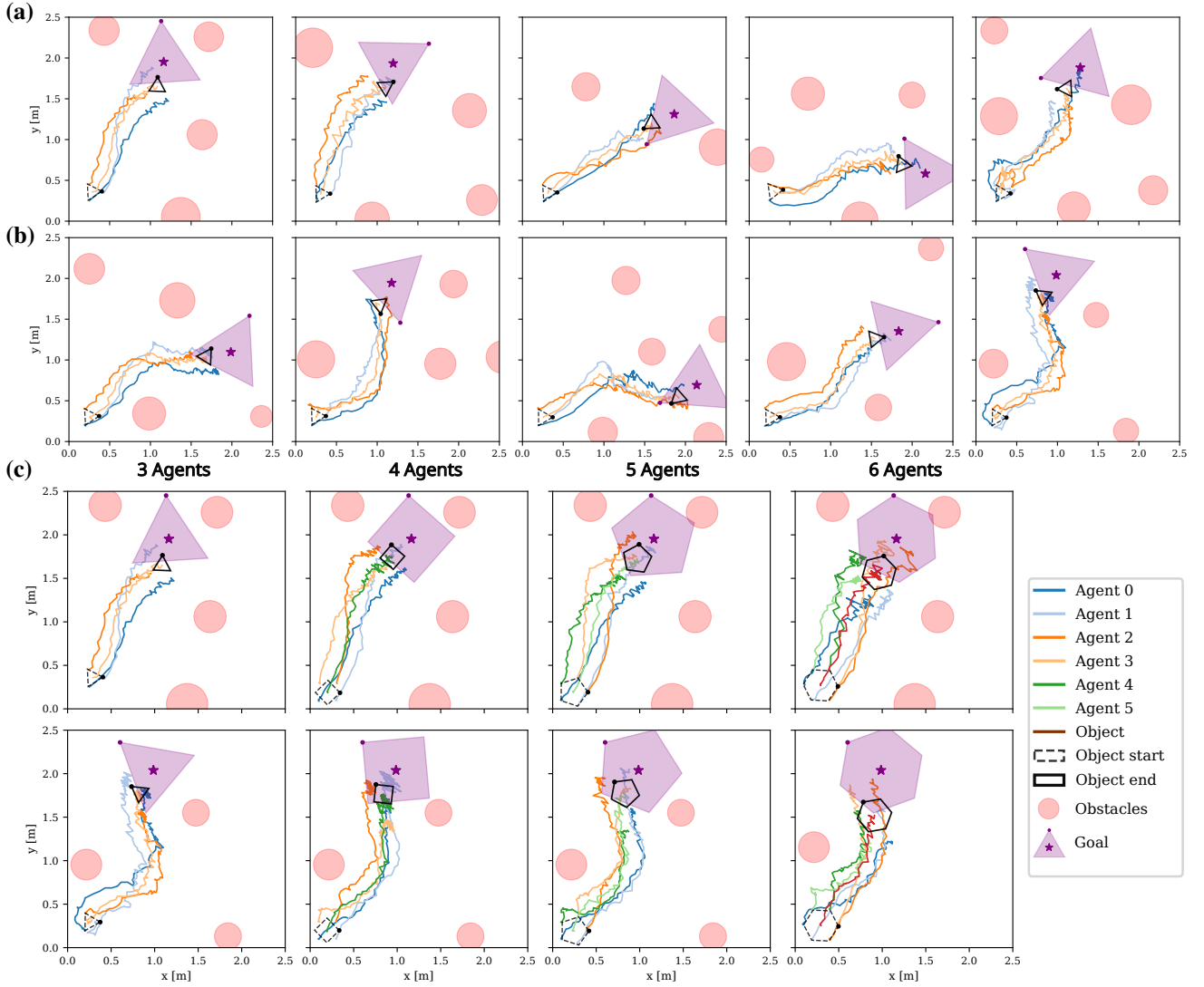


Fig. 3: **Hardware validation of distributed multi-drone transportation.** (a) Easy scenarios where obstacles do not significantly affect the payload trajectory. (b) Hard scenarios where obstacles actively constrain the transport trajectories. (c) Scalability and generalization of a single learned policy evaluated with 3–6 agents.

While DGPPPO and the $\beta = 0.5$ penalty variant are similar in aggregate, under a stricter 0.1 m goal threshold DGPPPO outperforms at every team size, with an average hit rate of 84% vs. 59% for the penalty variant. This justifies DGPPPO as the policy optimization backbone, while the proposed framework remains modular and could incorporate other safe RL methods.

5) Sensing and Communication Robustness

We evaluate the test-time robustness of the trained DGPPPO policy on teams of $n = 5$ agents under per-agent message dropout and additional Gaussian LiDAR noise, with 100 episodes per condition. The nominal safety rate is 76.2%, which decreases to 72.4% under 2% dropout and 71.4% under $\sigma = 0.03$ m LiDAR noise, and further drops to 61.4% under 5% dropout and 52.6% under $\sigma = 0.045$ m LiDAR noise.

B. Hardware Experiments

We deploy the learned policies on a real-world Crazyflie 2.1 quadrotor platform equipped with the thrust-upgrade bundle to provide sufficient thrust margin for stable hovering and payload transport. In all experiments, the drones are uniformly

distributed around the payload with an inter-agent spacing of 0.20 m. The payloads are either 3D-printed or laser-cut rigid polygons. To localize the payload, an additional Crazyflie unit without motors is mounted at the payload’s center of mass. All experiments use cables of length 0.50 m, and the payload mass is approximately 27 g.

During hardware experiments, a centralized computer is used for state estimation and logging, while decentralized execution is emulated by providing local observation to each agent’s policy. The policy outputs planar accelerations in the x and y directions at a rate of 10 Hz. These accelerations are integrated to generate the next desired planar waypoint. A low-level onboard PID controller then tracks this waypoint. The commanded accelerations are scaled by a factor of 15. This scaling was selected empirically and was found to yield stable closed-loop behavior under the hierarchical control architecture. The desired altitude (z -axis) is held constant throughout the experiment. Across all hardware trials, the mean position tracking error remained below 5.8 cm, with most experiments exhibiting errors between 3–5 cm.

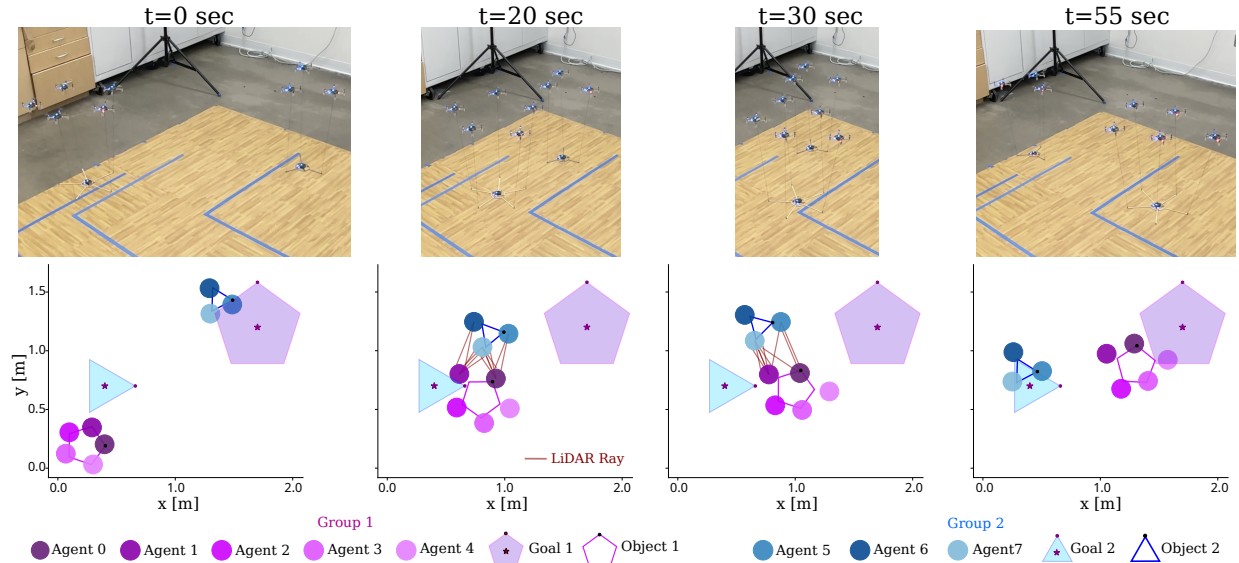


Fig. 4: **Multi-group multi-drone transportation experiment.** Time-lapse snapshots and recorded trajectories show two independent teams transporting separate payloads while perceiving the other team as dynamic obstacles via LiDAR.

To validate policy performance, we design 10 hardware test scenarios, comprising 5 *Easy* scenarios (Fig. 3a) and 5 *Hard* scenarios (Fig. 3b). In each scenario, the initial positions of the drones and payload are fixed, while the goal pose and orientation are randomly sampled such that the distance between the initial and goal payload positions is 2.0 m. Circular obstacles are randomly placed within the workspace. In the *Easy* scenarios, no obstacles lie directly between the initial and goal positions, such that obstacle avoidance does not significantly alter the nominal transport trajectory. In contrast, the *Hard* scenarios contain obstacles that obstruct the direct path, requiring nontrivial maneuvering. In all tested scenarios, teams of three drones successfully transport the payload to the desired goal position and orientation. Task completion is declared when the payload is within 0.30 m of the goal position and the orientation error is less than 0.2 rad.

We further evaluate the scalability and generalization of the learned policy to varying team sizes (Fig. 3c). Specifically, we test the first *Easy* scenario and the final *Hard* scenario using teams of 4, 5, and 6 drones. Notably, although domain randomization during training was restricted to team sizes in the range 3–5, the learned policy generalizes reliably to the unseen 6-drone configuration. In the 6-drone *Hard* scenario, the obstacle density is intentionally reduced to accommodate the larger formation footprint. This modified trial tests zero-shot detouring with six agents and is not directly comparable to the smaller-team *Hard* results; the controlled cross-team comparison is provided in simulation (Fig. 2). All scenarios are successfully completed, except for one 6-drone *Hard* scenario, where the payload reaches the target region but plateaus near the goal, resulting in a final position error of 0.5 m and an orientation error of 0.2 rad. This suggests a terminal pose-refinement limitation from reward shaping, not a safety failure.

C. Multi-Group Multi-Drone Transportation

A key advantage of our approach is that it does not assume static obstacles. Since obstacles are represented implicitly

through LiDAR observations, the learned policy can naturally respond to dynamic obstacles without explicit modeling.

We evaluate this capability in a hardware experiment (Fig. 1.c). Without any additional fine-tuning, we conduct multi-group tests involving two teams with different sizes, specifically a 5-agent group and a 3-agent group. As shown in fig. 4, agents treat drones from the other group as dynamic obstacles using LiDAR sensing and successfully avoid inter-group collisions during execution (e.g., at $t = 20$ s and $t = 30$ s). Both groups reliably transport their respective payloads to the target positions and orientations without interference, demonstrating the robustness, scalability, and generality of the proposed distributed policy. For this experiment, the communication radius is increased to 0.30 m to ensure reliable inter-agent coordination.

VII. CONCLUSION

We demonstrated scalable and safety-aware cooperative multi-drone payload transportation through extensive real-world experiments, including teams larger than those encountered during training and dynamic multi-group scenarios in which other drone teams act as moving obstacles. These results establish the practical feasibility of fully distributed, learning-based aerial collaboration in complex physical environments.

The successful zero-shot transfer from simulation to hardware provides empirical support for modeling cable-suspended transport using a minimal planar decision abstraction, and suggests that learned safety certificates with appropriate margins can be preserved during real-world execution.

This work is currently limited to planar payload transport with evenly distributed cable attachment points. Future work will investigate broader attachment configurations, cable properties, and payload variations, as well as extend the framework to fully three-dimensional transportation.

REFERENCES

- [1] C. R. Kube and E. Bonabeau, “Cooperative transport by ants and robots,” *Robotics and autonomous systems*, vol. 30, no. 1-2, pp. 85–101, 2000.

- [2] E. Tuci, M. H. Alkilabi, and O. Akanyeti, "Cooperative object transport in multi-robot systems: A review of the state-of-the-art," *Frontiers in Robotics and AI*, vol. 5, p. 59, 2018.
- [3] X. An, C. Wu, Y. Lin, M. Lin, T. Yoshinaga, and Y. Ji, "Multi-robot systems and cooperative object transport: Communications, platforms, and challenges," *IEEE Open Journal of the Computer Society*, vol. 4, pp. 23–36, 2023.
- [4] E. Arbel, L. Buise, C. van Waes, N. Oppenheimer, Y. Lahini, and M. Y. Ben Zion, "A mechanical route for cooperative transport in autonomous robotic swarms," *Nature Communications*, vol. 16, no. 1, p. 7519, 2025.
- [5] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *The International Journal of Robotics Research*, vol. 36, no. 9, pp. 1000–1021, 2017.
- [6] S. K. A. De Sousa, R. C. S. Freire, E. A. N. Carvalho, L. Molina, P. C. Santos, and E. O. Freire, "Two-layers workspace: A new approach to cooperative object transportation with obstacle avoidance for multi-robot system," *IEEE Access*, vol. 10, pp. 6929–6939, 2022.
- [7] S. C. Barakou, C. S. Tzafestas, and K. P. Valavanis, "A review of real-time implementable cooperative aerial manipulation systems," *Drones*, vol. 8, no. 5, p. 196, 2024.
- [8] B. E. Jackson, T. A. Howell, K. Shah, M. Schwager, and Z. Manchester, "Scalable cooperative transport of cable-suspended loads with uavs using distributed trajectory optimization," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3368–3374, 2020.
- [9] G. Li and G. Loianno, "Nonlinear model predictive control for cooperative transportation and manipulation of cable suspended payloads with multiple quadrotors," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 5034–5041.
- [10] H. Lee, H. Kim, and H. J. Kim, "Planning and control for collision-free cooperative aerial transportation," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 1, pp. 189–201, 2016.
- [11] J. Wehbeh, S. Rahman, and I. Sharf, "Distributed model predictive control for uavs collaborative payload transport," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 11 666–11 672.
- [12] M. Gassner, T. Cieslewski, and D. Scaramuzza, "Dynamic collaboration without communication: Vision-based cable-suspended load transport with two quadrotors," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5196–5202.
- [13] M. Sarvaiya, G. Li, and G. Loianno, "Hpa-mpc: Hybrid perception-aware nonlinear model predictive control for quadrotors with suspended loads," *IEEE Robotics and Automation Letters*, vol. 10, no. 1, pp. 358–365, 2025.
- [14] G. Li, X. Liu, and G. Loianno, "Human-aware physical human-robot collaborative transportation and manipulation with multiple aerial robots," *IEEE Transactions on Robotics*, vol. 41, pp. 762–781, 2025.
- [15] H.-N. Nguyen, S. Park, J. Park, and D. Lee, "A novel robotic platform for aerial manipulation using quadrotors as rotating thrust generators," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 353–369, 2018.
- [16] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, 2011.
- [17] M. Bernard, K. Kondak, and G. Hommel, "Load transportation system based on autonomous small size helicopters," *The aeronautical journal*, vol. 114, no. 1153, pp. 191–198, 2010.
- [18] D. Sanalidro, H. J. Savino, M. Tognon, J. Cortés, and A. Franchi, "Full-pose manipulation control of a cable-suspended load with multiple uavs under uncertainties," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2185–2191, 2020.
- [19] S. Sun and A. Franchi, "Nonlinear mpc for full-pose manipulation of a cable-suspended load using multiple uavs," in *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2023, pp. 969–975.
- [20] A. Pallar, G. Li, M. Sarvaiya, and G. Loianno, "Optimal trajectory planning for cooperative manipulation with multiple quadrotors using control barrier functions," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 2808–2814.
- [21] S. Sun, X. Wang, D. Sanalidro, A. Franchi, M. Tognon, and J. Alonso-Mora, "Agile and cooperative aerial manipulation of a cable-suspended load," *Science Robotics*, vol. 10, no. 107, p. eadu8015, 2025.
- [22] R. C. Sundin, P. Roque, and D. V. Dimarogonas, "Decentralized model predictive control for equilibrium-based collaborative uav bar transportation," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4915–4921.
- [23] K. Oishi, Y. Amano, and T. Jimbo, "Cooperative transportation using multiple single-rotor robots and decentralized control for unknown payloads," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2024–2030.
- [24] K. Mohammadi, M. Jafarinasab, S. Sirouspour, and E. Dyer, "Decentralized motion control in a cable-based multi-drone load transport system," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4198–4203.
- [25] G. Li, R. Ge, and G. Loianno, "Cooperative transportation of cable suspended payloads with mavs using monocular vision and inertial sensing," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5316–5323, 2021.
- [26] A. Tagliabue, M. Kamel, S. Verling, R. Siegwart, and J. Nieto, "Collaborative transportation using mavs via passive force control," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 5766–5773.
- [27] T. Lee, "Geometric control of quadrotor uavs transporting a cable-suspended rigid body," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 1, pp. 255–264, 2017.
- [28] G. Wu and K. Sreenath, "Geometric control of multiple quadrotors transporting a rigid-body load," in *53rd IEEE Conference on Decision and Control*. IEEE, 2014, pp. 6141–6148.
- [29] S. Zhang, O. So, M. Black, Z. Serlin, and C. Fan, "Solving multi-agent safe optimal control with distributed epigraph form MARL," in *Proceedings of Robotics: Science and Systems*, 2025.
- [30] A. Faust, I. Palunko, P. Cruz, R. Fierro, and L. Tapia, "Automated aerial suspended cargo delivery through reinforcement learning," *Artificial Intelligence*, vol. 247, pp. 381–398, 2017.
- [31] R. Li, F. Yang, Y. Xu, W. Yuan, and Q. Lu, "Deep reinforcement learning-based swing-free trajectories planning algorithm for uav with a suspended load," in *2022 China Automation Congress (CAC)*. IEEE, 2022, pp. 6149–6154.
- [32] J. Estevez, J. Manuel Lopez-Guede, J. del Valle-Echavarri, and M. Graña, "Reinforcement learning based trajectory planning for multi-uav load transportation," *IEEE Access*, vol. 12, pp. 144 009–144 016, 2024.
- [33] K. Mo, L. Chu, X. Zhang, X. Su, Y. Qian, Y. Ou, and W. Pretorius, "Dral: Deep reinforcement adaptive learning for multi-uavs navigation in unknown indoor environment," in *2024 7th International Conference on Mechatronics and Computer Technology Engineering (MCTE)*, 2024, pp. 859–863.
- [34] S. A. Khursheed, "Cooperative payload transportation by uavs: A model-based deep reinforcement learning (mbdrl) application," Master's thesis, Virginia Tech, 2024.
- [35] S. Li and D. Zanotto, "Multi-uav cooperative transportation using dynamic control allocation and a reinforcement learning compensator," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 85468. American Society of Mechanical Engineers, 2021, p. V009T09A034.
- [36] J. Zeng, A. M. Gimenez, E. Vinitzky, J. Alonso-Mora, and S. Sun, "Decentralized aerial manipulation of a cable-suspended load using multi-agent reinforcement learning," in *Conference on Robot Learning*. PMLR, 2025, pp. 3850–3868.
- [37] S. Zhang, O. So, M. Black, and C. Fan, "Discrete GCBF proximal policy optimization for multi-agent safe optimal control," in *The Thirteenth International Conference on Learning Representations*, 2025.
- [38] S. Zhang, O. So, K. Garg, and C. Fan, "GCBF+: A neural graph control barrier function framework for distributed safe multiagent control," *IEEE Transactions on Robotics*, vol. 41, pp. 1533–1552, 2025.
- [39] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European control conference (ECC)*. Ieee, 2019, pp. 3420–3431.
- [40] A. Alan, A. J. Taylor, C. R. He, A. D. Ames, and G. Orosz, "Control barrier functions and input-to-state safety with application to automated vehicles," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 6, pp. 2744–2759, 2023.
- [41] S. Nayak, K. Choi, W. Ding, S. Dolan, K. Gopalakrishnan, and H. Balakrishnan, "Scalable multi-agent reinforcement learning through intelligent information aggregation," in *International conference on machine learning*. PMLR, 2023, pp. 25 817–25 833.
- [42] S. Gu, J. G. Kuba, M. Wen, R. Chen, Z. Wang, Z. Tian, J. Wang, A. Knoll, and Y. Yang, "Multi-agent constrained policy optimisation," *arXiv preprint arXiv:2110.02793*, 2021.
- [43] S. Gu, J. G. Kuba, Y. Chen, Y. Du, L. Yang, A. Knoll, and Y. Yang, "Safe multi-agent reinforcement learning for multi-robot control," *Artificial Intelligence*, vol. 319, p. 103905, 2023.