2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Tying Knots in the Air: Reducing the Number of Quadrotors for Aerial Knot Formation

Tongshu Wu[1], Edward Lopez[1], Diego S. D'Antonio[2], and David Saldaña[1]

***Abstract*— Knots provide compact, lightweight, and mechanically stable configurations that are invaluable for aerial transportation and construction. However, autonomous knot formation in midair remains an open challenge due to the dexterity and complexity of manipulating flexible cables. In this paper, we present a method for midair knot formation that employs two types of aerial robots: lifting robots, which hold the cable endpoints, and support robots, which stabilize intermediate spans to enable interlacing. Our approach focuses on minimizing the number of support robots required while ensuring that the knot's topology is preserved. Our method proceeds in three stages: (i) encode the knot projection as a grid of directional segments and crossings, (ii) apply our Loop Consistency Filter (LCF) to identify the minimal set of support robots required to preserve topology, and (iii) reconstruct continuous Cartesian trajectories using a cable model governed by a spring–damper force and a straightening force. Our results show a reduction in the required robots to form a knot of at least fifty percent compared to the baseline grid-based method. We demonstrate that our method is effective on actual robots, enabling the formation of knots with multiple quadrotors.**

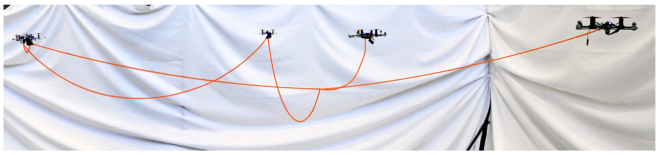
I. Introduction

Ropes and cables are widely used in transportation, manipulation, and securing tasks across domains such as shipping, delivery, construction, and medicine [1]–[3]. Their flexibility, low weight, and adaptability make them well-suited for aerial applications, where payload limits and compliance are critical. Instead of relying on rigid mechanisms for grasping [4], aerial robots have the potential to use cables and knots to secure payloads. Knots are particularly attractive since they provide compact, mechanically stable configurations without the need for rigid connectors, enabling numerous applications, including transportation, logistics, and construction.

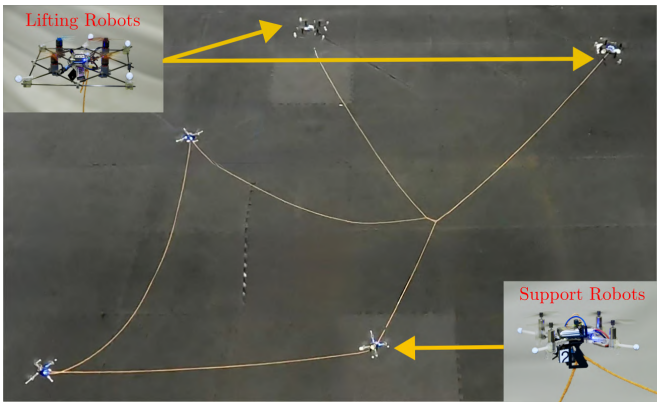
Although folding knots has been studied using arm robots, those approaches require grasping and re-grasping the cable multiple times while making the knot [5]–[10]. Other approaches are focused on topological planning [11], [12] and learning actions from knotting primitives from human demonstrations [13], [14]. However, folding knots using autonomous robots is still an active research field because it involves complex dynamics and high dexterity.

Recent work in aerial robotics has explored cable manipulation for construction [1], shape control [15], and knot making [16], [17]. Notably, multiple quadrotors have been used to collaboratively manipulate flexible payloads for building tensile structures [18], [19]. Complementary to knot folding, Inoue et al. [20] proposed a topology-preserving anchoring approach using multi-flying anchors for autonomous wire attachment in unstructured environments. Cable–cable interaction has also been investigated in [21], [22], where multiple aerial robots interlace their suspended cables in midair to form polygonal hitches for object grasping. Similarly, Cao et al. introduced a braid-theoretic path planning method that guarantees entanglement-free trajectories for teams of tethered aerial robots [23]. In parallel, D'Antonio and Saldaña proposed the multi-catenary framework in [16], which represents knot diagrams as grid-based catenary spans to generate topologically correct knot plans in simulation. However, this approach required a large number of robots (e.g., nine quadrotors for a single overhand knot) and has not yet been validated experimentally.

Despite these advances, tying knots autonomously in midair remains an open challenge. Existing approaches either rely on large robot teams, are limited to anchoring or hitching primitives, or focus on avoiding entanglements



(a) Side view of the robot team forming a knot.



(b) Top view of the robot team with two lifting robots and three support robots.

Fig. 1: Aerial robots forming a 4-crossing knot in midair. **Video:** https://youtu.be/4qkVbkzWQlo

[1]Autonomous and Intelligent Robotics Laboratory (AIRLab), Lehigh University, Bethlehem, PA 18015, USA. {tow225, edl325, saldana}@lehigh.edu

[2]The author is with the Distributed Multi-Agent Robotic Systems (ΔMARS Lab), Department of Electrical and Computer Engineering, Oakland University, Rochester, MI 48309, USA. dantonio@oakland.edu

* The authors gratefully acknowledge the support of the NSF Awards 2322840 and 2442475.

without achieving full knot closure. In this paper, we propose a topology-preserving framework for midair knot formation that reduces the number of required robots (see Fig. 1). Our method proceeds in three stages: *i)* encode the knot projection as a grid of directional segments and crossings, *ii)* extract a minimal set of turning points needed to maintain topology, and *iii)* reconstruct continuous Cartesian trajectories from this reduced representation. The result is a scalable framework for midair knot formation that remains physically feasible and topologically consistent.

The main contribution of this paper is twofold. First, we propose a topology-aware framework for midair knot formation that reduces the number of quadrotors required relative to prior grid-based methods. The framework incorporates a loop filter that converts a node-based knot representation into a reduced execution shape while preserving the underlying knot topology. Second, we present the first experimental demonstration of midair knot formation using physical aerial robots. This work establishes a scalable baseline for aerial transportation. Although we do not yet consider tying knots around objects or transporting payloads, the proposed framework provides a foundation for future studies on aerial manipulation tasks involving securing and transporting objects.

II. Problem Statement

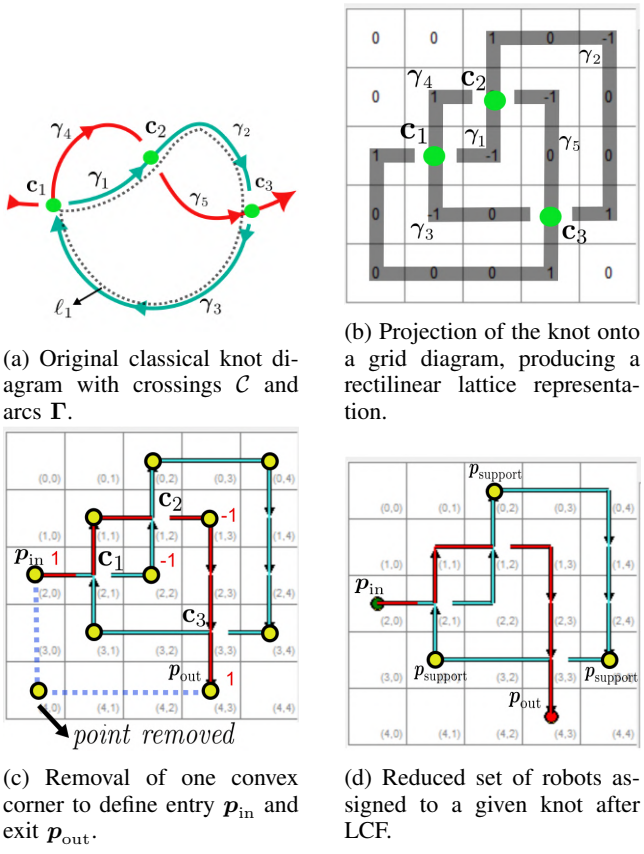
In this work, we address the problem of forming knots in midair using a team of aerial robots. We propose a method that leverages a grid-based knot representation [16] together with topology-preserving selection rules to identify a compact set of support points that preserve the physical realization of the knot while holding it in midair. The grid diagram for knot representation [24] has proven to be an excellent mathematical tool for describing knots in midair; however, current approaches require a high number of robots [16]. The grid size increases with the number of crossings, directly increasing the number of robots required to realize the knot. To address this limitation, we transform the grid representation into a reduced form that captures the requirements for the cable configuration, allowing robots to pass and interlace cables. The resulting structure is then mapped into executable Cartesian trajectories, enabling midair knot formation without requiring regrasp of the cable.

Definition 1 (Knot)**.** *A knot K is a simple closed curve embedded in the three-dimensional Euclidean space $\mathbb{R}^3$ that does not intersect itself [25].*

To make a simple representation of $\mathbb{R}^3$ space knots, we study knots through their planar projections in $\mathbb{R}^2$, which yield diagrams that capture both connectivity and crossing information. Each crossing can be either an over-crossing or an under-crossing, ensuring that the topological type of the knot is preserved. The set of all crossings is

$$\mathcal{C} = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_n\}, \tag{1}$$

where each $\mathbf{c}_i \in \mathbb{R}^2$ is a signed crossing with fixed over/under assignment.



(a) Original classical knot diagram with crossings $\mathcal{C}$ and arcs $\mathbf{\Gamma}$.

(b) Projection of the knot onto a grid diagram, producing a rectilinear lattice representation.

(c) Removal of one convex corner to define entry $\boldsymbol{p}_{\text{in}}$ and exit $\boldsymbol{p}_{\text{out}}$.

(d) Reduced set of robots assigned to a given knot after LCF.

Fig. 2: Pipeline illustrating how a knot diagram is converted into a grid representation with defined entry and exit points, and further reduced to a minimal set of robot-assigned points for execution.

Definition 2 (Grid Representation)**.** *A grid representation of a knot K is a planar, rectilinear diagram drawn on an $N \times N$ square grid, where the knot is represented by a closed path consisting of axis-aligned horizontal and vertical segments.*

A grid representation transforms a smooth knot curve into a piecewise connected loop of vertical and horizontal lines on a rectangular lattice in $\mathbb{R}^2$. As illustrated in Fig. 2(a), the process begins with a classical planar knot, which is then transformed into the grid representation shown in Fig. 2(b). The resulting knot structure is encoded in a matrix, $\mathbf{A} \in \{-1, 0, +1\}^{N \times N}$, where nonzero entries mark turning points. Each row and column contains exactly two nonzero entries, producing a uniquely traversable path of alternating horizontal and vertical segments. To preserve the knot topology during physical execution, we assign orientation values to each segment, a segment from -1 to $+1$ corresponds to an over-crossing, while a segment from $+1$ to -1 corresponds to an under-crossing. These crossings partition the knot into a finite set of m arcs,

$$\mathbf{\Gamma} = \{\gamma_1, \gamma_2, \ldots, \gamma_m\}, \tag{2}$$

where each arc γ_i is an open curve that connects two crossings.

A **loop** in the diagram is defined as a closed path obtained by chaining together a subset of arcs from $\mathbf{\Gamma}$. Traversing these arcs sequentially determines the regions where subsequent arcs must pass, ensuring that the cable can be continuously followed to form the knot. In other words, the loops encode the structural constraints that must be preserved to maintain knot feasibility. We denote the set of loops by

$$\mathcal{L} = \{\ell_1, \ell_2, \ldots, \ell_r\}, \tag{3}$$

where each loop ℓ_i is the union of arcs that connect end-to-end in a cycle, producing a piecewise closed curve in the planar diagram (see Fig. 2(a)). Among these, we are interested in the subset of *active loops*, denoted by $\mathcal{L}_{\text{active}}$, which contain at least one arc through which the remaining cable must pass after the loop is formed.

In this setting, n robots are attached to the cable. Each robot $i = 1, ..., n$ is located at point $\boldsymbol{p}_i \in \mathbb{R}^3$ in the world frame $\{W\}$ and its motion is governed by the firs-order dynamics with velocity as a control input, $\dot{\boldsymbol{p}}_i = \boldsymbol{u}$. A detailed dynamic model for two quadrotors jointly holding a cable was presented in [16].

To make the knot, each robot starts attached to the cable at a location corresponding to a turning point in the grid diagram of the knot, where the cable is initially laid out flat on the ground. We label the robots sequentially, and express their positions as $\boldsymbol{p}_i$, for $i = 1, 2, \ldots, n$. Since the grid diagram is planar, we study their projected positions in two dimensions, treating $\boldsymbol{p}_i \in \mathbb{R}^2$. Let

$$\mathcal{P}_{\text{grid}} = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n\} \tag{4}$$

denote the ordered set of all turning points of the grid diagram corresponding to the target knot. These points represent where the cable changes direction and are assigned to robots.

The robots at $\boldsymbol{p}_1$ and $\boldsymbol{p}_n$ are designated as *lifting robots* (see Fig. 3), responsible for initiating and completing the folding process. The intermediate points

$$\mathcal{P}_{\text{mid}} = \{\boldsymbol{p}_2, \ldots, \boldsymbol{p}_{n-1}\} \subset \mathcal{P}_{\text{grid}} \tag{5}$$

represent the set of locations eligible to be selected as *support robots*. From this candidate set, we define the actual set of support robots as

$$\mathcal{P}_{\text{support}} \subset \mathcal{P}_{\text{mid}}. \tag{6}$$

The main problem that we address in this paper is the following.

Problem 1 (Forming a Knot in Midair)**.** *Design a strategy for a team of aerial robots to form a topologically valid knot in midair using a reduced number of robots.*

To solve this problem, we focus on solving the following subproblems.

Subproblem 1 (Robot Minimization)**.** *Select a minimal subset $\mathcal{P}_{support} \subset \mathcal{P}_{mid}$ to assign support robots, and remove unnecessary robots, preserving the physical knot practicability.*

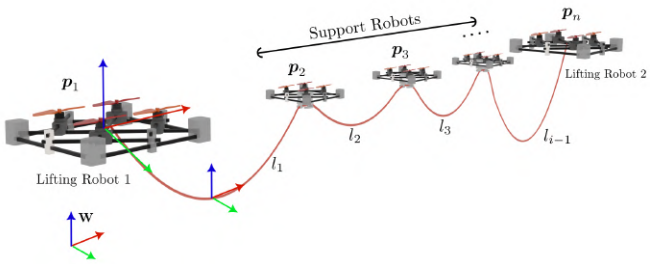


Fig. 3: Illustration of lifting and support robot. The team of aerial robots consists of a lifting robot and support robots attached to a cable to fold it into a target knot shape.

Subproblem 2 (Trajectory Planning)**.** *Given selected robot positions, compute continuous trajectories that fold the cable into the desired knot around the object.*

III. Making a Knot

Forming knots using aerial robots requires that certain openings remain accessible to allow cables to be threaded through. When humans tie knots, these openings are stabilized by the fingers, which prevent surrounding strands from collapsing and obstructing the passage. Analogously, in robotic systems, support robots maintain the geometry of critical cable configurations, thereby enabling pulling robots to complete the threading process. We begin by identifying the cable configurations, which we characterize as critical loops.

A. Loop Consistency Filter

The *Loop Consistency Filter* (LCF) allocates support robots by exploiting the structure of the grid representation. LCF identifies critical loops in the knot as enclosed regions, analyzes their interiors, and assigns support robots only where necessary, thereby reducing redundancy while preserving topology. The procedure consists of four steps.

1) Region Detection: The input to region detection is the ordered sequence of turning points $\mathcal{P}_{\text{grid}} = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_n\}$ traversed by the cable path.

Two types of enclosed regions are detected:

(i) Loops. If a turning point $\boldsymbol{p}_i$ is revisited at index $j > i$, then the intervening subsequence of segments

$$\ell = \gamma_{i+1} \cup \gamma_{i+2} \cup \cdots \cup \gamma_j, \tag{7}$$

forms a closed cycle, recorded as a loop. This method captures nested and overlapping loops naturally, as each repeated point generates a distinct region. As illustrated in Fig. 4(b), a threaded loop region ℓ_2 requires a support robot when an additional arc passes through its interior.

(ii) Pockets. A pocket arises when two segments γ_p and γ_q connect two crossings and enclose a finite region of the grid:

$$\ell = \gamma_p \cup \gamma_q. \tag{8}$$

Although not a strict cycle in the combinatorial sense, such a bounded region plays the same structural role as a loop

in maintaining cable geometry. As shown in Fig. 4(c), a threaded pocket region ℓ_3 also requires a support robot to preserve the opening.

In summary, both loops and pockets are treated uniformly as regions $\ell \in \mathcal{L}$. For each detected region ℓ, we define

$$\mathcal{I}(\ell) \subset \mathbb{Z}^2 \tag{9}$$

as the set of discrete grid cells enclosed by the boundary of ℓ.

2) Enclosure Analysis and Classification: For each region $\ell \in \mathcal{L}$, representing either a loop or a pocket, we analyze whether its interior encloses any future turning points in the cable path. The interior of region ℓ, denoted by $\mathcal{I}(\ell)$, is computed using a flood-fill algorithm over the discrete grid. Starting from an exterior cell, the algorithm marks all reachable cells, terminating at the boundary of ℓ. The unmarked cells define the enclosed area.

To determine whether a region plays a structural role in the knot, we examine whether any turning point $\boldsymbol{p}_j \in \mathcal{P}_{\text{grid}}$, with index $j > i$, lies within $\mathcal{I}(\ell)$, where $\boldsymbol{p}_i$ is the earliest point associated with the formation of region ℓ.

The presence of a turning point within the interior of ℓ indicates that the cable both enters and exits the region again in the future. Since a turning point corresponds to a change in direction, its presence inside the region implies that the cable forms a local bend within $\mathcal{I}(\ell)$. This necessarily creates an incoming and outgoing arc through the region boundary, confirming that the region is topologically threaded by a future segment of the cable.

The region ℓ is then classified as follows:

- **Active:** If there exists a turning point $\boldsymbol{p}_j$, with $j > i$, such that $\boldsymbol{p}_j \in \mathcal{I}(\ell)$, then ℓ is labeled *active*. Active regions must be kept open and require support robots to preserve their geometry. We record such regions in the active set $\mathcal{L}_{\text{active}}$.
- **Passive:** If no such future turning point lies within $\mathcal{I}(\ell)$, then ℓ is labeled *passive*. Passive regions are not threaded and do not require support.

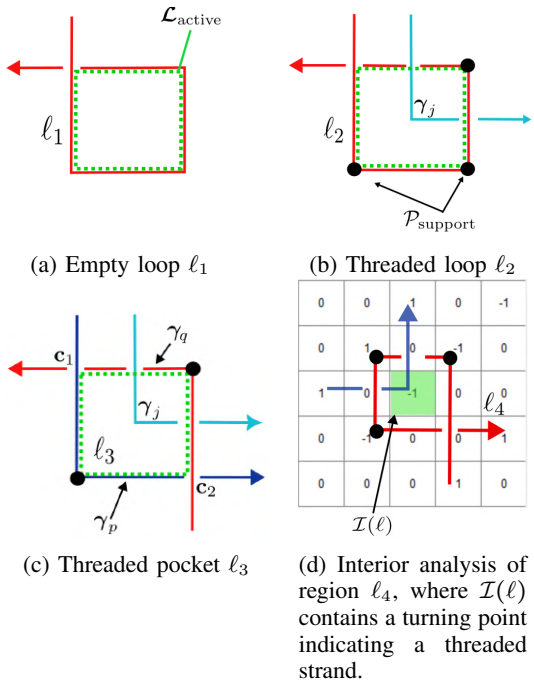
To prevent redundant assignment of support robots, we implement a layered support model. Each discrete grid cell maintains a counter indicating the number of overlapping active regions that enclose it. This hierarchical representation ensures that support is allocated only where necessary, while preserving the topological correctness of the knot.

3) Support Selection: For each active region $\ell \in \mathcal{L}_{\text{active}}$, the algorithm identifies key boundary turning points where supports may be placed. Let

$$B(\ell) = \left\{ \boldsymbol{p}_i \in \partial\ell \;\middle|\; \boldsymbol{p}_i \in \mathcal{P}_{\text{grid}} \right\}, \tag{10}$$

where $\partial\ell$ denotes the boundary of region ℓ. Thus, $B(\ell)$ is the set of turning points that lie directly on the enclosing loop or pocket.

From $B(\ell)$, four *extremal points* are selected: those with the maximum and minimum x-coordinates and those with the maximum and minimum y-coordinates. These points span the region and offer the most geometric clearance for maintaining its open shape.



(a) Empty loop ℓ_1 (b) Threaded loop ℓ_2

(c) Threaded pocket ℓ_3 (d) Interior analysis of region ℓ_4, where $\mathcal{I}(\ell)$ contains a turning point indicating a threaded strand.

Fig. 4: Enclosed regions in grid-based knot diagrams. (a) Empty region ℓ_1 requires no support. (b) Threaded loop ℓ_2 with arc γ_j requires support. (c) Threaded pocket ℓ_3, bounded by crossings $\mathbf{c}_1, \mathbf{c}_2$ and arcs γ_p, γ_q, also requires support. (d) Interior analysis of region ℓ_4: the enclosed cell set $\mathcal{I}(\ell)$ contains a turning point, confirming that the region is threaded and requires support.

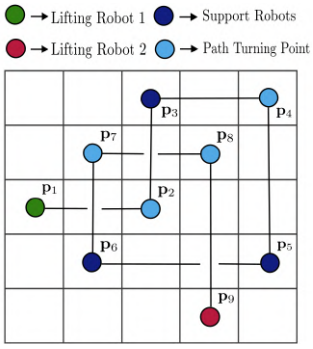
A post-processing step removes redundancy: if two adjacent boundary segments contain no interior strands, the intermediate extremal point is discarded. In such cases, the cable can be supported directly by the two outer points, and additional support is unnecessary.

To compute the final set of support points, the algorithm aggregates all selected boundary points from active regions and merges duplicates across overlapping regions. This ensures that every structurally active region (Fig. 4(b)–(c)) is adequately supported. By contrast, passive regions (Fig. 4(a)) remain stable without additional support.

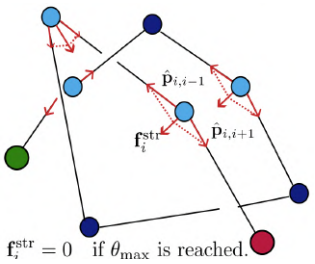
The output of the Loop Consistency Filter is a compact set of support points sufficient to preserve the knot's topology (Fig. 5(a)). With fewer supports, the cable can deviate from the original grid-aligned path, forming geometric shortcuts between fixed points. This motivates the need to recompute a continuous trajectory that maintains topological correctness despite the reduced configuration.

B. From Grid Representation to Robot Trajectories

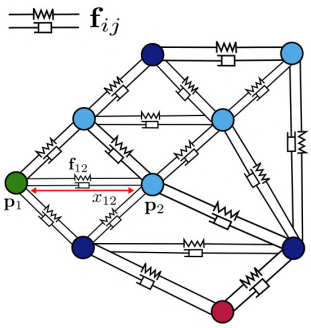
A cable suspended between stationary robots naturally forms catenary curves that, when viewed from above, appear as straight segments. If intermediate robots are removed, these apparent segments collapse into a single straight line,



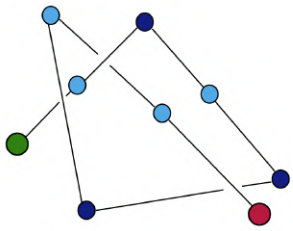
(a) LCF output with assigned support robots



(b) Straightening force apply to each turning point



(c) Example of single point Spring–damper model



(d) Final robot-executed path

Fig. 5: Cable adjustment pipeline after robot reduction. (a) Output of LCF with assigned support robots. (b) Spring–damper simulation. (c) Path refinement via straightening force. (d) Final smoothed trajectory used by the robot team.

effectively replacing the grid-aligned path with shortcuts that alter the overall topology. To address this phenomenon, we model the LCF robot formation as a dynamical system governed by two forces: a spring–damper force that preserves inter-robot spacing, and a straightening force that enforces segment alignment.

1) Straightening Force: Suspended cables naturally relax into straight configurations between fixed endpoints, a behavior extensively studied in the context of elastic rods [26]. In our approach, we reposition the free nodes, also referred to as unassigned points, while treating the positions of the support robots as fixed anchors. This results in a straightening effect applied exclusively at unassigned turning points located between two consecutive robot-assigned anchors. The anchors themselves remain fixed and are not influenced by the straightening force.

Let $\hat{\mathbf{p}}_{j,k}$ be the unit vector that points from point $\mathbf{p}_j$ to point $\mathbf{p}_k$, i.e., $\hat{\mathbf{p}}_{j,k} = \dfrac{\mathbf{p}_k - \mathbf{p}_j}{\|\mathbf{p}_k - \mathbf{p}_j\|}$. For an unassigned turning point i, the indices $i-1$ and $i+1$ denote the positions of its immediate neighbors along the cable path (as shown in Fig. 5(b)). The applied force is

$$\mathbf{f}_i^{\text{str}} = \alpha \frac{\hat{\mathbf{p}}_{i,i-1} + \hat{\mathbf{p}}_{i,i+1}}{\|\hat{\mathbf{p}}_{i,i-1} + \hat{\mathbf{p}}_{i,i+1}\|}, \qquad \mathbf{f}_i^{\text{str}} = \mathbf{0} \text{ for } i \in \mathcal{P}_{\text{robot}}. \tag{11}$$

where the parameter $\alpha > 0$ is a constant to adjust the intensity of the force.

The force $\mathbf{f}_i^{\text{str}}$ points inwards when the three points $(\mathbf{p}_{i-1}, \mathbf{p}_i, \mathbf{p}_{i+1})$ form a bend, and is equal to zero when they are collinear and due to the normalization, the magnitude of the force remains equal to α. We use this force to reflect the relaxation tendency of suspended cables by driving each unassigned turning point toward the straight span between its two neighbors.

2) Spring-Damper Model: While the straightening force stabilizes individual spans, a global mechanism is still needed to preserve knot topology and maintain formation cohesion. We therefore adopt a spring-damper model from multi-robot formation [27], [28]. Each point along the cable is treated as a mass node. Virtual springs with linear damping connect pairs of nodes to regulate their relative distances and dissipate oscillations. The coupling is applied globally (Fig. 5(c)): every point interacts with every other point through spring damper connections. This dense network stabilizes the configuration and preserves the intended shape during motion.

Let $\mathbf{p}_i \in \mathbb{R}^2$ denote the position of point i, and let $\dot{\mathbf{p}}_i$ denotes its velocity. For each pair (i, j), a virtual spring is defined with rest length x_{ij}^0, given by their initial separation. The force on i from j is

$$\mathbf{f}_{ij} = k(x_{ij} - x_{ij}^0)\hat{\mathbf{d}}_{ij} + c\left((\dot{\mathbf{p}}_j - \dot{\mathbf{p}}_i) \cdot \hat{\mathbf{d}}_{ij}\right)\hat{\mathbf{d}}_{ij}, \tag{12}$$

where $\mathbf{d}_{ij} = \mathbf{p}_j - \mathbf{p}_i$ is the displacement vector, $x_{ij} = \|\mathbf{d}_{ij}\|$ is the Euclidean distance, and $\hat{\mathbf{d}}_{ij}$ is the corresponding unit direction vector. Parameters k and c denote spring stiffness and damping, respectively.

This formulation constrains motion along interpoint directions and accounts for both displacement and relative velocity. It prevents collisions, stabilizes the global formation, and ensures that the reduced robot set preserves the knot's topological structure during robot motion.

To compute the adjusted node positions, we model each non-anchor point as a virtual point mass with constant mass $m > 0$, while the lifting and support robot locations remain fixed. For each free node i, the resulting dynamics are

$$m\ddot{\mathbf{p}}_i = \mathbf{f}_i^{\text{str}} + \sum_{j \neq i} \mathbf{f}_{ij}, \tag{13}$$

where $\mathbf{f}_i^{\text{str}}$ denotes the straightening force defined in (11), and $\mathbf{f}_{ij}$ is the spring-damper interaction force defined in (12).

The process is terminated when both the maximum node displacement and the maximum node velocity fall below prescribed thresholds, yielding a static equilibrium configuration that is used as the final smoothed trajectory.

3) Resulting Path: The straightening force pulls unassigned turning points toward the segment defined by the nearest robot-assigned anchors, removing unnecessary bends introduced by grid discretization. In parallel, the spring–damper model maintains spacing between points and dissipates oscillations, preventing collapse or distortion of the

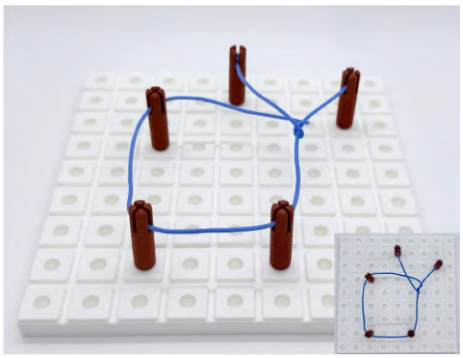


Fig. 6: Physical testbed for validating reduced-robot knotting. A 3D-printed grid represents the workspace, with removable red posts marking robot positions and serving as support points for the cable.

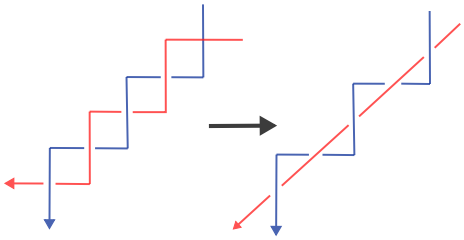


Fig. 7: Staircase braid pattern. Left: baseline assignment where each corner requires a robot. Right: reduced configuration where one robot circulates along a straight cable, greatly reducing the robot count.

overall structure. Together, these forces relax the reduced configuration into a stable equilibrium that preserves knot topology while producing a smooth, physically plausible cable shape. The resulting equilibrium, shown in Fig. 5(d), defines the final trajectory for the aerial robot team.

To execute the trajectory, the robots begin in a single-line formation at the entry waypoint $\mathbf{p}_{\text{in}}$, with the leader lifting robot in front, the support robots in the middle, and the final lifting robot at the rear. They then track the smoothed path toward their assigned anchor positions. The motion is sequential: each robot follows the one ahead at a fixed spacing until reaching its designated position, where it hovers as a support point that maintains the intended knot topology. At each crossing, the robots follow a smooth arc to adjust their height, ensuring a clear, continuous path during the over/under motion.

IV. Experiments and Validation

A. Theoretical Comparison with Baseline

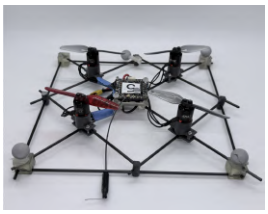
We evaluate our framework by measuring the reduction in robot count across knots of varying complexity[1]. Unlike fixed deployment strategies, our method adapts the allocation of robots to the loop and pocket structure of each knot.

Table I summarizes the results. For each knot, we show the original grid diagram, the reduced configuration, the resulting robot path, and the physical implementation. The table also reports the number of crossings and the robot counts before and after reduction. Across all cases, our approach substantially decreases the required number of robots while preserving both geometry and topology.

Feasibility was verified using a 3D-printed grid testbed (see Fig. 6, implementation column of Table I). Removable red posts mark anchor positions and act as support and lifting points, replicating how aerial robots would hold the cable. By recreating the resulting trajectories for all knots in Table I, we confirmed that our framework not only reduces the number of required robots but also preserves the intended knot topology and ensures practical realizability.

A key insight is the identification of the *staircase braid pattern*, observed in the 5, 7 (Var. 1), and 9 crossing knots. In the baseline, each corner is assigned a robot, whereas our method treats the entire structure as a linear segment stabilized by a single circulating robot (Fig. 7). This illustrates how our framework places robots only where structural support is essential rather than at every corner or crossing. Knots with the same number of crossings can still differ greatly in loop and threading patterns, as seen in the two 7-crossing variants of Table I. In such cases, simple spans receive minimal support, while nested or threaded regions are reinforced.

[1]https://github.com/swarmslab/Midair-Knot-Making



(a) Lifting robot: a customized platform built on Crazyflie Bolt, designed to grasp the cable endpoints and maintain global tension.

(b) Support robot: Crazyflie 2.1 nano quadrotor, used as an intermediate robot to regulate cable spans.

Fig. 8: Robotic platforms used in experiments. (a) The lifting robot enforces global tension by holding the cable endpoints. (b) Support robots shape the internal spans and maintain the knot structure.

B. Midair Knot Formation with Aerial Robots

To validate our approach, we implement a physical testbed equipped with two types of aerial robots (see Fig. 8). A customized Crazyflie Bolt serves as the *lifting robot*, holding the cable endpoints and maintaining global tension. Crazyflie 2.1 nano quadrotors act as *support robots*, positioned along the cable to regulate internal spans. Robot motion is tracked in real time with an OptiTrack system, integrated with ROS for closed-loop control.

The workflow consists of offline trajectory generation followed by online waypoint tracking. First, LCF selects the robot support locations, and the reduced knot configuration is relaxed using the straightening and spring–damper model

TABLE I: Comparison of knot diagrams before and after robot reduction, with corresponding paths and physical implementations for varying numbers of crossings.

# Crossings	Robots Before Reduction	Robots After Reduction	Path	Implementation
3	10	5		
4	12	5		
5	14	5		
7 (Var. 1)	18	7		
7 (Var. 2)	18	8		
9	22	6		

to obtain a smooth equilibrium cable shape. This spring–damper model is used only as a virtual geometric planner and is not used as the online formation controller. The resulting equilibrium is then sampled into Cartesian waypoints and assigned to the lifting and support robots. During execution, each robot tracks its own waypoint sequence using motion-capture feedback and the onboard low-level controller, while the support robots hover at their assigned positions to preserve the knot openings and the lifting robots pull the cable endpoints through the required threaded regions.

We conduct experiments on knots with 3, 4, and 5 crossings (variant 1) from Table I. The execution of the 5-crossing knot is shown in Fig. 9, where sequential snapshots capture the cable being folded into the final configuration. In addition, Fig. 10 shows the completed 3- and 4-crossing knots, which were successfully tied and stabilized in midair.

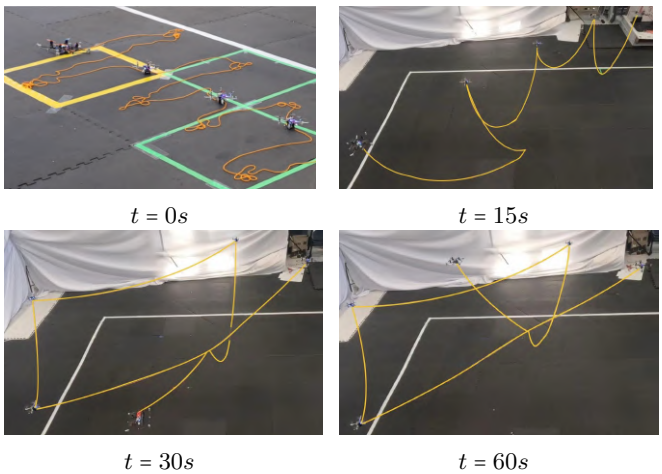
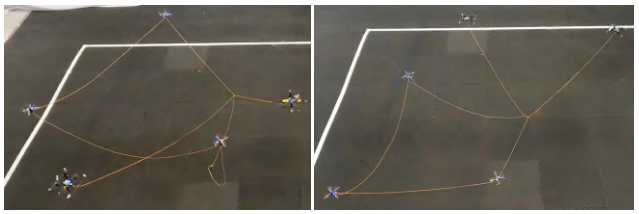


Fig. 9: Snapshots of the aerial robot team folding a 5-crossing knot over time. The frames show the evolution of the cable configuration at different stages of execution.

V. Conclusion and Future Work

This paper presented a method for midair knot formation that reduces the number of aerial robots required compared



3-crossing knot 4-crossing knot

Fig. 10: Completed 3 and 4 crossing knots suspended and stabilized mid-air after full execution by the aerial robot team.

with prior grid-based methods. In contrast to approaches that assign a robot to every turning point, the proposed Loop Consistency Filter (LCF) allocates only the supports needed to preserve topology, while a cable model generates executable trajectories for the reduced team. Across the knots considered in this work, our approach achieved robot-count reductions of more than 50% while preserving the intended knot structure, and experiments with physical aerial robots demonstrated the feasibility of the method in practice.

The experiments also revealed important hardware challenges, including rope vibrations caused by rotor downwash, limited spacing between neighboring robots, and sensitivity of the cable during threading. In our setup, these effects were mitigated through careful robot spacing, lightweight cable selection, and conservative flight motions, but they remain important practical constraints as knot complexity increases. In addition, the current framework relies on a simplified cable model and has not yet been validated on denser knot patterns or configurations with multiple lifting UAVs. Future work will therefore focus on richer cable dynamics, more complex knot families, and coordinated multi-lifting-robot knotting around objects and payloads.

REFERENCES

[1] J. Estevez, G. Garate, J. M. Lopez-Guede, and M. Larrea, "Review of aerial transportation of suspended-cable payloads with quadrotors," *Drones*, vol. 8, no. 2, 2024. [Online]. Available: https://www.mdpi.com/2504-446X/8/2/35

[2] A. Jarašūnienė, M. Išoraitė, and A. Petraška, "Developing small-cargo flows in cities using unmanned aerial vehicles," *Future Transportation*, vol. 4, no. 2, pp. 450–474, 2024. [Online]. Available: https://www.mdpi.com/2673-7590/4/2/22

[3] B. Alkouz, B. Shahzaad, and A. Bouguettaya, "Service-based drone delivery," *2021 IEEE 7th International Conference on Collaboration and Internet Computing (CIC)*, pp. 68–76, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:245650380

[4] A. Ollero, M. Tognon, A. Suarez, D. Lee, and A. Franchi, "Past, present, and future of aerial robotic manipulators," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 626–645, 2022.

[5] Y. Yamakawa, A. Namiki, and M. Ishikawa, "Motion planning for dynamic knotting of a flexible rope with a high-speed robot arm," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 49–54.

[6] Z. Ma and J. Xiao, "Robotic perception-motion synergy for novel rope wrapping tasks," *IEEE Robotics and Automation Letters*, vol. 8, no. 7, pp. 4131–4138, 2023.

[7] K. Suzuki, M. Kanamura, Y. Suga, H. Mori, and T. Ogata, "In-air knotting of rope using dual-arm robot based on deep learning," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6724–6731.

[8] H. Wakamatsu, A. Tsumaya, E. Arai, and S. Hirai, "Planning of one-handed knotting/raveling manipulation of linear objects," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 2, 2004, pp. 1719–1725 Vol.2.

[9] W. Wang and D. Balkcom, "Knot grasping, folding, and re-grasping," *The International Journal of Robotics Research*, vol. 37, no. 2-3, pp. 378–399, 2018. [Online]. Available: https://doi.org/10.1177/0278364918754676

[10] M. P. Bell, W. Wang, J. Kunzika, and D. Balkcom, "Knot-tying with four-piece fixtures," *The International Journal of Robotics Research*, vol. 33, no. 11, pp. 1481–1489, 2014. [Online]. Available: https://doi.org/10.1177/0278364914532217

[11] S. Bhattacharya and R. Ghrist, "Path homotopy invariants and their application to optimal trajectory planning," *Annals of Mathematics and Artificial Intelligence*, vol. 84, no. 3-4, p. 139–160, 2018.

[12] M. Saha and P. Isto, "Motion planning for robotic manipulation of deformable linear objects," in *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, 2006, pp. 2478–2484.

[13] M. Yan, G. Li, Y. Zhu, and J. Bohg, "Learning topological motion primitives for knot planning," 2020.

[14] T. V. Vinh, T. Tomizawa, S. Kudoh, and T. Suehiro, "A new strategy for making a knot with a general-purpose arm," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 2217–2222.

[15] L. Smolentsev, A. Krupa, and F. Chaumette, "Shape visual servoing of a cable suspended between two drones," *IEEE Robotics and Automation Letters*, vol. 9, no. 12, pp. 11 473–11 480, 2024.

[16] D. S. D'Antonio and D. Saldaña, "Folding knots using a team of aerial robots," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 3372–3377.

[17] S. Kudoh, T. Gomi, R. Katano, T. Tomizawa, and T. Suehiro, "In-air knotting of rope by a dual-arm multi-finger robot," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 6202–6207.

[18] F. Augugliaro, A. Mirjan, F. Gramazio, M. Kohler, and R. D'Andrea, "Building tensile structures with flying machines," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 3487–3492.

[19] J. Xu, L. Gao, R. Fierro, and D. Saldaña, "Airbender: Adaptive transportation of bendable objects using dual uavs," *IEEE Robotics and Automation Letters*, vol. 10, no. 3, pp. 2790–2797, 2025.

[20] S. Inoue, K. Kawaharazuka, K. Yoneda, S. Yuzaki, Y. Sahara, T. Suzuki, and K. Okada, "An rgb-d camera-based multi-small flying anchors control for wire-driven robots connecting to the environment," 2025. [Online]. Available: https://arxiv.org/abs/2508.02544

[21] D. S. D'Antonio, S. Bhattacharya, and D. Saldaña, "Forming and controlling hitches in midair using aerial robots," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 1270–1276.

[22] D. S. D'Antonio, T. Wu, S. Bhattacharya, and D. Saldaña, "From hitch to lift: Autonomous cable interlacing by multi-uav teams for aerial grasping and transportation," *IEEE Transactions on Robotics*, 2026.

[23] M. Cao, K. Cao, S. Yuan, K. Liu, Y. Wong, and L. Xie, "Path planning for multiple tethered robots using topological braids," *Robotics: Science and Systems XIX*, Jul 2023.

[24] C. Adams, *The Knot Book*. W.H. Freeman, 1994. [Online]. Available: https://books.google.com/books?id=M-B8XedeL9sC

[25] R. C. Read, "The knot book: An elementary introduction to the mathematical theory of knots," 1997.

[26] M. Bergou, M. Wardetzky, S. Robinson, B. Audoly, and E. Grinspun, "Discrete elastic rods," in *ACM SIGGRAPH 2008 Papers*, ser. SIGGRAPH '08. New York, NY, USA: Association for Computing Machinery, 2008. [Online]. Available: https://doi.org/10.1145/1399504.1360662

[27] Q. Chen, Y. Meng, and J. Xing, "Shape control of spacecraft formation using a virtual spring-damper mesh," *Chinese Journal of Aeronautics*, vol. 29, no. 6, pp. 1730–1739, 2016.

[28] E. Z. MacArthur and C. D. Crane, "Compliant formation control of a multi-vehicle system," in *2007 International Symposium on Computational Intelligence in Robotics and Automation*. IEEE, 2007, pp. 479–484.