Time-Aggregated Connectivity Maintenance for Multi-Robot Networks

Hao Liu*, Yupeng Yang†, Yanze Zhang*, Yiwei Lyu‡, and Wenhao Luo*
*Department of Computer Science, University of Illinois Chicago, Chicago, IL, USA
Email: {hliu232,yzhan361,wenhao}@uic.edu
†Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC, USA
Email: yyang52@charlotte.edu
‡Department of Computer Science and Engineering, Texas A&M University, College Station, TX, USA
Email: yiweilyu@tamu.edu

***Abstract*—Connectivity maintenance is critical for networked robot teams to exchange information and coordinate actions, yet enforcing *persistent global connectivity* can be unnecessarily restrictive when communication ranges are limited and teams operate at scale. In practice, robots must be allowed to temporarily disconnect in order to spread out efficiently, and then deliberately reconfigure their motions to achieve local reconnection and maintain *time-aggregated connectivity*, enabling information to flow across the team over a window of time. In this paper, we propose a novel motion coordination framework for robots to maintain time-aggregated connectivity while progressing to their individual goals. Built as a modular layer on top of robots' nominal motion plans, the framework encompasses (a) a time-window aggregated minimum spanning tree (TWA-MST) approach to dynamically decide which robot pairs should reconnect and when, and (b) a novel notion of adaptive prescribed-time control barrier functions (adaptive PT-CBF) that enforce reconnection through motion reconfiguration. This allows robots to follow their nominal plans while minimally adjusting motions to ensure time-aggregated connectivity over a prescribed time horizon. We provide both theoretical analysis and experimental results to demonstrate the effectiveness of the proposed methods. Project website is available at https://wenhaol.github.io/Time_Aggregated_Connectivity.**

I. Introduction

Multi-robot systems are efficient in spatially distributed missions, e.g., search-and-rescue, exploration, and environmental monitoring, where they can share information and make collective decisions within a multi-robot communication network. Ensuring information flow across the robot teams is often addressed by enforcing persistent *global connectivity*, which requires robots to coordinate motions and remain in a single connected component at all time [1]. While effective, this often yields conservative motion behaviors that conflict with spatial coverage and task progress, making them particularly inefficient for dispersing robots in challenging environments.

In practice, many multi-robot missions do not require continuous team-wide communication. Instead, periodic synchronization can be sufficient to maintain coordination and situational awareness as information propagates throughout the whole team within a period of time. This motivates intermittent connectivity [2, 3], where robots are allowed to temporarily disconnect and later reconnect, relaxing communication constraints so robots can focus on task execution and connect only when necessary. These efforts motivate the perspective that information flow across the team may be achieved through connectivity of a time-aggregated graph, i.e., the graph formed by accumulating communication edges between robots over a time horizon, rather than requiring a connected multi-robot graph at every instant. For example, many coordination objectives only require such a *time-aggregated connectivity* over recurring time intervals to achieve global consensus on information and coordinated behaviors [4–6].

To the best of our knowledge, prior work has mainly focused on persistent connectivity maintenance or rigidly defined local intermittent connectivity enforcement, but has rarely considered intermittent local reconnection for maintaining team-wide connectivity from a time-aggregated perspective. This paper considers the following setting: each robot has a high-level planner (assumed a prior) that produces a pre-planned or re-planned nominal trajectory, while a motion coordination layer will be developed to operate as an overlay service that refines motion execution to satisfy communication objectives [7–9]. Two challenges arise under this setting for time-aggregated connectivity maintenance. First, given pre-planned trajectories, intermittent connectivity requires jointly deciding which communication edges to restore, and when to activate them, while minimizing the overall reconnection efforts alongside with their original task. Existing intermittent-connectivity approaches typically rely on fixed periodic schedules [10], predefined intermittent rendezvous [11], or rigid reconnection strategies [2] that are disruptive and could dominate over robots' original plans. Second, once reconnection events are scheduled, robots must be able to reach the required proximity by the intended time. While Control Barrier Function (CBF) based methods have been widely used to enforce persistent connectivity maintenance [12–14] through set invariance, intermittent reconnection requires convergence to connectivity-satisfying set from connectivity-violating configurations by a deadline. Prescribed-time CBFs (PT-CBFs) provide a mechanism to enforce such convergence by an explicit deadline [15].

*This work was supported in part by the U.S. National Science Foundation under Grant CMMI-2528997.

However, in the context of intermittent reconnection for time-aggregated connectivity, it is difficult to *predefine* appropriate deadlines that ensure successful reconnection, as well as determine which robot pairs should reconnect and when, so that robots can remain close to their nominal task-related trajectories while achieving time-aggregated connectivity over recurring time intervals.

In this paper, we propose a time-aggregated connectivity maintenance framework that enables robots to disconnect temporarily while guaranteeing timely reconnection with minimal coordination overhead. The **contributions** are summarized as follows: (a) we introduce a Time-Window Aggregated Minimum Spanning Tree (TWA-MST) on the spatio-temporal multi-robot networks for reconnection scheduling with reduced efforts and time-aggregated connectivity guarantees, (b) we propose the adaptive prescribed-time control barrier function (adaptive PT-CBF), which automatically selects a feasible prescribed time T_{p} online and defines control constraints enforcing prescribed-time reconnection, and (c) we provide theoretical analysis and extensive experiments to validate the proposed approach, demonstrating effective reconnection, efficient task execution, and scalability to large robot teams.

II. Preliminary

We consider a team of N mobile robots operating in a d-dimensional workspace. The dynamics of each robot $i \in \{1, \ldots, N\}$ is denoted by:

$$\dot{\mathbf{x}}_i = f_i(\mathbf{x}_i) + g_i(\mathbf{x}_i)\mathbf{u}_i, \tag{1}$$

where $f_i : \mathbb{R}^d \mapsto \mathbb{R}^d$ and $g_i : \mathbb{R}^d \mapsto \mathbb{R}^{d\times q}$ are locally Lipschitz continuous. $\mathbf{u}_i \in \mathbb{R}^q$ is the control input.

Communication Model: We consider two proximity-based communication models in the robot team: (i) *short-range communication* with radius $R_{\mathrm{c}} \in \mathbb{R}_{>0}$ for intermittent high-fidelity data exchange, and (ii) *long-range communication* with radius $R_{\mathrm{d}} \in \mathbb{R}_{>0}$ ($R_{\mathrm{c}} \ll R_{\mathrm{d}}$) for lightweight, persistent coordination information exchange. This setup is motivated by practical multi-robot scenarios such as underground exploration [16], where robot teams often employ heterogeneous communication devices with different communication capabilities. For example, long-range link can be supported by Sub-GHz radios for lightweight state and control message exchange over extended distance (e.g., up to several miles), whereas short-range link can leverage Wi-Fi for higher-bandwidth data transmissions such as images, videos, and maps.

Let $d_{i,j}$ denote the Euclidean distance between robots i and j. The corresponding time-varying graphs induced by the robot motions are $\mathcal{G}^{\mathrm{c}}(t) = (\mathcal{V}, \mathcal{E}^{\mathrm{c}}(t))$ (short-range communication graph) and $\mathcal{G}^{\mathrm{d}}(t) = (\mathcal{V}, \mathcal{E}^{\mathrm{d}}(t))$ (long-range coordination graph), where an undirected edge (v_i, v_j) exists in $\mathcal{E}^{\mathrm{c}}(t)$ or $\mathcal{E}^{\mathrm{d}}(t)$ if $d_{i,j} \le R_{\mathrm{c}}$ or $d_{i,j} \le R_{\mathrm{d}}$ respectively. We distinguish the roles of the two graphs: the long-range coordination graph $\mathcal{G}^{\mathrm{d}}(t)$ is required to remain connected at all times to maintain continuous team-wide coordination information exchange, while short-range communication links in $\mathcal{G}^{\mathrm{c}}(t)$ only need to be activated intermittently for time-aggregated connectivity (defined in Sec. II-C) of $\mathcal{G}^{\mathrm{c}}(t)$.

A. Safety Barriers Certificates

Assume K static polyhedral obstacles can be commonly represented by L discretized obstacles modeled as rigid spheres along the boundary of the static obstacles [14, 17]. Each discretized obstacle can be denoted as $o \in \{1, ..., L\}$. Consider the joint robot states $\mathbf{x} = \{\mathbf{x}_1, ..., \mathbf{x}_N\} \in \mathbb{R}^{dN}$, the joint discretized-obstacle states $\mathbf{x}^{\mathrm{obs}} = \{\mathbf{x}_1^{\mathrm{obs}}, \ldots, \mathbf{x}_L^{\mathrm{obs}}\} \in \mathbb{R}^{dL}$, the minimum inter-robot safe distance as $R_{\mathrm{s}} \in \mathbb{R}$, and the minimum obstacle-robot safe distance as $R_{\mathrm{obs}} \in \mathbb{R}$. The desired sets for any pairwise robots i, j and obstacle o satisfying inter-robot or robot-obstacle collision avoidance can be defined as:

$$h_{i,j}^{\mathrm{s}}(\mathbf{x}) = ||\mathbf{x}_i - \mathbf{x}_j||^2 - R_{\mathrm{s}}^2, \forall i > j,$$
$$\mathcal{H}_{i,j}^{\mathrm{s}} = \{\mathbf{x} \in \mathbb{R}^{dN} | h_{i,j}^{\mathrm{s}}(\mathbf{x}) \ge 0\}, \tag{2}$$
$$h_{i,o}^{\mathrm{obs}}(\mathbf{x}, \mathbf{x}^{\mathrm{obs}}) = ||\mathbf{x}_i - \mathbf{x}_o^{\mathrm{obs}}||^2 - R_{\mathrm{obs}}^2, \forall i, o,$$
$$\mathcal{H}_{i,o}^{\mathrm{obs}} = \{\mathbf{x} \in \mathbb{R}^{dN},\ \mathbf{x}^{\mathrm{obs}} \in \mathbb{R}^{dL} | h_{i,o}^{\mathrm{obs}}(\mathbf{x}, \mathbf{x}_o^{\mathrm{obs}}) \ge 0\}. \tag{3}$$

By combining these pair-wise robot-robot safety sets and robot-obstacle safety sets, the overall safety sets can be expressed as:

$$\mathcal{H}^{\mathrm{s}} = \bigcap_{\{\forall i > j\}} \mathcal{H}_{i,j}^{\mathrm{s}},\ \mathcal{H}^{\mathrm{obs}} = \bigcap_{\{\forall i, o\}} \mathcal{H}_{i,o}^{\mathrm{obs}}. \tag{4}$$

To guarantee the safety of the multi-robot system, Control Barrier Function (CBF) [18] have been used to derive the admissible joint control $\mathbf{u} = \{\mathbf{u}_1, \cdots, \mathbf{u}_N\} \in \mathbb{R}^{qN}$ that render the forward invariance of the safe set in Eq. (2) and Eq. (3). The results can be summarized as:

Lemma 1. ***Control barrier function*** *[Summarized from [19]]. Given a team of robots with each robot's dynamics defined by Eq.* (1) *and the safe set $\mathcal{H}_{i,j}^{\mathrm{s}}$ as the 0-super level set of a continuously differentiable function $h_{i,j}^{\mathrm{s}}(\mathbf{x})$ for pair-wise robots i and j, the function $h_{i,j}^{\mathrm{s}}(\mathbf{x})$ is called a control barrier function if there exists an extended class-$\mathcal{K}$ function $\mathcal{K}(\cdot)$ such that $\sup_{\mathbf{u}\in\mathbb{R}^{qN}}\{\dot{h}_{i,j}(\mathbf{x}, \mathbf{u})\} \ge -\alpha(h(\mathbf{x}))$. The admissible control space for any Lipschitz continuous controller $\mathbf{u}$ enforcing the safe set forward invariant thus becomes:*

$$\mathcal{B}_{i,j}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^{qN} : \dot{h}_{i,j}^{\mathrm{s}}(\mathbf{x}, \mathbf{u}) + \mathcal{K}(h_{i,j}^{\mathrm{s}}(\mathbf{x})) \ge 0\}. \tag{5}$$

With this, the admissible control space render the safe set in Eq. (4) *forward invariant* (i.e, keep the system safe) can be summarized[1] as $\mathcal{B}^{\mathrm{s}}(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^{qN} : \dot{h}_{i,j}^{\mathrm{s}}(\mathbf{x}, \mathbf{u}) + \gamma h_{i,j}^{\mathrm{s}}(\mathbf{x}) \ge 0,\ \forall i > j\}$ and $\mathcal{B}^{\mathrm{obs}}(\mathbf{x}, \mathbf{x}^{\mathrm{obs}}) = \{\mathbf{u} \in \mathbb{R}^{qN} : \dot{h}_{i,o}^{\mathrm{obs}}(\mathbf{x}, \mathbf{x}^{\mathrm{obs}}, \mathbf{u}) + \gamma h_{i,o}^{\mathrm{obs}}(\mathbf{x}, \mathbf{x}^{\mathrm{obs}}) \ge 0,\ \forall i, o\}$.

Remark 1. *To enforce safety in environments with obstacles of irregular geometries, one can adopt other CBF variants, e.g., Minkowski-operation CBF [20], by replacing safety functions $h_{i,o}^{\mathrm{obs}}$ in Eq.* (3).

[1] In the rest of this paper, we select $\mathcal{K}(h(\cdot)) = \gamma h(\cdot)$ with γ as a user-defined parameter [19].

B. Global Connectivity of Coordination Graph

To enable reliable coordination, it is desirable to maintain *global connectivity* of the long-range coordination graph $\mathcal{G}^{\mathrm{d}}$.

Global Connectivity: A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is *globally connected* if, for every pair of vertices $v_i, v_j \in \mathcal{V}$, there exists a path over the graph that connects v_i to v_j. With this, we define pairwise *coordination distance condition* as:

$$h^{\mathrm{d}}_{i,j}(\mathbf{x}) = R^2_{\mathrm{d}} - \|\mathbf{x}_i - \mathbf{x}_j\|^2, \qquad \forall (v_i, v_j) \in \mathcal{E}^{\mathrm{d}}, \tag{6}$$
$$\mathcal{H}^{\mathrm{d}}_{i,j} = \left\{ \mathbf{x} \in \mathbb{R}^{dN} : \; h^{\mathrm{d}}_{i,j}(\mathbf{x}) \geq 0 \right\}.$$

Work in [12] introduced a sparse connectivity maintenance algorithm, referred to as Minimum Connectivity Constraint Spanning Tree (MCCST), that dynamically selects an optimal minimum spanning tree as robots move to achieve minimum-effort global connectivity maintenance. Denote the selected edge set by $\mathcal{E}^{\mathrm{d}*}$ and the resulting subgraph of $\mathcal{G}^{\mathrm{d}}$ by $\mathcal{G}^{\mathrm{d}*} = (\mathcal{V}, \mathcal{E}^{\mathrm{d}*})$, i.e., $\mathcal{G}^{\mathrm{d}*} \leftarrow \mathrm{MCCST}(\mathcal{G}^{\mathrm{d}})$. We redefine $\mathcal{G}^{\mathrm{d}*}$ as the *coordination graph* for the rest of the paper.

To maintain $\mathcal{G}^{\mathrm{d}*}$ so that the robot team can coordinate to achieve other objectives, we define the **desired coordination set** as $\mathcal{H}^{\mathrm{d}}(\mathcal{G}^{\mathrm{d}*}) = \bigcap_{(v_i,v_j)\in\mathcal{E}^{\mathrm{d}*}} \mathcal{H}^{\mathrm{d}}_{i,j}$. Based on Lemma 1, the corresponding admissible control set that renders $\mathcal{H}^{\mathrm{d}}(\mathcal{G}^{\mathrm{d}*})$ forward invariant becomes $\mathcal{B}^{\mathrm{d}}(\mathbf{x}, \mathcal{G}^{\mathrm{d}*}) = \left\{ \mathbf{u} \in \mathbb{R}^{qN} : \; \dot{h}^{\mathrm{d}}_{i,j}(\mathbf{x}, \mathbf{u}) + \gamma h^{\mathrm{d}}_{i,j}(\mathbf{x}) \geq 0, \; \forall (v_i, v_j) \in \mathcal{E}^{\mathrm{d}*} \right\}$.

C. Time-Aggregated Connectivity

We define pairwise short-range *communication condition* as:

$$h^{\mathrm{c}}_{i,j}(\mathbf{x}) = R^2_{\mathrm{c}} - \|\mathbf{x}_i - \mathbf{x}_j\|^2, \qquad \forall (v_i, v_j) \in \mathcal{E}^{\mathrm{c}}, \tag{7}$$
$$\mathcal{H}^{\mathrm{c}}_{i,j} = \left\{ \mathbf{x} \in \mathbb{R}^{dN} : \; h^{\mathrm{c}}_{i,j}(\mathbf{x}) \geq 0 \right\}.$$

In the following, we characterize the connectivity of the time-varying short-range communication graph $\mathcal{G}^{\mathrm{c}}(t)$ over a finite time window, referred to as time-aggregated connectivity.

Definition 2. ***Union of graph***. *Given a time-varying communication graph $\mathcal{G}^{\mathrm{c}}(t)$, a time window length $T > 0$ and a starting time $t_0 \geq 0$, we define the* ***union of graph*** *$\mathcal{G}^{\mathrm{u}}(t_0)$ of $\mathcal{G}^{\mathrm{c}}(t)$ over the interval $[t_0, t_0 + T]$ as:*

$$\mathcal{G}^{\mathrm{u}}(t_0) = \left(\mathcal{V}, \textstyle\bigcup_{\tau \in [t_0,\, t_0+T]} \mathcal{E}(\tau) \right) = (\mathcal{V}, \mathcal{E}^{\cup}). \tag{8}$$

An edge (v_i, v_j) exists in $\mathcal{G}^{\mathrm{u}}(t_0)$, if it is connected at least once during the time interval $[t_0, t_0 + T]$.

Definition 3. ***Time-aggregated connectivity*** *. Given the start time t_0 and a time window length T, the communication graph $\mathcal{G}(t)$ is said to satisfy* ***time-aggregated connectivity*** *over $[t_0, t_0 + T]$, if the collection of all edges that appear during $[t_0, t_0 + T]$ contains a path between any two vertices in communication graph $\mathcal{G}^{\mathrm{c}}(t)$ (i.e., the union of graph $\mathcal{G}^{\mathrm{u}}(t_0)$ of $\mathcal{G}^{\mathrm{c}}(t)$ is globally connected.).*

With that, given any prescribed globally connected union of a graph $\bar{\mathcal{G}}^{\mathrm{u}} = (\mathcal{V}, \bar{\mathcal{E}}^{\mathrm{u}})$ containing a predetermined combination of edges, we define the desired set that guarantees satisfaction of *time-aggregated connectivity* as follows:

$$\mathcal{H}^{\mathrm{c}}_T(\bar{\mathcal{G}}^{\mathrm{u}}, t_0) = \Big\{ \mathbf{x} \in \mathbb{R}^{dN} \;\Big|\; \forall (i,j) \in \bar{\mathcal{E}}^{\mathrm{u}}, \; \exists\, \tau \in [t_0, t_0 + T], \;\; \mathbf{x}(\tau) \in \mathcal{H}^{\mathrm{c}}_{i,j} \Big\}.$$

D. Problem Statement

We assume a high-level motion planner has been given that provides nominal task-related trajectories for all robots and the corresponding nominal trajectory tracking controllers, which remain accessible to all robots via the connected long-range coordination graph. Typically, the high-level planners aim to optimize for collision-free trajectories without accounting for more complicated constraints such as connectivity maintenance due to the evolving complexity. In this paper, our objective is to dynamically revise robots' motion from pre-planned or re-planned trajectories *in a post-hoc fashion* with minimum control deviation, and ensure (a) global connectivity of the long-range coordination graph, (b) time-aggregated connectivity of the short-range communication graph through intermittent, sequentially scheduled reconnections among robots, and (c) collision avoidance with other robots and obstacles.

III. Method

A. Time-Window Aggregated Minimum Spanning Tree

In this section, we seek to address a core question: *when* and *which* short-range communication edges should be triggered for reconnection so that the robot teams can achieve time-aggregated connectivity without overly compromising task execution. Rather than enforcing persistent connectivity, our scheduler looks ahead over a finite horizon, leverages *predicted time-varying edge costs*, and *aggregates* them within a rolling time window. Based on these aggregated costs, we then employ a lightweight *heuristic* to choose a cost-effective *combination of edges* for reconnection.

To quantify the reconnection effort, we assign a nonnegative weight to each candidate edge (v_i, v_j). In particular, we use the inter-robot distance as the instantaneous edge weight $w_{ij}(t) \triangleq d_{ij}(t)$, where $d_{ij}(t)$ denotes the Euclidean distance between robots i and j at time t. This choice captures the intuition that establishing (or maintaining) a short-range communication link requires the robots to come within the communication radius, and larger separations typically imply higher motion cost and stronger control intervention.

Given nominal trajectories and the coordination graph through which robots share their current locations and velocities, we roll out the corresponding *nominal (planned)* motions over a finite look-ahead horizon of length T_{w}. Accordingly, at each update time t, we evaluate the instantaneous edge weights $w_{ij}(\tau)$ along these nominal trajectories for all $\tau \in [t,\, t + T_{\mathrm{w}}]$. Since the coordination graph remains connected, the shared state and nominal-plan information can be disseminated team-wide; therefore, we perform reconnection scheduling on a complete *candidate* graph $\mathcal{G}^{\mathrm{s}} \triangleq (\mathcal{V}, \mathcal{E}^{\mathrm{s}})$ with $\mathcal{E}^{\mathrm{s}} \triangleq \{(v_i, v_j) \mid 1 \leq i < j \leq N\}$. Given this planned window, we then define a *horizon-aggregated* weight for each candidate edge based

on $w_{ij}(\tau)$ over $[t,\, t+T_{\mathrm{w}}]$, which serves as the graph weight used for reconnection scheduling, as formalized below:

$$\bar{w}_{ij}(t) \triangleq \min_{\tau\in[t,\, t+T_{\mathrm{w}}]} w_{ij}(\tau) = \min_{\tau\in[t,\, t+T_{\mathrm{w}}]} d_{ij}(\tau). \tag{9}$$

Intuitively, $\bar{w}_{ij}(t)$ captures the *best reconnection opportunity* for robots i and j within the upcoming horizon: it is the smallest inter-robot distance they would attain along the shared nominal plan over $[t,\, t+T_{\mathrm{w}}]$. Hence, a smaller $\bar{w}_{ij}(t)$ indicates that the nominal motion already brings the pair close at some time in the window, making this edge easier to restore, whereas a larger value suggests they remain far apart throughout the horizon. With these horizon-aggregated edge weights, we define the corresponding weighted time-window aggregated graph as $\bar{\mathcal{G}}^{\mathrm{s}}(t) \triangleq \big(\mathcal{V}, \mathcal{E}^{\mathrm{s}}, \bar{W}(t)\big)$, where $\bar{W}(t) = \{\bar{w}_{ij}(t)\}_{(v_i,v_j)\in\mathcal{E}^{\mathrm{s}}}$.

With this, because executing a reconnection action may deviate robots from the current nominal plan, once a reconnection is triggered we re-invoke the upstream planner at the *trigger time* of the selected edge(s). Specifically, for an edge (v_i, v_j) scheduled within the current look-ahead window $[t_k,\, t_k + T_{\mathrm{w}}]$, let $t_{ij}^{\mathrm{tr}} \in [t_k,\, t_k + T_{\mathrm{w}}]$ denote its trigger time. When the trigger occurs (possibly for multiple edges), we replan at $t_{k+1} = t_{ij}^{\mathrm{tr}}$, thereby updating the nominal trajectories and recomputing the aggregated weights on the next look-ahead window $[t_{k+1},\, t_{k+1} + T_{\mathrm{w}}]$.

Consequently, to ensure the *consistency* of the reconnection across replanning iterations, we should enforce that any edge previously selected and triggered must remain in the tree. To this end, we maintain a set $\mathcal{F}(t_k) \subseteq \mathcal{E}^{\mathrm{s}}$ that collects all previously selected edges. At the same time, given the updated horizon-aggregated weights at each replanning instant t_k, we select the remaining edges to minimize the total aggregated distance. This yields a *constrained* minimum spanning tree (MST) problem on $\bar{\mathcal{G}}^{\mathrm{s}}(t_k)$:

$$\mathcal{G}^{s*} = \arg\min_{\mathcal{T}\in\mathbb{T}(\bar{\mathcal{G}}^{\mathrm{s}})} \sum_{(i,j)\in\mathcal{E}(\mathcal{T})} \bar{w}_{ij}(t_k) \quad \text{s.t.} \quad \mathcal{F}(t_k) \subseteq \mathcal{E}(\mathcal{T}), \tag{10}$$

where $\mathbb{T}(\bar{\mathcal{G}}^{\mathrm{s}})$ denotes the set of all spanning trees over $\mathcal{G}^{\mathrm{s}}$.

Finally, since any spanning tree contains exactly $N-1$ edges, establishing team-wide connectivity via sequential reconnections requires committing (and eventually realizing) at least $N-1$ edges in total; thus, in the worst case the system may need to replan at least $N-1$ times to complete all required reconnections. Moreover, because each trigger time satisfies $t_{ij}^{\mathrm{tr}} \in [t_k,\, t_k + T_{\mathrm{w}}]$, the subsequent planning window naturally shifts to $[t_k + t_{ij}^{\mathrm{tr}},\, t_k + t_{ij}^{\mathrm{tr}} + T_{\mathrm{w}}]$ (or equivalently $[t_{k+1},\, t_{k+1} + T_{\mathrm{w}}]$). To transfer the constrained MST in Eq. (10) into a regular MST problem, we propose the following *reweighting* mechanism.

Definition 4. ***Reweighted time-window aggregated graph.*** *Let $\mathcal{V} = \{v_1, \ldots, v_N\}$ be the robot set and let $\mathcal{E}^{\mathrm{s}} = \{(v_i, v_j) \mid 1 \le i < j \le N\}$ denote the complete candidate edge set. At time t, suppose the upstream planner provides nominal trajectories over $[t,\, t+T_{\mathrm{w}}]$, which are updated in a receding-horizon manner and shared among robots via the connected coordination graph. Let $\bar{w}_{ij}(t)$ denote the time-window aggregated weight defined in Eq.* (9)*. With this, we define the effective (reweighted) edge cost:*

$$\tilde{w}_{ij}(t) \triangleq \begin{cases} \bar{w}_{ij}(t)/\lambda, & \sigma_{ij}(t) = 1, \\ \bar{w}_{ij}(t), & \sigma_{ij}(t) = 0, \end{cases} \tag{11}$$

where $\sigma_{ij}(t) \in \{0, 1\}$ indicates whether edge (v_i, v_j) has been triggered, and $\lambda \in \{\lambda : \frac{\bar{w}_{ij}(t)}{\lambda} \ll \bar{w}_{i'j'}(t),\ \forall (i,j) : \sigma_{ij}(t) = 1,\ \forall (i',j') : \sigma_{i'j'}(t) = 0\}$. The induced reweighted time-window aggregated graph is $\tilde{\mathcal{G}}^{\mathrm{s}}(t) \triangleq \big(\mathcal{V}, \mathcal{E}^{\mathrm{s}}, \tilde{W}(t)\big)$, where $\tilde{W}(t) = \{\tilde{w}_{ij}(t)\}_{(v_i,v_j)\in\mathcal{E}^{\mathrm{s}}}$.

Definition 4 constructs a *reweighted* time-window aggregated graph $\tilde{\mathcal{G}}^{\mathrm{s}}(t) = \big(\mathcal{V}, \mathcal{E}^{\mathrm{s}}, \tilde{W}(t)\big)$ that embeds the edge-persistence requirement into an ordinary MST computation. Specifically, once an edge (v_i, v_j) is *triggered* (i.e., $\sigma_{ij}(t) = 1$), its aggregated cost is discounted by a large factor λ as in Eq. (11). The choice of λ guarantees a strict separation of priorities: every triggered edge has an effective cost $\bar{w}_{ij}(t)/\lambda$ that is *much smaller* than the cost of any untriggered edge, regardless of how $\bar{w}_{ij}(t)$ changes under receding-horizon replanning. As a result, the MST computed on $\tilde{\mathcal{G}}^{\mathrm{s}}(t)$ is driven to *respect the persistence constraint* by construction: it must retain all previously triggered edges, and then completes the spanning tree by optimizing over the remaining (untriggered) edges. We next establish this property formally. Lemma 5 proves the *consistency over time* of the resulting tree sequence, i.e., under receding-horizon replanning at update instants $\{\tau_k\}$, it can be guaranteed to preserve all edges triggered prior to (and including) τ_k. Building on this consistency result, Theorem 6 further shows that solving a standard MST on the reweighted graph $\tilde{\mathcal{G}}^{\mathrm{s}}(t)$ is *equivalent* to solving the original constrained MST problem in Eq. (10). Namely, the computed solution is an optimal solution that satisfies the persistence constraint while minimizing total aggregated cost for future.

Lemma 5. ***Consistency over time (persistence of triggered edges).*** *Consider a sequence of replanning instants $\{\tau_k\}_{k\ge 0}$. At each τ_k, the reweighted time-window aggregated graph is formed according to Definition 4. Let $\mathcal{F}(\tau_k) \triangleq \{(v_i, v_j) \in \mathcal{E}^{\mathrm{s}} \mid \sigma_{ij}(\tau_k) = 1\}$ denotes the set of edges that have been triggered up to time τ_k. Then, at every replanning instant, the newly computed minimum spanning tree on the reweighted graph must contain all edges in $\mathcal{F}(\tau_k)$. In other words, once an edge is triggered, it is preserved in all subsequent tree selections despite replanning updates.*

Proof: See the detailed proof in Appendix A. ∎

Theorem 6. ***Time-window aggregated minimum spanning tree (TWA-MST).*** *Given the reweighted time-window aggregated graph $\tilde{\mathcal{G}}^{\mathrm{s}}(t_k) = \big(\mathcal{V}, \mathcal{E}^{\mathrm{s}}, \tilde{W}(t_k)\big)$ as in Definition 4, where $\tilde{W}(t_k) = \{\tilde{w}_{ij}(t_k)\}$ is given by Eq.* (11)*. We denote the minimum spanning tree of $\tilde{\mathcal{G}}^{\mathrm{s}}(t_k)$ as:*

$$\tilde{\mathcal{T}}^*(t_k) = \arg\min_{\mathcal{T}\in\mathbb{T}(\tilde{\mathcal{G}}^{\mathrm{s}})} \sum_{(i,j)\in\mathcal{E}(\mathcal{T})} \tilde{w}_{ij}(t_k). \tag{12}$$

Then, $\tilde{\mathcal{T}}^*(t_k)$ is exactly an optimal solution to the constrained MST problem in Eq. (10) *(i.e., $\tilde{\mathcal{T}}^*(t_k) = \mathcal{G}^{s*}$). With this, We call $\tilde{\mathcal{T}}^*(t_k)$ as Time-Window Aggregated Minimum Spanning Tree (TWA-MST) of the original aggregated graph $\bar{\mathcal{G}}^{\mathrm{s}}(t_k)$.*

Proof: Let $\mathcal{F}(t_k) = \{(i,j) \in \mathcal{E}^{\mathrm{s}} \mid \sigma_{ij}(t_k) = 1\}$ be the set of previously triggered (fixed) edges. By Definition 4, all edges in $\mathcal{F}(t_k)$ are assigned discounted costs $\tilde{w}_{ij}(t_k) = \bar{w}_{ij}(t_k)/\lambda$, while all other edges keep $\tilde{w}_{ij}(t_k) = \bar{w}_{ij}(t_k)$, and the choice of λ guarantees a strict priority separation: every triggered edge has smaller effective cost than any untriggered edge. Therefore, in any MST construction on $\tilde{\mathcal{G}}^{\mathrm{s}}(t_k)$ (e.g., Kruskal's algorithm), all triggered edges are considered before any untriggered edge and, since $\mathcal{F}(t_k)$ is acyclic, they are all included, which partitions the vertices into sub-groups induced by $\mathcal{F}(t_k)$. After these "small" edges are fixed, the remaining task is to connect the sub-groups using only untriggered edges, for which $\tilde{w}_{ij}(t_k) = \bar{w}_{ij}(t_k)$ holds. Hence minimizing the total reweighted cost is equivalent to minimizing the total original aggregated cost over the added (untriggered) edges subject to containing $\mathcal{F}(t_k)$, which is exactly the constrained MST problem in Eq. (10). Thus, the MST $\tilde{\mathcal{T}}^*(t_k)$ of the reweighted graph is an optimal solution to Eq. (10). ∎

With the computed time-window aggregated minimum spanning tree in Theorem 6, we are able to provide a feasible solution to decide how the information is scheduled to propagate through the team of robots. To guarantee a timely reconnection whenever a pairwise communication edge is selected to be restored, we continue to propose a novel reconnection mechanism, adaptive PT-CBF, that forces the selected edge to be connected before a feasible, non-conservative deadline.

B. Adaptive Prescribed-Time CBF

We first review two variants of control barrier function formulations that can drive a system from an unsafe set into a desired safe set: finite-time CBF (FT-CBF) [21] and prescribed-time CBF (PT-CBF) [15]. FT-CBF guarantee reaching the safe set within a finite upper bound determined by the initial condition and design parameters, whereas PT-CBF aim to enforce safety satisfaction within a user-specified horizon. We build on these ideas and develop an adaptive PT-CBF that automatically selects a feasible prescribed time for reconnection. PT-CBF commonly rely on time-varying scaling terms that become unbounded at the prescribed deadline. We first introduce the finite-time control barrier function.

Lemma 7. ***Finite-time control barrier function*** *[Summarized from [21]]. Given a team of robots with each robot's dynamics defined by Eq.* (1) *and a desired set $\mathcal{C} = \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}$ defined by a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$. The function $h(\mathbf{x})$ is called FT-CBF on a set $\mathcal{D}$ if there exist real parameters $\rho \in [0,1)$ and $c > 0$, such that for all $\mathbf{x} \in \mathcal{D}$, and $\dot{h}(\mathbf{x}) + c \cdot \mathrm{sign}(h(\mathbf{x})) \cdot |h(\mathbf{x})|^\rho \geq 0$. Then any locally Lipschitz continuous controller $u(\mathbf{x})$ that satisfies the above inequality renders $\mathcal{C}$ forward invariant when $\mathbf{x}_0 \in \mathcal{C}$, and drives the state into $\mathcal{C}$ in finite time when $\mathbf{x}_0 \in \mathcal{D} \setminus \mathcal{C}$. In particular, the finite-time CBF guarantees that the state reaches $\mathcal{C}$ within $T_{\mathrm{FTCBF}} = \frac{1}{c(1-\rho)} \left|h(\mathbf{x}_0)\right|^{1-\rho}$, and remains in $\mathcal{C}$ thereafter.*

Lemma 7 provides an explicit upper bound on the reaching time. However, in practice, using this method, the progress toward the threshold can become increasingly slow when the state approaches the boundary of the safe set (e.g., near $h(\mathbf{x}) = 0$), which may delay the moment when the system actually crosses the desired communication threshold in reconnection tasks. To explicitly regulate the completion time, PT-CBF introduces a time-varying barrier gain (often constructed from a blow-up function as in Definition 8) so that safety satisfaction is enforced no later than a prescribed deadline.

Definition 8. ***Blow-up function****[Summarized from [15]] Let $[a,b)$ be a finite time interval with $a < b$. A function $\eta(t) : [a,b) \to \mathbb{R}_{\geq 0}$ is called a **blow-up function** if: 1) $\eta(t)$ is continuous on $[a,b)$, 2) $\eta(t)$ is strictly increasing on $[a,b)$, and 3) $\lim_{t \to b^-} \eta(t) = +\infty$.*

Lemma 9. ***Prescribed-time control barrier function*** *[Summarized from [15]]. Given a team of robots with each robot's dynamics defined by Eq.* (1) *and let $h : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function. The objective is to drive $h(\mathbf{x}_0)$ from an unsafe value ($h(\mathbf{x}_0) < 0$) to a safe value ($h(\mathbf{x}) \geq 0$) within a prescribed horizon $[t_0, t_0 + T_{\mathrm{p}})$. The function $h(\mathbf{x})$ is a PT-CBF if there exist $c > 0$ and a blow-up function $\mu(t)$ such that the locally Lipschitz continuous controller $\mathbf{u}(\mathbf{x})$ can enforce $\dot{h}(\mathbf{x}) + c\,\mu(t)\,h(\mathbf{x}) \geq 0, \quad \forall t \in [t_0, t_0 + T_{\mathrm{p}})$. Then the system is guaranteed to enter the safe set $\mathcal{C} = \{\mathbf{x} \mid h(\mathbf{x}) \geq 0\}$ at or before the prescribed time $t_0 + T_{\mathrm{p}}$, even when starting from an unsafe state.*

While PT-CBF provides a direct mechanism to enforce a user-specified completion time, its performance critically depends on the prescribed horizon T_{p}. If T_{p} is chosen too small, the resulting constraint can become overly aggressive and may render the Quadratic programming (QP) infeasible or prevent the robots from entering the communication range. Conversely, choosing a large T_{p} can make the reconnection behavior overly conservative, leading to a delayed reconnection and increased task disruption.

This motivates an adaptive mechanism that selects a feasible and effective prescribed time online. Motivated by the above limitations, we propose an adaptive PT-CBF that retains the prescribed-time reachability property while automatically adjusting T_{p} according to the initial barrier value and design parameters.

Theorem 10. ***Adaptive prescribed-time control barrier function (Adaptive PT-CBF)****. Given a team of robots with each robot's dynamics defined by Eq.* (1)*, let $h : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function defining the desired set $\mathcal{C} := \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}$. Then h is an **adaptive prescribed-time control barrier function** if there exists real parameter $\rho \in (0,1)$ and $c > 0$, so that*

$$\dot{h}(\mathbf{x}) + c\,\mu(t)\,\mathrm{sign}\big(h(\mathbf{x})\big)\,\big|h(\mathbf{x})\big|^{\rho} \;\geq\; 0, \tag{13}$$

holds for all $t \in [0, T_{\rm p})$, where the blow-up function $\mu(t)$ is defined as $\mu(t) = \left(\frac{T_{\rm p}}{T_{\rm p}-t}\right)^m + \alpha, \quad t \in [0, T_{\rm p})$, with $\alpha > 0$ and $m \in 2\mathbb{N}$ (i.e., m is a positive even integer). The prescribed time $T_{\rm p}$ is chosen adaptively according to

$$T_{\rm p} = \frac{|h(\mathbf{x}_0)|^{1-\rho}}{c(1-\rho)\,\mu_0}, \qquad \mu_0 = 1 + \alpha > 1. \tag{14}$$

*With this, the following properties hold for the corresponding closed-loop solution $\mathbf{x}(t)$: 1) (*Forward invariance*) If $\mathbf{x}_0 \in \mathcal{C}$, then $h(\mathbf{x}(t)) \geq 0$ for all $t \geq 0$; that is, the safe set $\mathcal{C}$ is forward invariant. 2) (*Prescribed-time reachability*) If $\mathbf{x}_0 \notin \mathcal{C}$, i.e., $h(\mathbf{x}_0) < 0$, then there exists a time $t^\star \leq T_{\rm p}$ such that $h(\mathbf{x}(t^\star)) = 0$, and $h(\mathbf{x}(t)) \geq 0$ for all $t \geq t^\star$.*

Proof: When $h(\mathbf{x}) < 0$, Eq. (13) with $\text{sign}(h) = -1$ becomes $\dot{h}(\mathbf{x}) \geq c\,\mu(t)\,|h(\mathbf{x})|^\rho$. Define $V(\mathbf{x}) := |h(\mathbf{x})| = -h(\mathbf{x})$ in this region. Then $\dot{V}(\mathbf{x}) = -\dot{h}(\mathbf{x}) \leq -c\,\mu(t)\,V(\mathbf{x})^\rho$. Since $\mu(t) \geq \mu_0$ for all $t \in [0, T_p(\mathbf{x}_0))$, we obtain the conservative inequality $\dot{V}(\mathbf{x}) \leq -c\,\mu_0\,V(\mathbf{x})^\rho$, $0 < \rho < 1$. By separation of variables and integration from $V(\mathbf{x}_0) = |h(\mathbf{x}_0)|$ down to 0, one obtains the convergence bound $T_{\rm p} \leq \frac{|h(\mathbf{x}_0)|^{1-\rho}}{c(1-\rho)\,\mu_0}$, which, by construction, equals $T_{\rm p}$ in Eq. (14). This proves starting outside $\mathcal{C}$, the trajectory reaches $h(\mathbf{x}) = 0$ within the prescribed time $T_{\rm p}(\mathbf{x}_0)$ and remains in $\mathcal{C}$ thereafter. When $h(\mathbf{x}) \geq 0$ we have $\text{sign}(h) = 1$ and Eq. (13) yields $\dot{h}(\mathbf{x}) \geq -c\,\mu(t)\,h(\mathbf{x})^\rho$, which is a standard CBF as stated in Lemma 1. Thus, if $h(\mathbf{x}_0) \geq 0$, the trajectory remains in $\mathcal{C}$ for all $t \geq 0$, establishing set forward invariance property. ■

Based on the reconnection schedule returned by TWA-MST $\tilde{\mathcal{T}}^*(t_k)$ in Theorem 6, we introduce the following *reconnection admissible set* enforced by adaptive PT-CBF, which transfers state constraints in Eq. (9) into following control constraints:

$$\mathcal{B}^{\rm c}(\mathbf{x}, \mathcal{G}^{\rm c*}(t), t) = \Big\{\mathbf{u} \in \mathbb{R}^{qN} \,\Big|\, \dot{h}^{\rm c}_{ij}(\mathbf{x}(t)) + c\,\mu(t)\,\text{sign}\big(h^{\rm c}_{ij}(\mathbf{x}(t))\big)\,|h^{\rm c}_{ij}(\mathbf{x}(t))|^\rho \geq 0, \forall (v_i, v_j) \in \mathcal{E}(\tilde{\mathcal{T}}^*(t_k))\Big\}.$$

With all components established above, we now summarize the overall framework in Algorithm 1. The algorithm describes the proposed control framework, clarifying how replanning, edge selection, and pairwise reconnection actions interact over time to realize the complete reconnection process.

C. Algorithm Design

Algorithm 1 summarizes our receding-horizon reconnection scheme that couples time-window aggregation, TWA-MST scheduling, and adaptive PT-CBF control. There is an upstream MAPF planner that provides nominal trajectories over a look-ahead window $[t, t + T_{\rm w}]$, which are shared through the currently connected coordination layer. Using these trajectories, we compute time-window aggregated edge costs and update the weighted aggregated graph. We then (i) maintain an in-range coordination tree via MCCST under radius $R_{\rm d}$, and (ii) compute a reweighted TWA-MST to obtain a compact reconnection schedule with $|\mathcal{V}|-1$ candidate edges. Importantly, whenever a reconnection edge is triggered and the robots deviate from the nominal plan, we replan nominal

Algorithm 1 Adaptive PT-CBF with TWA-MST Scheduling

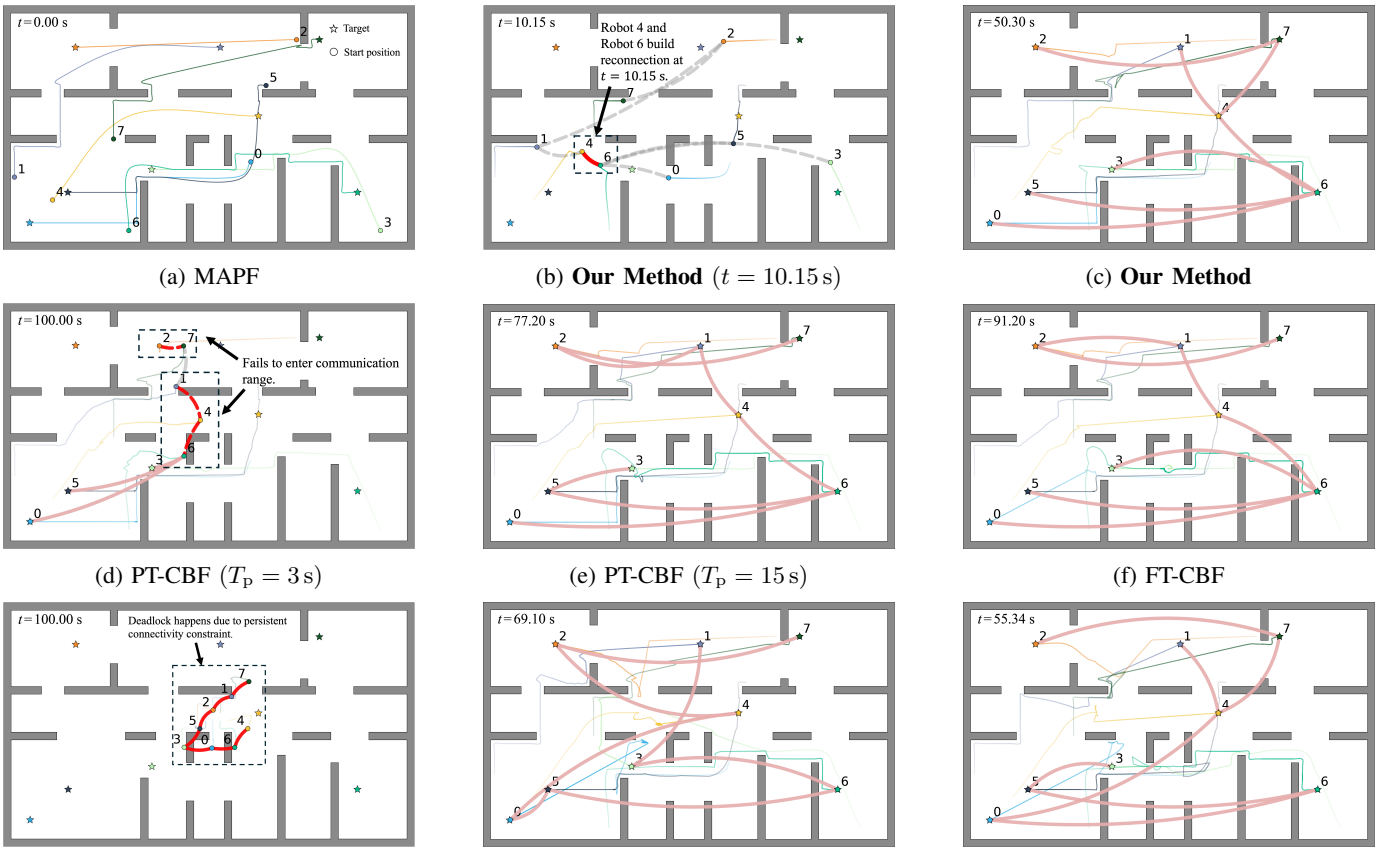
Require: Start-Goal configurations, rolling time window $T_{\rm w}$, communication radius $R_{\rm c}$, safety radius $R_{\rm s}$, coordination radius $R_{\rm d}$, upstream planner (e.g. an MAPF planner [22])
Ensure: Executed trajectory $\mathbf{x}(\cdot)$, applied control $\mathbf{u}^*(\cdot)$
Initialize coordination graph $\mathcal{G}^{\rm d*}(t)$, Activation state $\sigma(0)$
Plan Trajectories for the time window $[0, T_{\rm w}]$, with trajectories and nominal controls: $\Pi = \{\pi_i\}_{i=1}^N$ with $\pi_i^{[0,T_{\rm w}]} = \{\tilde{\mathbf{x}}_i, \mathbf{u}_i^{\rm nom}\}^{[0,T_{\rm w}]}$, $t_k = 0$
Calculate short-range communication graph $\tilde{\mathcal{G}}^{\rm s}(t_k)$
Calculate TWA-MST $\tilde{\mathcal{T}}^*(t_k)$
Reconnection $\leftarrow$ False,
for $t = 0, \ldots, T$ **do**
  if t == $argmin_t \mathcal{E}(\tilde{\mathcal{T}}^*(0))$ **then**:
    selected edge $= v_i, v_j$,
    Reconnection $\leftarrow$ True
    $t_k += 1$
  end if
  if Reconnection **then**:
    Replan: update trajectories and nominal controls with $\Pi = \{\pi_i\}_{i=1}^N$ with $\pi_i^{[t,T_{\rm w}]} = \{\tilde{\mathbf{x}}_i, \mathbf{u}_i^{\rm nom}\}^{[t,t+T_{\rm w}]}$
    Update Short-Range Communication Graph: $\tilde{\mathcal{G}}^{\rm s}(t_k)$
    Update TWA-MST: $\tilde{\mathcal{T}}^*(t_k)$
  end if
  if $dist(\mathbf{x}_i, \mathbf{x}_j) \leq R_{\rm c}$ **then**:
    Reconnection $\leftarrow$ False
  end if
  Control synthesis: $\mathbf{u}^*(t) \leftarrow$ CBF-QP in (15):
  Apply $\mathbf{u}^*(t)$ and update $\mathbf{x}(t)$
  Update $\sigma(t)$ and Log $\{\mathbf{x}(t), \mathbf{u}^*(t), \sigma(t)\}$
end for

trajectories and refresh aggregated weights and TWA-MST accordingly, ensuring the schedule remains consistent with the updated motion plan. With above mechanism, at each time t we compute coordination backbone and reconnection schedule, and then solve standard QP to obtain control input:

$$\begin{aligned} \mathbf{u}^*(t) = \arg\min_{\mathbf{u}} \ & \|\mathbf{u} - \mathbf{u}^{\rm nom}(t)\|^2 \\ \text{s.t.} \ & \mathcal{G}^{\rm d*}(t) \leftarrow \text{MCCST}\big(\mathcal{G}^{\rm d}(t)\big), \\ & \mathcal{G}^{\rm c*}(t) \leftarrow \arg\min_{\mathcal{T} \in \mathbb{T}(\tilde{\mathcal{G}}^{\rm s}(t))} \sum_{(i,j)\in\mathcal{E}(\mathcal{T})} \tilde{w}_{ij}(t), \\ & \mathbf{u} \in \mathcal{B}^{\rm s}(\mathbf{x}) \cap \mathcal{B}^{\rm d}\big(\mathbf{x}, \mathcal{G}^{\rm d*}(t)\big) \cap \mathcal{B}^{\rm c}\big(\mathbf{x}, \mathcal{G}^{\rm c*}(t), t\big), \\ & \|\mathbf{u}_i\| \leq \alpha_i, \quad i = 1, \ldots, N. \end{aligned} \tag{15}$$

Here $\mathbf{u}^{\rm nom}(t) \triangleq [(\mathbf{u}_1^{\rm nom}(t))^\top, \ldots, (\mathbf{u}_N^{\rm nom}(t))^\top]^\top$ is the nominal tracking input provided by upstream planner, and $\alpha_i > 0$ is control bound of robot i. The set $\mathcal{B}^{\rm d}(\cdot)$ enforces coordination-link maintenance for all edges in $\mathcal{G}^{\rm d*}(t)$, whereas $\mathcal{B}^{\rm c}(\cdot)$ enforces reconnection constraints only for currently activated/triggered edge(s) selected from $\mathcal{G}^{\rm c*}(t)$. All constraints are linear in $\mathbf{u}$, hence Eq. (15) can be solved efficiently.

Proposition 11. ***Bounded-time time-aggregated connectivity**. Let $\tilde{\mathcal{T}}^*(t_k)$ be the TWA-MST computed at time t_k (Theorem 6) with edge set $\mathcal{E}_{\rm mst}(t_k) = \mathcal{E}(\tilde{\mathcal{T}}^*(t_k))$. Following Algorithm 1, the edges in $\mathcal{E}_{\rm mst}(t_k)$ are triggered sequentially. By Lemma 5, once an edge (v_i, v_j) is triggered, it is preserved in all subsequent scheduled trees and the prescribed-time guarantee ensures that the triggered edge is formed within $T_{\rm p}^{ij}$. Then at most $|\mathcal{V}| - 1$ distinct edges are triggered, each after waiting at most one planning window of length $T_{\rm w}$, and thus time-aggregated connectivity is achieved within $T_{\rm tree}(t_k) \leq$*



(a) MAPF (b) **Our Method** ($t = 10.15\,\mathrm{s}$) (c) **Our Method**

(d) PT-CBF ($T_\mathrm{p} = 3\,\mathrm{s}$) (e) PT-CBF ($T_\mathrm{p} = 15\,\mathrm{s}$) (f) FT-CBF

(g) MCCST (h) Randomly Selected Spanning Tree (i) TSMST

Fig. 1: Colored curves show robot trajectories, and the overlaid arcs indicate the selected communication edges at the shown time. Pink arcs denote edges that have been previously connected; solid red arcs denote edges whose connections are successfully established; dashed red arcs denote edges currently being established but not-yet reconnected; and gray dashed arcs denote selected but not-yet-triggered edges.

$(|\mathcal{V}|-1)T_\mathrm{w} + \sum_{(v_i,v_j)\in\mathcal{E}_\mathrm{mst}(t_k)} T_\mathrm{p}^{ij}$. *By time* $t_k + T_\mathrm{tree}(t_k)$, *the union graph of established links contains the spanning tree* $\tilde{\mathcal{T}}^*(t_k)$ *and is therefore connected.*

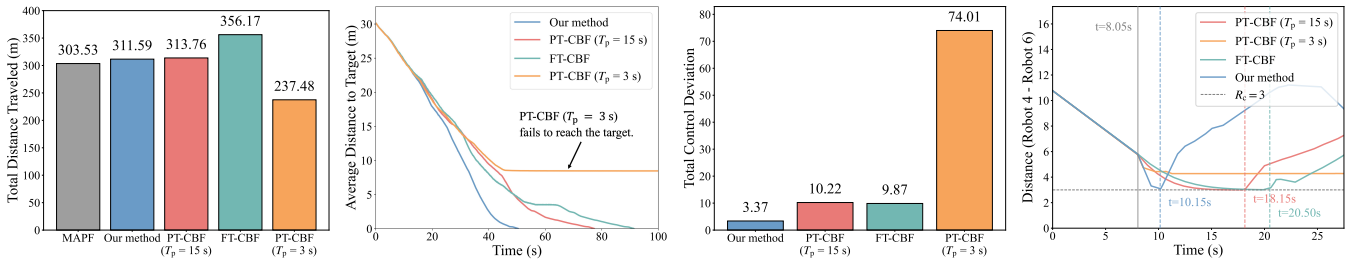
Proof: See the detailed proof in Appendix B. ∎

IV. Results

A. Simulation Example

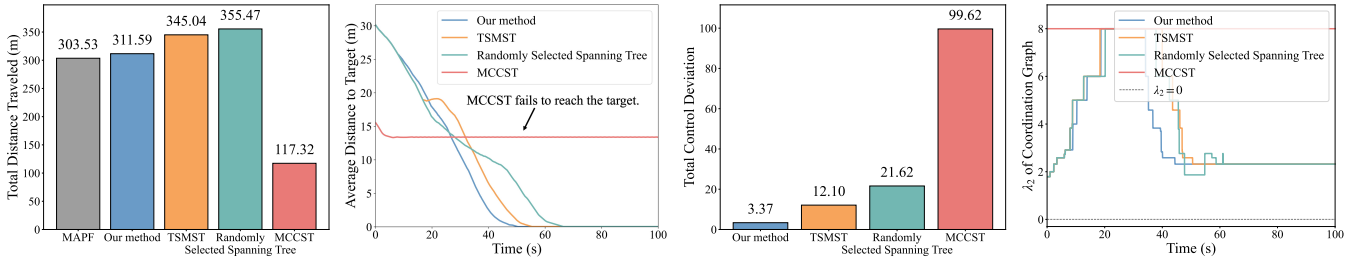
We evaluate our proposed method in a cluttered $32\,\mathrm{m}\times 54\,\mathrm{m}$ indoor environment with obstacles, using $N = 8$ robots. The communication radius is set to $R_\mathrm{c} = 3\,\mathrm{m}$ and the safety radius is set to $R_\mathrm{s} = 0.9\,\mathrm{m}$. The coordination range is set to $R_\mathrm{d} = 30\,\mathrm{m}$, under which the long-range coordination graph should remain connected throughout the execution. Nominal reference trajectories are generated by a Multi-Agent Path Finding (MAPF) planner [22]. The original planned trajectory is as shown in Fig. 1a, where each color corresponds to the planned path of each robot. The pre-planed trajectory by the MAPF planner requires $T_\mathrm{MAPF} = 47.45\,\mathrm{s}$ to finish all the tasks for this scenario. To ensure a fair comparison, we initialize all methods with the same start-goal configuration and evaluate them under an identical experimental setup.

Case Study 1: To evaluate the efficiency of the proposed adaptive PT-CBF (**our method**) in Sec. III-B, we compare it with PT-CBF [21] and FT-CBF [15], while using our method in Sec. III-A to compute the reconnection graph for all baselines. All methods share the same nominal plan and matched parameters. For PT-CBF, we choose two prescribed times $T_\mathrm{p} = 3\,\mathrm{s}$ and $T_\mathrm{p} = 15\,\mathrm{s}$. Fig. 2 summarizes the results. As shown in Fig. 2 (a)–(c), our method achieves a shorter traveled distance, lower control deviation, and faster task completion than the baselines. Fig. 2d further illustrates the reconnection behavior on a representative edge (Robot 4–Robot 6). As shown in Fig. 2d, all methods trigger the reconnection process between Robot 4 and Robot 6 at the same time $t = 8.35\,\mathrm{s}$. Starting from this *identical* trigger time, adaptive PT-CBF drives the inter-robot distance below R_c in a timely manner and faster than PT-CBF ($T_\mathrm{p} = 15\,\mathrm{s}$) and FT-CBF. PT-CBF ($T_\mathrm{p} = 3\,\mathrm{s}$) fails to bring the robots within R_c due to the overly aggressive prescribed time. When T_p is set larger, PT-CBF becomes less aggressive, but the actual time required for the robots to enter R_c increases accordingly. In contrast, FT-CBF exhibits increasingly slow distance reduction as the robots approach R_c. Overall, adaptive PT-CBF achieves



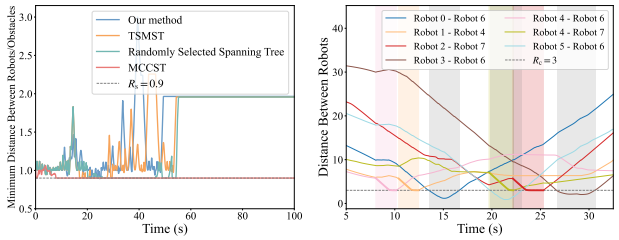
(a) Total distance traveled (b) Average distance to target (c) Total control deviation (d) Inter-robot distance between Robot 4 and Robot 6

Fig. 2: Comparison of our adaptive PT-CBF with PT-CBF and FT-CBF under the same setup.



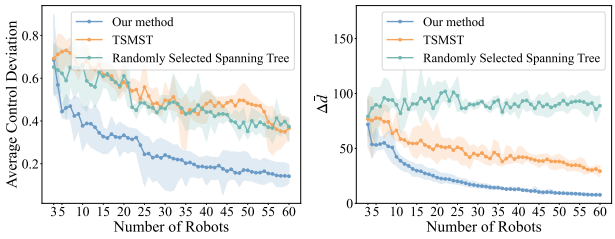
(a) Total distance traveled (b) Average distance to target (c) Total control deviation (d) λ_2 of Coordination graph

Fig. 3: Comparison of TWA-MST with MCCST, TSMST, and Randomly Selected Spanning Tree under the same setup.



(a) (b)

Fig. 4: (a) Minimum separation distance over time across robot–robot and robot–wall pairs. (b) Distances of TWA-MST-selected edges during reconnection. Gray shaded intervals indicate task-driven reconnection via planned motion, while colored intervals indicate our adaptive PT-CBF enforced reconnection. All edges reconnect within $R_c = 3\,\mathrm{m}$.

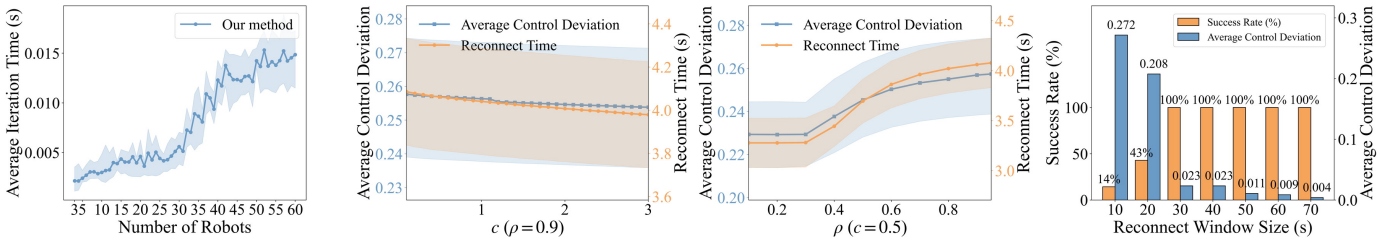


(a) (b)

Fig. 5: Performance comparison versus the number of agents. (a) Average control deviation. (b) Average additional travel distance $\Delta\bar{d} \triangleq \frac{\sum_{i=1}^{N} d_i - d_{\mathrm{MAPF}}}{N}$, where d_i represents total travel distance of robot i and d_{MAPF} is the nominal total travel distance planned by MAPF planner. Solid lines denote average over trials and shaded regions indicate the min–max range.

a better trade-off between reconnection timeliness and task efficiency under the same experimental setup.

Case Study 2: To evaluate the reconnection scheduling component, we compare our scheduler against Minimum Connectivity Constraint Spanning Tree (MCCST) [12], Time-sub-interval minimum spanning tree (TSMST), and the Randomly Selected Spanning Tree method baselines. For a fair comparison, TSMST and the Randomly Selected Spanning Tree method adopt our **adaptive PT-CBF** constraints to enforce their triggered reconnections, while MCCST [12] intrinsically enforces an always-connected requirement through its connectivity-maintenance mechanism. **TSMST** computes an MST using the instantaneous edge weights $w_{ij}(t)$ at each time instant $t \in [t_k,\, t_k + T_{\mathrm{w}}]$. Consequently, the entire tree is selected from a single-time snapshot (i.e., all edges are evaluated at the same t), rather than selecting edges from multiple time instants within $[t_k,\, t_k + T_{\mathrm{w}}]$. Among all those MST, we select one with minimum total weight as Time-sub-interval MST (TSMST) and the tree is triggered to be reconnected at the same time. **Randomly selected spanning tree** randomly selects a spanning tree from the set of all spanning trees of $\tilde{\mathcal{G}}^{\mathrm{s}}(t_k)$ with the minimum total cost. Fig. 3 summarizes the performance comparison. Our scheduler achieves the lowest total control deviation and the smallest traveled distance while completing the task, indicating that our scheduling yields a more consistent edge set for reconnection with reduced disruption to the nominal task execution. In contrast, MCCST enforces persistent connectivity constraints, which can become overly restrictive in cluttered environments, it fails to reach the goals within the evaluation horizon. Compared with our method, both TSMST and the Randomly Selected Spanning Tree method incur substantially higher traveled distance and



(a) (b) (c) (d)

Fig. 6: (a) Average per-iteration computation time of the proposed method as the number of robots increases. (b) Sensitivity of the proposed method to parameter c in terms of average control deviation and reconnect time. (c) Sensitivity of the proposed method to parameter ρ in terms of average control deviation and reconnect time. (d) Success rate (i.e., whether time-aggregated connectivity is successfully satisfied) and average control deviation under different reconnect window sizes.

control deviation, with the Randomly Selected Spanning Tree method consistently worse than TSMST. Besides, Fig. 2b and Fig. 3b further verify the efficiency of the proposed control constraints in enforcing the scheduled reconnection. Fig. 4a shows that all methods satisfy the desired safety requirement. Fig. 4b illustrates the reconnection schedule produced by our scheduler, where the shaded gray regions indicate reconnections induced by the original MAPF planner, while the colored regions indicate reconnections between different robot pairs enforced by the adaptive PT-CBF. Overall, our scheduler better balances task efficiency and connectivity requirements without enforcing always-connected behavior.

B. Quantitative Results

We first conduct *large-scale* quantitative evaluations in a $128\,\mathrm{m} \times 128\,\mathrm{m}$ environment. We randomly generate 30 obstacles and sample start and goal configurations for up to $N = 60$ robots. For each method, we repeat the experiment five times with independent start and goal pairs. In all runs, the desired aggregated reconnection procedure is successfully completed. Fig. 5 reports the performance comparison. Our method achieves lower average additional travel distance $\Delta\bar{d}$ and lower total control deviation across different team sizes. Overall, both metrics decrease as the number of robots increases. In contrast, TSMST and Randomly selected spanning tree incur substantially higher $\Delta\bar{d}$ and control deviation, indicating less efficient coordination under the same task setup. We also report runtime scaling in Fig. 6a. In the 60-robot case on a MacBook Air (M4), the *average* per-step computation time for solving Eq. (15) is about 15 ms, corresponding to roughly 67 Hz. Updating TWA-MST takes about 66 ms, but this computation is only required during replanning rather than at every control step, indicating favorable real-time performance. We further evaluate the sensitivity of the proposed adaptive PT-CBF controller to its main parameters, c and ρ, in Eq. (13), and the effect of reconnect window length in Fig. 6(b)–(d). As shown in Eq. (14), our Adaptive PT-CBF with a larger c or smaller ρ would allow a relatively lower prescribed time upper bound T_p, thus enforcing shorter actual reconnect time. While a smaller c or a larger ρ instead increases the prescribed time upper bound, thereby enabling the robots to have a longer reconnection time. Motivated by this theoretical relationship, we isolate the controller behavior from multi-robot coordination effects to enable a fair comparison of the effects of c and ρ in a pairwise robot setting, as shown in Fig. 6(b)–(c). Fig. 6b confirms that increasing c leads to an approximately linear decrease in reconnect time and the average control deviation over the tested range. Fig. 6c verifies that increasing ρ causes both reconnect time and average control deviation to increase nonlinearly. We continue to evaluate different reconnection window sizes in the scenario of Fig. 1. As shown in Fig. 6d, very short windows provide insufficient time to realize the desired union graph, resulting in lower success rates and larger control deviation. While very long windows tend to reduce control deviation by enabling better future reconnection opportunities, they require increased computation and memory for MAPF replanning and trajectory storage. Thus, window length presents a practical trade-off governed by available computing resources and the configuration of the robot team.

V. Conclusion and Future Work

In this paper, we propose a motion coordination framework for multi-robot systems that balances task execution and intermittent communication with time-aggregated connectivity guarantee. We introduced a time-window aggregated minimum spanning tree (TWA-MST) on the spatiotemporal multi-robot communication graph to identify a compact set of low-effort reconnection edges, together with an adaptive prescribed-time CBF (adaptive PT-CBF) that enforces timely reconnection of triggered robot pairs while minimally perturbing nominal motions. Theoretical analysis proves the reconnection and time-aggregated connectivity guarantees, and simulation results show improved performance compared with baselines while scaling to large robot teams.

Future work includes developing decentralized implementation of our framework and extending to connectivity restoration under failures and uncertainties (e.g., localization errors), as well as incorporating more realistic communication models (e.g., line-of-sight [17] and received signal strength based models [23]) to capture directional and stochastic communication effects, such as bandwidth limits, packet loss, and delays.

References

[1] Lorenzo Sabattini, Cristian Secchi, Nikhil Chopra, and Andrea Gasparri. Distributed control of multirobot systems with global connectivity maintenance. *IEEE Transactions on Robotics*, 29(5):1326–1332, 2013. URL https://ieeexplore.ieee.org/abstract/document/6548083.

[2] Maira Saboia, Lillian Clark, Vivek Thangavelu, Jeffrey A Edlund, Kyohei Otsu, Gustavo J Correa, Vivek Shankar Varadharajan, Angel Santamaria-Navarro, Thomas Touma, Amanda Bouman, et al. Achord: Communication-aware multi-robot coordination with intermittent connectivity. *IEEE Robotics and Automation Letters*, 7(4):10184–10191, 2022. URL https://ieeexplore.ieee.org/document/9837416.

[3] Yiannis Kantaros, Meng Guo, and Michael M Zavlanos. Temporal logic task planning and intermittent connectivity control of mobile robot networks. *IEEE Transactions on Automatic Control*, 64(10):4105–4120, 2019. URL https://ieeexplore.ieee.org/abstract/document/8612974.

[4] Ali Jadbabaie, Jie Lin, and A Stephen Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on automatic control*, 48(6):988–1001, 2003. URL https://ieeexplore.ieee.org/abstract/document/1205192.

[5] Luc Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on automatic control*, 50(2):169–182, 2005. URL https://ieeexplore.ieee.org/abstract/document/1393134.

[6] Reza Olfati-Saber, J Alex Fax, and Richard M Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007. URL https://ieeexplore.ieee.org/abstract/document/4118472.

[7] Alejandro Cornejo, Fabian Kuhn, Ruy Ley-Wild, and Nancy Lynch. Keeping mobile robot swarms connected. In *International Symposium on Distributed Computing*, pages 496–511. Springer, 2009. URL https://link.springer.com/chapter/10.1007/978-3-642-04355-0_50.

[8] Michal Čáp, Peter Novák, Alexander Kleiner, and Martin Seleckỳ. Prioritized planning algorithms for trajectory coordination of multiple mobile robots. *IEEE transactions on automation science and engineering*, 12(3):835–849, 2015. URL https://ieeexplore.ieee.org/abstract/document/7138650.

[9] Luke Antonyshyn, Jefferson Silveira, Sidney Givigi, and Joshua Marshall. Multiple mobile robot task and motion planning: A survey. *ACM Computing Surveys*, 55(10):1–35, 2023. URL https://dl.acm.org/doi/full/10.1145/3564696.

[10] Viswanath Gunturi, Shashi Shekhar, and Arnab Bhattacharya. Minimum spanning tree on spatio-temporal networks. In *International Conference on Database and Expert Systems Applications*, pages 149–158. Springer, 2010. URL https://link.springer.com/chapter/10.1007/978-3-642-15251-1_11.

[11] Alysson Ribeiro Da Silva, Luiz Chaimowicz, Thales C Silva, and M Ani Hsieh. Communication-constrained multi-robot exploration with intermittent rendezvous. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3490–3497. IEEE, 2024. URL https://ieeexplore.ieee.org/abstract/document/10802343.

[12] Wenhao Luo, Sha Yi, and Katia Sycara. Behavior mixing with minimum global and subgroup connectivity maintenance for large-scale multi-robot systems. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9845–9851. IEEE, 2020. URL https://ieeexplore.ieee.org/abstract/document/9197429.

[13] Beatrice Capelli and Lorenzo Sabattini. Connectivity maintenance: Global and optimized approach through control barrier functions. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5590–5596. IEEE, 2020. URL https://ieeexplore.ieee.org/abstract/document/9197109.

[14] Yupeng Yang, Yiwei Lyu, Yanze Zhang, Sha Yi, and Wenhao Luo. Decentralized Multi-Robot Line-of-Sight Connectivity Maintenance under Uncertainty. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.005. URL https://arxiv.org/abs/2406.12802.

[15] Imoleayo Abel, Drew Steeves, Miroslav Krstić, and Mrdjan Janković. Prescribed-time safety design for strict-feedback nonlinear systems. *IEEE Transactions on Automatic Control*, 69(3):1464–1479, 2023. URL https://ieeexplore.ieee.org/abstract/document/10288378.

[16] Tomáš Rouček, Martin Pecka, Petr Čížek, Tomáš Petříček, Jan Bayer, Vojtěch Šalanskỳ, Daniel Heřt, Matěj Petrlík, Tomáš Báča, Voječh Spurnỳ, et al. Darpa subterranean challenge: Multi-robotic exploration of underground environments. In *International Conference on Modelling and Simulation for Autonomous Systems*, pages 274–290. Springer, 2019. URL https://link.springer.com/chapter/10.1007/978-3-030-43890-6_22.

[17] Yupeng Yang, Yiwei Lyu, and Wenhao Luo. Minimally constrained multi-robot coordination with line-of-sight connectivity maintenance. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7684–7690. IEEE, 2023. URL https://ieeexplore.ieee.org/document/10161401.

[18] Li Wang, Aaron D Ames, and Magnus Egerstedt. Safety barrier certificates for collisions-free multirobot systems. *IEEE Transactions on Robotics*, 33(3):661–674, 2017. URL https://ieeexplore.ieee.org/document/7857061.

[19] Aaron D Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications. In *2019 18th European control conference (ECC)*, pages 3420–3431. Ieee, 2019. URL https://ieeexplore.ieee.org/document/8796030.

[20] Yi-Hsuan Chen, Shuo Liu, Wei Xiao, Calin Belta, and Michael Otte. Control barrier functions via minkowski

operations for safe navigation among polytopic sets. In *2025 IEEE 64th Conference on Decision and Control (CDC)*, pages 4481–4488. IEEE, 2025. URL https://ieeexplore.ieee.org/document/11312188.

[21] Anqi Li, Li Wang, Pietro Pierpaoli, and Magnus Egerstedt. Formally correct composition of coordinated behaviors using control barrier certificates. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3723–3729. IEEE, 2018. URL https://ieeexplore.ieee.org/document/8594302.

[22] Yimin Tang, Sven Koenig, and Jiaoyang Li. Ita-ecbs: A bounded-suboptimal algorithm for combined target-assignment and path-finding problem. In *Proceedings of the International Symposium on Combinatorial Search*, volume 17, pages 134–142, 2024. URL https://arxiv.org/abs/2404.05223.

[23] Yupeng Yang, Yiwei Lyu, Yanze Zhang, Ian Gao, and Wenhao Luo. Integrating online learning and connectivity maintenance for communication-aware multi-robot coordination. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5770–5776. IEEE, 2024. URL https://ieeexplore.ieee.org/document/10802189.

VI. APPENDIX

In this appendix, we provide detailed proofs of Lemma 5 and Proposition 11.

A. Proof of Lemma 5

Lemma 5. ***Consistency over time (persistence of triggered edges)****. Consider a sequence of replanning instants $\{\tau_k\}_{k\geq 0}$. At each τ_k, the reweighted time-window aggregated graph is formed according to Definition 4. Let $\mathcal{F}(\tau_k) \triangleq \{(v_i, v_j) \in \mathcal{E}^{\mathrm{s}} \mid \sigma_{ij}(\tau_k) = 1\}$ denotes the set of edges that have been triggered up to time τ_k. Then, at every replanning instant, the newly computed minimum spanning tree on the reweighted graph must contain all edges in $\mathcal{F}(\tau_k)$. In other words, once an edge is triggered, it is preserved in all subsequent tree selections despite replanning updates.*

Proof: Fix any k and any triggered edge $e = (v_i, v_j) \in \mathcal{F}(\tau_k)$. Since $\mathcal{F}(\tau_k)$ is a forest (acyclic), every edge in it is a bridge. Removing e disconnects the component containing e into two vertex sets; let $S_e \subseteq \mathcal{V}$ be the set containing v_i after removal (hence $v_j \in \mathcal{V} \setminus S_e$). Let $\delta(S_e)$ denote the edges crossing the cut $(S_e, \mathcal{V} \setminus S_e)$. Clearly $e \in \delta(S_e)$, and any other edge $e' \in \delta(S_e) \setminus \{e\}$ cannot belong to $\mathcal{F}(\tau_k)$; otherwise it would create a cycle, contradicting acyclicity. Thus, $\delta(S_e) \setminus \{e\} \subseteq \mathcal{E}^{\mathrm{s}} \setminus \mathcal{F}(\tau_k)$. By Eq. (11), for $e \in \mathcal{F}(\tau_k)$ we have $\tilde{w}_{ij}(\tau_k) = \bar{w}_{ij}(\tau_k)/\lambda$, while for any $e' = (v_{i'}, v_{j'}) \in \mathcal{E}^{\mathrm{s}} \setminus \mathcal{F}(\tau_k)$ we have $\tilde{w}_{i'j'}(\tau_k) = \bar{w}_{i'j'}(\tau_k)$. With the choice of λ in Definition 4, it follows that $\tilde{w}_{ij}(\tau_k) < \tilde{w}_{i'j'}(\tau_k)$ for all $e' = (v_{i'}, v_{j'}) \in \delta(S_e) \setminus \{e\}$. Hence e is the unique minimum-weight edge crossing the cut $(S_e, \mathcal{V} \setminus S_e)$. By the cut property, e must belong to every MST of $\tilde{\mathcal{G}}^{\mathrm{s}}(\tau_k)$, and thus $(v_i, v_j) \in \mathcal{E}(\mathcal{T}(\tau_k))$. Since $e \in \mathcal{F}(\tau_k)$ is arbitrary, we obtain $\mathcal{F}(\tau_k) \subseteq \mathcal{E}(\mathcal{T}(\tau_k))$ for all $k \geq 0$, which completes the proof. ∎

B. Proof of Proposition 11

Proposition 11. ***Bounded-time time-aggregated connectivity****. Let $\tilde{\mathcal{T}}^*(t_k)$ be the TWA-MST computed at time t_k (Theorem 6) with edge set $\mathcal{E}_{\mathrm{mst}}(t_k) = \mathcal{E}(\tilde{\mathcal{T}}^*(t_k))$. Following Algorithm 1, the edges in $\mathcal{E}_{\mathrm{mst}}(t_k)$ are triggered sequentially. By Lemma 5, once an edge (v_i, v_j) is triggered, it is preserved in all subsequent scheduled trees and the prescribed-time guarantee ensures that the triggered edge is formed within T_{p}^{ij}. Then at most $|\mathcal{V}| - 1$ distinct edges are triggered, each after waiting at most one planning window of length T_{w}, and thus time-aggregated connectivity is achieved within $T_{\mathrm{tree}}(t_k) \leq (|\mathcal{V}| - 1)T_{\mathrm{w}} + \sum_{(v_i,v_j)\in\mathcal{E}_{\mathrm{mst}}(t_k)} T_{\mathrm{p}}^{ij}$. By time $t_k + T_{\mathrm{tree}}(t_k)$, the union graph of established links contains the spanning tree $\tilde{\mathcal{T}}^*(t_k)$ and is therefore connected.*

Proof: Following Algorithm 1, at the beginning of a reconnection process, i.e., in the first look-ahead window starting at t_0, our method computes the initial TWA-MST $\tilde{\mathcal{T}}^*(t_0)$ over the interval $[t_0,\, t_0 + T_{\mathrm{w}}]$. By construction of the time-window aggregated edge weights, it is guaranteed that at least one edge in $\tilde{\mathcal{T}}^*(t_0)$ is scheduled to be triggered within this window. In the worst case, such a triggering occurs at the end of the window, i.e., at $t_0 + T_{\mathrm{w}}$.

Once an edge (v_i, v_j) is triggered, Theorem 10 guarantees that the corresponding robot pair can be driven back into the communication range within a bounded time T_{p}^{ij}. At the next replanning instant t_1, the algorithm recomputes a new TWA-MST $\tilde{\mathcal{T}}^*(t_1)$ based on the updated trajectories and edge weights. By Lemma 5, any edge that has been realized within a previous look-ahead window is guaranteed to remain contained in all subsequent TWA-MST solutions. As a result, all triggered edges are monotonically preserved across the sequence $\{\tilde{\mathcal{T}}^*(t_k)\}$ and are never removed or reselected in future replanning steps.

Consequently, the scheduled TWA-MST evolves as a tree-construction process: in each look-ahead window, at least one new edge is triggered and added, while all previously realized edges are retained. Although a new TWA-MST $\tilde{\mathcal{T}}^*(t_k)$ is computed at each replanning time, all previously triggered edges are retained in the evolving tree solely to guarantee time-aggregated connectivity at the reconnection graph. Note that, once an edge has been triggered and realized, it is not triggered again within the same reconnection process. Instead, it remains in the reconnection graph as a realized connection until the current time-aggregated connectivity objective is achieved and the process is reset. Since a minimum spanning tree over $|\mathcal{V}|$ nodes contains exactly $|\mathcal{V}| - 1$ edges, the full tree is formed after at most $|\mathcal{V}| - 1$ look-ahead windows. Therefore, the time required to achieve time-aggregated connectivity is bounded by:

$$T_{\mathrm{tree}}(t_k) \;\leq\; (|\mathcal{V}| - 1)T_{\mathrm{w}} + \sum_{(v_i,v_j)\in\mathcal{E}_{\mathrm{mst}}(t_0)} T_{\mathrm{p}}^{ij}.$$

Since the final TWA-MST $\tilde{\mathcal{T}}^*(t_k)$ obtained after the tree construction is formed within bounded time, the **union graph** over the corresponding time interval contains this spanning tree and is therefore connected.

In the current formulation, once a selected robot pair reconnects using the adaptive PT-CBF, the corresponding constraint can be released, allowing the robots to resume their nominal tasks. If sustained communication is required (e.g., to allow sufficient information exchange for a longer time), the constraint can instead be enforced for a user-defined "*stay*" time T_{stay}. In this case, by Theorem 10, the robots are guaranteed to remain within the communication range during this period, and time-aggregated connectivity is achieved within:

$$T_{\mathrm{tree}}(t_k) \;\leq\; (|\mathcal{V}| - 1)(T_{\mathrm{w}} + T_{\mathrm{stay}}) + \sum_{(v_i,v_j)\in\mathcal{E}_{\mathrm{mst}}(t_0)} T_{\mathrm{p}}^{ij}.$$

Note that, once time-aggregated connectivity of the multi-robot is achieved , the Algorithm 1 can be reset and reapplied to initiate a new reconnection process, enabling repeated rounds of time-aggregated connectivity over the time horizon. ∎