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

Dynamics-Aware Critical Neighbor Selection for Distributed Connectivity Maintenance in Multi-Agent Systems*

Wei Tan and Xin Chen

***Abstract*— Maintaining connectivity in multi-agent systems often compromises task performance. Current strategies are frequently hampered by heavy communication loads and overly restrictive motion constraints. Furthermore, their local decision-making relies on static geometric information, neglecting agent dynamics. To address these shortcomings, this paper proposes a scalable, distributed framework centered on a novel dynamics-aware connection cost metric. This metric enables agents to prospectively select dynamically stable, task-compatible links, which are then enforced using control barrier functions (CBFs) within a cooperative optimization scheme. In multi-agent target-reaching tasks, simulations show our dynamics-aware metric reduces the final average goal distance by up to 26.1% compared to a static distance-based selection heuristic. Furthermore, our framework maintains persistent connectivity in highly dynamic scenarios, whereas a state-of-the-art algebraic connectivity-based method fails under limited communication bandwidth.**

***Index Terms*— Multi-Agent Systems, distributed control, connectivity maintenance.**

I. INTRODUCTION

Coordinated multi-agent systems offer remarkable advantages in applications ranging from environmental monitoring [1], [2] to search and rescue [3]. The success of these systems hinges on persistent inter-agent communication, which enables the distributed coordination necessary for complex tasks. However, maintaining this communication network, especially in obstacle-rich environments or when tasks require agents to disperse, creates a fundamental conflict. The challenge lies in designing a distributed control strategy that preserves global network connectivity while imposing minimal restrictions on task performance.

Early approaches to this problem were either centralized [4], [5], suffering from poor scalability and single points of failure [6], or overly conservative distributed methods that restricted mobility by preserving all initial communication links [7]. To achieve greater flexibility, many modern strategies now focus on maintaining the algebraic connectivity (λ_2) of the network's Laplacian graph above a positive threshold [8]–[11]. While theoretically sound, the reliance on a global network property like λ_2 introduces significant practical hurdles for real-time, distributed control.

*This work was supported in part by the National Natural Science Foundation of China under Grant 62333019, the Fundamental Research Funds for the Central Universities, China University of Geosciences, and the 111 project under Grant B17040.

The authors are with the School of Automation, and the Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, China University of Geosciences, Wuhan, Hubei 430074, China (e-mail: {tanwei, chenxin}@cug.edu.cn).

Corresponding author: Xin Chen.

These hurdles manifest in several critical ways. The underlying estimation protocols, for instance, require iterative, multi-round communication to estimate λ_2 [12], incurring significant computational and communication overhead [13], [14]. Furthermore, the estimation can fail in configurations where λ_2 is not a simple eigenvalue [14]. Although λ_2 is a continuous function of the system state, it is not continuously differentiable. Specifically, its gradient changes discontinuously at points where its multiplicity is greater than one. This lack of smoothness violates the core assumptions of frameworks like CBFs, which can lead to discontinuous controllers and a forfeiture of formal safety guarantees [15]. These limitations motivate the search for an alternative paradigm.

A promising alternative that circumvents the pitfalls of λ_2 estimation is the concept of critical neighbor selection. By identifying and maintaining connections to only a minimal subset of neighbors essential for global connectivity, this approach reformulates the problem into a set of local, tractable constraints. However, existing methods often rely on myopic selection criteria, such as static hop counts and Euclidean distance [16], which ignore agent dynamics. This can lead to poor choices, where a selected link is topologically sound but dynamically unstable, requiring excessive control effort to maintain and unnecessarily hindering task execution.

To address these limitations, this paper proposes a novel distributed control framework that synergizes a dynamics-aware critical neighbor selection process with a distributed optimization scheme. Instead of relying on reactive or static metrics, our agents select critical neighbors based on a connection cost that predicts link stability under their intended task-oriented motion. We then formalize connectivity and collision avoidance constraints using CBFs. However, a critical challenge arises because these CBF constraints are inherently coupled: an agent's safe control action explicitly depends on its neighbors' actions. To cooperatively resolve this, we integrate a state-of-the-art distributed optimization framework [17]. This method employs continuously updated auxiliary variables to achieve shared responsibility. Consequently, each agent can solve a local quadratic program while maintaining violation-free safety guarantees throughout the process.

The key contributions of this paper are:

1) A dynamics-aware critical neighbor selection algorithm based on a novel connection cost metric that predicts link stability under nominal motion.
2) A CBF-based control architecture that enforces connectivity with the selected critical neighbors, ensuring

the integrity of the communication backbone.

3) The integration of a violation-free distributed optimization framework to cooperatively resolve coupled CBF safety constraints among agents via local computation.

The remainder of this paper is organized as follows. Section II introduces preliminaries on our system model and CBFs. Section III details our proposed framework, including the dynamics-aware cost metric and the critical neighbor selection algorithm. Section IV presents the distributed optimization method for safe control synthesis. The theoretical analysis of connectivity maintenance is provided in Section V. Finally, Section VI presents simulation results, followed by conclusions in Section VII.

II. PRELIMINARIES

This section introduces the system model, problem formulation, and the fundamentals of CBFs that form the basis of our control framework.

A. System Model and Problem Formulation

We consider a team of N mobile robots, indexed by the set $\mathcal{R} = \{1, 2, \ldots, N\}$. Each robot $i \in \mathcal{R}$ operates in a d-dimensional Euclidean space $\mathbb{R}^d$ for $d \in \{2, 3\}$, and its state is characterized by its position $\mathbf{p}_i(t) \in \mathbb{R}^d$ and actual velocity $\mathbf{v}_i(t) \in \mathbb{R}^d$. We assume each robot is equipped with a low-level controller capable of tracking a commanded velocity $\mathbf{v}_{i,\mathrm{c}}(t) \in \mathbb{R}^d$ generated by our higher-level control framework using a reduced-order model (ROM)

$$\dot{\mathbf{p}}_i(t) = \mathbf{v}_{i,\mathrm{c}}(t). \tag{1}$$

This reduced-order model assumes a sufficiently competent low-level controller is available to track the commanded velocity $\mathbf{v}_{i,\mathrm{c}}(t)$. Our simulations are conducted in $\mathbb{R}^2$, and the framework is directly applicable to $\mathbb{R}^3$.

The communication capabilities are represented by a time-varying undirected graph $G(t) = (\mathcal{V}, \mathcal{E}(t))$, where $\mathcal{V} = \mathcal{R}$ and $\mathcal{E}(t)$ is the set of active communication links. In this paper, the terms *edge* and *link* are used interchangeably to denote a valid communication channel. An edge $(i, j) \in \mathcal{E}(t)$ exists if

$$\|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|^2 \leq D_{\max}^2, \tag{2}$$

where $D_{\max}$ is the maximum communication range. The set of communication neighbors for agent i is

$$\mathcal{N}_i(t) = \{j \in \mathcal{R} \setminus \{i\} \mid \|\mathbf{p}_i(t) - \mathbf{p}_j(t)\|^2 \leq D_{\max}^2\}. \tag{3}$$

Our objective is to design a distributed control input $\mathbf{v}_{i,\mathrm{c}}(t)$ for each robot i that enables the team to achieve a desired task, represented by a nominal velocity command $\mathbf{v}_{i,\mathrm{nom}}(t)$, while primarily ensuring that global network connectivity is maintained. The controller must be distributed, using only local information from $\mathcal{N}_i(t)$.

B. Control Barrier Functions for Connectivity Maintenance

To formally guarantee connectivity maintenance, we employ CBFs. A CBF is a function $h(\mathbf{p})$ that defines a safe set $\mathcal{C} = \{\mathbf{p} \mid h(\mathbf{p}) \geq 0\}$. Safety is maintained if any controller applied to the system renders the set $\mathcal{C}$ forward invariant, meaning if the system starts in $\mathcal{C}$, it never leaves it.

For our multi-agent system with integrator dynamics $\dot{\mathbf{p}}_i = \mathbf{v}_{i,\mathrm{c}}$, this invariance is typically enforced by constraining the control inputs $\mathbf{v}_{i,\mathrm{c}}$ such that for any state on the boundary of the safe set ($\partial\mathcal{C} = \{\mathbf{p} \mid h(\mathbf{p}) = 0\}$), the vector field points into or along the boundary. A sufficient condition for all $\mathbf{p} \in \mathcal{C}$ is:

$$\dot{h}(\mathbf{p}, \mathbf{v}_\mathrm{c}) = \frac{\partial h}{\partial \mathbf{p}} \dot{\mathbf{p}} \geq -\alpha(h(\mathbf{p})), \tag{4}$$

where $\alpha(\cdot)$ is an extended class $\mathcal{K}$ function (e.g., a linear function $\alpha(x) = \alpha x$ for some constant $\alpha > 0$). This condition ensures that as the system state approaches the boundary of the safe set (i.e., as $h(\mathbf{p}) \to 0$), the control action prevents violation.

To maintain a communication link between agents i and j within range $D_{\max}$, we define the safe set using the function:

$$h_{ij}(\mathbf{p}_i, \mathbf{p}_j) = D_{\max}^2 - \|\mathbf{p}_i - \mathbf{p}_j\|^2. \tag{5}$$

The corresponding CBF constraint on their commanded velocities is:

$$-2(\mathbf{p}_i - \mathbf{p}_j)^\top (\mathbf{v}_{i,\mathrm{c}} - \mathbf{v}_{j,\mathrm{c}}) \geq -\alpha h_{ij}. \tag{6}$$

While this constraint is ultimately used to synthesize the final safe control action, the CBF framework also provides a powerful tool for predicting the behavior of links. We leverage this predictive capability to inform our critical neighbor selection, as detailed next.

III. PROPOSED FRAMEWORK: DYNAMICS-AWARE CRITICAL NEIGHBOR SELECTION

This section presents our core contribution: a distributed control framework for maintaining global connectivity. The fundamental challenge addressed is enabling each agent to make local control decisions that preserve network-wide connectivity. Our solution integrates a novel critical neighbor selection strategy, which proactively predicts link stability, with the CBF framework. This approach avoids the need for global information exchange or iterative estimation of network properties such as the algebraic connectivity $\lambda_2(L(G(t)))$.

A. Dynamics-Aware Connection Cost Metric

A key innovation in our approach is a connection cost metric that evaluates the desirability of maintaining a link. Instead of relying on static geometric distance, our metric proactively predicts how a link will evolve under the agents' task-oriented nominal velocities. This is achieved by evaluating the connectivity CBF condition from Section II-B prospectively.

For a pair of communicating agents (i, j), we use the connectivity CBF candidate $h_{ij}(t) = D_{\max}^2 - \|\mathbf{p}_{ij}(t)\|^2$. The CBF condition (4) requires that $\dot{h}_{ij} + \alpha h_{ij} \geq 0$. We repurpose the left-hand side of this inequality to score a connection's quality by calculating it with the *nominal* relative velocity $\mathbf{v}_{ij,\text{nom}}(t) \triangleq \mathbf{v}_{i,\text{nom}}(t) - \mathbf{v}_{j,\text{nom}}(t)$.

Let $\mathbf{p}_{ij}(t) \triangleq \mathbf{p}_i(t) - \mathbf{p}_j(t)$. The time derivative of h_{ij} under these nominal velocities is $\dot{h}_{\text{nom},ij}(t) = -2\mathbf{p}_{ij}(t)^\top \mathbf{v}_{ij,\text{nom}}(t)$. We define a scoring function $H_{\text{nom},ij}(t)$ that quantifies this prospective safety margin:

$$
\begin{aligned}
H_{\text{nom},ij}(t) &= \dot{h}_{\text{nom},ij}(t) + \alpha_{\text{cost}} h_{ij}(t) \\
&= -2\mathbf{p}_{ij}(t)^\top \mathbf{v}_{ij,\text{nom}}(t) + \alpha_{\text{cost}} \left(D_{\max}^2 - \|\mathbf{p}_{ij}(t)\|^2\right),
\end{aligned} \tag{7}
$$

where $\alpha_{\text{cost}} > 0$ is a weighting factor. This scoring function balances two key factors: the link's immediate rate of change, represented by the Lie derivative, and its current safety buffer, given by the CBF value. Here, *link stability* is quantitatively defined as the value of this prospective safety margin $H_{\text{nom},ij}(t)$. A high positive value of $H_{\text{nom},ij}(t)$ indicates that the nominal controls are highly favorable to maintaining the link, or that the link has a large initial safety margin $h_{ij}(t)$. We define the connection cost as the negative of this score:

$$
\text{cost}_{ij}(t) \triangleq -H_{\text{nom},ij}(t). \tag{8}
$$

Thus, minimizing this cost corresponds to selecting neighbors with whom the connection is either intrinsically stable (large h_{ij}) or is actively being strengthened by the nominal task controllers (large positive $\dot{h}_{\text{nom},ij}$). By selecting neighbors that minimize this cost, agents prioritize connections that are predicted to be the most stable and require the least control intervention to preserve.

B. Critical Neighbor Selection

The core principle of our selection strategy is to dynamically construct a communication spanning tree rooted at a designated seed agent s_0. Instead of preserving all available links, each agent proactively seeks the most dynamically stable connection pointing toward the root (Case 1), while establishing lateral connections to prevent topological fragmentation (Case 2).

To avoid notation conflicts with the CBF h_{ij}, we denote the hop count (distance to the seed) of agent i as $l_i(t)$.

Definition 1 (Critical Link): A communication edge $(i, j) \in \mathcal{E}(t)$ is defined as a *critical link* for agent i if j is actively selected or maintained by i. These links constitute the minimal topological backbone and are the only edges strictly enforced via CBFs.

1) Hop Count Estimation: Each agent i estimates its hop count level $l_i(t)$ relative to a designated seed agent s_0, for which $l_{s_0}(t) \equiv 0$. Non-seed agents initialize their levels as $l_i(0) = N$ (a safe upper bound) and iteratively update them via local communication:

$$
l_i(t) = \min\left(\{N\} \cup \{l_j(t - \Delta t) + 1 \mid j \in \mathcal{N}_i(t)\}\right), \tag{9}
$$

where Δt is the discrete update interval.

2) Selection Rules: Let $l_i^{\min}(t) = \min\left(\{l_j(t) \mid j \in \mathcal{N}_i(t)\} \cup \{N\}\right)$ represent the closest available hop-level to the seed among agent i's neighbors. Each robot i updates its active critical neighbor set $C_i(t)$ according to the following mutually exclusive cases:

Case 1: $l_i^{\min}(t) < l_i(t)$. Robot i can reduce its hop count. This case contains our primary contribution, where the connection cost proactively guides the selection.

1) Define the set of potential gradient neighbors:

$$
\mathcal{N}_i^g(t) \triangleq \left\{j \in \mathcal{N}_i(t) \mid l_j(t) = l_i^{\min}(t)\right\}. \tag{10}
$$

2) Select the primary gradient neighbor $g_i(t)$ from $\mathcal{N}_i^g(t)$ with the minimum connection cost:

$$
g_i(t) \triangleq \arg\min_{k \in \mathcal{N}_i^g(t)} \text{cost}_{ik}(t). \tag{11}
$$

3) To prevent selection of dynamically unstable paths—such as links subject to rapid spatial divergence—the algorithm evaluates 'anchor' neighbors. An anchor neighbor j offers a more robust indirect route to the seed by maintaining a lower connection cost to the gradient neighbor $g_i(t)$ than agent i itself. As outlined in Algorithm 1, we formally define the set of candidate anchor neighbors as:

$$
\begin{aligned}
\mathcal{N}_i^{b'}(t) \triangleq \{&j \in \mathcal{N}_i(t) \mid (j \in \mathcal{N}_{g_i(t)}(t)) \wedge (l_j(t) \leq l_i(t)) \\
&\wedge (\text{cost}_{j,g_i(t)}(t) < \text{cost}_{i,g_i(t)}(t))\}.
\end{aligned} \tag{12}
$$

4) The comprehensive candidate set is $\mathcal{N}_i^b(t) \triangleq \mathcal{N}_i^{b'}(t) \cup \{g_i(t)\}$. Agent i selects the single neighbor from this set that minimizes the connection cost:

$$
C_i(t) = \left\{\arg\min_{j \in \mathcal{N}_i^b(t)} \text{cost}_{ij}(t)\right\}. \tag{13}
$$

Case 2: $l_i^{\min}(t) = l_i(t)$. To prevent fragile, chain-like topologies from breaking due to unpredicted disturbances, agent i connects to all neighbors at the same hop-level:

$$
C_i(t) = \{j \in \mathcal{N}_i(t) \mid l_j(t) = l_i(t)\}. \tag{14}
$$

Case 3: $l_i^{\min}(t) > l_i(t)$ or $\mathcal{N}_i(t) = \emptyset$. This case applies strictly to the designated seed node s_0 (which inherently has a lower level than all neighbors) or to a completely isolated agent. No new active critical neighbors are selected:

$$
C_i(t) = \emptyset. \tag{15}
$$

After robot i determines its set $C_i(t)$, it informs all robots $k \in C_i(t)$. Each robot k then records i in its passive critical neighbor set $P_k(t) = \{i \mid k \in C_i(t)\}$. The complete set of critical neighbors for robot i, which dictates the active CBF constraints, is $K_i^{\text{all}}(t) = C_i(t) \cup P_i(t)$.

The hierarchical selection process is formally structured to prioritize link stability while maintaining a path to the seed. To facilitate reproducibility and provide a concise overview of the local decision-making process executed at each time step, the complete dynamics-aware critical neighbor selection procedure is summarized in Algorithm 1.

Algorithm 1 Dynamics-Aware Critical Neighbor Selection (Executed locally by Agent i at time t)

Require: Local state $\mathbf{p}_i, \mathbf{v}_{i,\mathrm{nom}}$, previous hop-level $l_i(t-\Delta t)$, and neighbor information from $\mathcal{N}_i(t)$.
Update hop-level $l_i(t)$ via (9).
Determine minimum neighbor hop-level $l_i^{\min}(t)$.
if $l_i^{\min}(t) < l_i(t)$ **then** {Case 1: Gradient connection available}
  Identify potential gradient neighbors $\mathcal{N}_i^g(t)$.
  Select primary gradient neighbor: $g_i(t) \leftarrow \arg\min_{k\in\mathcal{N}_i^g(t)} \mathrm{cost}_{ik}(t)$.
  Identify anchor neighbors $\mathcal{N}_i^{b'}(t)$ evaluating $\mathrm{cost}_{j,g_i(t)}(t) < \mathrm{cost}_{i,g_i(t)}(t)$.
  Construct candidate set: $\mathcal{N}_i^b(t) \leftarrow \mathcal{N}_i^{b'}(t) \cup \{g_i(t)\}$.
  Finalize active selection: $C_i(t) \leftarrow \left\{\arg\min_{j\in\mathcal{N}_i^b(t)} \mathrm{cost}_{ij}(t)\right\}$.
else if $l_i^{\min}(t) = l_i(t)$ **then** {Case 2: Lateral connections}
  $C_i(t) \leftarrow \{j \in \mathcal{N}_i(t) \mid l_j(t) = l_i(t)\}$.
else {Case 3: Seed node or isolated}
  $C_i(t) \leftarrow \emptyset$.
end if
Broadcast active set $C_i(t)$ to all neighbors $j \in \mathcal{N}_i(t)$.
Receive selections from neighbors to update passive set $P_i(t)$.
return Comprehensive critical neighbor set $K_i^{\mathrm{all}}(t) = C_i(t) \cup P_i(t)$.

3) Communication Requirements: This algorithm requires each agent to broadcast its position, hop count, and nominal velocity. For the anchor neighbor selection in Case 1, 2-hop topology information is needed, which can be obtained if agents also broadcast their neighbor lists. This overhead is still significantly lower than methods requiring iterative estimation of global network properties.

IV. DISTRIBUTED SAFE CONTROL SYNTHESIS

Enforcing the CBF constraint for a critical link (i,j) imposes a linear inequality on the concatenated control vector $(\mathbf{v}_{i,\mathrm{c}}, \mathbf{v}_{j,\mathrm{c}})$. When agent i has multiple critical neighbors, its safe control set becomes dependent on the actions of all of them, leading to coupled constraints across the network. Resolving these couplings in a distributed manner is non-trivial.

To address this, we employ the distributed optimization algorithm from Tan et al. [17]. We first explicitly map the CBF constraints into the local QP format. For each critical link, indexed by $m = (i, j_m)$, the CBF inequality from (6) is:

$$\underbrace{2(\mathbf{p}_i - \mathbf{p}_{j_m})^\top}_{A_{i,m}} \mathbf{v}_{i,\mathrm{c}} + \underbrace{(-2(\mathbf{p}_i - \mathbf{p}_{j_m})^\top)}_{A_{j_m,m}} \mathbf{v}_{j_m,\mathrm{c}} \le \underbrace{\alpha h_{ij_m}(t)}_{b_m}. \tag{16}$$

This shared constraint, $A_{i,m}\mathbf{v}_{i,\mathrm{c}} + A_{j_m,m}\mathbf{v}_{j_m,\mathrm{c}} \le b_m$, is then decoupled using auxiliary variables, y_i^m, that dynamically allocate responsibility for satisfying the constraint. This allows each agent i to compute its control input by solving a local quadratic program (QP) at each time step:

$$\begin{aligned} \min_{\mathbf{v}_{i,\mathrm{c}}} \quad & \frac{1}{2}\|\mathbf{v}_{i,\mathrm{c}} - \mathbf{v}_{i,\mathrm{nom}}(t)\|^2 \\ \text{s.t.} \quad & A_{i,m}\mathbf{v}_{i,\mathrm{c}} + y_i^m(t) - y_{j_m}^m(t) \le b_m/2, \\ & \qquad \forall m \in \mathcal{M}_i(t) \\ & -v_{\max}\mathbf{1} \le \mathbf{v}_{i,\mathrm{c}} \le v_{\max}\mathbf{1}, \end{aligned} \tag{17}$$

where $\mathcal{M}_i(t)$ indexes the active critical links corresponding to the neighbors in $K_i^{\mathrm{all}}(t)$.

Symmetrically, agent j_m solves its own QP with the constraint $A_{j_m,m}\mathbf{v}_{j_m,\mathrm{c}} + y_{j_m}^m(t) - y_i^m(t) \le b_m/2$. The auxiliary variables y_i^m are updated based on the local Lagrange multipliers c_i^m and a constant gain $k_0 > 0$:

$$\dot{y}_i^m = -k_0 \cdot \mathrm{sign}\left(c_i^m(t) - c_{j_m}^m(t)\right), \tag{18}$$

where this update rule applies for all of agent i's active constraints, $m = (i, j_m) \in \mathcal{M}_i(t)$. As proven in [17], summing the two local constraints in (17) recovers the original coupled constraint (16), meaning the method is violation-free. This property is essential for guaranteeing system safety.

V. THEORETICAL ANALYSIS

Let $G_{\mathrm{crit}}(t) = (\mathcal{R}, E_{\mathrm{crit}}(t))$ denote the *critical subgraph*, where $E_{\mathrm{crit}}(t) = \{(i,j) : j \in K_i^{\mathrm{all}}(t)\}$ represents the minimal backbone of critical connections preserved by CBF constraints.

Assumption 1: **(Initial Connectivity)** At the initial time $t = 0$, the underlying communication graph $G(0)$ is connected.

Lemma 1: **(Hop Count Convergence)** Given an initially connected graph and a uniquely designated seed agent s_0 with $l_{s_0}(t) \equiv 0$, the update mechanism in (9) guarantees that every agent $i \in \mathcal{R}$ converges to a finite hop count level $l_i(t) < N$ within finite time steps, provided the topology changes slower than the communication rate.

Observation 1: **(Critical Neighbor Selection Properties)** The selection algorithm ensures that for any non-seed agent $i \ne s_0$ with a finite $l_i(t)$: (1) In Case 1, agent i selects exactly one neighbor $m^* \in C_i(t)$ strictly satisfying $l_{m^*}(t) \le l_i(t)$; (2) In Case 2, agent i selects all neighbors sharing the same hop-level.

Theorem 1: **(Connectivity Maintenance)** Under Assumption 1, if the distributed QP controllers successfully render the safe set forward invariant for all critical connections $(i,j) \in E_{\mathrm{crit}}(t)$, then the critical subgraph $G_{\mathrm{crit}}(t)$, and consequently the global network $G(t)$, remains connected for all $t \ge 0$.

Proof: The proof proceeds by strong induction on the hop count level $k = l_i(t)$. Based on Lemma 1, we show that any agent i with $l_i(t) < N$ maintains a continuous path to the seed s_0 strictly within $G_{\mathrm{crit}}(t)$.

Base Case ($k = 1$): Any agent i with $l_i(t) = 1$ must have the seed s_0 (with $l_{s_0}(t) = 0$) in its neighbor set. According to the algorithm (Case 1), i selects a critical neighbor m^*

that provides a path towards a lower hop level. Since s_0 is the only agent at level 0, a critical path (i, s_0) is instantiated in $G_{\text{crit}}(t)$.

Inductive Step: Assume that any agent j with $l_j(t) < k$ has a verified path to s_0 in $G_{\text{crit}}(t)$. Consider an agent i with $l_i(t) = k$.

- In *Case 1*, agent i selects a neighbor m^* where $l_{m^*}(t) \leq k$. If $l_{m^*}(t) < k$, the critical edge (i, m^*) connects i to an agent that has a path to s_0 by the induction hypothesis. If $l_{m^*}(t) = k$, m^* must be an anchor neighbor which, by definition, is structurally connected to a gradient agent $g_i(t)$ with $l_{g_i(t)}(t) < k$. In both subcases, the proactive selection enforces a critical edge that connects agent i to an agent at a lower hop level, which inductively connects to s_0.
- In *Case 2*, agent i connects to all available neighbors in the set $\mathcal{V}_k = \{j \mid l_j(t) = k\}$, ensuring the subgraph of $G_{\text{crit}}(t)$ induced by $\mathcal{V}_k$ is tightly coupled. Since $G(t)$ is connected, there exists at least one edge (u, v) such that $u \in \mathcal{V}_k$ and $l_v(t) = k - 1$. Agent u, triggering Case 1, forms a critical link toward this lower level. Consequently, all agents in $\mathcal{V}_k$ (including i) possess a path to u in $G_{\text{crit}}(t)$, thus inheriting the path to the $k-1$ level, and ultimately to s_0.

By induction, every agent converges to a state where it possesses a path to s_0 within $G_{\text{crit}}(t)$. Because the cooperative QP formulation continuously enforces the CBF constraints to strictly preserve all edges in $E_{\text{crit}}(t)$, the connectivity of $G_{\text{crit}}(t)$ is mathematically guaranteed for all $t \geq 0$. ■

The algorithm has complexity $O(D)$ for hop count estimation and $O(|\mathcal{N}_i(t)|^2)$ per agent for neighbor selection.

VI. SIMULATION RESULTS

The proposed framework is evaluated through a series of batch simulations. These simulations are designed to benchmark our method against state-of-the-art approaches and to conduct targeted ablation studies that quantify the distinct benefits of our two primary contributions: the dynamics-aware connection cost metric and the cooperative distributed optimization scheme.

A. Baseline Methods

Our proposed method, denoted as CBF-CN, is compared against three baselines to rigorously evaluate its performance and structural components:

1) **ACE-based [18]:** A state-of-the-art decentralized algebraic connectivity estimation and maintenance method. It utilizes distributed power iteration to estimate the global algebraic connectivity λ_2, and restricts control actions via CBFs to ensure λ_2 remains above a predefined positive threshold.
2) **Distance-CN [16]:** A variant of our method that replaces our dynamics-aware connection cost with a purely geometric heuristic, selecting the critical neighbor based solely on minimum Euclidean distance. This baseline isolates and validates the benefit of our proposed predictive cost metric.
3) **Decoupled CBF-CN (Ablation):** An ablation study that retains the cost-based neighbor selection but disables the cooperative distributed optimization. Each agent solves its local quadratic program (QP) greedily without the auxiliary variable updates. This highlights the critical role of our coordination scheme in mitigating conflicting actions.

B. Simulation Setup

To ensure rigorous evaluation and statistical significance, we conducted batch simulations over multiple independent random network topologies. For each combination of algorithm and swarm size, where $N \in \{6, 8, 12, 16\}$, the experiment was repeated across 8 distinct random seeds.

All validations were performed in a 2D environment for a duration of $100.0\,\text{s}$. Agents were initially deployed at random positions within a $[-15.0, 15.0] \times [-15.0, 15.0]\,\text{m}$ region. To evaluate the proposed framework in a multi-agent target-reaching task, unique target positions were assigned using a circular distribution within a larger bounding box of $[-20.0, 20.0] \times [-20.0, 20.0]\,\text{m}$. This setup deliberately forces agents to disperse radially, creating a severe spatial conflict between the primary navigation task and the safety-critical necessity to maintain the communication network.

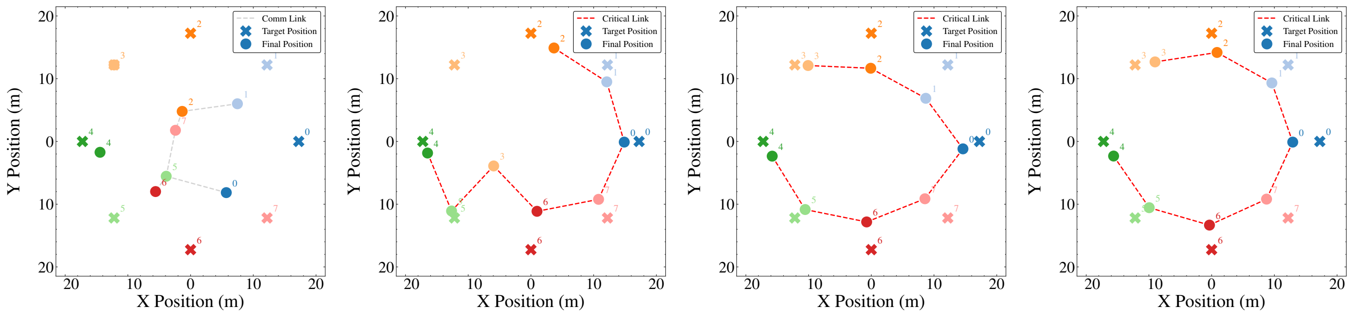
The maximum communication range was defined as $D_{\max} = 12.0\,\text{m}$, with a desired connectivity buffer parameter $d_{\text{conn}} = 10.0\,\text{m}$. The nominal controller gain was set to 0.5, and our proposed connection cost weighting factor was $\alpha_{\text{cost}} = 1.0$.

A fundamental distinction between the evaluated methods lies in their computational frequency and allowable velocity limits. For our critical neighbor approaches, namely CBF-CN, Distance-CN, and Decoupled-CN, the control loop operated efficiently at $\Delta t = 0.1\,\text{s}$, and the agents were permitted a maximum velocity of $v_{\max} = 2.0\,\text{m/s}$. In contrast, the ACE-based method necessitated a significantly finer control timestep of $0.01\,\text{s}$ and an estimator timestep of $0.001\,\text{s}$ to ensure the convergence of the global eigenvalue estimation. Furthermore, to prevent the iterative estimator from diverging due to rapid topological changes, the maximum velocity for ACE agents had to be strictly capped at a highly conservative $v_{\max} = 0.1\,\text{m/s}$. The minimum required algebraic connectivity for ACE was set to $\lambda_{\min} = 0.1$.

C. Results and Discussion

Table I summarizes the statistical outcomes of the Monte Carlo simulations. The primary metrics evaluated are the average distance to the final targets, which indicates task performance, and the minimum algebraic connectivity $(\min \lambda_2)$ observed during the full $100\,\text{s}$ duration.

1) Connectivity Maintenance and Conservatism: The empirical results demonstrate a significant difference in connectivity maintenance among the evaluated algorithms. While our proposed framework (CBF-CN) and the local heuristic baselines (Distance-CN, Decoupled-CN) successfully maintained the physical connectivity of the network (100% Success Rate) across all swarm sizes and random initializations,



(a) ACE-based (b) Distance-CN (c) Decoupled CBF-CN (d) CBF-CN (Ours)

Fig. 1: Final agent positions (circles) and targets (crosses) for a representative 8-agent task. Our proposed method (d) enables agents to achieve a significantly smaller average final target distance, demonstrating superior task execution flexibility while strictly maintaining global network connectivity.

TABLE I: Quantitative Performance Comparison Across Varying Swarm Sizes (Averaged over 8 Random Seeds)

Swarm	Method	Avg. Target Dist. (m) ↓	Success Rate (%) ↑
$N = 6$	ACE-based	6.12	75.0
	Distance-CN	3.20	100.0
	Decoupled-CN	3.78	100.0
	CBF-CN (Ours)	**2.80**	**100.0**
$N = 8$	ACE-based	10.00	25.0
	Distance-CN	5.63	100.0
	Decoupled-CN	5.01	100.0
	CBF-CN (Ours)	**4.29**	**100.0**
$N = 12$	ACE-based	17.12	37.5
	Distance-CN	10.55	100.0
	Decoupled-CN	9.45	100.0
	CBF-CN (Ours)	**8.06**	**100.0**
$N = 16$	ACE-based	25.01	87.5
	Distance-CN	14.70	100.0
	Decoupled-CN	13.35	100.0
	CBF-CN (Ours)	**10.87**	**100.0**

Note: ↓ lower is better, ↑ higher is better.

the global ACE-based method exhibited significant failures. As detailed in Table I, the success rate of the ACE method dropped significantly to 25.0% and 37.5% for 8-agent and 12-agent swarms, respectively.

As visualized in the Fig. 2a, our CBF-CN framework seamlessly maintains connectivity without suffering structural fragmentation. In contrast, the ACE-based method exhibits poor robustness in this highly dynamic scenario. Because it relies on a global estimation of λ_2 that is vulnerable under rapid topology changes, it frequently fails to maintain network connectivity despite imposing the severe velocity limitation (0.1 m/s) detailed in Section VI-B. Consequently, it not only suffers from network fractures but also largely fails the primary mission, leaving agents stranded far from their intended goals (reflected by the large final target distances).

2) Task Performance and Scalability: By reformulating the global connectivity requirement into local, computationally tractable constraints via critical neighbor selection, our CBF-CN framework enables agents to operate at their full dynamic capabilities (2.0 m/s) without fracturing the network.

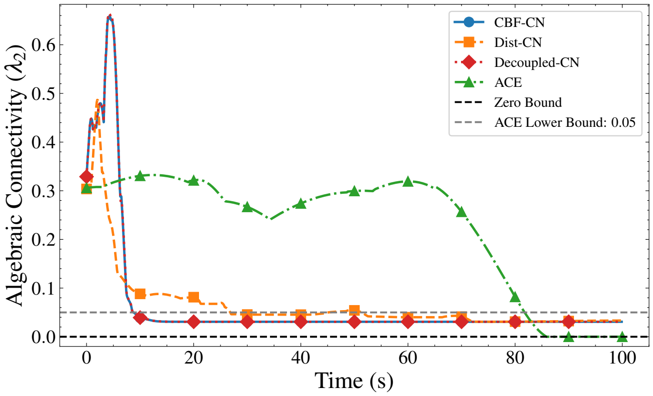
As evidenced in Table I, the proposed CBF-CN method consistently achieves the lowest average final target distance across all scenarios. This quantitative superiority is further supported by the final spatial configurations shown in Fig. 1. For instance, in the 8-agent task (Fig. 1d), agents utilizing our framework successfully navigate to locations much closer to their respective radial goals compared to the baselines. The temporal evolution of this task execution is further illustrated in the Fig. 2b, where our method demonstrates a faster and deeper convergence toward the targets.

This performance gap underscores the specific value of our two main contributions. First, compared to the purely geometric Distance-CN baseline, our dynamics-aware connection cost metric proactively selects dynamically stable links, providing the team with greater flexibility to disperse. Second, the outperformance over the Decoupled-CN baseline validates the necessity of the cooperative distributed optimization scheme. By dynamically sharing the responsibility of the coupled CBF constraints, agents effectively avoid greedy, conflicting maneuvers that otherwise degrade team-wide progress.

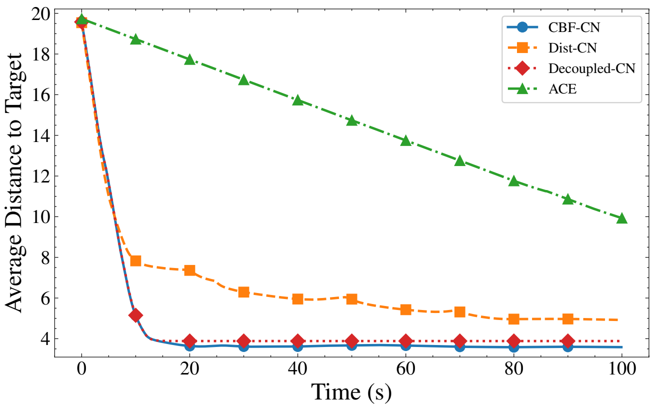
Finally, Fig. 3 visually reinforces the scalability data from Table I. As the team size increases up to $N = 16$, the computational and communication bottlenecks of global-metric methods like ACE force them into severe performance degradation. Our distributed CBF-CN framework, conversely, scales highly favorably, maintaining robust and superior target-reaching performance regardless of the expanding network scale.

VII. CONCLUSION

This paper introduced a distributed framework that ensures persistent connectivity by shifting the paradigm of neighbor selection from static geometry to dynamic prediction. Our core contribution, a novel dynamics-aware connection cost metric, allows agents to form a dynamically stable communication topology that minimally interferes with task execution. The integration with CBFs and a cooperative optimization scheme provides formal safety guarantees in a scalable manner. Experimental results confirmed that this predictive approach not only ensures network integrity under



(a) Algebraic connectivity over time.



(b) Average distance to target over time.

Fig. 2: Performance comparison for an 8-agent system. (Top) Algebraic connectivity (λ_2) over time. A value above $\lambda_2 = 0$ indicates the network remains connected. (Bottom) Average distance to target over time. The steeper descent and lower final value of our CBF-CN method indicate superior task performance.

challenging conditions but also yields improved task performance.

While our method excels in local, real-time decision-making, its decentralized nature may lead to globally sub-optimal network configurations or potential deadlocks. In contrast, approaches based on global metrics like algebraic connectivity, despite their practical limitations, can offer a better perspective on the overall network topology. Therefore, a significant avenue for future research is the development of a hybrid switching control strategy. This would leverage our efficient local controller for nominal, real-time operation, while strategically invoking a global topology optimization mechanism to escape local minima, thus combining the strengths of both distributed reactivity and global oversight.

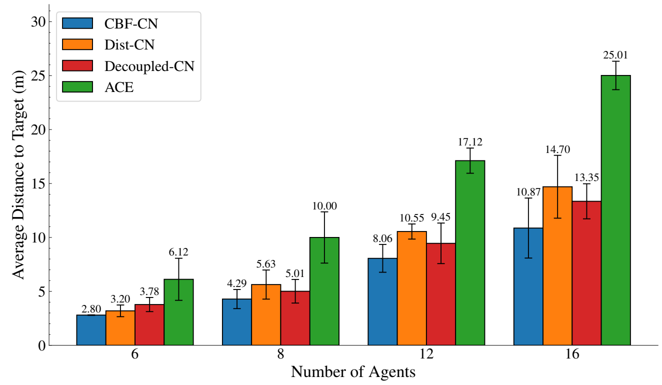


Fig. 3: Average final distance to target across multiple runs for different agent counts (N). The ACE-based method is omitted from this plot for visual clarity, as its final distances significantly exceed the scale of the other methods (refer to Table I for exact values).

REFERENCES

[1] M. Calvo-Fullana, M. Gerasimenko, D. Mox, L. Agorio, M. d. Castillo, V. Kumar, A. Ribeiro, and J. A. Bazerque, "A networked multiagent system for mobile wireless infrastructure on demand," *IEEE Transactions on Robotics*, vol. 40, pp. 4598–4614, 2024.

[2] E. Sebastián, E. Montijano, and C. Sagüés, "Adaptive multirobot implicit control of heterogeneous herds," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3622–3635, 2022.

[3] Y. Tian, K. Liu, K. Ok, L. Tran, D. Allen, N. Roy, and J. P. How, "Search and rescue under the forest canopy using multiple uavs," *The International Journal of Robotics Research*, vol. 39, no. 10-11, pp. 1201–1221, 2020.

[4] Y. Kim and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph laplacian," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 116–120, 2006.

[5] D. Mox, V. Kumar, and A. Ribeiro, "Learning connectivity-maximizing network configurations," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5552–5559, 2022.

[6] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," *Proceedings of the IEEE*, vol. 99, no. 9, pp. 1525–1540, 2011.

[7] E. Restrepo, A. Loría, I. Sarras, and J. Marzat, "Edge-based strict lyapunov functions for consensus with connectivity preservation over directed graphs," *Automatica*, vol. 132, p. 109812, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109821003320

[8] L. Sabattini, C. Secchi, N. Chopra, and A. Gasparri, "Distributed control of multirobot systems with global connectivity maintenance," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1326–1332, 2013.

[9] L. Sabattini, N. Chopra, and C. Secchi, "Decentralized connectivity maintenance for cooperative control of mobile robotic systems," *The International Journal of Robotics Research*, vol. 32, no. 12, pp. 1411–1423, 2013.

[10] H. Fang, Y. Wei, J. Chen, and B. Xin, "Flocking of second-order multiagent systems with connectivity preservation based on algebraic connectivity estimation," *IEEE Transactions on Cybernetics*, vol. 47, no. 4, pp. 1067–1077, 2017.

[11] B. Capelli and L. Sabattini, "Connectivity maintenance: Global and optimized approach through control barrier functions," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 5590–5596.

[12] P. Yang, R. Freeman, G. Gordon, K. Lynch, S. Srinivasa, and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, no. 2, pp. 390–396, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109809005366

[13] I. Malli, C. P. Bechlioulis, and K. J. Kyriakopoulos, "Robust distributed estimation of the algebraic connectivity for networked multi-robot systems," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 9155–9160.

[14] Y. Zhang, S. Li, and J. Weng, "Distributed estimation of algebraic connectivity," *IEEE Transactions on Cybernetics*, vol. 52, no. 5, pp. 3047–3056, 2022.

[15] P. Ong, B. Capelli, L. Sabattini, and J. Cortés, "Nonsmooth control barrier function design of continuous

constraints for network connectivity maintenance," *Automatica*, vol. 156, p. 111209, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109823003709

[16] X. Li, R. Zhou, G. Sun, and J. Zhang, "Connectivity-preserving flocking of multiagent systems via selecting critical neighbors," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 6, pp. 3779–3792, 2023.

[17] X. Tan, C. Liu, K. H. Johansson, and D. V. Dimarogonas, "A continuous-time violation-free multiagent optimization algorithm and its applications to safe distributed control," *IEEE Transactions on Automatic Control*, vol. 70, no. 8, pp. 5114–5128, 2025.

[18] N. De Carli, P. Salaris, and P. R. Giordano, "Distributed control barrier functions for global connectivity maintenance," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 12 048–12 054.