Controllability Backbone in Networks

Obaid Ullah Ahmad, Mudassir Shabbir, and Waseem Abbas

Abstract— This paper studies the controllability backbone problem in dynamical networks defined over graphs. The main idea of the controllability backbone is to identify a small subset of edges in a given network such that any subnetwork containing those edges/links has at least the same network controllability as the original network while assuming the same set of input/leader vertices. We consider the strong structural controllability (SSC) in our work, which is useful but computationally challenging. Thus, we utilize two lower bounds on the network's SSC based on the zero forcing notion and graph distances. We provide algorithms to compute controllability backbones while preserving these lower bounds. We thoroughly analyze the proposed algorithms and compute the number of edges in the controllability backbones. Finally, we compare and numerically evaluate our methods on random graphs.

Index Terms— Strong structural controllability, network control, zero forcing, graph distances.

I. INTRODUCTION

Network structure profoundly influences the dynamical behavior of networked multiagent systems. For instance, network controllability, connectivity, robustness to failures, information dissemination, and influence evolution in networks rely on the underlying network topology [1]. Therefore, any changes to the network's structural organization, such as adding or removing links between agents, may alter the system-level properties of the network, which could be either beneficial or detrimental. Thus, for a survivable network design and avoid the deterioration in the desired network behavior, a practical approach is to identify a sparse subnetwork (or backbone) whose maintenance would guarantee the preservation of the desired network property in the face of modifications. For example, to maintain connectivity, preserving edges in the minimum spanning tree ensures a path between every pair of agents. Similarly, in communication infrastructure networks, connected dominating sets are used to identify the minimum number of agents necessary to form the backbone network [2].

This paper studies the *controllability backbone* problem in a networked dynamical system defined over a graph $G = (V, E)$. Network controllability concerns the ability to manipulate the agents within a network as desired through external control signals injected via a subset of agents called *input agents* or *leaders*. The network controllability depends on the choice of leaders $V_\ell \subseteq V$ and the interconnections between agents [3]. Moreover, the network controllability may deteriorate if the connections/edges between agents change [4], [5]. The main idea of the controllability backbone is to determine a small subset of edges $E_B \subseteq E$ such that *any* subnetwork of G containing E_B has at least the same network controllability as G with the same leaders. In other words, maintaining E_B implies that the minimum network controllability is preserved despite edge modifications.

We consider the *strong structural controllability (SSC)* for the backbone problem. SSC is advantageous as it depends on the edge set E and not on the edge weights (which represent the coupling strengths between vertices and often are not precisely known). However, determining the SSC of a network is a challenging computational problem [5]–[7]. So a typical approach is to obtain tight lower bounds. Therefore, we aim to identify a controllability backbone for a given network $G = (V, E)$ and leader set V_ℓ, where the backbone preserves a tight lower bound on the network's SSC. As for the SSC lower bounds, we consider two widely used bounds based on the zero forcing sets and distances in graphs [8]–[10]. Our main contributions are as follows:

1) We present a novel approach to identifying a sparse subgraph in a graph that guarantees the same level of controllability (SSC) as the original graph. We call this subgraph the *controllability backbone* (Section II).
2) We provide a polynomial algorithm to compute a minimum controllability backbone, which preserves a lower bound on the network's SSC based on zero forcing sets in graphs (Section III).
3) Additionally, we consider a distance-based lower bound on SSC and compute a controllability backbone preserving the distance bound. We derive tight bounds on the number of edges in the distance-based backbone (Section IV).
4) Finally, we illustrate our results and compare different controllability backbones (Section V).

There are previous works dealing with the densification problem, i.e., how can we add edges to a graph while maintaining its controllability (e.g., [11], [12])? In contrast, this paper studies an inverse, i.e., the sparsification problem, to identify a small subset of crucial edges whose existence within any subgraph guarantees the same controllability as the original graph. While some studies have considered identifying edges whose removal from the graph does not deteriorate the network controllability of the remaining graph (e.g., [13]–[15]), our problem setup is distinct. We require

that *any* subgraph containing the backbone edges be at least as controllable as the original graph, resulting in a more general problem formulation. Furthermore, our formulation considers the concept of strong structural controllability, which adds to its generality.

The rest of the paper is organized as follows: Section II introduces preliminaries and sets up the controllability backbone problem. Section III reviews the zero forcing ideas and then studies the zero forcing-based controllability backbone problem. Section IV describes the distance-based bound on network SSC and then employs it to compute the distance-based backbone. Section V compares the two controllability backbones and numerically evaluate the proposed methods. Finally, Section VI concludes the paper.

II. PRELIMINARIES AND PROBLEM FORMULATION

A. Notations and System

An undirected graph $G = (V, E)$ models a multiagent network. The vertex set V, and the edge set $E \subset V \times V$ represent agents and interactions between them, respectively. The edge between vertices u and v is denoted by an unordered pair (u, v). The *neighborhood* of u in graph G is the set $\mathcal{N}_G(u) = \{v \in V : (u, v) \in E\}$ and the *degree* of u is $\deg(u) = |\mathcal{N}_G(u)|$. A *path* P in a graph G is defined as a sequence of vertices $(v_1, v_2, v_3, \cdots, v_k)$, where $v_1, v_2, v_3, \cdots, v_k$ are distinct vertices in the graph, and for every i from 1 to $k-1$, there exists an edge between v_i and $v_i + 1$. The *distance* between vertices u and v, denoted by $d(u, v)$, is the number of edges in the shortest path between u and v. A graph $\hat{G} = (V, \hat{E})$ is a *subgraph* of $G = (V, E)$, denoted by $\hat{G} \subseteq G$, if $\hat{E} \subseteq E$, and G will be a super graph of $\hat{G}$.

We consider a network of n agents, denoted by $V = \{v_1, v_2, \cdots, v_n\}$, of which m are input/leader vertices, which are represented by $V_\ell = \{\ell_1, \ell_2, \cdots, \ell_m\} \subseteq V$, and the rest are followers. We consider the following liner time-invariant system on G.

$$\dot{x}(t) = Mx(t) + Hu(t). \tag{1}$$

Here, $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^m$ is the external input injected into the system through m leaders. $M \in \mathcal{M}(G)$ is the system matrix, where $\mathcal{M}(G)$ is a family of symmetric matrices associated with G defined as:

$$\mathcal{M}(G) = \{M \in \mathbb{R}^{n \times n} : M = M^\top, \text{ and for } i \neq j, \\ M_{ij} \neq 0 \Leftrightarrow (i, j) \in E(G)\}. \tag{2}$$

The matrix $H \in \mathbb{R}^{n \times m}$ in (1) is the input matrix, such that $H_{ij} = 1$, if $v_i = \ell_j$; and 0 otherwise. We note that the input matrix H is defined by the selection of leader agents. Here, $\mathcal{M}(G)$ denotes a broad class of system matrices defined on graphs, including the adjacency, Laplacian, and signless Laplacian matrices.

B. Strong Structurally Controllable Networks

The system (1) is *controllable* if there exists an input $u(t)$ that can drive the system from an arbitrary initial state $x(t_0)$

to any desired state $x(t_f)$ in a finite amount of time. If the system is controllable for a given system and input matrices, we say that (M, H) is a *controllable pair*. Moreover, (M, H) is a controllable pair if and only if the controllability matrix $\mathcal{C}(M, H) \in \mathbb{R}^{n \times nm}$ is full rank, i.e., $\mathrm{rank}(\mathcal{C}(M, H)) = n$. The controllability matrix is defined as:

$$\mathcal{C}(M, H) = \begin{bmatrix} H & MH & M^2H & \cdots & M^{n-1}H \end{bmatrix}. \tag{3}$$

Definition (*Strong Structural Controllability (SSC)*) A graph $G = (V, E)$ with a given set of leaders $V_\ell \subseteq V$ (and the corresponding H matrix) is *strong structurally controllable* if and only if (M, H) is a controllable pair *for all* $M \in \mathcal{M}(G)$.

If the network G is strong structurally controllable for a given set of leaders, then the rank of the controllability matrix does not depend on the edge weights (as long as they satisfy (2)). For the rest of the paper, we refer to strong structural controllability simply as *controllability*. The dimension of strong structurally controllable subspace, denoted by $\gamma(G, V_\ell)$, is the smallest possible rank of the controllability matrix under feasible weights.

Definition (*Dimension of SSC*) For a fixed leader set V_ℓ, the dimension of strong structurally controllable subspace, denoted by $\gamma(G, V_\ell)$, is the smallest possible rank of the controllability matrix over all $M \in \mathcal{M}(G)$, i.e.,

$$\gamma(G, V_\ell) = \min_{M \in \mathcal{M}(G)} (\mathrm{rank}(\mathcal{C}(M, H))). \tag{4}$$

$\gamma(G, V_\ell)$ quantifies 'how much' of the network G can always be controlled through the leaders V_ℓ.

C. Controllability Backbone Problem

We are interested in identifying a small subset of edges among vertices within a network that would maintain its strong structural controllability in its subnetworks. This entails identifying the sparsest subgraph, referred to as the *controllability backbone*, that guarantees at least the same level of controllability as the original network in any subnetwork that encompasses the controllability backbone. In essence, the controllability backbone represents the minimum structure that must be preserved within the network to ensure its minimum controllability despite structural perturbations.

Definition (*Controllability Backbone B*) For a given $G = (V, E)$ and leaders $V_\ell \subseteq V$, the controllability backbone (or simply *backbone*) $B = (V, E_B)$, is a subgraph of G with $E_B \subseteq E$, such that any subgraph $\hat{G} = (V, \hat{E})$ containing E_B, i.e., $E_B \subseteq \hat{E} \subseteq E$ satisfies

$$\gamma(\hat{G}, V_\ell) \geq \gamma(G, V_\ell). \tag{5}$$

In other words, any subgraph $\hat{G} = (V, \hat{E})$ of G containing backbone edges E_B, has at least the same controllability as G. Thus, preserving backbone edges guarantees that controllability does not deteriorate in a subgraph $\hat{G}$. A backbone with the minimum edge set is referred to as the *minimum backbone graph* $B^* = (V, E^*)$. We aim to compute B^*.

Problem 1: Given a graph $G = (V, E)$ and a leader set V_ℓ, find the minimum controllability backbone graph.

Figure 1 illustrates the idea of a controllability backbone. For a given G and $V_\ell = \{v_4, v_6, v_7\}$, the dimension of SSC is $\gamma(G, V_\ell) = 8$. A minimum backbone B^* is shown in Figure 1(b). Any subgraph $\hat{G}$ (of G) containing B^* also has $\gamma(\hat{G}, V_\ell) = 8$.
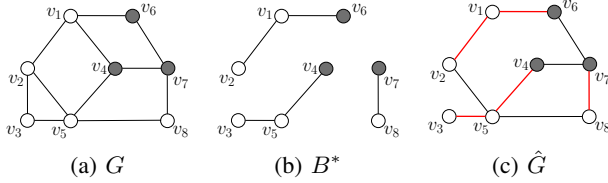

(a) G (b) B^* (c) $\hat{G}$

Fig. 1: (a) A graph G. (b) A minimum controllability backbone B^* of G. (c) A subgraph $\hat{G}$ of G containing the backbone (red edges).

The minimum backbone problem relies on the computation $\gamma(G, V_\ell)$ (as in (4)), which is a computationally arduous task. To address this challenge, it is common to compute tight lower bounds on $\gamma(G, V_\ell)$ instead of $\gamma(G, V_\ell)$ when dealing with SSC-related problems. As a result, we also modify the controllability backbone problem and focus on obtaining a sparse subgraph of a given G whose existence within any subgraph $\hat{G} \subseteq G$ guarantees that $\hat{G}$ has the same or greater value of lower bounds on the dimension of SSC as G. To accomplish this, we consider two widely used lower bounds, including (1) a zero forcing set-based bound and (2) a bound based on the distances between vertices. In the forthcoming sections, we will elaborate on these bounds and their application to the controllability backbone problem.

III. Zero forcing for Controllability backbone

Zero forcing is a rule-based coloring of vertices in a graph. The main idea is to initiate the coloring process with a small subset of initially colored vertices which eventually color other vertices based on some rules. Zero forcing has several network applications and provides a tight lower bound on the network's SSC, as we explain below [8].

A. Zero Forcing-based Lower Bound on SSC

First, we define the zero forcing process and related terms and then explain the SSC bound based on the zero forcing phenomenon.

Definition (*Zero forcing (ZF) Process*) Consider a graph $G = (V, E)$, such that each $v \in V$ is colored either BLACK or WHITE initially. The ZF process is to iteratively change the color of WHITE vertices to BLACK using the following rule until no further color changes are possible.

Color change rule: If $v \in V$ is colored BLACK and has exactly one WHITE neighbor u, change the color of u to BLACK.

We say that v *infected* u if the color of WHITE vertex u is changed to BLACK by some BLACK vertex v.

Definition (*Derived Set*) Consider a graph $G = (V, E)$ with $V_\ell \subseteq V$ as the set of initial BLACK vertices. Then, the set of BLACK vertices obtained at the end of the ZF process is the derived set [16], denoted by $dset(G, V_\ell)$, and $|dset(G, V_\ell)| = \zeta(G, V_\ell)$. When the context is clear, we will drop the parameter V_ℓ.

The set of initial BLACK vertices V_ℓ is also referred to as the *input or leader set*. For a given V_ℓ, $dset(G, V_\ell)$ is unique [16]. Now, we define the zero forcing set.

Definition (*Zero Forcing Set (ZFS)*) For a graph $G = (V, E)$, $V_\ell \subseteq V$ is a ZFS if and only if $dset(G, V_\ell) = V$. We denote a ZFS of G by $Z(G)$.

Figure 2 illustrates zero forcing through a set of input vertices and the corresponding derived set.
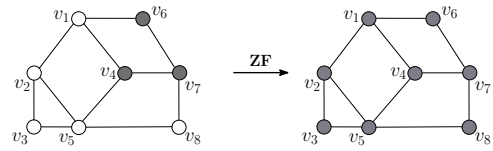

Fig. 2: $V_\ell = \{v_4, v_6, v_7\}$ is the input set. After the ZF process, $dset(G, V_\ell) = V$, as indicated by the black vertices. Hence, V_ℓ is a ZFS.

The zero forcing phenomenon is significant in characterizing the network's SSC [8], [17], [18]. In particular, the size of the derived set for a given set of input vertices provides a lower bound on the dimension of SSC.

Theorem 3.1: [19] For any network $G = (V, E)$ with the leaders $V_\ell \subseteq V$,

$$\zeta(G, V_\ell) \leq \gamma(G, V_\ell),$$

where $\zeta(G, V_\ell)$ is the size of the derived set with V_ℓ as input vertices, and $\gamma(G, V_\ell)$ is the dimension of SSC (as in (4)).

Proof: Proof follows from Lemma 4.2 in [19], which shows that for a set of state matrices, the controllable subspace always contains a $|dset(G, V_\ell)|$-dimensional subspace. ∎

B. ZFS-based Backbone

We are interested in finding a controllability backbone that will maintain the zero forcing bound $\zeta(G, V_\ell)$ for a given leader set V_ℓ. The idea is to identify a subset of edges E_{B_z} in a given $G = (V, E)$ with a leader set V_ℓ such that the ZFS-based controllability bound in *any* subgraph of $\hat{G} = (V, \hat{E})$ containing those edges, (i.e., $E_{B_z} \subseteq \hat{E}$) is preserved. We formally define the ZFS-based backbone as follows:

Definition (*ZFS-based Backbone*) Given a graph $G = (V, E)$ and a leader set V_ℓ, the ZFS-based backbone is a subgraph $B_z = (V, E_{B_z})$, such that any subgraph $\hat{G} = (V, \hat{E})$, where $E_{B_z} \subseteq \hat{E} \subseteq E$ satisfies the following:

$$\zeta(\hat{G}, V_\ell) \geq \zeta(G, V_\ell).$$

Thus, the dimension of SSC in any subgraph of G containing the ZFS-based backbone is at least $\zeta(G, V_\ell)$, or in

other words, $\gamma(\hat{G}, V_\ell) \geq \zeta(G, V_\ell)$. Our goal is to find the ZFS-based backbone with the minimum number of edges.

Problem 2: Given a graph $G = (V, E)$ and a leader set V_ℓ, find a minimum ZFS-based backbone.

In [8], authors show that a leader set V_ℓ renders the network strong structurally controllable if and only if V_ℓ is a zero forcing set (ZFS) of the network graph G. Thus, the ZFS-based backbone is essentially the controllability backbone (as defined in Section II-C) if V_ℓ is a ZFS of G. Algorithm 1 solves Problem 2 and computes a minimum ZFS-based backbone. The main idea is to run the ZF process and iteratively select an edge through which some BLACK vertex colors its WHITE neighbor (thus, increasing the size of the derived set).

Algorithm 1 Computing ZF-based controllability backbone

Input: G, V_ℓ
Output: ZFS-based backbone $B_z = (V, E_{B_z})$
1: **Initialize:** $E_{B_z} \leftarrow \emptyset$, $dset(G, V_\ell) \leftarrow V_\ell$ (The set of initial BLACK vertices)
2: **while** there exits a BLACK vertex v with exactly one WHITE neighbor u **do**
3:     $dset(G, V_\ell) \leftarrow dset(G, V_\ell) \cup \{u\}$
4:     $E_{B_z} \leftarrow E_{B_z} \cup \{(u, v)\}$
5: **end while**

Theorem 3.2: Consider a graph $G = (V, E)$ and a leader set $V_\ell \subseteq V$, where $|V| = n$ and $|V_\ell| = m$. Algorithm 1 returns a minimum ZFS-based backbone $B_z = (V, E_{B_z})$ in $O(n^2)$ time. Moreover, $|E_{B_z}| = \zeta(G, V_\ell) - m$.

    Proof: Available in [20]. ∎

An example illustrating implementation of Algorithm 1 is presented in [20].

We note that when V_ℓ is a ZFS of G (i.e., $dset(G, V_\ell) = V$), then $\zeta(G, V_\ell) = \gamma(G, V_\ell) = |V|$, implying that a minimum ZFS-based backbone is also a minimum controllability backbone (as in Problem 1). However, when the leader set V_ℓ is not a ZFS, then the distance-based bound on the dimension of SSC is typically better than the ZFS-based bound [18]. Next, we discuss the distance-based bound and apply it to the controllability backbone problem.

IV. GRAPH DISTANCES FOR CONTROLLABILITY BACKBONE

In this section, we design a controllability backbone using a bound on the network SSC based on the graph distances between vertices in the underlying network graph. First, we introduce the distance-based bound on the dimension of SSC [9]. We then frame the notion of the distance-based controllability backbone and provide an algorithm to compute such a backbone. In Section V, we compare the ZFS-based and distance-based backbones.

A. Distance-based Lower Bound

Assuming m leaders $V_\ell = \{\ell_1, \ell_2, \cdots, \ell_m\}$ in a leader-follower network $G = (V, E)$, we define the *distance-to-leader (DL) vector* for each $v_i \in V$ as

$$D_i = \begin{bmatrix} d(\ell_1, v_i) & d(\ell_2, v_i) & \cdots & d(\ell_m, v_i) \end{bmatrix}^T \in \mathbb{Z}^m.$$

The jth component of D_i, denoted by $[D_i]_j$, is $d(\ell_j, v_i)$, i.e., the distance between leader ℓ_j and vertex v_i. Figure 3 shows DL vectors of vertices in a graph G with leaders $V_\ell = \{v_4, v_6\}$. Next, we define a *sequence* of distance-to-leader vectors, called *pseudo-monotonically increasing sequence* [9].

Definition (*Pseudo-monotonically Increasing Sequence (PMI)*) A sequence $\mathcal{D} = [\mathcal{D}_1 \ \mathcal{D}_2 \ \cdots \ \mathcal{D}_k]$ of distance-to-leader vectors is a PMI if for any vector $\mathcal{D}_i$ in the sequence, there is some coordinate $\pi(i) \in \{1, 2, \cdots, m\}$ such that

$$[\mathcal{D}_i]_{\pi(i)} < [\mathcal{D}_j]_{\pi(i)}, \ \forall j > i. \tag{6}$$

We say $[\mathcal{D}_i]_{\pi(i)}$ satisfies the PMI property at coordinate $\pi(i)$.

The PMI property (6) essentially gurantees that for each vector $\mathcal{D}_i$ in the PMI sequence, there is some index/coordinate $\pi(i)$ such that the values of all the subsequent vectors at the coordinate $\pi(i)$ are strictly greater than $[\mathcal{D}_i]_{\pi(i)}$. An example of PMI sequence of six vectors is shown in (7), where the coordinates of circled values are the ones where the PMI property is satisfied.

Next, we note that each vector in a PMI sequence is a DL vector of some vertex in the graph. However, multiple vertices can have the same DL vectors. For example, the DL vectors of v_2 and v_8 are the same. Thus, to explicitly specify the vertex whose DL vector appears in the PMI sequence, we introduce the *distance-to-leader* mapping.

Definition (*Distance-to-Leader Mapping (DLM)*) Let $\mathcal{D}$ be a PMI sequence. For each $\mathcal{D}_i \in \mathcal{D}$, a Distance-to-Leader Mapping (DLM), denoted by $f(\mathcal{D}_i)$, is a vertex whose DL vector is $\mathcal{D}_i$, i.e., $\mathcal{D}_i = D_{f(\mathcal{D}_i)}$.

To further clarify, note the following notations:
$\mathcal{D}_i$: i^{th} vector in the PMI sequence,
D_v: DL vector of vertex v.
Figure 3 illustrates these ideas. For the graph G in Figure 3 and $V_\ell = \{v_4, v_6\}$, a PMI sequence of length six can be constructed as

$$\mathcal{D} = \left[\begin{bmatrix} \textcircled{0} \\ 2 \end{bmatrix}, \begin{bmatrix} 2 \\ \textcircled{0} \end{bmatrix}, \begin{bmatrix} 1 \\ \textcircled{1} \end{bmatrix}, \begin{bmatrix} \textcircled{1} \\ 3 \end{bmatrix}, \begin{bmatrix} 2 \\ \textcircled{2} \end{bmatrix}, \begin{bmatrix} \textcircled{2} \\ 3 \end{bmatrix} \right]. \tag{7}$$

A PMI sequence of DL vectors is related to the network SSC. In fact, the length of PMI sequence provides a tight lower bound on the dimension of SSC $\gamma(G, V_\ell)$, as stated in the following result.

Theorem 4.1: [9] If $\delta(G, V_\ell)$ or simply $\delta(G)$, is the length of the longest PMI sequence of distance-to-leader vectors in a network $G = (V, E)$ with V_ℓ leaders, then

$$\delta(G, V_\ell) \leq \gamma(G, V_\ell), \tag{8}$$

where $\gamma(G, V_\ell)$ is the dimension of SSC (as in (4)).

B. Distance-based Backbone

Here, we will use the distance-based bound to formulate a controllability backbone problem. Then, we will provide
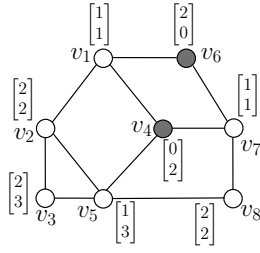
Fig. 3: A network with two leaders $V_\ell = \{\ell_1, \ell_2\} = \{v_4, v_6\}$, along with the DL vectors of vertices. A PMI sequence of length six is $\mathcal{D} = [\mathcal{D}_1 \; \mathcal{D}_2 \; \cdots \; D_6] = [D_{v_4} \; D_{v_6} \; D_{v_7} \; D_{v_5} \; D_{v_8} \; D_{v_3}]$.

Algorithm 2 Computing distance-based backbone
___
Input: G, V_ℓ, PMI sequence $\mathcal{D} = [\mathcal{D}_1 \; \mathcal{D}_2 \; \cdots \; \mathcal{D}_{\delta(G,V_\ell)}]$
Output: Distance-based backbone $B_d = (V, E_{B_d})$
1: **for** $i = 1$ to $\delta(G, V_\ell)$ **do**
2:     compute $f(\mathcal{D}_i)$, % DLM of $\mathcal{D}_i$.
3:     compute $d(\ell_{\pi(i)}, f(\mathcal{D}_i))$ % distance between leader $\ell_{\pi(i)}$ and vertex $f(\mathcal{D}_i)$.
4:     $\mathcal{P}_i \leftarrow$ edges on the shortest path between $\ell_{\pi(i)}$ and $f(\mathcal{D}_i)$. % If there are multiple, choose any shortest path.
5: **end for**
6: $E_{B_d} \leftarrow \bigcup_i \mathcal{P}_i$
___

and analyze an algorithm for computing such a backbone.

Definition (*Distance-based Backbone*) Given a graph $G = (V, E)$ and a leader set V_ℓ, the distance-based backbone is a subgraph $B_d = (V, E_{B_d})$ such that any subgraph $\hat{G} = (V, \hat{E})$, where $E_{B_d} \subseteq \hat{E} \subseteq E$ satisfies the following:

$$\delta(\hat{G}, V_\ell) \geq \delta(G, V_\ell).$$

It basically means that any subgraph $\hat{G}$ containing backbone edges E_{B_d} has the longest PMI sequence with V_ℓ leaders of length more or equal to the longest PMI sequence as in G, and thus, has at least the same controllability bound as in G. As a result, any $\hat{G}$ containing the backbone edges satisfies $\delta(G, V_\ell) \leq \gamma(\hat{G}, V_\ell)$. So, our goal is to find the minimum distance-based backbone.

___
Problem 3: Given a graph $G = (V, E)$ and a leader set V_ℓ, find a minimum distance-based backbone.
___

Algorithm 2 presents a scheme to compute a minimal distance-based backbone of a given $G = (V, E)$ and leader set $V_\ell = \{\ell_1, \ell_2, \cdots, \ell_m\}$. The input to the algorithm is a PMI sequence $\mathcal{D} = [\mathcal{D}_1 \; \mathcal{D}_2 \; \cdots \; \mathcal{D}_\delta]$ of length $\delta(G, V_\ell) = \delta$. Additionally, we also know the corresponding distance-to-leader mapping $f(\mathcal{D}_i)$ for each $\mathcal{D}_i \in \mathcal{D}$ (i.e., vertices whose DL vectors appear in $\mathcal{D}$). We note that PMI sequence and DLM can be easily computed using methods in [7]. The main idea of Algorithm 2 is to maintain the edges such that for each $\mathcal{D}_i \in \mathcal{D}$, the distance between $f(\mathcal{D}_i)$ and the leader $\ell_{\pi(i)} \in V_\ell$ is preserved. Recall that $\pi(i)$ is the coordinate of $\mathcal{D}_i$ at which the PMI property is satisfied. The details are outlined below and an example is in [20].

Next, we show that the above algorithm always returns a distance-based backbone.

Theorem 4.2: Consider a graph $G = (V, E)$ and a leader set $V_\ell \subseteq V$, where $|V| = n$ and $|V_\ell| = m$. Algorithm 2 returns a distance-based backbone $B_d = (V, E_{B_d})$ in $O(n^3)$ time.

    Proof: Available in [20].   ∎

We note that for a given G and leader set V_ℓ, multiple distinct PMIs of the same length can exist. Moreover, there can be multiple distance-to-leader mappings for a given PMI $\mathcal{D}$. At the same time, for a given $\mathcal{D}$, there can be multiple ways to assign a coordinate $\pi(i)$ to $\mathcal{D}_i \in \mathcal{D}$ while satisfying

the PMI property. Thus, for a given G and V_ℓ, multiple PMIs can have the same length $\delta(G, V_\ell)$ but distinct distance-to-leader mappings or $\pi(i)$ combinations. The distance-based backbone returned by Algorihm 2 depends on the above-mentioned factors, i.e., the input PMI sequence, $\pi(i)$ combinations in the PMI sequence, and the corresponding distance-to-leader mappings. Hence, the distance-based backbone returned by Algorihm 2 may be distinct for different PMI sequences with the same lentgh $\delta(G, V_\ell)$. An instance that demonstrates how the backbone B_d relies on various factors is exemplified in [20].

Next, we state the lower and upper bounds on the number of edges in the distance backbone returned by Algorithm 2.

Proposition 4.3: For a given graph G and a leader set V_ℓ, the lower bound on the number of edges in the distance-based backbone B_d computed using Algorithm 2 is $\delta(G, V_\ell) - m$, where m is the number of leaders.

    Proof: Available in [20].   ∎

An upper bound on the number of edges in the distance backbone due to Algorithm 2 is given below.

Proposition 4.4: For a given graph G, a leader set V_ℓ, and a maximum length PMI $\mathcal{D}$, the upper bound on the number of edges in B_d computed using Algorithm 2 is $\binom{m}{2} + \binom{\delta(G, V_\ell) - m + 1}{2}$.

    Proof: Available in [20].   ∎

In Section V, we perform a numerical analysis and observe that the number of edges in the distance-based backbone is typically very close to the lower bound. Further, by carefully selecting the PMI sequence of the desired length, the number of edges in the distance-based backbone due to Algorithm 2 can be minimized. Next, we compare the ZFS- and distance-based backbones numerically in graphs.

V. Comparison and Illustration

The ZFS- and distance-based bounds have their own merits in computing the controllability backbones, and the number of edges in the ZFS- and distance-based backbones depends on the values of the respective bounds (as in Theorem 3.2 for the ZFS backbone, and Propositions 4.3 and 4.4 for the distance-based backbones). As discussed previously, the ZFS bound works best when the leader set V_ℓ is a ZFS, implying that the network is strong structurally controllable. In such a

scenario, the ZFS bound on the dimension of SSC is at least as good as the distance bound, i.e., $\zeta(G, V_\ell) \geq \delta(G, V_\ell)$ [18]. Moreover, the number of edges in the ZFS-based backbone obtained using Algorithm 1 will be minimum. Thus, the ZFS-based backbone would be a better choice than the distance-based backbone. An example is presented in [20].

On the other hand, if the leader set is not a ZFS, the distance-based bound is typically superior to the ZFS bound [18]. Hence, in such cases, the distance-based backbone is better than the ZFS backbone as the subgraphs that include the distance-based backbone will have higher controllability (i.e., greater dimension of SSC) than those containing the ZFS backbone.

Finally, we perform a numerical evaluation of the ZFS and distance backbones on Erdős-Rényi (ER) graphs with $n = 50$ vertices and varying average density p. For each graph G, we randomly select $m = 12$ leader vertices and find a derived set and maximum length PMI sequence for the selected leader set. We use Algorithms 1 and 2 to find the respective backbone graphs for $50,000$ different ER graphs for each value of p. Figure 4(a) plots the average value of the ZFS and distance bounds on the dimension of SSC as a function of p. We observe that the distance bound $\delta(G)$ is significantly better than the ZFS bound $\zeta(G)$. Figure 4(b) plots the number of edges E_{B_z} in the ZFS-based backbone (computed using Algorithm 1) as a function of p. Similarly, Figure 4(c) plots the the number of edges E_{B_d} in the distance-based backbone (using Algorithm 1). The plot also shows the lower and upper bounds on the number of edges in the distance-based backbone as described in Propositions 4.3 and 4.4, respectively. We observe that the number of edges in distance-based backbones is much closer to the lower bound, $\delta(G) - m$. For instance, for $p = 0.1$, the upper bound on E_{B_d} is 276 compared to the lower bound value of 23. However, the actual value of $|E_{B_d}| \approx 29$, is much closer to the lower bound.
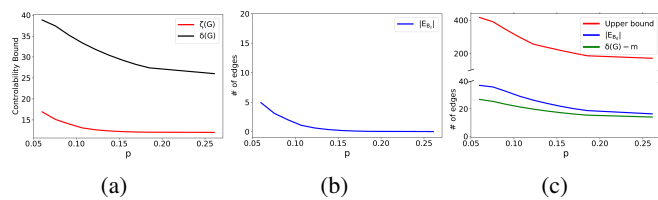

Fig. 4: (a) Comparison of the ZFS and distance bounds on SSC. (b) Number of edges in the ZFS-based backbones. (c) Number of edges in the distance-based backbones.

VI. Conclusion

The controllability of a network can be compromised due to changes in the network's connections. To address this issue, we proposed a method to identify a subset of edges, referred to as the "backbone edges," that are crucial for preserving the minimum network controllability even when the connections in the network are perturbed. Specifically, we designed an algorithm to compute an optimal backbone (containing the minimum number of edges) when the leader set is a zero forcing set. Moreover, we presented an algorithm

utilizing the distance-based bound on the network SSC to identify a controllability backbone when the leader set is not a zero forcing set. Finally, we conducted numerical evaluations on random graphs to demonstrate the effectiveness of each algorithm. As a future direction, we plan to explore the possibility of backbone identification in networks that preserve other network properties, such as energy-based controllability parameters.

References

[1] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010.

[2] J. Yu, N. Wang, G. Wang, and D. Yu, "Connected dominating sets in wireless ad hoc and sensor networks–a comprehensive survey," *Computer Communications*, vol. 36, no. 2, pp. 121–134, 2013.

[3] F. Pasqualetti, S. Zampieri, and F. Bullo, "Controllability metrics, limitations and algorithms for complex networks," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 40–52, 2014.

[4] W. Abbas, M. Shabbir, A. Y. Yazıcıoğlu, and A. Akber, "Tradeoff between controllability and robustness in diffusively coupled networks," *IEEE Transactions on Control of Network Systems*, vol. 7, no. 4, pp. 1891–1902, 2020.

[5] S. S. Mousavi, M. Haeri, and M. Mesbahi, "On the structural and strong structural controllability of undirected networks," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 2234–2241, 2017.

[6] A. Chapman and M. Mesbahi, "On strong structural controllability of networked systems: A constrained matching approach," in *American control conference (ACC)*, 2013, pp. 6126–6131.

[7] M. Shabbir, W. Abbas, Y. Yazicioglu, and X. Koutsoukos, "Computation of the distance-based bound on strong structural controllability in networks," *IEEE Transactions on Automatic Control*, 2022.

[8] N. Monshizadeh, S. Zhang, and M. K. Camlibel, "Zero forcing sets and controllability of dynamical systems defined on graphs," *IEEE Transactions on Automatic Control*, vol. 59, pp. 2562–2567, 2014.

[9] A. Yazıcıoğlu, W. Abbas, and M. Egerstedt, "Graph distances and controllability of networks," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 4125–4130, 2016.

[10] S. Zhang, M. Cao, and M. K. Camlibel, "Upper and lower bounds for controllable subspaces of networks of diffusively coupled agents," *IEEE Transactions on Automatic control*, vol. 59, pp. 745–750, 2013.

[11] W. Abbas, M. Shabbir, H. Jaleel, and X. Koutsoukos, "Improving network robustness through edge augmentation while preserving strong structural controllability," in *American Control Conference (ACC)*, 2020, pp. 2544–2549.

[12] S. S. Mousavi, M. Haeri, and M. Mesbahi, "Strong structural controllability of networks under time-invariant and time-varying topological perturbations," *IEEE Transactions on Automatic Control*, vol. 66, no. 3, pp. 1375–1382, 2020.

[13] M. A. Rahimian and A. G. Aghdam, "Structural controllability of multi-agent networks: Robustness against simultaneous failures," *Automatica*, vol. 49, no. 11, pp. 3149–3157, 2013.

[14] S. Jafari, A. Ajorlou, A. G. Aghdam, and S. Tafazoli, "On the structural controllability of multi-agent systems subject to failure: A graph-theoretic approach," in *IEEE Conference on Decision and Control (CDC)*, 2010, pp. 4565–4570.

[15] C.-L. Pu, W.-J. Pei, and A. Michaelson, "Robustness analysis of network controllability," *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 18, pp. 4420–4425, 2012.

[16] AIM Minimum Rank Special Graphs Work Group, "Zero forcing sets and the minimum rank of graphs," *Linear Algebra and its Applications*, vol. 428, no. 7, pp. 1628–1648, 2008.

[17] M. Trefois and J.-C. Delvenne, "Zero forcing number, constrained matchings and strong structural controllability," *Linear Algebra and its Applications*, vol. 484, pp. 199–218, 2015.

[18] Y. Yazıcıoğlu, M. Shabbir, W. Abbas, and X. Koutsoukos, "Strong structural controllability of networks: Comparison of bounds using distances and zero forcing," *Automatica*, vol. 146, p. 110562, 2022.

[19] N. Monshizadeh, K. Camlibel, and H. Trentelman, "Strong targeted controllability of dynamical networks," in *IEEE Conference on Decision and Control (CDC)*, 2015, pp. 4782–4787.

[20] O. U. Ahmad, W. Abbas, and M. Shabbir, "Controllability backbone in networks," *arXiv:2309.02649*, 2023.