Distributed Biconnecitvity Achievement and Preservation in Multi-agent Systems

Esteban Restrepo and Paolo Robuffo Giordano *Senior Member, IEEE*

***Abstract*—We propose a distributed control law to increase the robustness of a multi-agent network to node failure or removal. More precisely, our approach is able to maintain biconnectivity of an initially biconnected graph. Remarkably, if the graph is not initially biconnected or if the property is lost after a node removal or failure, our approach is also able to render the graph biconnected after a time instant which can be tuned by the user. The proposed control algorithm is completely distributed using only locally available information and requires the estimation of a single global parameter akin to existing connectivity-maintenance algorithms in the literature. Numerical simulations illustrate the effectiveness of our approach.**

***Index Terms*—Multi-robot systems, distributed control, network connectivity.**

I. Introduction

IT is well known that connectivity of the graph representing the interconnections in a multi-agent system, that is, the existence of a path from any node to any other node in the network, is a necessary and sufficient condition for convergence to the agreement subspace in consensus-based coordination algorithms [1]. In light of this, a great number of algorithms have been proposed in order to maintain the connectivity of the graph in networks of autonomous multi-agent systems where the interaction among the nodes is limited to a finite region of the space around each agent—see e.g. [2]–[5].

Mobile agents in realistic scenarios, such as, e.g., robotic vehicles, are prone to multiple kinds of failures or may be required to leave the network in order to fulfill some specific task. These situations, however, may lead to disconnection of the graph and subsequently to the failure of the system-wise coordination mission. Therefore, it is necessary to design algorithms that not only maintain connectivity but that render the graph robust to node removals. In other words, the graph must remain connected even after one (or several) of the nodes and all its incident edges are removed. Such a property is the so-called *biconnectivity* [6]. In the present paper we propose a control law that is able to render the graph biconnected, possibly starting from a non biconnected graph, and to maintain said property for all time.

Biconnectivity maintenance has been much less studied than the simple connectivity counterpart. In [7] a control law is proposed that enforces biconnectivity of the graph by using the gradient of the third smallest eigenvalue of the perturbed Laplacian. However, the third smallest eigenvalue is difficult to estimate in a distributed way and each agent would need to estimate a different eigenvalue for every critical node in the graph, increasing the amount of computations each agent needs to perform. Moreover, this approach is able to render a graph biconnected but no guarantees are given on the preservation of the biconnectivity in the presence of, e.g. disturbances or other control objectives. In [8] a biconnectivity-maintenance-and-restoration approach is proposed. However, it needs that each node i continuously estimates the values of the second smallest eigenvalue of the perturbed Laplacian at *every* node, that is, each node needs to estimate N different second smallest eigenvalues, making this approach not scalable. An extension of the latter is proposed in [9] where the authors define a set of *frangible* nodes, i.e., critical nodes that may soon fail or be removed from the graph. A perturbed Laplacian is computed multiplying the edges of all the frangible nodes by a small constant. Then, the so-called *perturbed algebraic connectivity* is calculated. Based on a gradient control of the perturbed algebraic connectivity the robustness of the graph with respect to the removal of the frangible nodes is improved. However, it only increases the number of edges of the critical nodes and holds only if the initial condition on the perturbed algebraic connectivity is satisfied. That is, no guarantees are given on the maintenance or achievement of biconnectivity.

In this letter we propose a gradient-based biconnectivity controller that solves two of the open challenges not addressed by the previous works in the literature mentioned above, namely, scalability and guarantees for biconnectivity maintenance *and* recovery. Indeed, on one hand, in contrast to the works mentioned above, we guarantee both that the graph becomes and remains biconnected, even if it is not initially so or if biconnectivity is lost after, e.g., a node failure. Moreover, biconnectivity is achieved after a specified time instant, which can be tuned by the user. On the other hand, our approach relies only on locally-available information and only *one* estimation needs to be performed by the agents, that of the perturbed algebraic connectivity, akin to other global-connectivity-maintenance approaches in the literature, which greatly reduces the number of computations needed with respect to the approaches mentioned above, making our approach more scalable.

This letter is organized as follows. In Section II the model and the problem statement are presented. The biconnectivity

The authors are with CNRS, Univ Rennes, Inria, IRISA, Campus de Beaulieu, 35042 Rennes, France, email: {esteban.restrepo@inria.fr, prg@irisa.fr}.

This work was supported by the ANR-20-CHIA-0017 project "MULTI-SHARED"

control is presented in Section III. Finally, the main results are illustrated via numerical simulations in Section IV and some concluding remarks are given in Section V.

II. Model and graph theory

We consider N agents modeled by the single-integrator dynamics

$$\dot{p}_i = u_i, \tag{1}$$

where p_i and u_i denote, respectively, the position and control input of agent i. The agents communicate with each other and this communication is described by an *undirected* graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t))$, where $\mathcal{V} = \{1, 2, \ldots, N\}$ is the index set and $\mathcal{E}(t) \subseteq \mathcal{V}^2$ is the edge set characterizing the information exchange between agents, that is, an edge $e_k := (i, j) \in \mathcal{E}(t)$, $k = \{1, \ldots, M(t)\}$, is an ordered pair indicating that node j has access to information from node i.

We assume that the communication between the agents is proximity-based, that is, and edge e_{ij} connecting robots i and j exists if and only if $||p_i - p_j|| \leq R$, with R denoting the upper-bound on the distance under which the communication is reliable. Hence, the set of neighbors of agent i is given by

$$\mathcal{N}_i := \{j \in \mathcal{V}, \mid j \neq i \text{ and } ||p_i - p_j|| < R\}. \tag{2}$$

To model this proximity-based communication we use the adjacency matrix $A \in \mathbb{R}^{N \times N}$, where the element a_{ij}, represents the weight of edge e_{ij}. Namely, $a_{ij} > 0$ if and only if $j \in \mathcal{N}_i$, and $a_{ij} = 0$ otherwise. More precisely, given $r < R$ and denoting $d_{ij} := ||p_i - p_j||$, we let

$$a_{ij} = \begin{cases} 1, & \text{if } d_{ij} \leq r \\ \frac{1}{2}\left(1 + \cos\left(\frac{\pi(d_{ij} - r)}{R - r}\right)\right), & \text{if } r < d_{ij} \leq R \\ 0, & R < d_{ij}. \end{cases} \tag{3}$$

Since $\mathcal{G}(t)$ is undirected, it follows that $a_{ij} = a_{ji}$.

The Laplacian matrix $L \in \mathbb{R}^{N \times N}$ is a symmetric positive semi-definite matrix given by $L = \text{diag}\{A\mathbf{1}\} - A$, where $\mathbf{1} \in \mathbb{R}^N$ is the vector of all ones. It is well known that some fundamental properties of the graph are associated with the Laplacian matrix. Specifically, denoting λ_2 as the second smallest eigenvalue of L, we have that $\lambda_2 > 0$ if and only if $\mathcal{G}(t)$ is connected and $\lambda_2 = 0$ otherwise—see [10] for more details on graph theory. Furthermore, we let ν_2 denote the unit-norm eigenvector of L associated with λ_2. Under the weights definition (3), λ_2 becomes a connectivity measure of the proximity-based graph and, in particular, a function of the system state (e.g. of the agents relative positions).

Although maintaining the connectivity of the graph, i.e. $\lambda_2(t) > 0$ for all t, is a sufficient condition for most coordination tasks in multi-agent systems, it may prove lacking in certain scenarios where individual agents disconnect from the group due to, e.g., failures, attacks, recharging or user commands. In such scenarios, the removal of one of the robots may lead the remaining graph to become disconnected.

Biconnectivity is the property of a graph to be robust to a node removal [6]. More precisely, let $\mathcal{G}_{-i}$ be the graph remaining after the removal of node i. Then, $\mathcal{G}$ is said to be biconnected if, for any $i \in \mathcal{V}$, $\mathcal{G}_{-i}$ is connected. If $\mathcal{G}_{-i}$ is disconnected, node i is called an *articulation* point. Therefore, an equivalent definition for a biconnected graph is for it to be a connected graph with no articulation points.

A. Perturbed algebraic connectivity

In order to consider the robustness of a graph with respect to a node removal we define the perturbed adjacency matrix $A^i(\varepsilon)$, whose elements are given by

$$a^i_{kj}(\varepsilon) = \begin{cases} \varepsilon a_{kj}, & \text{if } k = i \text{ or } j = i \\ a_{kj}, & \text{otherwise}, \end{cases} \tag{4}$$

where ε is a small positive constant. The perturbed adjacency matrix $A^i(\varepsilon)$ captures the case in which the node i is *almost* removed from the graph. Indeed, note that $\varepsilon \to 0$ would correspond to the case where node i is effectively removed.

Now, the perturbed Laplacian matrix (at node i) is defined as $L^i(\varepsilon) = \text{diag}\{A^i(\varepsilon)\mathbf{1}\} - A^i(\varepsilon)$. The second smallest eigenvalue of the perturbed Laplacian at node i is denoted $\lambda^i_2(\varepsilon)$ and its associated eigenvector $\nu^i_2(\varepsilon)$.

If node i is an articulation point, then the graph $\mathcal{G}_{-i}$ is composed of M^i remaining connected components $\mathcal{C}^i_m$, $m \in \{1, \ldots, M^i\}$. In [11] it is shown that the connectivity of $\mathcal{G}_{-i}$ and the size of $\mathcal{C}^i_m$ is related to the value

$$\hat{\lambda}^i_2 = \lim_{\varepsilon \to 0^+} \frac{\lambda^i_2(\varepsilon)}{\varepsilon}, \tag{5}$$

which is called the *perturbed algebraic connectivity*. Indeed, letting $\nu^i_{2,i}(\varepsilon)$ be the ith component of $\nu^i_2(\varepsilon)$ and denoting by C^i_m the number of nodes in $\mathcal{C}^i_m$ and by D^i_m the sum of their weights, i.e., $D^i_m = \sum_{j \in \mathcal{N}_i \bigcap \mathcal{C}^i_m} a_{ij}$, it is shown in [11] that

$$\hat{\lambda}^i_2 = \frac{\sum_{j=1}^N a_{ij}(\nu^i_{2,i}(0) - \nu^i_{2,j}(0))^2}{||\nu^i_2(0)||}, \tag{6}$$

$$\hat{\lambda}^i_2 \nu^i_{2,j} = \frac{D^i_m}{C^i_m}(\nu^i_{2,j} - \nu^i_{2,i}), \quad \forall j \in \mathcal{C}^i_m. \tag{7}$$

From (7) and the definition of ν^i_2, namely $\sum_j \nu^i_2 = 0$, if $\nu^i_{2,i} \neq 0$, we get the algebraic equation

$$\sum_{m=1}^{M^i} \frac{D^i_m C^i_m}{D^i_m - C^i_m \hat{\lambda}^i_2} + 1 = 0. \tag{8}$$

Based on the algebraic equation (8), it is shown in [9] that there exists a lower bound $\bar{\lambda}$ such that if the perturbed algebraic connectivity satisfies $\hat{\lambda}^i_2 > \bar{\lambda}$ then the remaining graph $\mathcal{G}_{-i}$ is connected.

III. Biconnectivity control

A. Perturbed graph

In this paper, the objective is to design a control law that renders the graph biconnected and maintains said property for all time. For this purpose we make use of the perturbed Laplacian and the perturbed algebraic connectivity. However, instead of considering the perturbed graph only at a single critical node, we set a general perturbed graph that accounts simultaneously for all the articulation points in the graph.

In light of the above, we associate each agent with a dynamic parameter, which will be defined later, denoted $\rho_i \in [0, 1-\delta]$, with $\delta > 0$ a small constant, and we let $\varepsilon_i = 1 - \rho_i$. Then, we define the perturbed adjacency matrix $\tilde{A}$, whose elements are given by

$$\tilde{a}_{ij}(d_{ij}) = \min\{\varepsilon_i, \varepsilon_j\} a_{ij}(d_{ij}). \tag{9}$$

Note that if either node i or j are critical points, then the edge-weights a_{ij} and a_{ji} are multiplied by a small constant to express that this node is almost disconnected. Now, the perturbed Laplacian matrix is given by $\tilde{L} = \text{diag}\{\tilde{A}\mathbf{1}\} - \tilde{A}$. Let $\tilde{\lambda}_2$ be the second smallest eigenvalue of $\tilde{L}$ and $\tilde{\nu}_2$ its associated eigenvector. Then, with a slight abuse of notation, we denote the *perturbed algebraic connectivity* of $\tilde{L}$ by $\hat{\lambda}_2$, which is approximated by

$$\hat{\lambda}_2 \approx \frac{\tilde{\lambda}_2}{\delta}. \tag{10}$$

The main idea is to set $\rho_i = 1 - \delta$ (equivalently $\varepsilon_i = \delta$) if node i is an articulation point, and to apply a control law based on the gradient of the perturbed algebraic connectivity to maintain $\hat{\lambda}_2$ above a threshold $\bar{\lambda}$, defined in Section III-B below, in order to guarantee the biconnectivity of the graph. Then, when the node stops being an articulation point, ρ_i smoothly decreases to 0, so that it is no longer a "critical node" and the normal edge weights are recovered with $\varepsilon_i = 1$.

The authors in [12] proposed a way in which each node can verify if it is an articulation node based on the value of the third smallest eigenvalue λ_3^i of the perturbed Laplacian $L^i(\varepsilon)$ (at a single node i). This approach, however, requires the knowledge, by every robot, of the third smallest eigenvalue (and its associated eigenvector) of $L^i(\varepsilon)$ at each articulation point. However, the third smallest eigenvalue is difficult to estimate in a distributed way and, moreover, each agent would need to perform a different estimation of λ_3^i for every articulation node in the graph, increasing the amount of computations performed by each agent.

In contrast, in this paper we propose to check if a node is *locally biconnected*. Let $\mathcal{G}_i^l \subset \mathcal{G}$ denote the local subgraph centered at node i and formed by the neighbors of node i, without itself. That is, $\mathcal{G}_i^l = (\mathcal{V}_i^l, \mathcal{E}_i^l)$, where $\mathcal{V}_i^l = \mathcal{N}_i$ and an edge $e_{kj} \in \mathcal{E}_i^l$ exits if and only if $e_{kj} \in \mathcal{E}$, with $k,\ j \in \mathcal{N}_i$. Then, a node is called locally biconnected if the second smallest eigenvalue $\lambda_{2,i}^l$ of the local graph $\mathcal{G}_i^l$ is positive.

Remark 1: In an undirected graph, to characterize the local subgraph, each node only needs to receive the positions of its neighbors. Then, based on these, the local Laplacian matrices can be determined. •

A sufficient condition for a graph to be biconnected is that all its nodes are locally biconnected [12]. Therefore, we define a dynamic law for the parameter ρ_i based on the connectivity of the local graph. More precisely we set

$$\dot{\rho}_i = -\kappa_1 \rho_i + \frac{\kappa_2}{2}\left(1 + \text{sign}(\sigma_\lambda - \lambda_{2,i}^l)\right). \tag{11}$$

The dynamic system (11) is an exponentially stable system with an additive disturbance which is different from zero only when $\lambda_{2,i}^l$ is smaller than a small threshold σ_λ. This means that if a node i is not locally biconnected, i.e. $\lambda_{2,i}^l = 0$, the value of ρ_i increases and, equivalently, ϵ_i decreases, thereby "perturbing" the graph. We set $\rho_i(t_0) = 0$ if node i is locally biconnected at t_0, and $\rho_i(t_0) = 1 - \delta$ otherwise.

Remark 2: From (11) we have that $\rho_i(t) \in [0, 1-\delta]$ for all $t \geq t_0$. To see this, note that the second term on the right-hand side of (11) is always positive. Therefore, $\dot{\rho}_i(t) \geq -\kappa_1\rho_i(t)$, together with $\rho_i(t_0) \in [0, 1-\delta]$, means that $\rho_i(t) \geq 0$ for all t. Then, the derivative of $V_\rho = 0.5\rho_i^2$ yields $\dot{V} \leq 0$ for $\kappa_1\rho_i \geq \kappa_2$, hence, the set $\mathcal{R}_i := \{\rho_i \in \mathbb{R} : 0 \leq \rho_i \leq \frac{\kappa_2}{\kappa_1}\}$ is forward invariant. Choosing κ_1 and κ_2 such that $\frac{\kappa_2}{\kappa_1} = 1 - \delta$ we get the result. •

Remark 3: Designing a control law based on the perturbation parameter ρ_i given in (11) would render every node locally biconnected, which is more conservative than just achieving biconnectivity, since the graph may become biconnected even if every node is not locally so. However, using $\lambda_{2,i}^l$ instead of, e.g., λ_3^i, offers a good trade-off between conservativeness and complexity as it requires less computations and can be calculated locally. •

B. Lower bound of the perturbed algebraic connectivity

In order to derive a lower bound for $\hat{\lambda}_2$ that guarantees that the graph remains biconnected we use the algebraic equation (8). The following result, originally proposed in [8, Th. 3.1], is adapted hereafter to better fit the notations of this paper.

First let us assume that there is a single agent i such that $\rho_i > 0$, that is, there is a single "critical" node. Assume further that the graph $\mathcal{G}_{-i}$ is disconnected with two connected components. Let us define $\alpha, \beta \in (0,1)$, then, without loss of generality, the number of nodes in each connected component and the sum of their weights are given, respectively, by $C_1^i = \beta(N-1)$, $C_2^i = (1-\beta)(N-1)$, and $D_1^i = \alpha D_i$, $D_2^i = (1-\alpha)D_i$. Then, from (8), we have

$$\frac{D_i\alpha\beta(N-1)}{\beta(N-1)\hat{\lambda}_2 - D_i\alpha} + \frac{D_i(1-\alpha)(1-\beta)(N-1)}{(1-\beta)(N-1)\hat{\lambda}_2 - D_i(1-\alpha)} = 1, \tag{12}$$

where

$$\beta \in \left\{\frac{1}{N-1}, \frac{2}{N-1}, \ldots, \frac{N-2}{N-1}\right\} \tag{13}$$

since $C_1^i = \beta(N-1)$ should be an integer. Hence, from (12) we have that the perturbed algebraic connectivity satisfies

$$\frac{\alpha D_i}{\beta(N-1)} \leq \hat{\lambda}_2 \leq \frac{(1-\alpha)D_i}{(1-\beta)(N-1)}. \tag{14}$$

The maximal algebraic connectivity is then found by computing the gradient of $\hat{\lambda}_2$ with respect to the parameters α and β (see [8, Th. 3.1] for more details), and is given by

$$\hat{\lambda}_2 = \frac{num(D_i, N)}{den(D_i, N)}, \tag{15}$$

$$\begin{aligned} num(D_i, N) &= 2D_i^2N(N-1) - D_iN(N-6) - 9D_i \\ &\quad + D_i(N-3)\sqrt{4D_i^2N^2 - 4D_iN(N+1) + N^2 - 2N + 9}, \\ den(D_i, N) &= 8D_i^2N(N-2) + 4D_iN(N-6) + 36D_i \\ &\quad -2N^2 + 12N - 18). \end{aligned}$$

The value in (15) corresponds to the maximum algebraic connectivity if the node i is an articulation point. Moreover, since in the presence of multiple critical nodes the perturbed algebraic connectivity is smaller or equal than for a single agent, if we make $\hat{\lambda}_2 > \bar{\lambda}$, where $\bar{\lambda}$ is set to the right-hand side of (15), then the graph remains biconnected.

Note that in this paper, unlike in [8], the expression in (15) is a function of the degree of node i, D_i, since we do not consider the normalized Laplacian. However, from the definition of the weights (3), the value of the degree is time-varying and depends on the number of neighbors and robot's i proximity to them. In light of the latter, we simplify (15) by taking the maximum value for the degree, given by $D_i = N - 1$ when the node i is completely connected and sufficiently close to its neighbors, and we obtain

$$\begin{aligned}\bar{\lambda} &= \frac{\overline{num}(N)}{8N^4 - 28N^3 + 11N^2 + 50N - 45}, \qquad (16)\\ \overline{num}(N) &:= (N-1)\left(2N^3 - 5N^2 + 8N - 9\right.\\ &\quad \left. +(N-3)\sqrt{4N^4 - 12N^3 + 5N^2 + 2N + 9}\right).\end{aligned}$$

The bound (16) is more conservative than (15) and, in principle, would drive the critical nodes to become completely connected. However, the latter does not occur since, as soon as a node i becomes locally biconnected, the last term on the right-hand side of (11) vanishes and ρ_i converges to 0 exponentially, and the node stops being a critical node. Moreover, as can be seen in Fig. 1, $\bar{\lambda}$ as a function of the number of agents N ranges from 0.47 to 0.5 with an asymptote at 0.5. Therefore, without loss of generality, one could take $\bar{\lambda} = 0.5$. This makes our approach, compared to the works in the literature, more scalable since it does not depend on the number of agents in the system and is useful in situations where the number of agents changes or is unknown.

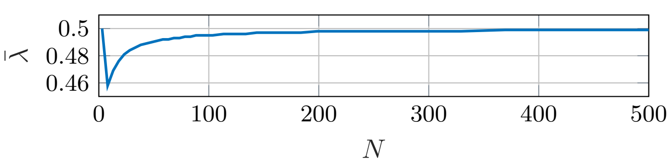

Fig. 1: Variation of $\bar{\lambda}$ with respect to N.

C. Control law for biconnectivity

The design of the control law is inspired by the connectivity preserving controller proposed in [13]. Thus, we introduce the scalar function $V_\lambda : \mathcal{D} \to \mathbb{R}_{\geq 0}$, $\hat{\lambda}_2 \mapsto V_\lambda(\hat{\lambda}_2)$, which is $\mathcal{C}^1$ over its domain and has the property that $V_\lambda(\hat{\lambda}_2) \to \infty$ as $\hat{\lambda}_2 \to \partial\mathcal{D}$. Then, the gradient controller is set to be

$$u_i^c = -\kappa \frac{\partial V_\lambda(\hat{\lambda}_2)}{\partial p_i} = -\kappa \frac{\partial V_\lambda}{\partial \hat{\lambda}_2} \frac{\hat{\lambda}_2}{\partial p_i}, \quad \kappa > 0. \qquad (17)$$

From [13] and recalling (10), we have that

$$\begin{aligned}\frac{\partial \hat{\lambda}_2}{\partial p_i} &= \frac{1}{\delta} \hat{\nu}_2^\top \frac{\partial \tilde{L}}{\partial p_i} \hat{\nu}_2 \\ &= \frac{1}{\delta} \sum_{j=1}^{N} (\hat{\nu}_i - \hat{\nu}_j)^2 \frac{\partial \tilde{a}_{ij}}{\partial p_i}. \qquad (18)\end{aligned}$$

Replacing (18) into (17) we obtain

$$u_i^c = -\kappa \frac{\partial V_\lambda}{\partial \hat{\lambda}_2} \sum_{j=1}^{N} \frac{(\hat{\nu}_i - \hat{\nu}_j)^2}{\delta} \frac{\partial \tilde{a}_{ij}}{\partial p_i}. \qquad (19)$$

Moreover choosing

$$V_\lambda(\hat{\lambda}_2) = \coth(\hat{\lambda}_2 - \lambda^*) - 1, \qquad (20)$$

where λ^* is a properly chosen bound, the gradient control law becomes

$$u_i^c = -\kappa \operatorname{csch}^2(\hat{\lambda}_2 - \lambda^*) \sum_{j=1}^{N} \frac{(\hat{\nu}_i - \hat{\nu}_j)^2}{\delta} \frac{\partial \tilde{a}_{ij}}{\partial p_i}. \qquad (21)$$

The objective of biconnectivity maintenance is equivalent to rendering forward invariant the set

$$\mathcal{D} := \{\hat{\lambda}_2 \in \mathbb{R}_{\geq 0} \mid \hat{\lambda}_2 > \lambda^*\}. \qquad (22)$$

It can be shown that under the gradient control (17), with $\lambda^* = \bar{\lambda}$ in (16), the multi-agent system remains biconnected for all time. However, biconnectivity may be lost after a node is removed due to, e.g., a failure. Hence, the initial graph may not be biconnected, i.e., there is at least one articulation point, and it may be possible that $\hat{\lambda}_2 \leq \bar{\lambda} = \lambda^*$. But, since V_λ is only defined in the set (22), the controller (17) would not be able to render the graph biconnected, indeed it may not be well defined. In order to overcome this problem, we set instead λ^* as the solution to the dynamic system

$$\dot{\lambda}^*(t) = -\kappa_\lambda(\lambda^*(t) - (\bar{\lambda} + \epsilon)), \qquad (23)$$

where $\epsilon > 0$ is a small constant, $\bar{\lambda}$ is given by (16), and $\lambda^*(t_0) = 0$. Note that, under (23) and $\lambda^*(t_0) = 0$, $\lambda^*(t)$ is positive for all t and exponentially converges to $\bar{\lambda} + \epsilon$ with a rate determined by $\kappa_\lambda > 0$. Since we assume that the initial graph is at least connected and $\varepsilon_i \geq \delta$, for all i, we have that $\hat{\lambda}_2(t_0) > 0$ and therefore, given $\lambda^*(t_0) = 0$, we have that $\hat{\lambda}_2(t_0) \in \mathcal{D}$ so that V_λ and the input (21) are well defined.

Under the control law (21), setting $\lambda^*(t)$ as the solution to (23), we guarantee that there exists a (tunable) time instant $T(\kappa_\lambda) > t_0$ such that for all $t \geq T(\kappa_\lambda)$ the graph is biconnected. This is stated in the following result.

Proposition 1 (Main result): Assume the initial network $\mathcal{G}(t_0)$ is connected. Then, under the gradient control law (17) with $\lambda(t)^*$ as the solution to (23) with $\lambda(t_0)^* = 0$ and $\bar{\lambda}$ as in (16), the graph $\mathcal{G}(t)$ becomes and remains biconnected. More precisely, the (22) is rendered forward invariant and there exists a (tunable) time instant $T(\kappa_\lambda) > t_0$ such that $\mathcal{G}(t)$ is biconnected for all $t \geq T(\kappa_\lambda)$. □

Proof: Assume that $\hat{\lambda}_2(t) \in \mathcal{D}$ (this assumption will be relaxed later). Define the potential function $V_\lambda : \mathbb{R}_{\geq 0} \times \mathcal{D} \to \mathbb{R}_{>0}$, given by $V_\lambda(t, \hat{\lambda}_2) = \coth(\hat{\lambda}_2 - \lambda^*(t)) - 1$, which is positive, i.e., $V_\lambda(t, \hat{\lambda}_2) > 0$, in its domain of definition. Its derivative yields

$$\begin{aligned}\dot{V}_\lambda(t, \hat{\lambda}_2) &= \frac{\partial V_\lambda}{\partial \hat{\lambda}_2} \frac{\hat{\lambda}_2}{\partial p}^\top \dot{p} + \frac{\partial V_\lambda}{\partial t} \\ &= \operatorname{csch}^2(\hat{\lambda}_2 - \lambda^*(t)) \left[-\kappa \left| \frac{\hat{\lambda}_2}{\partial p} \right|^2 - \dot{\lambda}^*(t) \right]. \qquad (24)\end{aligned}$$

Given (23) and $\lambda^*(t_0)$, we have that $\dot{\lambda}^*(t) \geq 0$ for all $t \geq t_0$. Therefore, we have

$$\dot{V}_\lambda(t, \hat{\lambda}_2) \leq -\kappa \operatorname{csch}^2(\hat{\lambda}_2 - \lambda^*(t)) \left| \frac{\hat{\lambda}_2}{\partial p} \right|^2 \leq 0. \tag{25}$$

From (25) we conclude that V_λ is bounded along the trajectories of $p(t)$. Now, in order to prove the forward invariance of $\mathcal{D}$ we proceed by contradiction. Let us assume that there exists a τ such that $\hat{\lambda}_2(t) \in \mathcal{D}$ for all $t \in [t_0, \tau)$ and $\hat{\lambda}_2(\tau) \notin \mathcal{D}$. Therefore, from continuity of the solutions we have that $\hat{\lambda}_2(t) \to \partial\mathcal{D}$ as $t \to \tau$. From the definition of V_λ this implies that $V_\lambda(t, \hat{\lambda}_2(t)) \to \infty$. However, the latter is in contradiction with (25) which implies that $V_\lambda(t, \hat{\lambda}_2(t))$ is bounded. Therefore, we conclude that $\mathcal{D}$ is forward invariant.

Finally, since from (23) λ^* exponentially converges to $\bar{\lambda} + \epsilon$ with a rate determined by κ_λ, there exists a (tunable) time instant $T(\kappa_\lambda) > t_0$ such that $\lambda^*(T) = \bar{\lambda}$. From the forward invariance of $\mathcal{D}$, $\hat{\lambda}_2(t) > \lambda^*(t)$ holds for all $t \geq t_0$. Therefore, $\hat{\lambda}_2(t) > \bar{\lambda}$ for all $t \geq T(\kappa_\lambda)$ implying that the graph becomes and remains biconnected for all $t \geq T(\kappa_\lambda)$. ∎

IV. Numerical examples

In this section we present some numerical examples to validate our theoretical results. The multi-agent system consists of $N = 21$ agents that interact over a proximity-based undirected graph where the edge weights are given by (3) with $R = 2.2$ m and $r = 1.2$ m. The initial configuration with the initial graph are represented in Fig. 2. From the latter, it is clear that the initial graph is not biconnected, therefore $\hat{\lambda}_2(t_0) < \bar{\lambda}$. However, since $\lambda^*(t)$ is the solution of (23), we have that $\hat{\lambda}_2(t_0) > \lambda^*(t_0)$—see Fig. 4.

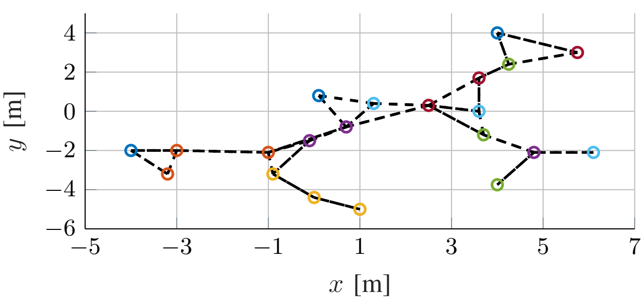

Fig. 2: Initial configuration and initial graph.

The parameters of the dynamic law (11) are set to $\delta = 0.02$, $\kappa_1 = 10$, $\kappa_2 = 9.8$, and $\sigma_\lambda = 0.005$, and the gain of the controller (21) is set to $\kappa = 0.25$. Moreover we set $\kappa_\lambda = 0.3$ in (23) and $\bar{\lambda} \approx 0.48$, obtained from (16) by replacing $N = 21$. Figs. 3-5 show the results of applying the controller (23). It can be seen from Fig. 4 that the value of the perturbed algebraic connectivity $\hat{\lambda}_2$ remains above the imposed time-varying lower bound $\lambda(t)^*$. The latter guarantees that the graph becomes biconnected as can be seen in Figs. 3 and 5. Indeed note that the parameters ρ_i become 0 after 3s.

For comparison Fig. 6 shows the results when the agents' inputs are given by the connectivity maintenance control in

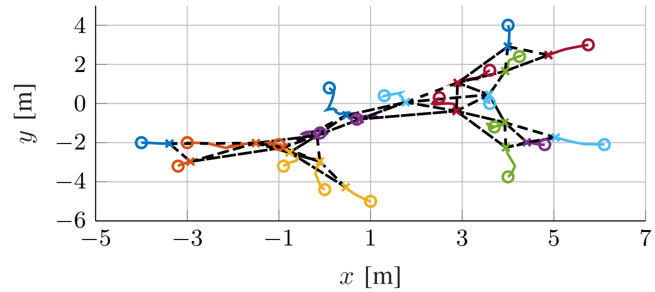

Fig. 3: Trajectories of the agents and final graph.

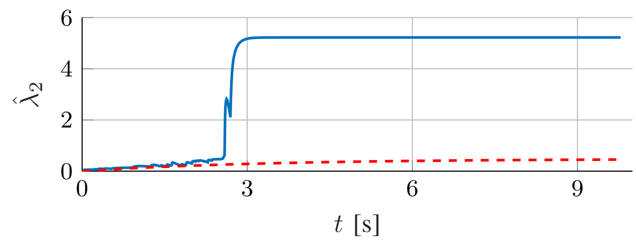

Fig. 4: Perturbed algebraic connectivity. The dashed red line represents the bound $\lambda^*(t)$.

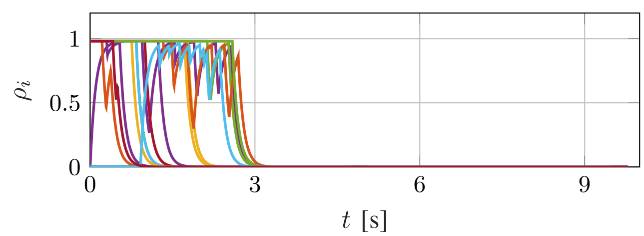

Fig. 5: Dynamic indicator of local biconnectivity. $\rho_i = 0$ means that node i is locally biconnected.

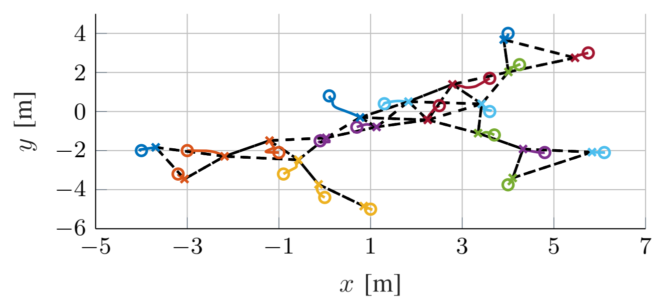

Fig. 6: Trajectories of the agents and final graph.

[2]. Although the connectivity is maintained, it is clear from Fig. 6 that the graph is not biconnected.

Finally, to demonstrate the applicability of the proposed approach in a more meaningful example we studied the scenario where agents may join and leave the network. In this scenario, the agents start from the same initial configuration as before (Fig. 2). As the agents evolve under the biconnectivity law described above, three new agents (initial positions marked with a triangle in Fig. 7) join the network at around 6s, 10s, and 17s. Moreover, at 20s and 24s, respectively, two agents (initial positions marked with a square in Fig. 7), leave the network. Figure 8 depicts the evolution of the number of agents over time. Whenever an agent leaves or joins, the value of λ^* is reset to zero in order to account for the possible loss of biconnectivity. This could be distributedly achieved in practice using a flooding algorithm [14] to update the value of λ^*. Then, biconnectivity is recovered under our proposed controller. This can be seen in Fig. 9 where the perturbed algebraic connectivity remains at all times above the prescribed bound. Figure 10 reports the values of the parameters ρ_i, where it can be seen that the local biconnectivity is recovered after each addition or removal.

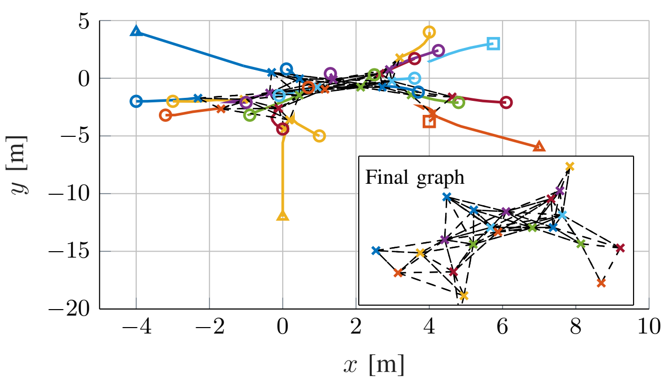

Fig. 7: Trajectories of the agents and final graph.

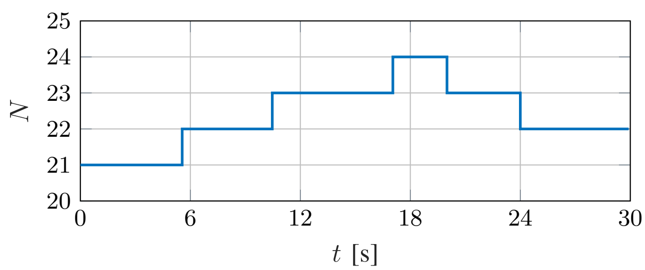

Fig. 8: Evolution of the number of agents in the network.

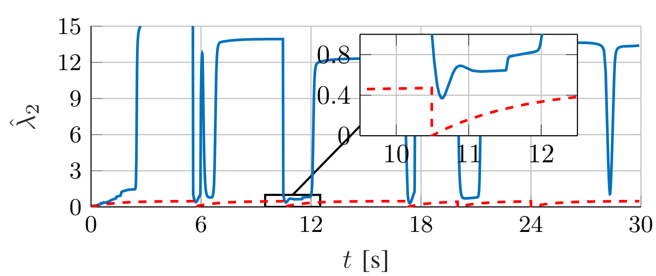

Fig. 9: Perturbed algebraic connectivity. The dashed red line represents the bound $\lambda^*(t)$.

V. Conclusions

In realistic multi-agent scenarios with limited-range communication, it is necessary to account for potential disconnections of the network due to the removal of one or multiple nodes following, e.g., a failure or an attack. In such cases, the more robust property of biconnectivity is preferred to guarantee robustness with respect to node removal. We propose

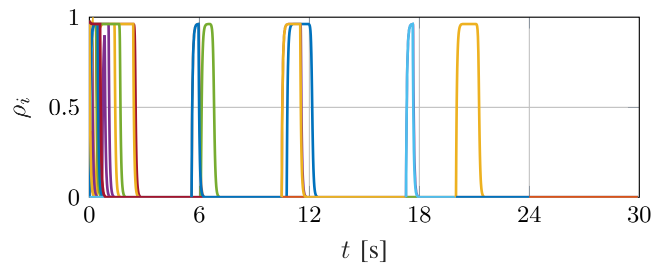

Fig. 10: Dynamic indicator of local biconnectivity ρ_i .

a gradient-based control law that is able, using only locally available information, to not only maintain biconnectivity but to also recover it, after it is lost, in tunable pre-specified time. Current and future work focuses, on one hand, on a quantitative and experimental comparison between the local-biconnectivity-based approach proposed in this letter and the approaches based on computing λ_3, and on the other hand, considering field-of-view and collision-avoidance constraints in "generalized biconnectivity" framework, and applying the results to the context of open multi-robot systems.

References

[1] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*, ser. Princeton series in applied mathematics. Princeton: Princeton University Press, 2010, oCLC: ocn466341412.

[2] L. Sabattini, C. Secchi, and N. Chopra, "Decentralized Estimation and Control for Preserving the Strong Connectivity of Directed Graphs," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2273–2286, Oct. 2015.

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

[4] D. V. Dimarogonas and K. J. Kyriakopoulos, "Connectedness preserving distributed swarm aggregation for multiple kinematic robots," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1213–1223, Oct. 2008.

[5] A. Ajorlou, A. Momeni, and A. G. Aghdam, "A class of bounded distributed control strategies for connectivity preservation in multi-agent systems," *IEEE Trans. Autom. Control*, vol. 55, no. 12, pp. 2828–2833, 2010.

[6] M. C. Golumbic, *Algorithmic graph theory and perfect graphs*. Elsevier, 2004.

[7] M. Zareh, L. Sabattini, and C. Secchi, "Enforcing biconnectivity in multi-robot systems," in *2016 IEEE 55th Conf. Decision Control (CDC)*, 2016, pp. 1800–1805.

[8] T. Murayama, "Distributed control for bi-connectivity of multi-robot network," *SICE Journal of Control, Measurement, and System Integration*, vol. 0, no. 0, pp. 1–10, 2022.

[9] ——, "Dynamical improvement of network connectivity for multi-robot robustness using perturbed algebraic connectivity," in *2021 IEEE/SICE Int. Symp. Syst. Integration (SII)*, 2021, pp. 781–785.

[10] C. Godsil and G. F. Royle, *Algebraic graph theory*. Springer Science & Business Media, 2001, vol. 207.

[11] T. Murayama, "Articulation node importance estimation and its correctness for robustness of multi-robot network," *IFAC-PapersOnLine*, vol. 51, no. 22, pp. 166–171, 2018, 12th IFAC Symp. Robot Control (SYROCO).

[12] M. Zareh, L. Sabattini, and C. Secchi, "Decentralized biconnectivity conditions in multi-robot systems," in *2016 IEEE 55th Conf. Decision Control (CDC)*, 2016, pp. 99–104.

[13] L. Sabattini, C. Secchi, N. Chopra, and A. Gasparri, "Distributed Control of Multirobot Systems With Global Connectivity Maintenance," *IEEE Trans. Robot.*, vol. 29, no. 5, pp. 1326–1332, Oct. 2013.

[14] G. Oliva, R. Setola, and C. N. Hadjicostis, "Distributed finite-time average-consensus with limited computational and storage capability," *IEEE Trans. on Control of Netw. Syst.*, vol. 4, no. 2, pp. 380–391, 2017.