Improving the fault tolerance of multi-robot networks through a combined control law strategy

Cinara Ghedini
Computer Science Division
Technological Institute of Aeronautics
São José dos Campos – SP – Brazil
Email: cinara@ita.br

Carlos H. C. Ribeiro
Computer Science Division
Technological Institute of Aeronautics
São José dos Campos – SP – Brazil
Email: carlos@ita.br

Lorenzo Sabattini
Department of Sciences and
Methods for Engineering (DISMI)
University of Modena and Reggio Emilia – Italy
Email: lorenzo.sabattini@unimore.it

***Abstract*—Applications based on groups of self-organized mobile robots and — more generically — agents are becoming pervasive in communication, monitoring, traffic and transportation systems. Their advantage is the possibility of providing services without the existence of a previously defined infrastructure and with a high degree of autonomy. On the other hand, physical agents, in general, are prone to failures, adding uncertainty and unpredictability in the environments in which they operate. Therefore, a robust topology regarding failures is an imperative requirement. In this paper, we show that mechanisms based solely on connectivity maintenance are not enough to obtain a sufficiently resilient network, and a robustness-oriented approach is necessary. Thus, we propose a local combined control law that aims at maintaining the overall network connectivity while improving the network robustness via actions that reduce vulnerability to failures that might lead to network disconnection. The combined control law performance was validated from two perspectives: as a reactive and as a proactive mechanism. As a reactive mechanism, it was able to accommodate ongoing failures and postpone or avoid network fragmentation. As a proactive mechanism, the network topology was able to evolve from a potentially vulnerable topology w.r.t. failures to a more robust one.**

***Index Terms*—Fault-tolerant networks, Multi cooperative robot control, Adaptive networks**

I. Introduction

Network services are becoming pervasive, and network infrastructure is widely available: this makes it possible to access the Internet, cloud technologies, and effective communication services in many circumstances. Nevertheless, there are wide areas of the world where network infrastructure is not available, either because it is not possible from an economic or technological point of view, or because such a structure can get damaged in a disaster event [1].

Mobile robots, if equipped with appropriate communications devices, can be exploited to create an infrastructure network. For instance, interconnected mobile robots can provide rescuing devices in an exploration task, or generic clients (e.g., mobile phones, laptops, tablets, etc.) with communication services. Providing network services for unstructured environments employing interconnected mobile robotic systems involves several issues, from deployment and communication efficiency to topology control. In particular, it is necessary to guarantee the possibility of exchanging data among all the nodes in the network. Along these lines, connectivity maintenance is a well-studied topic in the field of decentralized multi-robot systems. The main approaches provide solutions for ensuring that, if the communication graph is initially connected, then it will remain connected, that is, if a link among two robots is active at time $t = 0$, than it will continue to be active as the system evolves, for time $t > 0$. Examples can be found in [2]–[6]. More recently, a few strategies reported in the literature solve the connectivity maintenance problem from a global point of view: single links are allowed to be added or removed, as long as the overall communication graph remains connected [7], [8].

However, the literature on connectivity maintenance does not generally consider robot failures, and hence does not provide robust solutions in this respect. Robots are prone to failures due to hardware or communication issues, and — as it is well known from the literature of Complex Networks — successive failures, particularly of agents playing a central role in the network topology, may easily lead to an inoperative or reduced service [9]–[12]. Despite that, the detection and mitigation of vulnerable topological configurations w.r.t. failures are mostly disregarded.

In the context of this work, a vulnerable state means that the network is prone to get disconnected if some nodes fail. Moreover, we consider robustness to failures as the system capacity to mitigate the effects of node failures through predictive actions that avoid topological configurations vulnerable to such effects. This contrasts with the standard definition of robustness used in the field of Robust Control, namely satisfactory (under some performance metric) operation, regardless of bounded parametric uncertainties.

Considering the problem relevance, we propose here a model that intends to overcome the downside of the connectivity maintenance approaches, using as reference the mechanism proposed in [7] and combining it with a recent approach that aims at enhancing the network robustness to failures based only on locally available information [13]. For this purpose, we reformulate the robustness improvement model, which is based on discrete-time systems, to a continuous time single integrator model to describe the motion of each robot.

The new combined control law strategy was validated comparing its performance with each of the individual control laws. The results demonstrate that the combined control law

was able to evolve the network topology to a more robust one regarding failures. In addition, its performance in a failure scenario was assessed, demonstrating its effectiveness to adapt the network topology to accommodate failures, postponing or even avoiding network fragmentation.

The rest of this paper is organized as follows. The system model, the background on network evaluation and the algebraic connectivity control law are presented in Section II. The relevance of the problem addressed here is discussed in details in Section II-D. Section III describes the combined model, and Section IV presents the simulation model and discusses the results.

II. Preliminaries

A. System model

Consider a multi-robot system composed of N robots that are able to communicate with other robots within the same communication radius R. The resulting communication topology is represented by an undirected graph $\mathcal{G}$ where each robot is a vertex (node) of the graph, and each communication link between two robots is an edge of the graph. Let each robot state be the position $p_i \in \mathbb{R}^m$ of the robot itself and $p = \left[p_1^T \dots p_N^T\right]^T \in \mathbb{R}^{Nm}$ be the state vector of the multi-robot system. Let each robot be modeled as a single integrator system, whose velocity can be directly controlled. Namely,

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

where $u_i \in \mathbb{R}^m$ is a control input.

Let $\mathcal{V}(\mathcal{G})$ and $\mathcal{E}(\mathcal{G}) \subset \mathcal{V}(\mathcal{G}) \times \mathcal{V}(\mathcal{G})$ be the vertex set and the edge set of graph $\mathcal{G}$, respectively. Moreover, let $W \in \mathbb{R}^{N\times N}$ be the weight matrix: each element w_{ij} is such that

$$w_{ij} = \begin{cases} w_{ij} > 0 & \text{if} \quad (i,j) \in \mathcal{E}(\mathcal{G}) \\ w_{ij} = 0 & \text{otherwise.} \end{cases} \tag{2}$$

Let $D = \text{diag}(\{k_i\})$ be the degree matrix, where k_i is the degree of the i-th node of the graph, i.e., $k_i = \sum_{j=1}^{N} w_{ij}$. The (weighted) Laplacian matrix of the graph is then defined as $L = D - W$.

B. Background on network properties

We will hereafter define some quantities that can be exploited for evaluating node and network connectivity and robustness to failures.

As is well known from algebraic graph theory, the Laplacian matrix of a graph $\mathcal{G}$ exhibits some remarkable properties regarding its connectivity [14]. Let λ_i, $i = 1, \dots, N$ be the eigenvalues of the Laplacian matrix, then:

- The eigenvalues can be ordered such that

$$0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N. \tag{3}$$

- Define now $\lambda = \lambda_2$. Then, $\lambda > 0$ if and only if the graph is connected. Therefore, λ is defined as the **algebraic connectivity** of the graph.
- Considering a weighted graph, λ is a non-decreasing function of each edge weight.

Even though the algebraic connectivity is more commonly used in the multi-robot systems literature for assessing the connectedness of a graph, it has been observed that in most real-world complex networks there is a large connected component together with a number of small components containing no more than a few percent of the nodes [15]. As the algebraic connectivity goes to zero as soon as the graph becomes disconnected, it does not provide further information about the network fragmentation. In this sense, the network connectivity $\mathcal{G}$ can be estimated by the relative size of the largest connected component, given by:

$$S(\mathcal{G}) = \frac{n_S}{N}, \tag{4}$$

where n_S is the number of nodes in the largest connected component and N is the number of nodes in the network. The *connected component* of a graph is a set of nodes such that a path exists between any pair of nodes in this set. For very large networks, this component is generally referred to as *giant component*. With a slight abuse of notation, we will hereafter define the **giant component** of a graph as its largest connected component, regardless of network size.

In addition to the algebraic connectivity and the giant component, we want to evaluate the number of node failures a network can stand before disconnecting. It is known that different nodes have different roles in maintaining the overall network connectivity. In special, the concept of *centrality* is usually exploited for identifying the most important nodes within a graph [16]. Several indicators can be found in the literature for defining centrality. In particular, referring to connectivity maintenance, we will consider the concept of *Betweenness Centrality* (BC) [17], which establishes higher scores for nodes that are contained in most of the shortest paths between every pair of nodes in the network.

For a given node i and pair of nodes j, l, the importance of i as a mediator of the communication between j and l can be established as the ratio between the number of shortest paths linking nodes j and l that pass through node i ($g_{jl}(i)$), and the total number of shortest paths connecting nodes j and l (g_{jl}). Then, the BC of a node i is simply the sum of this value over all pairs of nodes, not including i:

$$BC(i) = \sum_{j<l} \frac{g_{jl}(i)}{g_{jl}}. \tag{5}$$

Once the BC has been computed for all the nodes, it is possible to order them from the *most central* (i.e., the node with the highest BC value) to the *less central* (i.e., the node with the lowest BC value). Hence, let $[v_1, \dots, v_N]$ be the list of nodes ordered by descending value of BC.

According to the definition of centrality, removing the most central nodes might lead to network fragmentation. We therefore introduce the following definition of *Robustness level*.

Definition II.1 (Robustness level [13])**.** *Consider a graph $\mathcal{G}$ with N nodes. Let $[v_1, \ldots, v_N]$ be the list of nodes ordered by descending value of BC. Let $\varphi < N$ be the minimum index $i \in [1, \ldots, N]$ such that, removing nodes $[v_1, \ldots, v_i]$ leads to disconnecting the graph, that is, the graph including only nodes $[v_{\varphi+1}, \ldots, v_N]$ is disconnected.*

Then, the level of network robustness *of $\mathcal{G}$ is defined as:*

$$\Theta(\mathcal{G}) = \frac{\varphi}{N}. \tag{6}$$

$\diamond$

The level of robustness defines the fraction of central nodes that need to be removed from the network to obtain a disconnected network, i.e., $S(\mathcal{G}) < 1$. Small values of $\Theta(\mathcal{G})$ imply that the graph can lose connectivity in case of failures of a small fraction of its nodes. Therefore, increasing this value increases the robustness of the network.

Notice that $\Theta(\mathcal{G})$ is only an estimate of how far the network is from getting disconnected w.r.t. fraction of nodes removed. In fact, it might be the case that different orderings of nodes with the same BC produce different values of $\Theta(\mathcal{G})$.

From a local perspective, a heuristic for estimating the magnitude of the topological vulnerability of a node by means of information acquired from its 1-hop and 2-hops neighbors is proposed in [13].

Let $d(v, u)$ be the shortest path between nodes v and u, i.e., the minimum number of edges that connect nodes v and u. Subsequently, define $\Pi(v)$ as the set of nodes from which v can acquire information:

$$\Pi(v) = \{u \in V(G) : d(v, u) \leq 2\}.$$

Moreover, let $|\Pi(v)|$ be the number of elements of $\Pi(v)$. In addition, define $\Pi_2(v) \subseteq \Pi(v)$ as the set of the 2-hop neighbors of v, that comprises only nodes whose shortest path from v is exactly equal to 2 hops, namely

$$\Pi_2(v) = \{u \in V(G) : d(v, u) = 2\}.$$

Now define $L(v, u)$ as the *number of paths* between nodes v and u, and let $Path_\beta(v) \subseteq \Pi_2(v)$ be the set of v's 2-hop neighbors that are reachable through at most β paths, namely

$$Path_\beta(v) = \{u \in \Pi_2(v) : L(v, u) \leq \beta\}.$$

Thus, β defines the threshold for the maximal number of paths between a node v and each of its u neighbors that are necessary to include u in $Path_\beta(v)$. Therefore, using a low value for β allows to identify the most weakly connected 2-hop neighbors. Hence, the value of $|Path_\beta(v)|$ is an indicator of the magnitude of node fragility w.r.t. connectivity, and **the vulnerability level of a node regarding failures** is given by $P_\theta(v) \in (0, 1)$:

$$P_\theta(v) = \frac{|Path_\beta(v)|}{|\Pi(v)|}. \tag{7}$$

We will hereafter use $\beta = 1$, in order to identify 2-hop neighbors that are connected by a single path, which can represent a critical situation for network connectivity.

C. Algebraic connectivity control law

In [7], an approach was proposed to solve the global connectivity problem in a decentralized manner through the algebraic connectivity property. As it is well known, the algebraic connectivity approaches 0 when the network is poorly connected. Then, the control strategy ensures the network connectivity maintenance through a decentralized algebraic connectivity estimation.

Maintaining the connectivity entails designing a controller that ensures that, if the graph is connected at time $t = 0$, then it will remain connected $\forall t \geq 0$. Given the Laplacian matrix L, the connectivity is guaranteed if the algebraic connectivity λ is strictly greater than zero.

Define now $\epsilon > 0$ to be the desired lower-bound for the value of λ. The control strategy will then be designed to ensure that the value λ never goes below ϵ. An *energy function* is utilized for generating the decentralized connectivity maintenance control strategy.

Definition II.2. *An energy function*

$$V(\lambda) = V(\lambda(p)) : \mathbb{R}^{Nm} \mapsto \mathbb{R}$$

exhibits the following properties:

(P1) It is continuously differentiable $\forall \lambda > \epsilon$.
(P2) It is non-negative.
(P3) It is non-increasing with respect to λ, $\forall \lambda > \epsilon$.
(P4) It approaches a constant value, as λ increases.
(P5) It suddenly increases, as λ approaches $\epsilon > 0$, namely

$$\lim_{\lambda \to \epsilon} V(\lambda(p)) = \infty.$$

As an example, in [7] the following energy function was used:

$$V(\lambda) = \begin{cases} \coth(\lambda - \epsilon) & \text{if } \lambda > \epsilon \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

The control design essentially drives the robots to perform a gradient descent of $V(\cdot)$, in order to ensure connectivity maintenance. Namely, considering the dynamics of the system introduced in (1), the control law is defined as follows:

$$u_i = u_i^c = -\frac{\partial V(\lambda)}{\partial p_i} = -\frac{\partial V(\lambda)}{\partial \lambda}\frac{\partial \lambda}{\partial p_i}. \tag{9}$$

Since λ and its gradient are global quantities, the proposed control law is centralized. Decentralized implementation can be achieved replacing λ and its gradient with their estimates, computed by each robot in a decentralized manner utilizing the procedure proposed in [7].

D. Problem statement

The topological properties resulting from applications relying on dynamic networks are most of the time unknown, costly to estimate and change over time, adding uncertainty to the system behavior. Despite such variability, in general, these networks are often able to maintain most of the nodes into the giant component. However, this is not necessarily the case when such topological variability is biased towards nodes of high centrality that either leave or fail, or when the system

is facing successive or cascading failures. In these cases, the network can achieve a global state of vulnerability where its operation is completely compromised [10].

As an example, consider nodes 12 and 17 of the random network in Figure 1a. They are clearly playing a crucial role in the network communication. If any of them fails, the network efficiency to communicate will be degraded. If both fail, the network will get disconnected into two clusters, as illustrated in Figure 1b. Besides, notice that node 16 is in a vulnerable configuration since its communication with the entire network depends on node 9. Thus, for those applications where the interaction among nodes is critical to the system functionalities, such *harmful* topological configurations should be avoided to reduce the impact of possible node failures.

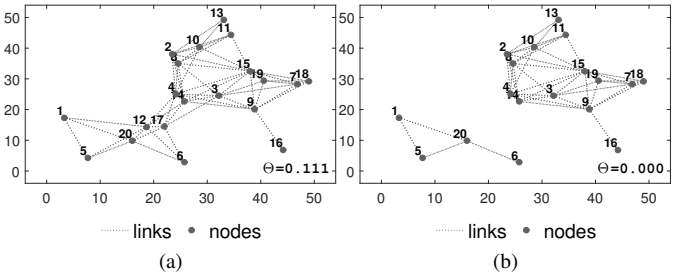

Fig. 1: A random network topology: initial configuration (a) and after two node failures (b).

As described in Section II-C, control strategies exist that aim at keeping nodes connected [7], [18]. This, however, does not ensure robustness to failures *per se*. For instance, despite the network shown on the right in Figure 2 exhibiting a larger algebraic connectivity than the network on the left, it is more affected by the central node failure.

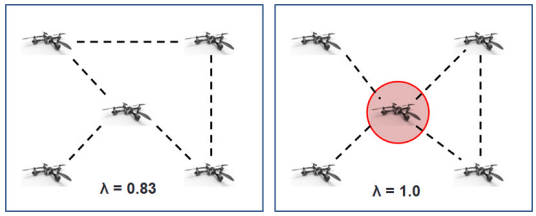

Fig. 2: Network with larger algebraic connectivity (right) is more affected w.r.t. central node failure.

For reinforcing that, the approach based on the algebraic connectivity maintenance described in Section II-C was applied to the network in Figure 1a, resulting in the topology illustrated in Figure 3a (see Section IV-A for model parametrization details). Despite a slight improvement in the network robustness, from Θ=0.11 to Θ=0.17, the vulnerable configurations highlighted before persist. We therefore argue that algebraic connectivity is not the most suitable property for evaluating or controlling the capacity of the network to accommodate failures.

In this regard, a continuous-time model based on the approach that aims at improving the network robustness proposed in [13] was also applied to the same network (Figure 3b) – see Section III for details. Notice that node 16 is now also directly connected to nodes 3, 17 and 18; and node 6 to node 14, improving the robustness from Θ=0.11 to Θ=0.33 and suppressing the vulnerable topological configurations. The downside of this strategy is not ensuring the network connectivity: this means that the network can become disconnected during (or as a result of) the adaptive process.

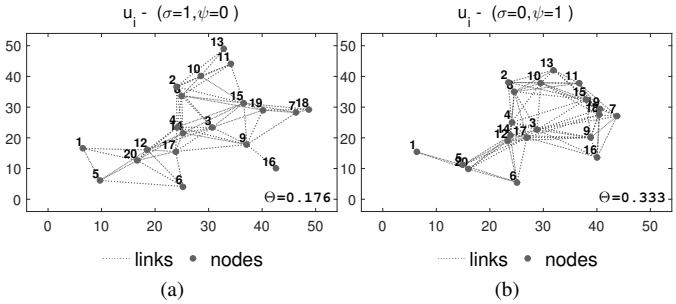

Fig. 3: Resulting network topology for the algebraic connectivity control law (a) and for the robustness improvement control law (b).

Summarizing, connectivity is a crucial requirement in decentralized multi-robot systems: in order to achieve a common objective, robots may need to exchange information. On the other hand, robots are likely to fail. The examples discussed in this Section demonstrate the importance of a robust connectivity maintenance strategy regarding failures. Given this evidence, in the next Section we aim at solving the following problem:

Problem. *Given a multi-robot system, design a local estimation procedure that allows each robot to assess its vulnerability level, based on locally available information, and subsequently exploit this estimate for controlling the motion of the robots in such a way that the overall robustness to failures is increased.*

III. Connectivity Maintenance and Robustness to Failure Improvement Control Law

This section describes the unified model that aims at improving the topological robustness of networks to failures while ensuring the connectivity maintenance. In particular, considering the dynamic introduced in (1), we define the following control law

$$u_i = \sigma u_i^c + \psi u_i^r \tag{10}$$

where u_i^c is the connectivity maintenance control law introduced in (9), and u_i^r is an additional control law that aims at improving robustness to failures. Moreover, $\sigma, \psi \geq 0$ are design parameters that represent control gains. Naturally, setting either σ or ψ to zero removes the effect of the corresponding control law. Conversely, if both parameters are larger than zero, both control actions are simultaneously active.

Based on the definition of the node vulnerability level, given by (7), we now introduce a control law u_i^r that leads to improving the network robustness.

Assume that the i-th robot identifies itself as vulnerable. In this case, the aim of the control strategy is to increase the number of links towards the 2-hop neighbors of i that are in $Path_\beta(i)$. Hence, define $x_\beta^i \in \mathbb{R}^m$ as the barycenter of the positions of the robots in $Path_\beta(i)$, namely

$$x_\beta^i = \frac{1}{|Path_\beta(i)|} \sum_{j \in Path_\beta(i)} x_j. \tag{11}$$

Considering the dynamics of the system introduced in (1), the control law is defined as follows:

$$u_i^r = \frac{x_\beta^i - x_i}{\left\| x_\beta^i - x_i \right\|} \alpha \tag{12}$$

where $\alpha \in \mathbb{R}$ is the linear velocity of the robots, that we assume constant for the sake of simplicity.

This control law drives vulnerable robots towards the barycenter of the positions of robots in their $Path_\beta$, thus decreasing the distance between them and eventually creating new edges in the communication graph. It is worth noting that (7) provides a decentralized methodology for each robot to evaluate its vulnerability level.

The control law in (12) is applied in a probabilistic manner, with higher probability for those robots i whose value $P_\theta(i)$ is high. This is obtained comparing $P_\theta(i)$ with a random number $r \in (0, 1)$: if $P_\theta(i) > r$, then the i-th robot considers itself as vulnerable, and applies the control law in (12).

We will hereafter analyze the performance of the proposed combined control law introduced in (10). It is worth remarking that the control law u_i^c was proven in [7] to guarantee connectivity maintenance in the absence of unexpected failures.

IV. Simulation

A. Simulation model

The proposed unified control strategy was validated using a simulation model, developed in Matlab by the authors. The model encompasses the benchmark networks, the control law parametrization, and the protocol to evaluate its performance.

The benchmark networks were generated through a random positioning of a variable number N of robots in $\mathbb{R}^2$, over a bounded area of size $\mathcal{A}$. Given the communication radius R, the communication network is then generated, based on the relative positions among the robots. For this simulation we set $N = 20$, $\mathcal{A} = 50$ and $R = 16$.

For evaluating the control law performance we consider the scenario where networks can evolve from a potentially vulnerable topology to a more robust one without perturbation and another where networks are experiencing disturbances (*i.e.*, node failures). For both scenarios, the total simulation time is defined as t seconds and at discrete time intervals t_i the network properties are measured. For the second, in addition to the properties computation, the most central node regarding the BC ranking, as defined in (5), is removed from the network. For this setup we consider the total simulation time as 10 seconds and the time interval between t_i and t_{i+1} as 1 second, $\forall i$. Besides, repeated experiments were carried out in order to assess the performance of the proposed methodology in a statistically sound manner (# in Table I indicates the number of repetitions). The parametrization settings for each control law is also presented in Table I.

TABLE I: Simulation settings

model			u_i^c	u_i^r	
t	t_i	#	ϵ	β	α
10	1	200	0.25	1	$0.15R$

In addition, the performance of the combined control law strategy was confronted with each individual control law by setting values for the σ and ψ gains. For instance, for the σ=0 and ψ=1 combination, only the robustness improvement control law is active, for σ=1 and ψ=0 only the algebraic connectivity control law is active, for σ=2 and ψ=1 both controllers are active but the algebraic connectivity control law is more relevant. Take into account that any combination of positive gains leads to the desired behavior, the gain settings should be defined according to the application requirement. Here, we assume that the robustness improvement must not overpower the network connectivity to produce a more robust network. Thus, the σ=2 and ψ=1 were considered to demonstrate the combined control law performance. We also evaluate the scenario where no control law is active (σ=0, ψ=0). Table II summarizes the gain combinations evaluated.

TABLE II: Gain settings

σ	ψ
0	0
0	1
1	0
2	1

B. Simulation results

This section presents the simulation results according to the model defined above. Regarding the robustness to failures improvement scenario, Figure 4 illustrates, for each gain combination, the robustness level and the algebraic connectivity values, averaged over 200 networks, at each time interval. As expected, the simulation results demonstrate the robustness improvement for ψ=1, i.e., when the robustness control law is active. Besides, the combined control law achieved a better performance than the robustness improvement control law, with the advantage of ensuring network connectivity during the adaptive process.

Figures 5 and 6 demonstrate the combined control law efficiency to evolve the initial network topologies (on the left) to a more robust one (on the right). In figure 5, notice that nodes 13, 15 and 17 are notably vulnerable. On the other hand, the communication in the network illustrated in Figure 6 is relying on nodes 11, 15 and 16. Failures of any of these nodes may lead the network to a critical state of connectivity. For both examples, the harmful topological configurations were

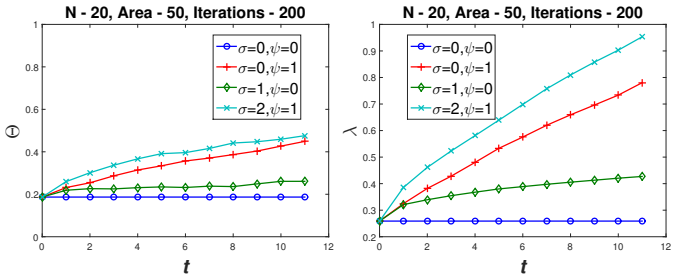

Fig. 4: The robustness level (Θ) and the algebraic connectivity (λ) evolution.

overcome by applying the combined control law. On the other hand, it does not take into account the coverage problem or a minimal distance between a pair of nodes. It means that, for a scenario with no failures some of the network nodes are likely to get close. For addressing that, a mechanism that aims at enhancing the environment coverage and/or a minimal distance parameter can be added to the control law introduced in (12).

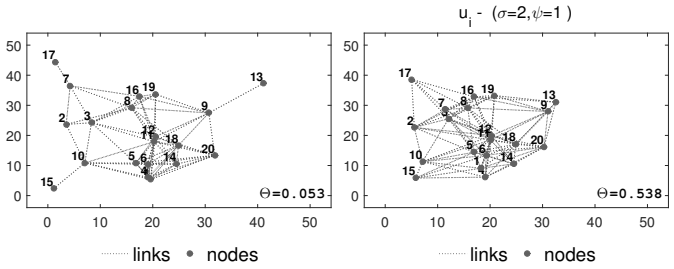

Fig. 5: Example of the combined control law performance.

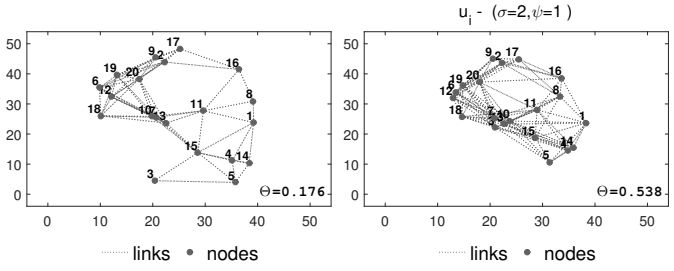

Fig. 6: Example of the combined control law performance.

Consider now the results for the failures scenario (Figure 7). Despite failures of central elements, the combined control strategy was able, on average, to maintain the robustness level and the algebraic connectivity values.

The giant component results highlight the impact of node failures on its value when there is no active adaptive mechanism and when the combined control law is active, blue line (σ=0, ψ=0) and cyan line (σ=2, ψ=1), respectively. For the first, a fraction of 0.32 nodes are into the giant component after 12 seconds of simulation, in contrast to a fraction 0.84 for the latter, clearly demonstrating the evolution to more resilient networks, able to react to unexpected failures through

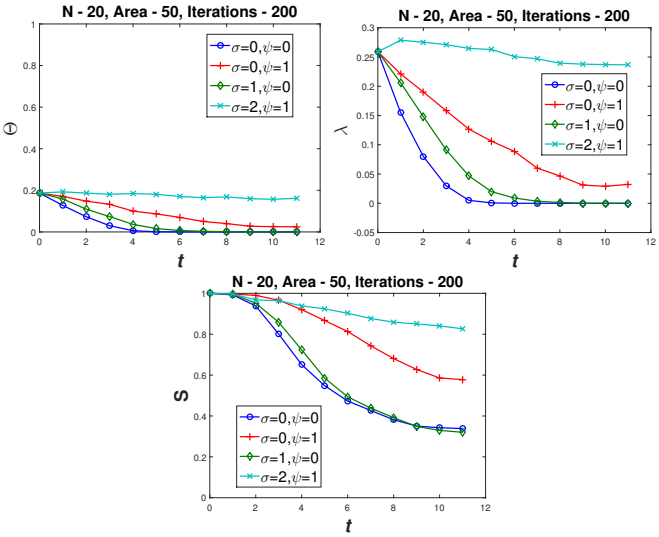

Fig. 7: The robustness level (Θ), the algebraic connectivity (λ) and the giant component (S) evolution under failures.

the adaptive mechanism. It is important to emphasize that in failure situations there is no way of ensuring that the network will remain connected because the failure probability distribution is usually unknown. Thus, the giant component evolution analysis is crucial to properly evaluate the mechanism performance.

For reinforcing the importance of failure mitigation, consider the network topologies shown in Figures 5 and 6 and their resulting topologies at the ending of the simulation time for σ=0, ψ=0 (on the left) and σ=2, ψ=1 (on the right) gain settings, illustrated in Figures 8 and 9, respectively. In such cases, the combined control law was able to maintain the networks connected even in the case of unexpected failures of central nodes, demonstrating its capacity to be reactive, despite a potentially vulnerable initial topological configuration.

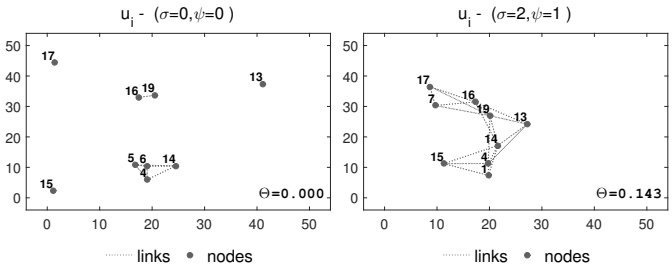

Fig. 8: Example of the combined control law performance under failures.

Finally, for exemplifying the network evolution during the attack simulation process, Figure 10 exhibits snapshots of the initial network shown in Figure 6 at t_i=4 simulation time. It is possible to observe the algebraic control law ineffectiveness to maintain the network nodes connected, i.e., the network is fragmented for σ=1 and ψ=0. Moreover, the robustness control law (σ=0, ψ=1) was not able to overcome

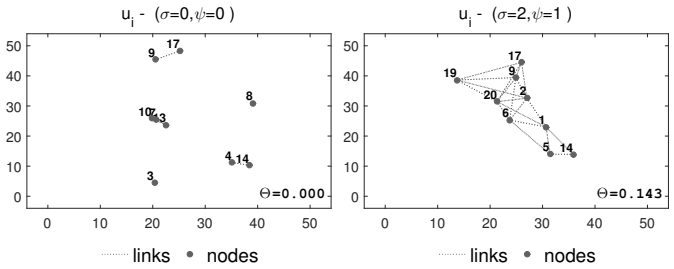

Fig. 9: Example of the combined control law performance under failures.

the vulnerable configuration, represented by the red nodes, i.e., those that defined themselves as vulnerable according to (1) and a random probability (see Section III). In contrast, the combined mechanism (σ=2, ψ=1) clearly provided a more resilient network. Some additional examples can be freely viewed online on https://youtu.be/dThj_-BdkJw.

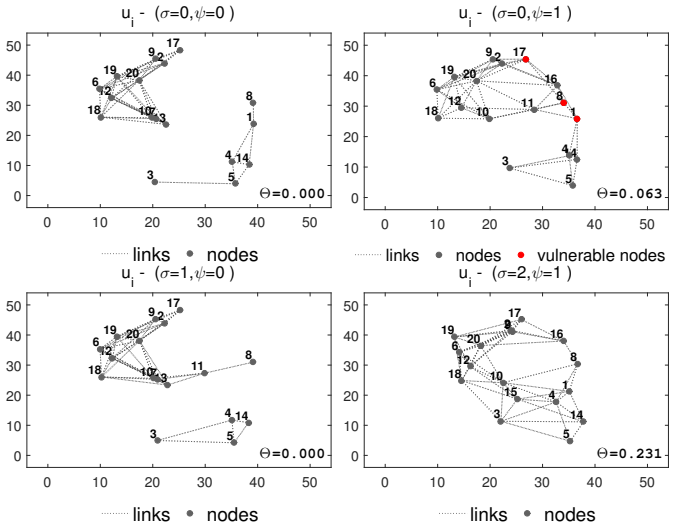

Fig. 10: Network topology at 4 seconds of simulation.

V. Conclusions

We proposed in this paper a mechanism that locally and autonomously combines connectivity maintenance and robustness w.r.t. node failures in a multi-robot setting. Both aspects are important in the design of cooperative multi-robot systems, the first for guaranteeing communication among all robots, the second for reducing the network vulnerability in the case of failures that can happen as a consequence of hardware unreliability or environmental hostility. The combined control law performance was experimentally validated from two perspectives: as a reactive mechanism, it was able to accommodate ongoing failures and postpone or avoid network fragmentation, and as a proactive mechanism, the network topology evolved towards a more robust topology w.r.t. failures. As ongoing work, we are now implementing the algorithms in a realistic multi-robot setting, also considering additional constraints such as obstacle avoidance. For future work, we intend to extend our approach considering additional requirements, such as coverage, energy consumption and heterogeneous robotic teams.

Acknowledgments

The authors thank FAPESP for the financial support to carry out this research (procs. no. 2012/25058-9 and 2014/13800-8). Carlos H. C. Ribeiro also thanks CNPq (proc. no. 303738/2013-8) and FAPESP (proc. no. 2013/13447-3).

References

[1] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue," *Robotics Automation Magazine, IEEE*, vol. 19, no. 3, pp. 46–56, Sept 2012.

[2] M. Ji and M. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Transactions on Robotics*, 2007.

[3] G. Notarstefano, K. Savla, F. Bullo, and A. Jadbabaie, "Maintaining limited–range connectivity among second–order agents," in *Proceedings of the American Control Conference*, 2006, pp. 2134–2129.

[4] Y. Cao and W. Ren, "Distributed coordinated tracking via a variable structure approach – part I: consensus tracking. part II: swarm tracking," in *Proceedings of the American Control Conference*, 2010, pp. 4744–4755.

[5] M. A. Hsieh, A. Cowley, V. Kumar, and C. J. Talyor, "Maintaining network connectivity and performance in robot teams," *Journal of Field Robotics*, vol. 25, no. 1, pp. 111–131, 2008.

[6] A. Ajorlou, A. Momeni, and A. G. Aghdam, "A class of bounded distributed control strategies for connectivity preservation in multi–agent systems," *IEEE Transactions on Automatic Control*, vol. 55, pp. 2828–2833, 2010.

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

[8] L. Sabattini, C. Secchi, and N. Chopra, "Decentralized estimation and control for preserving the strong connectivity of directed graphs," *IEEE Transactions on Cybernetics*, 2014.

[9] R. Albert, H. Jeong, and A.-L. Barabasi, "Error and attack tolerance of complex networks," *Nature*, vol. 406, no. 6794, pp. 378–382, July 2000.

[10] C. Ghedini and C. H. C. Ribeiro, "Rethinking failure and attack tolerance assessment in complex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 23–24, pp. 4684–4691, November 2011.

[11] Z. He, S. Liu, and M. Zhan, "Dynamical robustness analysis of weighted complex networks," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 18, pp. 4181 – 4191, 2013.

[12] M. Manzano, E. Calle, V. Torres-Padrosa, J. Segovia, and D. Harle, "Endurance: A new robustness measure for complex networks under multiple failure scenarios," *Computer Networks*, vol. 57, no. 17, pp. 3641 – 3653, 2013.

[13] C. Ghedini, C. Secchi, C. H. C. Ribeiro, and L. Sabattini, "Improving robustness in multi-robot networks," in *Proceedings of the IFAC Symposium on Robot Control (SYROCO)*, Salvador, Brazil, aug. 2015.

[14] C. Godsil and G. Royle, *Algebraic Graph Theory*. Springer, 2001.

[15] P.-Y. Chen and K.-C. Chen, "Information epidemics in complex networks with opportunistic links and dynamic topology," in *Global Telecommunications Conference (GLOBECOM)*. IEEE, Dec 2010, pp. 1–6.

[16] D. Koschützki, K. A. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl, and O. Zlotowski, "Centrality indices," in *Network analysis*. Springer, 2005, pp. 16–61.

[17] S. Wasserman, K. Faust, and D. Iacobucci, *Social Network Analysis : Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, November 1994.

[18] P. Robuffo Giordano, A. Franchi, C. Secchi, and H. H. Bülthoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 299–323, 2013.