Disruption of Connectivity Graphs in Uncertain Multi-Agent Systems

Brian Reily[1], Caden Coniff[2], John G. Rogers[1] and Christopher Reardon[2]

***Abstract*—Multi-agent systems have become ever-present in modern society, whether as multi-robot teams, sensor networks, or social networks. While ensuring the connectivity and robustness of multi-agent systems has seen extensive research, the problem of disrupting the connectivity of a multi-agent system has remained largely unaddressed. Yet, this capability can be essential in certain applications, such as responding to a hostile multi-robot system or controlling the flow of disinformation in a social network. In this paper we propose a novel method to disrupt the connectivity of a multi-agent system with uncertain relationships. We represent a multi-agent system as a graph, with edges denoting the probability of communication between agents. We introduce the problem of identifying a subgraph which minimizes the overall connectivity of the multi-agent system. We formulate a novel approach to identify optimal sets of vertices to remove by approximating a minimization of the algebraic connectivity, given constraints on the number of vertices to disconnect. We show through evaluation on simulated multi-agent systems that our approach is able to effectively disrupt the connectivity of a multi-agent system, and discuss its comparative complexity to existing approaches while attaining these superior results.**

***Keywords*—distributed systems, multi-agent systems, connectivity disruption**

I. Introduction

Multi-agent systems have become critical in many applications, and in order to function effectively or accomplish goals, they must be able to work collectively. Multi-robot systems can then provide sensor coverage and respond to disaster areas [1] or navigate collectively and orient themselves [2], and multi-agent systems such as social networks can efficiently transmit information [3]. To do all of this, these multi-agent systems must be able to communicate and transfer information. Multi-agent systems require the ability to effectively communicate, coordinate, and attain consensus in order to operate and accomplish their objectives. Even distributed algorithms often require at least partial information.

However, in many cases the disruption of connectivity in multi-agent systems can be a critical capability. Social media companies may want to disrupt the flow of false or misleading information by targeting a small number of influential users as opposed to mass removal. Security and defense systems may need to respond to unknown and possibly hostile multi-robot systems by prioritizing robots to target - consider a multi-robot system consisting of drones flying into an airport airspace, where certain agents need to be destroyed in order to interdict the attack. In both cases, the exact network structure of the multi-agent system may be uncertain - i.e., which social network users actually trust each other for information, which robots have the capability to communicate, or which robots have the need to communicate. The capability to disrupt the connectivity in uncertain multi-agent systems is critical.

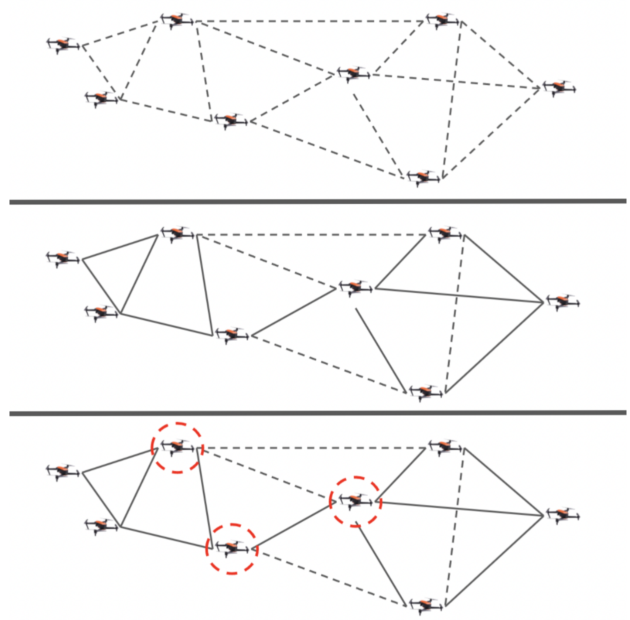


Fig. 1. [Best Viewed in Color] An uncertain multi-agent system can be represented by a number of possible connectivity edges (top frame). However, the actual connectivity graph for this multi-agent system may involve only a subset of these possible connections that actually exist (middle frame, in solid instead of dashed). Our approach then identifies a set of vertices to remove (circled in red) that optimally disrupt the connectivity graph of the uncertain multi-agent system (bottom frame).

The areas of multi-agent connectivity, coordination, and consensus have seen extensive research [4], [5]. In particular, connectivity maintenance has been addressed to maintain a certain k-connectivity [6] or algebraic connectivity [7] within a multi-agent or multi-robot system. Despite this focus, little attention has been given to the problem of disrupting connectivity in multi-agent systems. While some research has addressed the problem of removing redundant links [8] or slowing consensus rates [9], the problem of actively disrupting the connectivity in an uncertain multi-agent system has been largely unaddressed.

In this paper, we introduce a novel approach for the

[1]DEVCOM U.S. Army Research Laboratory. {brian.j.reily, john.g.rogers59}.civ@army.mil.
[2]Department of Computer Science, University of Denver. {caden.coniff, christopher.reardon}@du.edu.

disruption of connectivity in uncertain multi-agent systems. We first define a probabilistic representation of the multi-agent system based on observations of the positions of agents within it. We then formulate the problem of graph disruption by introducing the problem of selecting a subset of vertices that minimizes the algebraic connectivity of the multi-agent system. Then, we redefine and approximate this optimization problem by identifying the optimal selection of vertices based on their Fiedler connectivity cost. Through this, we identify an ordering of vertices to remove in order to optimally disrupt the graph. Illustrated in Figure 1, we first consider the distances between agents (top frame). We then probabilistically estimate the actual network connections between agents (middle frame). Finally, our approach then identifies vertices to remove in order to optimally disrupt the graph.

This paper has two important contributions.

- First, we present a principled formulation to select vertices based on their connectivity values in order to optimally disrupt a given graph.
- Second, we show through extensive simulation that our approach is able to significantly outperform other methods in graph disruption while operating at similar computational complexity.

II. Related Work

When we consider the problem of connectivity disruption, we must first consider the opposite problem of maintaining or increasing the connectivity of multi-agent systems, which has seen extensive research. Then, we look at the limited research into disrupting connectivity, which has primarily focused on removing redundant communication links, prioritizing network optimization over interrupting communications.

A. Multi-Agent Connectivity

Maintaining and increasing the connectivity of multi-agent systems is an area of active research. A variety of methods have been created to maintain minimum k-connectivity [6], efficiently do maintenance and perform repairs on k-connected networks without allowing network partitioning or longer routing paths from node failures [10], and maintain connectivity metrics under a variety of circumstances, such as walled [7] and adversarial [11] circumstances. The methods found above, in order to accomplish the variety of goals, employed and utilized sub-graph structure [6], [7], gradient ascent/descent [11]–[13], and game theory [14].

Sub-graph structure approaches create and maintain k-connectivity by enforcing additional constraints upon the multi-robot system to ensure robustness within the network [6]. Another approach builds efficient connectivity in a walled environment through use of a k-connectivity matrix and by optimizing the Fielder value of the weighted Laplacian [7]. This method maintains quality communication even in scenarios where a hop-count constraint cannot be satisfied.

Many other methods of maintaining or creating connectivity in multi-agent systems rely upon a gradient ascent- or descent-like procedure and measure of algebraic connectivity. Some control global connectivity by using gradient descent to control the local connectivity of 'critical' nodes [12]. Similar work instead uses a gradient control strategy, still based on algebraic connectivity, to maintain global connectivity without managing local connectivity [15]. In adversarial conditions, one metric combined a decentralized, iterative approach to maintaining algebraic connectivity with game theory, resulting in a Nash Equilibrium and equilibrium network [11]. Finally, [14] addressed the issue of maintaining connections across realistic networks with uniform-fading and disk based communication methods. While these provide insight to the ways graphs can stay connected, they do not provide algorithms or analysis of how these graphs may be disconnected.

B. Connectivity Disruption

The disruption of connectivity has mostly been researched in the cases of removing redundant links from multi-agent systems or general networks. Removing redundant links can be done for a variety of purposes. Some algorithms can improve the efficiency of resource usage based on graph spectra [8]. Others greedily remove links while maintaining synchronization speeds [16]. These works focus on removing links and nodes while preserving graph connectivity and therefore would not identify links that would suit the goals of our graph disruption work.

Work on identifying and ranking nodes and links may also be of assistance to us. One such method utilizes 'k-shell' (a measure of influence) methods of determining node coreness in a network [17]. This method aims to accurately identify nodes' positional importance and *spreading* influence, which can aid in disruption. Another method identifies a lower limit on transition probabilities, where values above this point no longer impact the convergence rate of the graph [18]. This work again does not focus on disruption, but is useful in identifying nodes and links that are possibly important to disrupt.

Despite this research, little has been done to actively disrupt the communication, coordination, and cohesion of multi-robot systems. Research is needed to effectively analyze the structure of multi-agent systems in order to most capably disrupt them.

III. Our Proposed Approach

Notation. In this paper, we denote matrices $\mathbf{X} \in \mathbb{R}^{n \times m}$ with boldface uppercase letters, and vectors $\mathbf{x} \in \mathbb{R}^n$ with boldface lowercase letters. A matrix element in the i-th row and j-th column is indicated as x_{ij} and the i-th vector element is x_i.

A. Background

A natural way to describe a multi-agent system is with a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where a vertex $v_i \in \mathcal{V}$ represents an agent and an edge $e_{ij} \in \mathcal{E}$ represents the relationship between the i-th and j-th agents. We can then represent the edge set $\mathcal{E}$ with an incidence matrix $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{n \times m}$, where n is the number of agents (or vertices) and m is the number of edges present in the multi-agent system. If the i-th agent is connected to the j-th agent via the k-th edge in the system, then $a_{ik} = 1$ and $a_{jk} = -1$. Given this, we introduce a final representation of the graph with the Laplacian matrix $\mathbf{L} \in \mathbb{R}^{n \times n}$. The Laplacian matrix is commonly defined as $\mathbf{L} = \text{degree}(\mathbf{A}) - \mathbf{A}$, with $\mathbf{A}$ being the adjacency matrix of the graph. When $\mathbf{A}$ instead represents the incidence matrix of the graph, the Laplacian is then defined as $\mathbf{L} = \mathbf{A}\mathbf{A}^\top$.

A key measure of a graph's connectivity is its Fiedler value [19], also known as its algebraic connectivity. The Fiedler value is denoted as the second-smallest eigenvalue of the Laplacian matrix of the graph. If we consider the eigenvalues to be ordered such that $\lambda(\mathbf{L}) = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$, where λ_i is the i-th smallest eigenvalue, then we can define $\lambda_2(\mathbf{L})$ to be the algebraic connectivity of the graph $\mathcal{G}$ described by $\mathbf{L}$. Algebraic connectivity reaches a minimum value of 0 when the graph is disconnected. The algebraic connectivity of a graph is also directly associated with its ability to reach consensus [20].

We can then formulate the problem of graph disruption by vertex removal as that of finding a subgraph that minimizes the algebraic connectivity:

$$\min_{\hat{\mathcal{G}}} \lambda_2(\hat{\mathbf{L}}) \quad \text{s.t. } \hat{\mathcal{G}} \subseteq \mathcal{G} \tag{1}$$

where $\hat{\mathbf{L}}$ describes the subgraph $\hat{\mathcal{G}}$ and the removed vertices being those in $\mathcal{G}$ but not in $\hat{\mathcal{G}}$.

B. Uncertain Graphs

In this paper, we specifically consider the more challenging condition of *uncertain* graphs, as opposed to graphs that represent certain communication links between agents. Given an observation of a multi-agent system consisting of n agents, we may only have access to the spatial positions of each agent, and not the internal communication or coordination structure or the specific capabilities of individual agents. I.e., we may know that agents i and j are some distance apart, but are unsure whether this means these agents can communicate or sense each other.

Due to this, we initially consider a graph describing spatial relationships between agents $\mathcal{G}^{distance}$, represented by an adjacency matrix $\mathbf{D} = [d_{ij}] \in \mathbb{R}^{n \times n}$ where d_{ij} denotes the actual observed physical distance between the i-th and j-th agents. From this, we need to generate a probabilistic incidence matrix given the distances in $\mathbf{D}$. We rely on two assumptions: a) when we observe the multi-agent system, it is at least 1-connected (that is, a k-connected graph is one that remains connected when fewer than k vertices are removed); and, b) the larger a distance d_{ij} is, the less likely it is to represent an actual communication or coordination link.

We define a probabilistic incidence matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$. Each column $\mathbf{a}_k$ represents an edge, where two elements are non-zero and the remaining elements are zero. Given a column $\mathbf{a}_k$ that describes the link between agents i and j, a_{ik} represents the probability that the i-th agent is able to communicate with the j-th agent, given the spatial distance d_{ij}. Accordingly, instead of this value being 1 as in a typical incidence matrix, this value is bounded such that $0 \leq a_{ik} \leq 1$ (with the corresponding entry a_{jk} is bounded $-1 \leq a_{jk} \leq 0$). All other values a_{mk} where $m \neq i$ and $m \neq j$ are equal to 0.

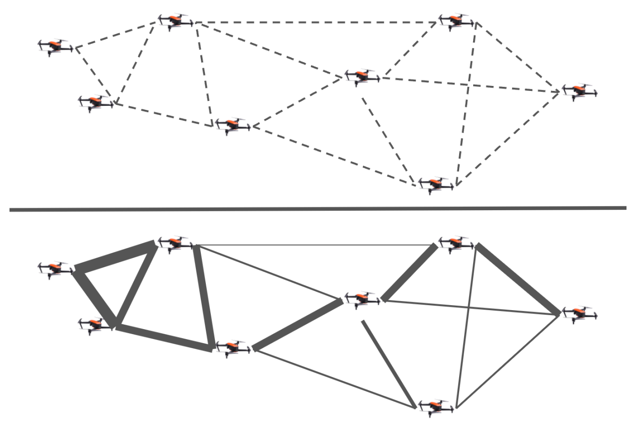


Fig. 2. Edges in the multi-agent system can be represented by their probability of providing communication and connectivity (possible edge probabilities depicted in the figure via line thickness).

We define elements in this probabilistic incidence matrix as:

$$a_{ik} = \frac{c_{max} - \frac{1}{d_{ij}}}{c_{max} - c_{min}} + \epsilon \tag{2}$$

visualized in Figure 2, where $c_{max} = \frac{1}{\max(\mathbf{D})}$, $c_{min} = \frac{1}{\min(\mathbf{D})}$ such that $\mathbf{D}$ remains k-connected, and where

$$a_{jk} = -a_{ik} \text{ if } j > i \tag{3}$$

to maintain the standard incidence matrix notation, where an edge k between agents i and j is represented by $a_{ik} = w$ and $a_{jk} = -w$, where w denotes the value of the edge.

C. Vertex Selection

Given the above problem definition, the goal of our approach is to identify a subgraph $\hat{\mathcal{G}} \subseteq \mathcal{G}$ that minimizes the connectivity of $\mathcal{G}$ while removing only a limited number of vertices, denoted as r. The number of vertices that must remain in the graph is then denoted as q, where $q = n - r$. We select this subgraph with a *selection* matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$, where off-diagonal elements are equal to 0 and diagonal elements x_{ii} represent the importance of the i-th vertex for the overall graph connectivity.

We can then consider minimizing the algebraic connectivity of the graph defined by $\mathbf{A}$, redefining Eq. (1) as:

$$\min_{\mathbf{X}} \lambda_2(\mathbf{X}\mathbf{A}\mathbf{A}^\top\mathbf{X}) \tag{4}$$

Here, we select a subgraph $\hat{\mathcal{G}}$ through the Laplacian matrix $\hat{\mathbf{L}} = \mathbf{X}\mathbf{A}\mathbf{A}^\top\mathbf{X}$.

The problem defined in Eq. (4) is difficult to directly minimize, so we instead solve an alternate approximate version of the problem. We introduce an incidence cost matrix $\mathbf{F} \in \mathbb{R}^{n \times m}$, where an element f_{ij} corresponds to the *Fiedler cost* of the corresponding incidence matrix entry a_{ij}. We calculate this cost from the Fiedler vector, which is the eigenvector associated with the Fiedler value. The Fiedler vector $\mathbf{z} \in \mathbb{R}^n$, due to its ties to the algebraic connectivity, can be used to partition a graph. We utilize it to approximate the connectivity represented by each edge in the graph:

$$\mathbf{F} = [f_{ij}] = \frac{1}{\text{abs}(\mathbf{z}_i - \mathbf{z}_j)} \tag{5}$$

The incidence cost f_{ij} of each edge in $\mathcal{G}$ is equal to the inverse of the difference between each node's value in the Fiedler vector.

To solve for the optimal $\mathbf{X}$ which minimizes the algebraic connectivity, we consider its diagonal as a selection vector $\mathbf{x}$:

$$\mathbf{x} = \text{diag}(\mathbf{X}) \tag{6}$$

As q vertices must remain in the selected subgraph, we enforce that the sum of values in $\mathbf{x}$ must equal q:

$$\|\mathbf{x}\|_1 = q$$

or, more easily calculated:

$$\mathbf{x}^\top \mathbf{1} = q \tag{7}$$

where $\mathbf{1}$ is a vector of 1s that is appropriately dimensioned.

We then reformulate the minimization problems in Eqs. (1) and (4) as that of identifying a minimum selection of vertices with the selection vector $\mathbf{x}$ from the incidence cost matrix $\mathbf{F}$:

$$\min_{\mathbf{x}} \|\mathbf{F}^\top \mathbf{x}\|_2^2 \tag{8}$$

We bound $\mathbf{x}$ such that $0 \leq x_i \leq 1$ and introduce Eq. (7) as a regularization term to ensure q vertices remain in the graph:

$$\min_{\mathbf{x}} \|\mathbf{F}^\top \mathbf{x}\|_2^2 + \beta(\mathbf{x}^\top \mathbf{1} - q)^2 \quad \text{s.t. } 0 \leq \mathbf{x} \leq 1, \mathbf{x} = \text{diag}(\mathbf{X}). \tag{9}$$

where β is a hyperparameter controlling the influence of the squared regularization term based on Eq. (7).

After finding the optimal $\mathbf{x}^*$ to minimize Eq. (9), it can be sorted to identify the order in which vertices should be removed. That is, an ideal scenario may consist of removing a single vertex and $\mathbf{x}^*$ results in the opposite of a one-hot vector (i.e, a single entry is 0 and all other entries are 1) - in this case, it is simple to see which vertex to remove. However, to remove multiple vertices and deal with the case where entries are somewhere between 0 and 1, we can sort $\mathbf{x}^*$:

$$\hat{\mathbf{x}} = \text{sort}(\mathbf{x}^*) \tag{10}$$

Now, the first vertex to remove corresponds to $\hat{\mathbf{x}}_1$, the second corresponds to $\hat{\mathbf{x}}_2$, and so on until we have reached $\hat{\mathbf{x}}_r$.

IV. Results

A. Experimental Setup

We compare our proposed approach to multiple other methods based on existing graph connectivity metrics.

- *Fiedler Low*: Inspired by the method of normalized cuts [21], this approach disrupts the graph by removing the vertex closest to the sign boundary in the Fiedler vector.

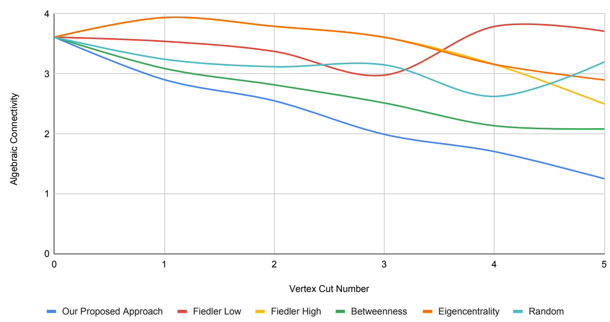


Fig. 3. [Best Viewed in Color] Algebraic connectivity over multiple vertex cuts for small ($n = 10$) multi-agent systems. Our approach, shown in blue, outpaces compared approaches.

- *Fiedler High*: Inspired by the heuristic presented to grow well-connected graphs in [22], this approach disrupts the graph by removing the vertex corresponding to the largest absolute value in the Fiedler vector.
- *Eigencentrality*: This approach disrupts the graph by removing the vertex with the highest eigenvector centrality, which is a measure of a vertex's influence in a graph. The eigenvector centrality of a vertex v is denoted as:

$$\text{eig}(v_i) = \frac{1}{\lambda} \sum_{v_j \in N(v_i)} \text{eig}(v_j) \tag{11}$$

where $N(v)$ denotes the neighborhood of a vertex v and λ is a constant.
- *Betweenness*: This approach removes the vertex with the highest betweenness centrality, which weights vertices by the number of times they appear on shortest paths between other pairs of vertices. Mathematically:

$$\text{betweenness}(v_i) = \sum_{k \neq i \neq j} \frac{\sigma_{kj}(v_i)}{\sigma_{kj}} \tag{12}$$

where σ_{kj} is the number of shortest paths between the h-th and j-th vertices and $\sigma_{kj}(v_i)$ is the number of shortest paths that pass through vertex i.
- *Random*: This approach selected a random permutation of vertices, with no weight given to connections.

These approaches are all calculated given the full approximated graph; e.g., edges are calculated by Eq. (2) and vertices are sorted for each method, resulting in a vertex list analogous to $\hat{\mathbf{x}}$.

We compare our proposed approach and these mentioned approaches based on the metric of algebraic connectivity. As defined earlier, this equals $\lambda_2(\hat{\mathbf{L}})$. Given that this is a typical representation of graph connectivity [19] and is an integral mathematical part of the compared methods *Fiedler Low*, *Fiedler High*, and *Eigencentrality*, this is an appropriate metric to evaluate the disruption of graph connectivity.

B. Evaluation

We evaluate our approach on three variously sized simulated multi-agent systems.

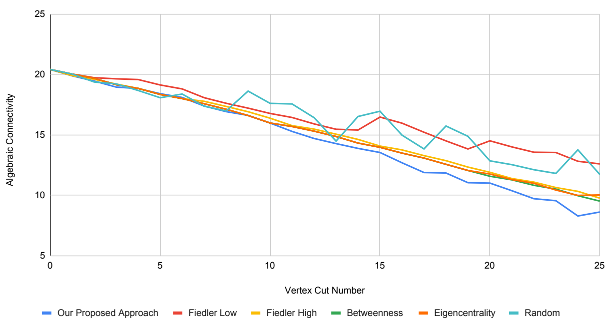


Fig. 4. [Best Viewed in Color] Algebraic connectivity over multiple vertex cuts for medium ($n = 50$) multi-agent systems. Our approach, shown in blue, outpaces compared approaches.

systems consisting of 10, 50, and 100 agents, generated statically in three dimensions, over 100 simulated multi-agent systems. While the algorithms are only aware of the probabilistic representation of the uncertain graph, each system has a ground truth communication threshold c_{gt} defined to be

$$c_{gt} = \frac{c_{max} + c_{min}}{2} \tag{13}$$

where c_{min} and c_{max} are defined as they are for Eq. (2). This ground truth communication threshold is used for the evaluation of the algebraic connectivity.

Thus, in evaluation we compute a probabilistic incidence matrix $\mathbf{A}$, based on the distance matrix $\mathbf{D}$. We utilize this incidence matrix $\mathbf{A}$ to compute all of our compared approaches, including our proposed one. For each, we use $\mathbf{A}$ to identify a vertex cut list, and then compute an evaluation metric on each step of the cut list.

We first present results for small-sized multi-agent systems, consisting of 10 simulated agents. The algebraic connectivity for these systems are seen in Figure 3 for 0 (indicating the average initial algebraic connectivity) to 5 vertex cuts. We can see that our approach, displayed in blue, provides an immediate improvement over other approaches. This advantage increases as the number of cuts increases, resulting in a significant disruption of the graph compared to traditional methods, most of which fail to outperform random choice until the fifth vertex removal. Note that the algebraic connectivity can actually increase after a vertex removal, as it does in Figure 3, whereas this would not happen after an edge removal. If an approach selects a vertex that is only loosely connected to the periphery of the graph, then removing that vertex results in the remainder of the graph becoming more tightly connected.

We next present results for medium-sized multi-agent systems, consisting of 50 simulated agents. Results for these are presented in Figure 4 for up to 25 vertex cuts. Again we see a similar performance pattern, with our approach in blue and able to attain superior performance. We again see the next best performance from the *Betweenness* approach, showing the value of this centrality measure in describing vertex connectivity. The *Fiedler Low* approach achieves worse performance than a random selection of vertices, as it also did for the small systems. The inspiration of this approach, used to divide graphs, means that the vertices selected by this approach are not tightly connected to any grouping of the graph, and thus not influential on the overall graph connectivity.

Finally, we present results for large-sized multi-agent systems in Figure 5, consisting of 100 agents each. We display these results in Figure 5(a), for up to 50 vertex cuts, with Figure 5(b) showing a detailed view of only up to 10 cuts. We again see a similar performance pattern, with our proposed approach outperforming any compared approaches. On these large graphs, the compared approaches struggle to separate from each other, with *Betweenness* performing the best after 10 cuts but being surpassed by *Fiedler High* in the full 50 cuts.

V. Discussion of Complexity

First, we propose that the formulation proposed by Eq. (9) can be solved by typical gradient descent. By solving for the derivative of Eq. (9), and given an initial guess of $\mathbf{x}$ (for example, $\mathbf{x}^0 = \mathbf{1}$), we can then iteratively solve for the optimal value $\mathbf{x}^*$ by:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k) \tag{14}$$

where α is a learning rate parameter and $\nabla f(\mathbf{x}^k)$ is the derivative of the formulation presented in Eq. (9). We can reconstruct Eq. (9) without mathematical difference into:

$$\min_{\mathbf{x}} \|\mathbf{F}^\top \mathbf{x}\|_2^2 + \beta(\mathbf{x}^\top \mathbf{1} - q)(\mathbf{1}^\top \mathbf{x} - q) \tag{15}$$

(as $\mathbf{x}^\top \mathbf{1} = \mathbf{1}^\top \mathbf{x}$), from which we can then compute the derivative with respect to $\mathbf{x}$ as

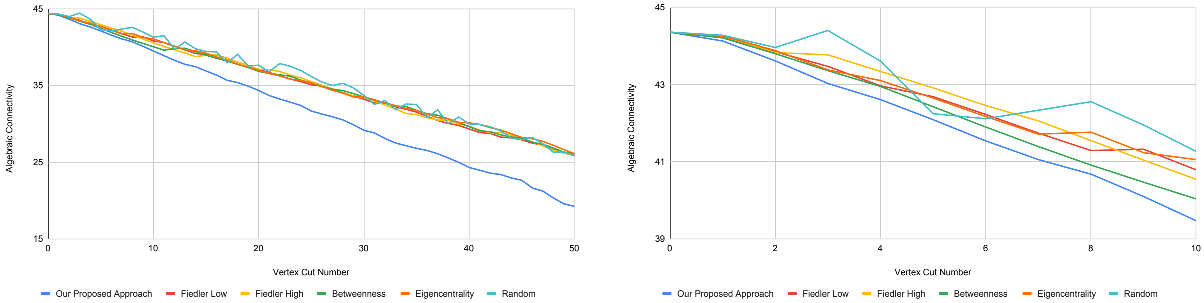
$$\nabla f(\mathbf{x}) = 2\mathbf{F}\mathbf{F}^\top \mathbf{x} + \beta(2\mathbf{1}\mathbf{1}^\top \mathbf{x} - 2q\mathbf{1}) \tag{16}$$

With this derivative, we can calculate Eq. (14). After each iteration, we adjust $\mathbf{x}$ such that

$$0 \leq \mathbf{x} \leq 1 \tag{17}$$

Each iteration of this approach is bounded by $n \times n$ complexity, and thus this overall optimization algorithm is upper bounded by $\mathcal{O}(kn^2)$, where k is the number of iterations. Therefore, the complexity of the minimization problem described by Eq. (9) is governed by either the number of iterations k or the number of agents n. If $n > k$, then this formulation is bounded by the cost to compute the *Fiedler cost* matrix, which requires an eigen-decomposition of the Laplacian matrix $\hat{\mathbf{L}}$. In this case, the computation of our proposed approach is bounded this type of computation, and thus by $\mathcal{O}(n^3)$, where n is the number of agents. If $k \geq n$, then the complexity of our approach is bounded by the number of iterations required to converge.

We then consider the complexity of the compared algorithms. The *Fiedler Low*, *Fiedler High*, and *Eigencentrality* approaches all depend on the calculation of the eigenvalues of the Laplacian, and thus similarly employ a computational complexity of $\mathcal{O}(n^3)$. Finally, the *Betweenness* approach is



(a) Full Results

(b) Detailed Results of First 10 Vertex Removals

Fig. 5. [Best Viewed in Color] Algebraic connectivity over multiple vertex cuts for large ($n = 100$) multi-agent systems. Our approach, shown in blue, outpaces compared approaches.

bounded by complexity $\mathcal{O}(nm + n^2 \log n)$ where m equals the number of edges in the system [23]. For sufficiently large and connected systems, where $m = \frac{n(n-1)}{2}$, this approach is closely upper-bounded by $\mathcal{O}(n^3)$.

Overall, our proposed approach achieves a similar complexity bound as the compared approaches. Considering the very similar complexity performance, our approach clearly achieves superior results.

VI. Conclusion

Multi-agent and multi-robot systems are pervasive in modern society. In order to operate effectively, these multi-agent systems must communicate and collaborate, even in adversarial environments. Hostile social networks and multi-drone systems need to be interrupted, with communication, connectivity, and cohesion disrupted. We present an optimization-based approach to identify optimal vertices to remove, based on formulating an approximate version of the problem of minimizing the Fiedler value of the multi-agent system. We show that our proposed approximation outperforms other methods, including both those focused on the Fiedler vector and those focused on the centrality of the graph, and is able to best reduce the algebraic connectivity of a graph, thus optimally disrupting it. Our approach also achieves these superior results with similar computational complexity to the compared approaches.

References

[1] B. Reily, C. Reardon, and H. Zhang, "Leading multi-agent teams to multiple goals while maintaining communication," in *Robotics: Science and Systems*, 2020.

[2] S. Zheng, Z. Li, Y. Yin, Y. Liu, H. Zhang, P. Zheng, and X. Zou, "Multi-robot relative positioning and orientation system based on uwb range and graph optimization," *Measurement*, p. 111068, 2022.

[3] U. Can and B. Alatas, "A new direction in social network analysis: Online social network analysis problems and applications," *Physica A: Statistical Mechanics and its Applications*, vol. 535, p. 122372, 2019.

[4] A. Dorri, S. S. Kanhere, and R. Jurdak, "Multi-agent systems: A survey," *IEEE Access*, vol. 6, pp. 28 573–28 593, 2018.

[5] F. Chen, W. Ren *et al.*, "On the control of multi-agent systems: A survey," *Foundations and Trends in Systems and Control*, vol. 6, no. 4, pp. 339–499, 2019.

[6] W. Luo and K. Sycara, "Minimum k-connectivity maintenance for robust multi-robot systems," in *International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 7370–7377.

[7] E. Stump, A. Jadbabaie, and V. Kumar, "Connectivity management in mobile robot teams," in *International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1525–1530.

[8] X. Chen, S. Zhang, L. Zhang, G. Yu, and J. Liu, "Determining redundant links of multiagent systems in keeping or improving consensus convergence rates," *IEEE Systems Journal*, 2021.

[9] S. Jafarizadeh, "Fastest mixing reversible markov chain: Clique lifted graphs and subgraphs," *Transactions on Signal and Information Processing over Networks*, vol. 6, pp. 88–104, 2020.

[10] P. Szczytowski, A. Khelil, and N. Suri, "Dkm: Distributed k-connectivity maintenance in wireless sensor networks," in *Conference on Wireless On-Demand Network Systems and Services (WONS)*, 2012, pp. 83–90.

[11] J. Chen and Q. Zhu, "Resilient and decentralized control of multi-level cooperative mobile networks to maintain connectivity under adversarial environment," in *Conference on Decision and Control (CDC)*, 2016, pp. 5183–5188.

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

[13] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, no. 2, pp. 390–396, 2010.

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

[15] L. Sabattini, N. Chopra, and C. Secchi, "On decentralized connectivity maintenance for mobile robotic systems," in *Conference on Decision and Control and European Control Conference*, 2011, pp. 988–993.

[16] C.-J. Zhang and A. Zeng, "Network skeleton for synchronization: Identifying redundant connections," *Physica A: Statistical Mechanics and its Applications*, vol. 402, pp. 180–185, 2014.

[17] Y. Liu, M. Tang, T. Zhou, and Y. Do, "Improving the accuracy of the k-shell method by removing redundant links: From a perspective of spreading dynamics," *Nature Scientific Reports*, vol. 5, no. 1, pp. 1–11, 2015.

[18] S. Jafarizadeh, "Fastest mixing reversible markov chain on friendship graph: Trade-off between transition probabilities among friends and convergence rate," *Systems & Control Letters*, vol. 130, pp. 13–22, 2019.

[19] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak mathematical journal*, vol. 23, no. 2, pp. 298–305, 1973.

[20] J. Zhu, "On consensus speed of multi-agent systems with double-integrator dynamics," *Linear Algebra and its Applications*, vol. 434, no. 1, pp. 294–306, 2011.

[21] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[22] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Conference on Decision and Control (CDC)*, 2006, pp. 6605–6611.

[23] S. Bhardwaj, R. Niyogi, and A. Milani, "Performance analysis of an algorithm for computation of betweenness centrality," in *International Conference on Computational Science and Its Applications*, 2011, pp. 537–546.