

Distributed Topology Manipulation to Control Epidemic Spreading over Networks

Dong Xue, *Member, IEEE*, and Sandra Hirche, *Senior Member, IEEE*

Abstract

This article considers the problem of network structure manipulation in the absence of a central coordinator which collects network information and makes decisions. The primary focus is on the spectral radius minimization problem by removing/rewiring links to control epidemic spreading over networks. The resulting optimizations are generally combinatorial and NP-hard. The lack of the central base prevents us from solving such problems in a centralized fashion, thus driving the need for distributed computation and collaborative decision-making to support effective topology modification. In this article, a distributed estimation scheme involving event-based communication and parallel algorithms is developed to enhance network capability against epidemics. Based on this low-complexity estimation algorithm, fully distributed strategies are proposed to enable individuals to sequentially discern the critical in-network contact, and to implement the desired link operation solely. We further extend the acquisitions to a more flexible framework based on a “coevolutionary networks” picture and discuss the optimality of solutions from an algebraic and topological perspective. Extensive simulation examples are presented to demonstrate the effectiveness of the proposed strategies.

Index Terms

epidemic spread, distributed computation, link operation, event-based control.

I. INTRODUCTION

Recent years have witnessed widespread interest in controlling the dissemination of undesired entities over networks in many areas such as human society, ecological systems, computer science. In its generality, the terminology “entity” includes contagious diseases in human communities [1], computer viruses on the Internet [2], instability in multi-agent systems [3], and cascading blackout in power grids [4].

Numerous studies [5]–[8] have highlighted the impact of the network structure on epidemic dissemination. These results have raised a growing concern about topology manipulation to curb the spreading of a disease in large-sized networks. In general, two aspects of topology operations help mitigate the effects of an epidemic. The first either inoculates [6] or quarantines [1] particular nodes, immunizing them against the disease and rendering them unable to propagate it. A problem intimately related to nodal operations is optimal protection resource allocation [9]. In realistic scenarios, for instance, individuals possessing more resources can afford more doctors or receive better medical treatment. Another method includes link removal [2], [5], [7], [10] and relocation [8], [11], [12]. The traditional approaches for shaping network structure are to execute independently of the states of dynamical processes occurring on networks. Nevertheless, it is already known that the evolution of many real-life network topology is inextricably coupled with the nodal dynamics and vice versa. Networks which display a feedback loop between system dynamics and network structures are called *adaptive* or *coevolutionary* networks [13]. See [14] for a comprehensive review of adaptive networks. The present article aims to proceed further in the link operation from the perspectives of adaptive networks and distributed optimization.

A. Prior and Related Work

The optimal topology-design problem to regulate epidemic spreading is generally combinatorial and NP-hard [2]. This kind of optimization problem is more likely to be solved by brute-force searching all possibility and selecting the best edge subset. However, computational complexity grows exponentially as the number of nodes increases and quickly becomes unfeasible even for moderate networks. Considerable research has addressed convex relaxation and heuristic algorithms to approximate the optimal solution. For example in [5], a semidefinite programming (SDP)-based relaxation is applied to tackle the combinatorics, whose computational complexity increases swiftly when applied to massive networks. Alternatively, a greedy algorithm, drawing upon perturbation theory, is employed in [15] to provide a local optimum in a fast, iterative manner. Many perturbation-based greedy heuristics that involve the computation of eigenvectors of network-induced matrices conduct in a centralized manner, relying on the knowledge of all the elements of the matrix in question. The unavailability of full network information has sparked an extensive interest in developing distributed methods to estimate the network-wide eigenvectors. Among others, significant attention has been dedicated to the power iteration (PI) method and especially its applicability to a distributed

This work was supported in part by the German Research Foundation within the Priority Program SPP 1914 “Cyber-Physical Networking”, and in part by the joint Sino-German Research Project “Control and optimization for event-triggered networked autonomous multi-agent systems (COVEMAS)”, which is funded through the German Research Foundation (DFG) and the National Science Foundation China (NSFC).

D. Xue and S. Hirche are with the Chair of Information-Oriented Control, Technische Universität München, D-80209 München, Germany; dongtonyxue@gmail.com

fashion [16]–[18]. In [17], a distributed method is proposed to find the Fiedler eigenvector of the Laplacian matrix based on power iteration and consensus averaging techniques. This algorithm requires nested loops for the intermediate normalization of the distributed vector, severely influencing its efficiency. To avoid many daunting lower-level iterations, a method based on the normalized least-mean-square algorithm is provided in [16] to circumvent the intermediate normalization. However, this method is indeed centralized because of the requirement for designating a global beacon node.

Research in distributed topology manipulation has mostly been focused on designing various algorithms to infer the network structure based on the locally available information. Considerably less attention has been dedicated to the subsequent implementation. Most of the existing results [9], [19] are to assume, implicitly or explicitly, the existence of a central entity that makes in-network decisions using local estimation. However, today's network systems are typically spatially scattered over a large district and may consist of hundreds of individuals in multi-agent systems or millions of individuals in social networks [20]. As such, they usually do not possess such a global center, thus necessitating the advancement of distributed strategy in response to this situation.

B. Motivation and Contributions

Though above-mentioned approaches, using centralized control, are superior in regulating epidemics for small-size networks, our novelty is in the development of a distributed strategy for the edge operations (removal and rewiring) while making the algorithm practical and scalable to real large-scale networks. Especially, the developed strategy is desired to distributively perform itself without complete information of network topology and a central decision-maker. Since the spectral radius of a graph is in intimate relation to the network capability against epidemic spreading, as such is terminus a quo for the optimization formulation of topology manipulation problems. In this article, we aim to harness the combinatorial and NP-hard nature of the optimization problems.

Our main contributions are multifold. First, eigenvalue sensitivity analysis is sought to approximate the spectral radius minimization problem to the one involving the eigenvector associated with the largest eigenvalue of the adjacency matrix. Since the approximated problem relies on an eigenvector, we develop a distributed estimation scheme in the advantage of in-network power iteration with intermittent mean correction. This obviates the requirement for nested loops, thus saving the utility of communication resources. In the meanwhile, a distributed identification process is implemented in parallel with the estimation mechanism and enables individual nodes to find self-perceptively the adequate links in the networks. The parallel design of the two distributed schemes reduces the computational complexity remarkably. This distributed strategy for link operation proceeds with locally retrievable information and without a centralized decision-maker. Second, we study the topology design problem in the context of coevolutionary networks. The paradigm shift from simultaneously manipulating the budget of links to sequentially operating the links allows the flexibility of the strategy in various applications. In particular, this analysis from an adaptive network perspective deepens our understanding of the efficiency of the distributed methods in terms of convergence speed and optimality quality. Subsequently, we examine the proposed distributed strategies from topological and algebraic points of view. As the number of links removed from the network increases, the link removal gradually eliminates the heterogeneity of the network such as degree difference and spectral gap. Besides, the spectral gap of the adjacency matrix has a strong impact on the effectiveness of the proposed algorithms. Finally, all results are illustrated and validated using numerical demonstrations and examples.

C. Paper Organization

The remainder of this article is organized as follows: Section II addresses basic notations from graph theory and the problem formulation. After providing an approximation algorithm to the original combinatorial problems in Section III, the detailed distributed strategies to manipulate the network structure are studied in Section V. Some further discussions provided in Section VI. Finally, the proposed approaches are evaluated via some numerical examples in Section VII.

Notation: Throughout the article, let \mathbb{R} be the set of real numbers and $\mathbb{N}_{>0}$ ($\mathbb{N}_{\geq 0}$) be the set of positive (non-negative) integers; $\mathbf{1}$ and $\mathbf{0}$ denote the column vector of all ones and all zeros with appropriate dimensions, respectively. An identity matrix (with dimensions inferred from context) is given by \mathbf{I} and \mathbf{e}_i is the i -th column vector of \mathbf{I} . For a given set \mathcal{S} , $|\mathcal{S}|$ denotes the cardinality of this set. For a symmetric matrix \mathbf{C} , $\lambda_i(\mathbf{C})$ denotes its i -th smallest eigenvalues sorted in the increasing order $\lambda_1(\mathbf{C}) \leq \dots \leq \lambda_n(\mathbf{C})$ and $\rho(\mathbf{C}) := \max_i |\lambda_i(\mathbf{C})|$ is defined as its spectral radius.

II. PROBLEM FORMULATION

In this section, we recall some necessary notions from graph theory and give the problem formulation.

A. Preliminaries

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an *undirected* graph with a set of vertices $\mathcal{V} = \{1, 2, \dots, n\}$ and a set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. In this article, we confine our attention to unweighted graph for the sake of simplicity and the outcomes can be easily extended to weighted cases. A graph is *connected* if there is a path between any pair of distinct nodes. If $(i; j) \in \mathcal{E}$, then the nodes i and j are

adjacent and the neighborhood set of node i is then denoted by $\mathcal{N}_i = \{j \in \mathcal{V} | (i; j) \in \mathcal{E}\}$. Furthermore, the complement $\bar{\mathcal{G}}$ of a graph \mathcal{G} is a graph with same vertex set \mathcal{V} as \mathcal{G} and its edge set $\bar{\mathcal{E}}$ has $(i; j) \in \bar{\mathcal{E}}$ if and only if $(i; j) \notin \mathcal{E}$.

A *adjacency* matrix of \mathcal{G} is given by $\mathbf{A}(\mathcal{G}) \in \mathbb{R}^{n \times n}$ and its elements are defined by $[\mathbf{A}(\mathcal{G})]_{ij} = 1$ if $(i; j) \in \mathcal{E}$; otherwise 0. For clarity of presentation, the eigenvector associated to the eigenvalue of adjacency matrix $\lambda_i(\mathbf{A})$ is symbolized with $\omega_i = [\omega_{i,1}, \dots, \omega_{i,n}]$. For a graph \mathcal{G} , a label $l_{ij}(\mathcal{G}) \in \{1, \dots, |\mathcal{E}|\}$ is assigned to edge $(i; j) \in \mathcal{E}$. One should note that relabeling the edges does not change the analysis, and hence the subscript in l_{ij} will be dropped if there is no risk of confusion. In addition, the *degree* matrix of a graph, $\mathbf{D}(\mathcal{G}) \in \mathbb{R}^{n \times n}$, is formed by $\mathbf{D}(\mathcal{G}) \triangleq \text{diag}(d_1, \dots, d_n)$ with $d_i = |\mathcal{N}_i|$. Consequently, the *Laplacian* matrix of a graph, $\mathbf{L}(\mathcal{G}) \in \mathbb{R}^{n \times n}$ is given by $\mathbf{L}(\mathcal{G}) \triangleq \mathbf{D}(\mathcal{G}) - \mathbf{A}(\mathcal{G})$. The second smallest eigenvalue $\lambda_2(\mathbf{L}(\mathcal{G})) \geq 0$, known as *algebraic connectivity* indicates the connectedness of the graph, i.e., $\lambda_2(\mathbf{L}) > 0$ if \mathcal{G} is connected, otherwise disconnected.

Furthermore, we say a $n \times n$ matrix $\mathbf{C} = [c_{ij}]$ is *compatible* with the graph \mathcal{G} if $c_{ij} = 0$ iff $(i; j) \notin \mathcal{E}$ and $j \neq i$. Clearly, adjacency matrix $\mathbf{A}(\mathcal{G})$ is compatible with graph \mathcal{G} . Matrix \mathbf{C} is nonnegative, i.e. $\mathbf{C} \geq 0$, if all its elements are nonnegative. A nonnegative symmetric matrix \mathbf{C} is irreducible if and only if it is compatible with a connected graph \mathcal{G} . A nonnegative matrix \mathbf{C} is said to be primitive if there exists a positive integer k such that $\mathbf{C}^k > 0$. In particular, a symmetric matrix \mathbf{C} is primitive if it is irreducible and has at least one positive diagonal element [21].

B. Research Motivation

Consider a network of n nodes represented by a connected undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with an adjacency matrix \mathbf{A} . Epidemic dynamics are commonly modeled by stochastic diffusion processes. Among others, a prominent model is the susceptible-infected-susceptible (SIS) model where each node in the network is either *infected* or *healthy*. Each infected individual recovers naturally with a curing rate γ and contaminates their healthy neighbors with an infection rate μ . The overarching point is that an epidemic dies out quickly if the effective spreading rate $\tau := \mu/\gamma$ is below a certain threshold that hinges on the model parameters; see [22] for a discussion on this threshold in real networks.

Lemma 1 ([23]). *Consider an infection process occurring on a graph \mathcal{G} with adjacency matrix \mathbf{A} . Initial infection will extinguish exponentially if the effective spreading rate satisfies $\tau \leq \frac{1}{\rho(\mathbf{A})}$, where $\rho(\mathbf{A})$ is the spectral radius of the graph \mathcal{G} .*

Central in the above lemma is the connection between the epidemic eradication and the spectral property of underlying graphs. More fundamentally, a smaller spectral radius leads to a higher probability of networks against the spread of viruses. Instead, the violation of the condition in Lemma 1 indicates that the disease may persist in the network for a fairly long time. Hence, a natural option to mitigate the spreading of a disease effectively is to make the spectral radius of the underlying graph as small as possible. In the following, we focus on studying a distributed strategy for link operations to reduce $\rho(\mathbf{A})$. Since the result of Lemma 1 does not depend on the modeling of epidemics, our strategy has the widest applicability to other epidemic models. For more details on modeling, readers are referred to [23].

C. Link-removing Strategy

Given a connected undirected graph $\mathcal{G}_0 = (\mathcal{V}, \mathcal{E}_0)$ with adjacency matrix \mathbf{A}_0 , the problem of link removal is defined as follows: for a fixed budget $|\Delta\mathcal{E}^-| = m_d$ ($m_d \in \mathbb{N}_{\geq 1}$), select a set of edges $\Delta\mathcal{E}^-$ from \mathcal{E}_0 such that the spectral radius of resulting graph $\mathcal{G}_{m_d}^- = \{\mathcal{V}, \mathcal{E}_0 \setminus \Delta\mathcal{E}^-\}$ is minimal. As claimed in Proposition 3.1.1. of [24], the adoption of Perron-Frobenius theory to adjacency matrix¹ provides $\rho(\mathbf{A}) = \lambda_n(\mathbf{A}) > 0$. With the notation $\mathbf{A}_{m_d}^-$ as the adjacency matrix of graph $\mathcal{G}_{m_d}^-$, the problem is cast as the following optimization

$$\begin{aligned} \min_{\Delta\mathcal{E}^- \subseteq \mathcal{E}_0} \quad & \lambda_n(\mathbf{A}_{m_d}^-) \\ \text{s.t.} \quad & |\Delta\mathcal{E}^-| = m_d. \end{aligned} \tag{P1a}$$

After labeling edges on graph \mathcal{G}_0 by $l_{ij} \sim (i; j) \in \mathcal{E}_0$, the optimization problem (P1a) can be rewritten as

$$\begin{aligned} \min_{\mathbf{y} \in \{0,1\}^{|\mathcal{E}_0|}} \quad & \lambda_n(\mathbf{A}_0 - \Delta\mathbf{A}_{m_d}^-) \\ \text{s.t.} \quad & \Delta\mathbf{A}_{m_d}^- = \sum_{l_{ij}=1}^{|\mathcal{E}_0|} y_{l_{ij}} (\mathbf{e}_i \mathbf{e}_j^\top + \mathbf{e}_j \mathbf{e}_i^\top), \\ & \mathbf{1}^\top \mathbf{y} = m_d \end{aligned} \tag{P1b}$$

where the vector $\mathbf{y} = [y_1, \dots, y_{|\mathcal{E}_0|}]^\top$ encodes the information in the sense that $y_{l_{ij}} = 1$ means the edge $(i; j) \in \mathcal{E}_0$ is removed from \mathcal{E}_0 .

¹For the sake of a concise presentation, we suppose that $\lambda_n(\mathbf{A}) > |\lambda_i(\mathbf{A})|$ for $i = 1, \dots, n-1$ throughout the article. Nevertheless, this pragmatic hypothesis is without loss of generality, as the developed distributed strategy in the sequel is applicable when this assumption is violated. This is because the proposed algorithms relies on power iterations of a primitive matrix derived from the adjacency, leading to a 1-dimensional dominant eigenspace.

In real-life networks, the link removal might physically be implemented, e.g., limiting traffic/travel between cities, or restricting interactions between individuals.

Remark 1. Note that the link-breaking optimization (P1a) can be reinterpreted equivalently to retard the information diffusion over networks. Since its convergence speed is a sub-modular set function of network topology, which means deleting an edge from the graph cannot accelerate the information diffusion [25]. Moreover, the distributed link-removing strategy can be extended to study transition behavior in large networks of coupled phase oscillators [26], as the largest eigenvalue of the adjacent matrix stands for a critical quantity.

D. Link-rewiring Strategy

Despite the instrumental effect on controlling the propagation of disease, the link-breaking scheme, in turn, degrades certain network performance dramatically, such as robustness and coherence. Instead, a moderate method, called link rewiring/rerouting/relocation, has also been employed to mitigate the effects of the disease [8], [11].

In specific, the nodes reconnect a portion of edges $1 \leq m_r \leq |\mathcal{E}_0|$ in graph \mathcal{G}_0 to minimize the spectral radius, which leads to a new network structure $\tilde{\mathcal{G}}_{m_r} = (\mathcal{V}, \tilde{\mathcal{E}})$, where $|\tilde{\mathcal{E}}| = |\mathcal{E}_0|$. This problem is formally stated as follows.

$$\begin{aligned} \min_{\tilde{\mathcal{E}}} \quad & \lambda_n(\tilde{\mathbf{A}}_{m_r}) \\ \text{s.t.} \quad & |\tilde{\mathcal{E}} \cap \mathcal{E}_0| = |\mathcal{E}_0| - m_r, \quad |\tilde{\mathcal{E}}| = |\mathcal{E}_0|, \end{aligned} \quad (\text{P2a})$$

where $\tilde{\mathbf{A}}_{m_r}$ is the adjacency matrix of the rewired graph $\tilde{\mathcal{G}}_{m_r}$. Likewise, the optimization problem (P2a) can be recast to

$$\begin{aligned} \min_{\bar{\mathbf{y}}, \mathbf{y}} \quad & \lambda_n(\mathbf{A}_0 - \Delta \mathbf{A}_{m_r}^- + \Delta \mathbf{A}_{m_r}^+) \\ \text{s.t.} \quad & \Delta \mathbf{A}_{m_r}^- = \sum_{l_{ij}=1}^{|\mathcal{E}_0|} y_{l_{ij}} (\mathbf{e}_i \mathbf{e}_j^\top + \mathbf{e}_j \mathbf{e}_i^\top), \quad \mathbf{1}^\top \mathbf{y} = m_r, \\ & \Delta \mathbf{A}_{m_r}^+ = \sum_{\bar{l}_{ij}=1}^{|\bar{\mathcal{E}}_{m_r}^-|} \bar{y}_{\bar{l}_{ij}} (\mathbf{e}_i \mathbf{e}_j^\top + \mathbf{e}_j \mathbf{e}_i^\top), \quad \mathbf{1}^\top \bar{\mathbf{y}} = m_r, \end{aligned} \quad (\text{P2b})$$

where $\bar{l}_{ij} \sim (i; j) \in \bar{\mathcal{E}}_{m_r}^-$ is the edge labeling on complement graph of $\mathcal{G}_{m_r}^-$, $l_{ij} \sim (i; j) \in \mathcal{E}_0$, $\mathbf{y} = [y_1, \dots, y_{|\mathcal{E}_0|}]^\top \in \{0, 1\}^{|\mathcal{E}_0|}$, $\bar{\mathbf{y}} = [\bar{y}_1, \dots, \bar{y}_{|\bar{\mathcal{E}}_{m_r}^-|}]^\top \in \{0, 1\}^{|\bar{\mathcal{E}}_{m_r}^-|}$.

The reformulation from problem (P2a) to (P2b) gives rise to disconnecting m_r links in the graph \mathcal{G}_0 first and then establishing m_r new connection in the graph $\mathcal{G}_{m_r}^-$. Note that the link-creating operation in link-rerouting strategy, adding m_r new edges into the graph $\mathcal{G}_{m_r}^-$ such that the increment of spectral radius is minimal, involves the link searching in the complement graph $\bar{\mathcal{G}}_{m_r}^- = (\mathcal{V}_0, \bar{\mathcal{E}}_{m_r}^-)$ of the graph $\mathcal{G}_{m_r}^-$. As shown in [27], reversing reciprocally the order of link deletion and addition in (P2b) results in the same minimum $\lambda_n^*(\mathbf{A}_{m_r})$.

E. Problem Statement

The optimization problems (P1b), (P2b) are of combinatorial nature and can be solved by a brute-force search. However, such an exhaustive search method to solve the problem (P1a) requires network-scale calculation of eigenvalues. It needs to examine the maximum eigenvalue of $\binom{|\mathcal{E}_0|}{m_d}$ different adjacency matrices and each eigenvalue calculation incurs $O(n^2)$ computational complexity. This complexity “explodes” while operating multiple edges collectively, where the solution space becomes exponential. Moreover, solving SDP for the convex relaxation by using, for example, the interior-point algorithm, costs approximately $(m_d^2 n^{2.5} + m_d n^{3.5})$ arithmetic operations [28]. This line of methodologies often needs a fusion center to gather the complete knowledge of network topology and to make collective decisions. In an attempt to eliminate this requirement of the central coordinator, we devote ourselves to developing new algorithms to solve (P1b), (P2b) independent of centralized management and with low computational complexity. More importantly, it is desirable that the developed strategy of link operation is itself performed by individual nodes based on the locally accessible information and cooperative decision making.

III. EIGENVALUE-SENSITIVITY-BASED APPROXIMATION

Eigenvalue sensitivity theory provides an insight into the behavior of the eigenvalues of a matrix while perturbing the matrix. The magnitude of the eigenvalue sensitivity informs about the range of the eigenvalue displacement in the complex plane. For more details on matrix perturbation theory, For proofs, readers are referred to [29] and the references therein.

Concerning the problem (P1b), the matrix $-\Delta \mathbf{A}_{m_d}^-$ can be regarded as a perturbation added to initial matrix \mathbf{A}_0 . As we shall see in the next lemma, the law of link removal is established by determining the links $l_{ij} \sim (i; j) \in \mathcal{E}_0$ with the largest product $\omega_{n,i} \omega_{n,j}$ of the components of the eigenvector ω_n corresponding to the largest adjacency eigenvalue $\lambda(\mathbf{A}_0)$.

Proposition 1 (Link removal). *Optimization problem (P1b) can be approximately solved by*

$$\begin{aligned} \max_{\mathbf{y} \in \{0,1\}^{|\mathcal{E}_0|}} \quad & \Delta \lambda_n(\mathbf{A}_0, \Delta \mathbf{A}_{m_d}^-) \\ \text{s.t.} \quad & \Delta \lambda_n(\mathbf{A}_0, \Delta \mathbf{A}_{m_d}^-) = \sum_{l_{ij}}^{|\mathcal{E}_0|} y_{l_{ij}} \omega_{n,i} \omega_{n,j}, \\ & \mathbf{1}^\top \mathbf{y} = m_d, \end{aligned} \quad (\text{P1c})$$

where $\omega_n(\mathbf{A}_0) = [\omega_{n,1}, \dots, \omega_{n,n}]^\top$ is the eigenvector associated to the eigenvalue $\lambda_n(\mathbf{A}_0)$ of the graph \mathcal{G}_0 .

Proof. Since the graph \mathcal{G}_0 is connected, Perron–Frobenius Theorem (see [30]) reads that the maximal eigenvalue of the adjacency matrix is unique and its corresponding eigenvector has all strictly positive entries. The application of matrix-perturbation to the objective function of optimization problem (P1b) yields the expansion (see e.g., [31, p.183])

$$\lambda_n(\mathbf{A}_0 - \Delta \mathbf{A}_{m_d}^-) = \lambda_n(\mathbf{A}_0) - \frac{\omega_n^\top(\mathbf{A}_0) \Delta \mathbf{A}_{m_d}^- \omega_n(\mathbf{A}_0)}{\omega_n^\top(\mathbf{A}_0) \omega_n(\mathbf{A}_0)} \quad (1a)$$

$$- \sum_{k=1}^{n-1} \frac{[\omega_k^\top(\mathbf{A}_0) \Delta \mathbf{A}_{m_d}^- \omega_n(\mathbf{A}_0)]^2}{\lambda_n(\mathbf{A}_0) - \lambda_k(\mathbf{A}_0)} + \mathbf{O}(\|\Delta \mathbf{A}_{m_d}^-\|^3). \quad (1b)$$

Obviously, the first-order perturbation expansion at the right-hand side of (1a) furnishes an upper bound for the largest eigenvalue of the perturbed adjacency. For graphs with a large spectral gap $\lambda_n(\mathbf{A}_0) - \lambda_{n-1}(\mathbf{A}_0)$ in which the high-order terms in (1b) can be neglected, this first-order approximation closes to the original objective function $\lambda_n(\mathbf{A}_{m_d}^-)$. The remaining proof can be derived from straightforwardly calculating. Instead of solving the NP-hard minimization (P1b), we embark on the quest of the maximization (P1c) and explore a suboptimal solution or an upper bound on the global optimizer. \square

Remark 2. *In the setting of optimization (P1c), deleting a link does not increase the spectral radius of graphs, but this intuition is essentially not correct for the primal problem (P1a) due to its non-convexity.*

As for the link-rewiring operation, there is also an approximation algorithm of the optimization problem (P2b) in a spirit similar to the link removal problem.

Proposition 2 (Link relocation). *The optimization (P2b) can be approximately solved by*

$$\begin{aligned} \max_{\mathbf{y}, \bar{\mathbf{y}}} \quad & \Delta \lambda_n(\mathbf{A}_0, \Delta \mathbf{A}_{m_r}^-) - \Delta \lambda_n(\mathbf{A}_{m_r}^-, \Delta \mathbf{A}_{m_r}^+) \\ \text{s.t.} \quad & \Delta \lambda_n(\mathbf{A}_0, \Delta \mathbf{A}_{m_r}^-) = \sum_{l_{ij}}^{|\mathcal{E}_0|} y_{l_{ij}} \omega_{n,i} \omega_{n,j}, \\ & \Delta \lambda_n(\mathbf{A}_{m_r}^-, \Delta \mathbf{A}_{m_r}^+) = \sum_{\bar{l}_{ij}=1}^{|\bar{\mathcal{E}}_{m_r}^-|} \bar{y}_{\bar{l}_{ij}} \nu_{n,i} \nu_{n,j}, \\ & \mathbf{1}^\top \bar{\mathbf{y}} = \mathbf{1}^\top \mathbf{y} = m_r, \end{aligned} \quad (\text{P2c})$$

where $l_{ij} \sim (i; j) \in \mathcal{E}_0$, $\mathbf{y} = [y_1, \dots, y_{|\mathcal{E}_0|}]^\top \in \{0,1\}^{|\mathcal{E}_0|}$, $\bar{l}_{ij} \sim (i; j) \in \bar{\mathcal{E}}_{m_r}^-$, and $\bar{\mathbf{y}} = [\bar{y}_1, \dots, \bar{y}_{|\bar{\mathcal{E}}_{m_r}^-|}]^\top \in \{0,1\}^{|\bar{\mathcal{E}}_{m_r}^-|}$. Moreover, $\omega_n = [\omega_{n,1}, \dots, \omega_{n,n}]^\top$ and $\nu_n = [\nu_{n,1}, \dots, \nu_{n,n}]^\top$ specify the eigenvector corresponding to the largest eigenvalue of adjacency matrix of graph \mathcal{G}_0 and graph $\mathcal{G}_{m_r}^-$, respectively.

Proof. The proof follows Proposition 1 and thus omitted. \square

In general, the solution to (P1c) and (P2c) is essentially an upper bound for the one to (P1b) and (P2b), respectively. So far, the original combinatorial problems (P1b) and (P2b) are heuristically approximated respectively by (P1c) and (P2c) which involve the eigenvectors of interests. Despite monitoring node-wise quantities only, yet, such approximations are indeed centralized owing to the need of complete topological information for computing the eigenvector of interest. Moreover, one often postulates the existence of a central coordinator determining the candidates based on the computed variables. Therefore, the sequel article is dedicated to relaxing those centralized requirements.

IV. DISTRIBUTED ESTIMATION OF EIGENVECTORS

Continued on the previous section, a method that allows nodes distributively to estimate eigenvectors of interests by only using local information is developed in this section.

Power iteration (PI) (see e.g., [29]) serves as an effective approach to estimate the dominant eigenvalue and its corresponding eigenvector of a symmetric matrix \mathbf{C} in the way:

$$\hat{\boldsymbol{\xi}}(t+1) = \frac{\mathbf{C}\hat{\boldsymbol{\xi}}(t)}{\|\mathbf{C}\hat{\boldsymbol{\xi}}(t)\|}, \quad t \in \mathbb{N}_{\geq 0} \quad (2)$$

where $\hat{\boldsymbol{\xi}}(t)$ is the vector estimate at t -th step, and the starting vector $\hat{\boldsymbol{\xi}}(0)$ has a nonzero component in the direction of the dominant eigenvalue. As time progresses, the sequence $\{\hat{\boldsymbol{\xi}}(t)\}_{t \in \mathbb{N}_{\geq 0}}$ exponentially converges to the dominant eigenvector of matrix \mathbf{C} at rate $|\lambda_{n-1}(\mathbf{C})/\lambda_n(\mathbf{C})|$.

In the application of PI method for sizable networks, an intermediate normalization is imperative at each iteration step. Even though intermediate operation helps to avoid the estimates of PI method converges to the zero vector or even diverges, it prevents the implementation of PI in a distributed fashion. In what follows, we develop a distributed PI method based on event-triggered normalization [32] and max-consensus protocol [33]. Because of event-based scheduling, mean correction dose not happen often, but occurs more than needed. We refer the interested reader to [34], [35] for recent advances in event-based control. Moreover, maximum-consensus is known to converge up to n steps with computational complexity $O(n)$ as opposed to average consensus that emerges a asymptotic convergence.

Let $\hat{\boldsymbol{\omega}}_n(t) = [\hat{\omega}_{n,1}(t), \dots, \hat{\omega}_{n,n}(t)]^\top$ be the estimate of the eigenvector $\boldsymbol{\omega}_n$ corresponding to $\lambda_n(\mathbf{A}(\mathcal{G}))$. Locally, each node $i \in \mathcal{V}$ maintains a component of the estimate $\hat{\omega}_{n,i}$ and exclusively accounts for computing the corresponding true value $\omega_{n,i}$. Since $\boldsymbol{\omega}_n$ has all strictly positive components, without loss of generality, the initialization is given randomly by a positive-value vector $\hat{\boldsymbol{\omega}}_n(0) > 0$.

Consider a modified distributed PI with $\mathbf{C} \triangleq \mathbf{I} + \mathbf{A}$ in (2)

$$\hat{\omega}_{n,i}(t+1) = r_i(t)[\hat{\omega}_{n,i}(t) + \sum_{j \in \mathcal{N}_i} [\mathbf{A}]_{ij} \hat{\omega}_{n,j}(t)] \quad (3)$$

where $r_i \in \mathbb{R}$ compensates the magnitude of node i and will be defined later. Obviously, the matrix \mathbf{C} is primitive and the leading eigenvalue of \mathbf{C} is simple. In particular, since $\rho(\mathbf{C}) = \lambda_n(\mathbf{C}) > 1$, $\hat{\boldsymbol{\omega}}_n$ may diverge infinitely as time evolves. It is, therefore, necessary to scale $\hat{\boldsymbol{\omega}}_n(t)$ felicitously.

First, each node is assigned a variable $h_i(t) \in \mathbb{R}_{>0}$ to locally estimate the growth rate $\|\mathbf{C}\hat{\boldsymbol{\omega}}_n(t)\|/\|\hat{\boldsymbol{\omega}}_n(t)\|$ in each iteration, which is updated according to the following rule

$$h_i(t+1) = h_i(t) + \frac{\hat{\omega}_{n,i}(t)}{\min(\sigma, \hat{\omega}_{n,i}^2(t))} \times [(1 - h_i(t))\hat{\omega}_{n,i}(t) + \sum_{j \in \mathcal{N}_i} [\mathbf{A}]_{ij} \hat{\omega}_{n,j}(t)] \quad (4)$$

where $\sigma > 0$ is a given threshold.

When $\hat{\omega}_{n,i}^2(t)$ exceeds the given upper threshold $\hat{\omega}_{n,i}^2(t) > \sigma$, node i triggers a timer to scale $\hat{\boldsymbol{\omega}}_n(t)$. Specifically, a flag signal of the event-trigger on node $i \in \mathcal{V}$, $\delta_i(t) \in \{0, 1\}$ indicates whether or not to shrink $\hat{\omega}_{n,i}(t)$ at time t , i.e.,

$$\delta_i(t) = \begin{cases} 1, & \text{if } \hat{\omega}_{n,i}^2(t) > \sigma \\ 0, & \text{otherwise} \end{cases}, \quad i \in \mathcal{V}.$$

To ensure all nodes reaching an agreement on r_i , nodes run a max-consensus algorithm for $h_i(t)$ across the network

$$p_i(t+s+1) = \max_{j \in \mathcal{N}_i \cup \{i\}} p_j(t+s), \quad p_i(t) = h_i(t), \quad (5)$$

where $s \in \mathbb{N}_{\geq 0}$. Algorithm (5) will end after a mixing time² T_d of a random walk on graph \mathcal{G}_0 and then, all nodes attain the maximum of the inputs $h_i(t)$ over the network

$$p_1(t+T_d) = \dots = p_n(t+T_d) = \max_{j \in \mathcal{V}} h_j(t).$$

As a consequence, the compensation variable $r_i(t)$ is given by

$$r_i(t) = \begin{cases} \frac{1}{\max_{j \in \mathcal{V}} h_j(t-T_d)} & \text{if } \delta_i(t-T_d) = 1 \\ 1 & \text{otherwise} \end{cases}. \quad (6)$$

Note that nodes keep on computing their local $h_j(t)$ ($j \in \mathcal{V}$) after $\delta_i(t) = 1$ since the max-consensus iteration (5) executes in parallel to the estimation algorithm (4) of the growth rate.

The following theorem summarizes that the output of the proposed method ultimately converges to $\boldsymbol{\omega}_n$.

²It can be chosen as the network diameter and could be a prior knowledge to every node since there are distributed algorithms to compute the diameter of a graph, e.g., [36], [37].

Theorem 1. Given a connected graph \mathcal{G} , the eigenvector ω_n corresponding to $\lambda_n(\mathbf{A}(\mathcal{G}))$ can be computed distributively and component-wisely by running (3)-(5).

Proof. As matrix \mathbf{C} is non-negative and primitive, the sequences $\{\max_{i \in \mathcal{V}} \hat{\omega}_{n,i}(t+1)/\hat{\omega}_{n,i}(t)\}$ converges asymptotically to $\rho(\mathbf{C})$, namely,

$$\rho(\mathbf{C}) \leq \dots \leq \max_{i \in \mathcal{V}} \frac{\hat{\omega}_{n,i}(2)}{\hat{\omega}_{n,i}(1)} \leq \max_{i \in \mathcal{V}} \frac{\hat{\omega}_{n,i}(1)}{\hat{\omega}_{n,i}(0)}; \quad (7)$$

see [38] and references therein. Note that between any consecutive event-triggered time instants, the update rule (4) exactly reduces to $h_i(t+1) = \hat{\omega}_{n,i}(t+1)/\hat{\omega}_{n,i}(t)$.

Since the magnitude scaling occurs only at designated event time by a common value, it does not change the direction of iteration in (3). In conjugation with max-consensus (5) and its convergence (7), it is not very difficult to deduce that

$$\lim_{t \rightarrow \infty} h_i(t) = \lim_{t \rightarrow \infty} \frac{\|\mathbf{C}\hat{\omega}_n(t)\|}{\|\hat{\omega}_n(t)\|} = 1 + \lambda_n(\mathbf{A}), \quad \forall i \in \mathcal{V}, \quad (8)$$

which indicates $\lim_{t \rightarrow \infty} r_i(t) = 1/(1 + \lambda_n(\mathbf{A}))$ for all $i \in \mathcal{V}$. Eventually, the distributed power-based iteration (3) in a compact form converges to the true eigenvector corresponding to the largest eigenvalue of adjacency matrix. That is,

$$\lim_{t \rightarrow \infty} \hat{\omega}_n(t+1) = \lim_{t \rightarrow \infty} \frac{(\mathbf{I} + \mathbf{A})\hat{\omega}_n(t)}{1 + \lambda_n(\mathbf{A})} = \omega_n.$$

The proof is completed. \square

In view of (8), the maximal eigenvalue of adjacency matrix is also available on each node when the algorithm establishes convergence, i.e., $\lambda_n(\mathbf{A}) \leftarrow h(t) - 1$ as $t \rightarrow \infty$. In network science [39], the principal adjacency eigenvalue and its associated eigenvector of a network suggest the topological structure of the network, as well as the behavior of dynamical systems on networks. Thus, our developed methodology has a wider applicability in the study of network systems.

In the special case when $\hat{\omega}_{n,i}^2(t) > \sigma$ holds for every $t = 1, 2, \dots$, the iteration (3) keeps on updating the norm compensation $r_i(t)$ over time after the first T_d steps. Thus, the proposed algorithm (3)-(5) in this scenario is identical to the conventional PI method (2) but with a distributed implementation of the intermediate normalization.

The remaining problem is how to choose the threshold σ . When an event is triggered at the time t due to $\hat{\omega}_{n,i}^2(t) = \sigma$, a plausible principle is that the estimates $\hat{\omega}_{n,i}(t+T_d)$ cannot grow infinitely or exceed a critical bound σ_{\max} , i.e., $\sigma \cdot n^{T_d} < \sigma_{\max}$, which is based on the fact $\lambda_n(\mathbf{A}(\mathcal{G})) \leq n - 1$. Furthermore, other bounds on spectral radius of graphs, e.g., in [24], can be introduced to explore the more precise value of σ . Evidently, a larger value σ_{\max} means a wider inter-event interval and thus less computation and fewer data transmissions across the network.

V. DISTRIBUTED STRATEGY FOR LINK MANIPULATION

This section is dedicated to present the overall distributed strategy for the edge removal/relocation within a formal computational context. At the heart of the strategy is that nodes are desired to solve the optimization (P1c) and (P2c) distributively and to identify the selected links collaboratively.

A. Iterative Implementation of Link Operations

The pioneering work of distributed topology manipulation with the main emphasis on distributed estimation of network topology often operates the whole budget of links simultaneously. By mimicking the gradient-based method, we tend to loosen this bundled implementation and sequentially remove/rewire one or multiple links instead of all at once.

The rationale behind the idea is elaborated on as below. With the link-removal case as an instance, the task of removing m_d links is unfolded along a sequence of issues $\mathcal{I} := \{1, \dots, K\}$ which generates a factorization of the edge-removing number m_d by $\{m_d^1, \dots, m_d^K\}$ such that $\sum_{k=1}^K m_d^k = m_d$. As such, nodes are required to delete m_d^k targeted links on each issue $k \in \mathcal{I}$ in a distributed manner. Obviously, the extreme case $m_d^1 = m_d$, as a special case, is reminiscent of the conventional approach widely adopted in the most related literature [5], [15]. It is, however, imperative to degrade the optimality performance in comparison to other link configuration. Recalling the perturbation expansion of the dominant eigenvalue in (1), we note that the number m_d^k can be treated as the step-size for the gradient method and thus a suitable selection of the step size is important. In a nutshell, a large step size m_d^k causes the problem of estimated value bouncing. Among others, the trivial case $m_d^k = 1$ for all $k \in \mathcal{I}$ seems to be the best configuration as it completes the next issue by estimating on the latest network topology, whereas it needs a larger consumption of computation and communication resources. Therefore, an appropriate optimal tradeoff between the degradation of network spectral-radius and the usage of network resources is to be determined in the practical applications.

The proposed methodology to sequentially solve the distributed edge manipulation problem falls into the research field of coevolutionary networks [14]. In particular, the successive link operation based on distributed computation is in a spirit similar to the coevolution phenomenon of opinion dynamics and influence structure in social systems [40]. As illustrated in Figure 1,

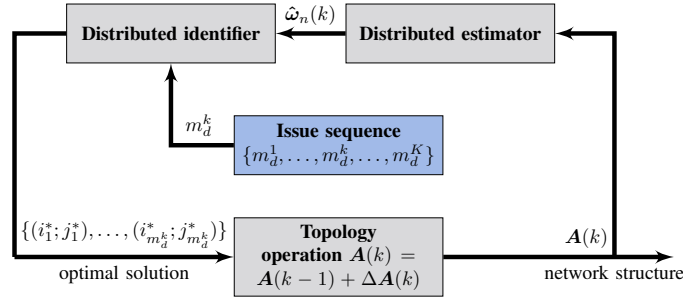


Fig. 1: The schematic diagram of the coevolution of distributed computation processes over networks and network topology.

the evolution of the topology is dependent on the states of network dynamics and vice versa. On the one hand, the topology of networks changes dynamically along the issue sequence $k \in \mathcal{I}$ according to the evolutionary rule $\mathbf{A}(k) = \mathbf{A}(k-1) + \Delta \mathbf{A}(k)$ where $\mathbf{A}(0) = \mathbf{A}_0$, and $\Delta \mathbf{A}(k) = -\Delta \mathbf{A}_{m_d^k}^-$ in the link-removing scenario. The variation of topology is determined by the outcomes of the distributed estimation dynamics. On the other hand, the distributed algorithm occurs on the renewed interconnection network and sequentially solves the optimization

$$\arg \max_{\{(i_1; j_1), \dots, (i_{m_d}; j_{m_d})\} \subset \mathcal{E}(k-1)} \sum_{l_{ij}(k-1)=1}^{|\mathcal{E}(k-1)|} y_{l_{ij}}(k) \psi(k-1) \quad (9)$$

where $\psi(k) = \omega_{n,i}(\mathbf{A}(k)) \omega_{n,j}(\mathbf{A}(k))$ and $\mathbf{1}^\top \mathbf{y}(k) = m_d^k$ on issue $k \in \mathcal{I}$. Here, $\mathcal{E}(k)$ is the edge set of graph $\mathcal{G}(\mathbf{A}(k))$ and $l_{ij}(k) \sim (i; j) \in \mathcal{E}(k)$ is the corresponding edge labeling. Therefore, the union of the optimal input arguments that sequentially solve the maximization (9) along the issue sequence provides a suboptimal solution to the link-removing optimization problem (P1b).

B. Link Removal

To unravel the design principle, we focus especially on “simple” situation where $m_d^k = 1$ for all $k \in \mathcal{I}$, because it can be very insightful and provides useful springboards for the study of more complicated scenarios. As discussed previously, this trivial configuration outperforms indeed in the optimality among others and exhibits the compelling distributed strategy for the link-removal problem.

The issue dependence of the variables will be omitted in general, but otherwise denoted for the sake of clarity. The meanings of some data transmitted among nodes are revisited:

- $\hat{\omega}_{n,i}(t)$ – estimate state of $\omega_{n,i}$ on node i at time t ;
- $r_i(t)$ – scale factor (6) for shrinking $\hat{\omega}_{n,i}(t)$ at time t ;
- b_i – timer with 0 means shrinking $\hat{\omega}_{n,i}(t)$ with $r_i(t) \neq 1$;
- $\delta_i(t)$ – flag bit for event trigger;
- $p_i(t)$ – variable of max-consensus algorithm.

Along the issue sequence, each node sequentially invokes a distributed algorithm to perform link removal and the detailed pseudo-code is exhibited in Algorithm 1.

In Algorithm 1, each node possesses a binary flag f_i that indicates whether node i is an endpoint of the selected optimal edge $l_{(i^*; j^*)}$ ($f_i = 1$) or not ($f_i = 0$) at the end of the algorithm. Lines 4–15 are the pseudo-codes of the compensation mechanism to avoid the infinity of estimation vector, while line 16 is the parallel distributed power iteration. The process encoded by lines 17–26 distributively identifies the candidate edge whose breaking suspends extremely the evolution of epidemic. In particular, the identification process is locally implemented in parallel to the distributed PI algorithm. That is to say, the identification of targeted links does not need to wait for the distributed computation to finish, and hence it is appealing to large-size networks. After the algorithm converges, the pair of nodes with $f_i = 1$ will self-consciously break the link indicated by the label $l_{(i^*; j^*)}$. Evidently, the proposed algorithm behaves indeed in a purely distributed fashion since it needs not a fusion center to gather information and to make a decision throughout the entire process.

To this end, the entire profile of the distributed strategy in a generic setting is concluded in the following.

Strategy 1 [Distributed Link Removal Strategy]: To remove m_d links from the network, nodes successively implement Algorithm 1 with proper parameter setup for K rounds and m_d^k edges are removed simultaneously at the end of each round.

Remark 3. *The development of per-node computation and data transmission endows the scalability of the network size to the distributed strategy. For each issue, Algorithm 1 runs distributed eigenvector estimation and decentralized edge selection parallelly and thus the worst computational complexity is equal to $O(n^2 + n)$. The overall complexity of the distributed strategy is given by $O(|\mathcal{I}|n^2)$. In contrast to centralized methods, the proposed algorithm entails compelling vantage points. For example,*

Algorithm 1: Distributed link-wise removal strategy

Input : \mathcal{G} is connected, $\sigma \cdot n^{T_d} < \sigma_{\max}$
Output: $f_i, l_{(i;j^*)}$

- 1 Initialization: set $t \leftarrow 0$, $r_i(0) \leftarrow 1$, $\delta_i \leftarrow 0$, $b_i \leftarrow -1$, $\beta_i \leftarrow 1$, $w_i \leftarrow 0$, $h_i(0) \leftarrow 1 + \max_{i,j \in \mathcal{V}} \sqrt{|\mathcal{N}_i||\mathcal{N}_j|}/2$, $\forall i \in \mathcal{V}$, and choose $\hat{\omega}_n(0) \in \mathbb{R}_{>0}^n$ randomly;
- 2 **while** 1 **do**
- 3 $t \leftarrow t + 1$;
- 4 **if** $\delta_i = 1$ **then**
- 5 $b_i \leftarrow b_i - 1$;
- 6 **if** $b_i = 0$ **then**
- 7 $r_i(t) \leftarrow p_i(t)$, $\delta_i \leftarrow 0$;
- 8 **else if** $|\hat{\omega}_{n,i}(t)| > \sigma$ **then**
- 9 $\delta_i \leftarrow 1$, $b_i \leftarrow T_d$, $p_i(t) = h_i(t)$;
- 10 **end**
- 11 **end**
- 12 **if** $\exists \delta_j = 1$ $j \in \mathcal{N}_i$ **then**
- 13 $k \leftarrow \arg \min_{j \in \mathcal{N}_i \cup \{i\}} \{\delta_j\}$,
- 14 $\delta_i \leftarrow \delta_k$, $p_i(t) \leftarrow p_k(t)$, $\delta_i = 1$;
- 15 **end**
- 16 each node transmits its data to neighbors and updates $\hat{\omega}_{n,i}$ and h_i according to (3) and (4), respectively;
- 17 **if** $t \bmod T_d = 0$ **then**
- 18 $f_i \leftarrow \beta_i$;
- 19 **else if** $t \bmod T_d = 1$ **then**
- 20 $w_i \leftarrow \max_{j \in \mathcal{N}_i} \hat{\omega}_{n,i} \hat{\omega}_{n,j}$;
- 21 $l_{(i;j^*)} \leftarrow l_{(i;\arg \max_{j \in \mathcal{N}_i} \hat{\omega}_{n,i} \hat{\omega}_{n,j})}$;
- 22 **end**
- 23 each node transmits w_i and $l_{(i;j^*)}$ to neighbors;
- 24 **if** $\max_{j \in \mathcal{N}_i} \hat{\omega}_{n,i} \hat{\omega}_{n,j} < \max_{j \in \mathcal{N}_i \cup \{i\}} w_j$ **then**
- 25 $\beta_i \leftarrow 0$, $w_i \leftarrow \max_{j \in \mathcal{N}_i \cup \{i\}} w_j$, $l_{(i;j^*)} \leftarrow l_{(\arg \max_{j \in \mathcal{N}_i \cup \{i\}} w_j; j^*)}$;
- 26 **end**
- 27 **end**

the greedy algorithm developed in [15] has the computational complexity $O(n^4)$, wherein the centralized computation of the dominant eigenvector leads to the complexity $O(n^2)$ and searching for the set of candidate edges incurs $O(n^2)$ computational complexity. Other approaches based on the distributed power-iteration but with the centralized edge selection, e.g., in [17], [19], approximately have the computational complexity $O(n^3)$ for a single-link operation.

C. Link Relocation

Next, we study how the network resilience evolves when a portion of existing links are sequentially rerouted in a distributed and iterative manner. In particular, individuals first cut the connection by using the link-breaking Algorithm 1 and a new connection is created simultaneously.

Notably, the requirement of the complete topological information is the main challenge arisen in the link-creating process while solving problem (P2c). In the link-removing case, the product $\omega_{n,i} \omega_{n,j}$ of existed links is accessible locally. However, to pursue this product of the nonexistent links in the present network amounts to a formidable computational task, especially, in a distributed fashion. In other words, the topological information of the complement graph of current interconnection becomes indispensable. To deal with this problem, a link addition algorithm with the weakest growth in spectral radius is introduced in lines 5-23 of Algorithm 2. Compared to link-breaking process, an extra period T_d corresponding to lines 7-14 is demanded to compute a local minimum $\min_{j \notin \mathcal{N}_i} \hat{\omega}_{n,j}$ for each node i . Lines 15-23 are intended to explore distributively the global minimum of the value $\hat{\omega}_{n,i} \min_{j \notin \mathcal{N}_i} \hat{\omega}_{n,j}$ wherein $(i; j) \notin \mathcal{E}_0$. At the end of Algorithm 2, the pair of nodes with $f_i = 1$ breaks up the link connecting them automatically. Meanwhile, the pair of nodes with $\tilde{f}_i = 1$ establishes their connection locally.

In summary, we show the whole picture of the distributed link-rewiring strategy treated in this article.

Strategy 2 [Distributed Link Rewiring Strategy]: To reroute $1 \leq m_r \leq |\mathcal{E}_0|$ links of the network, Algorithm 2 is implemented for m_r rounds and in each round, an existing edge is relocated according to the result of Algorithm 2.

Algorithm 2: Distributed link-wisely rewiring strategy

Input : \mathcal{G} is connected, $\sigma \cdot n^{T_d} < \sigma_{\max}$
Output: $f_i, l_{(i;j^*)}, \tilde{f}_i, \bar{l}_{(i;j^*)}$

- 1 Initialization: set $t \leftarrow 0$, $r_i(0) \leftarrow 1$, $\delta_i \leftarrow 0$, $b_i \leftarrow -1$, $\beta_i \leftarrow 1$, $w_i \leftarrow 0$, $h_i(0) \leftarrow 1 + \max_{i,j \in \mathcal{V}} \sqrt{|\mathcal{N}_i| |\mathcal{N}_j|} / 2$, $\tilde{\beta}_i \leftarrow 1$, $\tilde{w}_i \leftarrow 0$, $\pi_i \leftarrow i$, $\tilde{q}_i \leftarrow 0$, $\forall i \in \mathcal{V}$, and choose $\hat{\omega}_n(0) \in \mathbb{R}_{>0}^n$ randomly;
- 2 **while** 1 **do**
- 3 $t \leftarrow t + 1$;
- 4 execute line 4 – 26 in Algorithm 1;
- 5 **if** $t \bmod 2T_d = 0$ **then**
- 6 $\tilde{f}_i \leftarrow \tilde{\beta}_i$;
- 7 **else if** $t \bmod 2T_d \leq T_d$ **then**
- 8 **if** $t \bmod 2T_d = 1$ **then**
- 9 $\mathcal{W}_i \leftarrow \cup_{j \in \mathcal{N}_i \cup \{i\}} \{(\hat{\omega}_{n,j}, j)\}$;
- 10 **end**
- 11 each node transmits \mathcal{W}_i to neighbors and computes $\mathcal{W}_i \leftarrow \cup_{j \in \mathcal{N}_i \cup \{i\}} \mathcal{W}_j$;
- 12 search for $(\hat{\omega}_{n,j}, j) \in \mathcal{W}_i$ and compute
- 13 $\alpha_i \leftarrow \min_{j \notin \mathcal{N}_i} \hat{\omega}_{n,j}$, $\hat{\alpha}_i \leftarrow \arg \min_{j \notin \mathcal{N}_i} \hat{\omega}_{n,j}$;
- 14 **else**
- 15 **if** $t \bmod T_d = 1$ **then**
- 16 $w_i \leftarrow \hat{\omega}_{n,i} \alpha_i$, $\bar{l}_{(i;j^*)} \leftarrow \bar{l}_{(i;\hat{\alpha}_i)}$;
- 17 **end**
- 18 each node transmits w_i and $\bar{l}_{(i;\hat{\alpha}_i)}$ to neighbors and computes
- 19 $w_i \leftarrow \min_{j \in \mathcal{N}_i \cup \{i\}} w_j \neq 0$, $\pi_i \leftarrow \arg \min_{j \in \mathcal{N}_i \cup \{i\}} w_j \neq 0$;
- 20 **if** $\hat{\omega}_{n,i} \alpha_i > w_i$ **then**
- 21 $\tilde{\beta}_i \leftarrow 0$, $\bar{l}_{(i;j^*)} \leftarrow \bar{l}_{(\pi_i; \hat{\alpha}_{\pi_i})}$;
- 22 **end**
- 23 **end**
- 24 **end**

VI. FURTHER DISCUSSIONS

A. Spectral and Topological Analysis of Optimality

In this subsection, we focus on studying the optimality gap which amounts to the absolute error between the global minimum $\lambda_n^*(\mathbf{A}_{m_d}^-)$ for the optimization (P1b) and the local minimum with respect to the suboptimal solution derived from the proposed distributed strategy. Even though it is troublesome to provide a quantitative analysis of this optimality gap, the formula (1) provides prescient insights into the optimality performance. In addition to the step-size m_d^k , the approximation quality also depends jointly on the computation accuracy of the eigenvector estimation and the influence of the higher-order terms. Apparently, a large step size elevates the impact of higher-order terms and leads to a considerable deviation by using the first-order approximation (P1c), as opposed to the trivial case $m_d^k = 1$ that dampens this approximated error. By observing the higher-order terms in (1), the optimality performance is related to the spectral gap of the current adjacency matrix. When the spectral gap is large enough, the eigenvalues and eigenvectors other than the leading counterparts make little contribution to the approximation. Besides, the quality-of-service is primarily determined by the efficiency of the distributed estimation of the dominant eigenvector. As mentioned, the convergence rate of the distributed PI algorithm also relies a great deal on the spectral gap. More specifically, the convergence speed of distributed estimation accelerates as the spectral gap grows. In summary, we can argue that the developed distributed scheme of link operation offers a desired level of performance, so long as the networks are of a large spectral gap.

Furthermore, the leading eigenvector of the adjacency matrix, also known as eigenvector centrality, is often used to characterize the ability of nodes to spread epidemics over networks [1]: the larger value is, the higher power node has. Consequently, the product of each pair of eigenvector elements specifies the crucial role of the corresponding links in the propagation of disease. As such, the development in this article is intriguingly consistent with the intuitive instruction that to delete the edges connecting nodes with higher centrality increases the probability of rehabilitating a network from diseases. Therefore, untying the strong links is preferable than cutting weak links in controlling the epidemic spread. Empirical findings also show the epidemic scale is still dramatically wide by removing links connecting nodes with low centrality [2]. Nevertheless, it is necessary to remark that the efficiency of the proposed distributed strategy reduces gradually as the issue step progresses. Following the undock of edges among nodes with high centrality $\omega_{n,i}$ from the network, the discriminating power of the principal eigenvector weakens continuously. In connection with graph irregularity [41], the developed distributed strategy for

link operation tends to regularize the resulting network. A similar claim by empirical validations also appears in [2]. Good news is that networks extracting from real-world applications are far from being highly regular, so our approach is applicable to a significantly broader range.

B. Connectedness-preserving Distributed Link Removal

It is worth noting that when the number of deleted links m_d is closed to *edge connectivity*³, the resulting network after link removal may be disconnected. From the perspective of post-disaster reconstruction, networks tend to retain as a whole rather than splitting into isolated fractions, although the second scenario usually appears with a lower epidemic size. Therefore, a growing concern about preserving the connectedness of networks emerges in the course of topology operation [42].

Therefore, the optimization problem (P1c) has an additional constraint that $\mathcal{G}_{m_d}^-$ is connected. To solve this reformulated problem of (P1c), the main challenge is how to anticipate the connectedness of the resulting network in the event of link deletion, especially, in a distributed manner. Take removing a single edge from graph \mathcal{G}_0 as an example. Without loss of generality, assume that the edge to be deleted is $(i^*; j^*)$ and assign node i^* , j^* and others of $\mathcal{V} \setminus \{i^*, j^*\}$ a test signal 0, 1 and -1 , respectively. Then, the network runs simultaneously max-consensus and min-consensus protocol for the assigned signal variables over the graph $(\mathcal{V}, \mathcal{E}_0 \setminus \{(i^*; j^*)\})$. In the end, each node adds up the eventual outcomes of max- and min-consensus algorithm, respectively. If all the summations on nodes amount to zero, then the graph after deleting edge $(i^*; j^*)$ keeps connected; otherwise disconnected. See Algorithm 3 for the detailed pseudo-codes.

Algorithm 3: Distributed graph-connectivity examination

Input : \mathcal{G}_0 is connected, objective edge $(i^*; j^*) \in \mathcal{E}_0$

Output: connectedness of $(\mathcal{V}, \mathcal{E}_0 \setminus \{(i^*; j^*)\})$

- 1 Initialization: set $z_{i^*} \leftarrow 1$, $z_{j^*} \leftarrow -1$ and $z_j \leftarrow 0$ for all $j \in \mathcal{V} \setminus \{i^*, j^*\}$;
 - 2 execute max- and min-consensus algorithm for z_i ($i \in \mathcal{V}$) over $(\mathcal{V}, \mathcal{E}_0 \setminus \{(i^*; j^*)\})$;
 - 3 set $z_i \leftarrow \text{out}_i^{\max} + \text{out}_i^{\min}$ for all $i \in \mathcal{V}$;
 - 4 run max-consensus algorithm for z_i over \mathcal{G}_0 ;
 - 5 **if** output of max-consensus algorithm **then**
 - 6 | $(\mathcal{V}, \mathcal{E}_0 \setminus \{(i^*; j^*)\})$ is disconnected;
 - 7 **else**
 - 8 | $(\mathcal{V}, \mathcal{E}_0 \setminus \{(i^*; j^*)\})$ is connected;
 - 9 **end**
-

Algorithm 3 facilitates the extension of Strategy 1 for connectedness preservation during link removing. The detailed pseudo-codes are omitted due to space limitations.

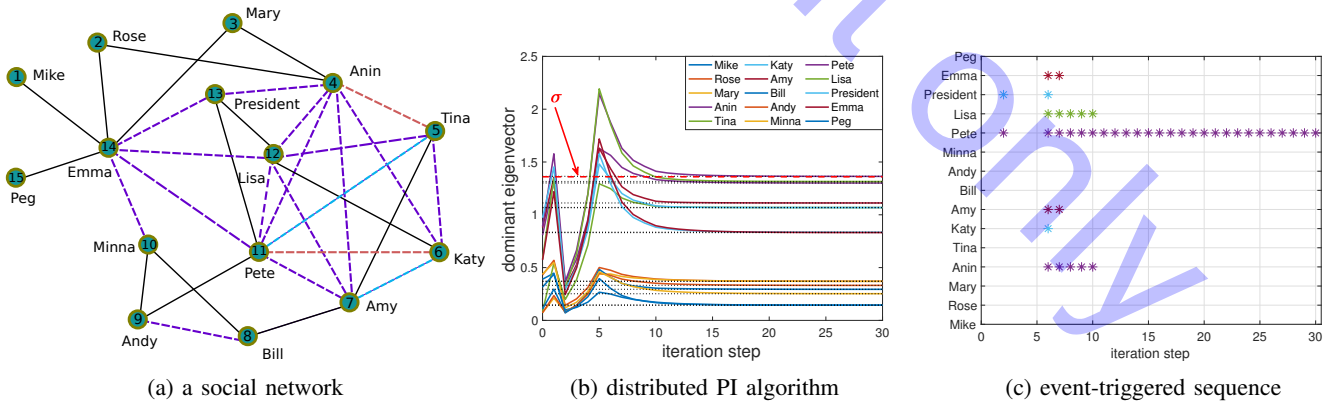


Fig. 2: Simulation on a small-size network: (a) Diagrammatic illustration of a social network [43] consisting of 15 members. The dashed lines in purple and red color are the global solution, while those in purple and blue color are the sub-optimal solution of Strategy 1. (b) The distributed PI estimation for the principal eigenvector $\omega_n(\mathbf{A}_0)$ provides a fast convergence to the real value (black dotted lines). (c) A diagram illustration of the event-triggered time instants.

VII. SIMULATIONS

In this section, we report simulation results for the proposed strategies of link operation over several empirical and real networks with different scales and topologies.

³Edge connectivity is denoted as the minimum number of edges whose deletion from a graph disconnects the graph.

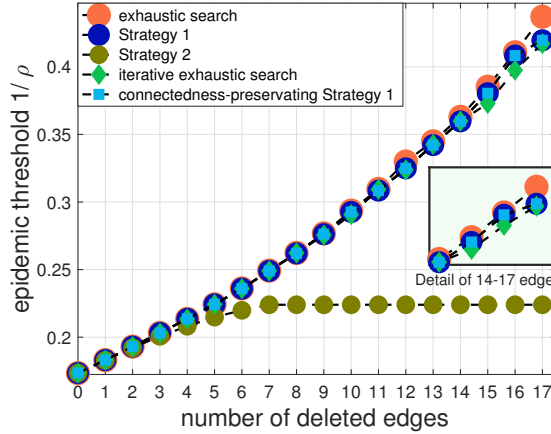


Fig. 3: Comparison of different strategies for link operation to control epidemic spreading, including exhaustive search, Strategy 1, Strategy 2, connectedness-preserving Strategy 1, and iterative exhaustive searching along the issue sequence.

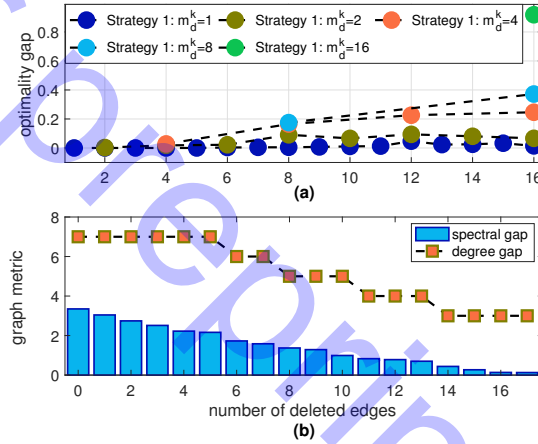


Fig. 4: Performance illustration: (a) The optimality gap between brute-force searching and distributed link-removal strategy 1 with different step size m_d^k . (b) The variation of network metrics in the link-removing process on small-size networks.

A. Small-size Networks

First, we elucidate the development of distributed protocols and compare them with the centralized method on the condition that the information of global network topology is accessible. The case studied here is that malicious rumors spread through a small social network consisting of 15 members and its interaction structure is shown in Fig. 2a. In order to control efficiently the dissemination of fake messages, we remove (rewire) $m_d = 17$ ($m_r = 17$) links from the network.

Central in the distributed strategy is a distributed PI algorithm which affords credible estimates of the centrality eigenvector. Therefore, we first examine the effectiveness of the event-triggered distributed PI algorithm over this small social network. The trajectories of the estimation variables $\hat{\omega}_{n,i}$ are depicted by colored solid lines in Fig. 2b, whereby the corresponding components of the true principal eigenvector are described by black dotted lines. It can be observed that the estimate states converge to their corresponding true values in a fast manner. In this process, the compensation for magnitudes in (3) does not occur continuously but acts only at some necessary moments. In this test, the event-trigger threshold is set to $\sigma = 1.36$ and the sequence of event triggered instants is plotted in Fig. 2c. In comparison to constant transmissions, event-based design indeed offers some overt advantages in computation and communication. Even so, we need to point out that the event switches of Pete continue to be on after convergence. This is because his estimate at equilibrium point exceed the threshold. An ad-hoc remedy is to endow a tailored threshold value σ_i for each node $i \in \mathcal{V}$, which, however, requisites for some prior knowledge of graph spectrum.

Under the same topological network, Fig. 3 compares the performance of the proposed distributed strategy and other approaches for link operation to control virus spreading. For link-deleting scheme, we focus on the proposed Strategy 1 with step size $m_d^k = 1$ for all $k \in \mathcal{I}$, as well as its connectedness-preserving version. At each round $k = \{1, \dots, m_d\}$, a single link is removed according to the output of Algorithm 1. The simulation results exhibit that Strategy 1 provides a near-optimal

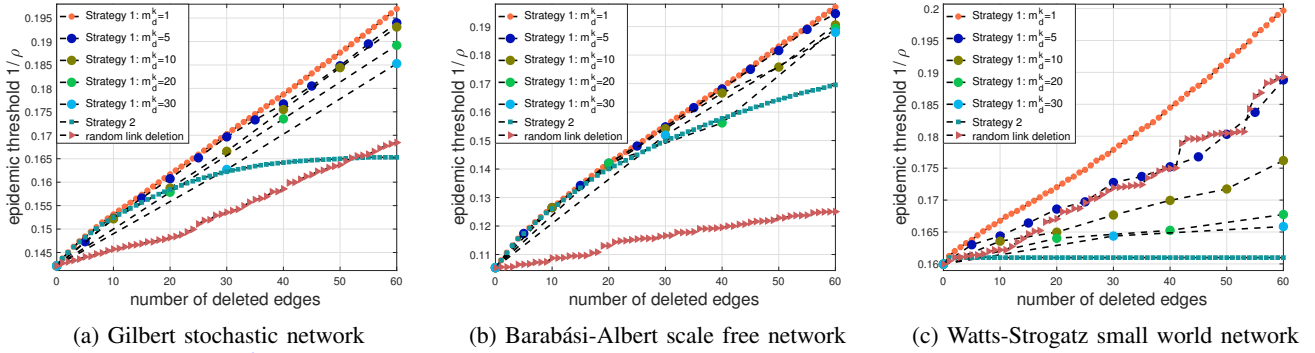


Fig. 5: Comparison of different strategies for link-removing scheme in three random networks.

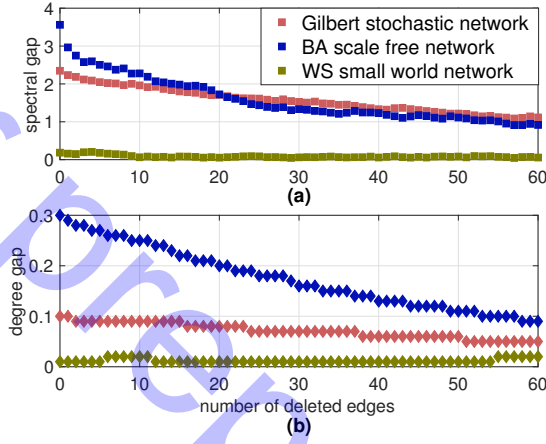


Fig. 6: The variation of spectral gap and degree gap in the link-removing process for random networks.

solution to the optimization problem (P1a), although the quality-of-service reduces slightly as the number m_d increases. The diagrammatic sketch of the resulting network after link removal in Fig. 2a corroborates also the efficiency of Strategy 1, whereby there is a mere two links difference between the results of Strategy 1 and brute-force search. In the meanwhile, we discuss the gap between the global solution to (P1b) and the solution obtained from Strategy 1 under different step-size configurations. From the upper plot of Fig. 4, a finer step size m_d^k results in a better optimality quality albeit more steps, agreeing with the previous discussion.

The lower plot of Fig. 4 shows the impact of link removal from the network on certain graph indices including spectral gap $\lambda_n(\mathbf{A}) - \lambda_{n-1}(\mathbf{A})$ and degree gap $\max_{i \in \mathcal{V}} d_i - \min_{j \in \mathcal{V}} d_j$. Both indices take on a descending tendency as the amount of the deleted links augments. This phenomenon implies, in some sense, that the link removal operation relieves irregularity of graphs. As discussed in Section VI-A, the performance loss of Strategy 1 is attributed to reducing the spectral gap.

B. Large-size Synthetic Networks

In this subsection, we report simulation results for the distributed strategies over three synthetic networks: 1). *Gilbert* stochastic model (Gi) ($|\mathcal{V}| = 100$, $|\mathcal{E}| = 268$), 2). *Barabási-Albert* Scale Free model (BA) ($|\mathcal{V}| = 100$, $|\mathcal{E}| = 281$), and 3). *Watts-Strogatz* model (WS) ($|\mathcal{V}| = 100$, $|\mathcal{E}| = 308$).

In this simulation study, we compare the proposed strategies with random link removal as the globally optimal configuration is inaccessible in this case. From the exposition of Fig. 5, the developed Strategy 1 with the trivial step-size $m_d^k = 1$ significantly outperforms the random link-removal strategy for all three random networks. Rewiring the existed links using Strategy 2 provides a poor performance. The worst occurs in WS network, as reported in Fig. 6, that has a minor discrepancy in nodal degrees.

Moreover, the results with other different step size are also depicted in Fig. 5. Despite saving computational efforts, the growth of the quantity m_d^k at each round deteriorates the effect of the link-removing operation for all random networks. In particular, it can be observed from Fig. 5c, somehow surprisingly, that the performance of Strategy 1 is worse than the random link removal for the WS model with the exception $m_d^k = 1$. As discussed before, a small spectral gap amplifies the inaccuracy of the distributed estimation algorithm. Moreover, Fig. 6 reveals that link-removing scheme makes the resulting random networks increasingly regular.

TABLE I: Normalized elapsed time

	$m_d^k = 1$	$m_d^k = 5$	$m_d^k = 10$	$m_d^k = 20$	$m_d^k = 30$
Gi network	0.7385	0.1457	0.0691	0.0278	0.0188
BA network	0.7364	0.1682	0.0664	0.0182	0.0108
WS network	0.7216	0.1536	0.0804	0.0276	0.0169

Finally, we test the elapsed time of the distributed link-removal Strategy 1 over random networks under different settings of the step size m_d^k . To address this issue concisely, Table I displays the normalized values of the elapsed time for each network. Naturally, the case of small step size comes at the cost of high iteration times for all three networks despite the evident benefits.

VIII. CONCLUSIONS

This article studies the distributed link operating problem of controlling epidemic spreading over networks in the absence of global network structure knowledge. Two key ingredients -distributed learning and collaborative decision making- are taken into account for the distributed topology operation. To bypass NP-hard barrier, we first approximate the original problem to the one involving the eigenvectors associated to the adjacency eigenvalues. With the design of event-triggered communication, a distributed PI estimation algorithm endows eventually a global knowledge of the eigenvectors in questions to local individuals. As such, the developed strategy enables us to acquire a suboptimal solution in a distributed fashion. Moreover, the proposed distributed computation algorithms has directed applications in network-induced problems including community detection [25], identification of critical nodes [18] and resource allocation [9].

REFERENCES

- [1] G. Moores, P. Shakarian, B. Macdonald, and N. Howard, "Finding near-optimal groups of epidemic spreaders in a complex network," *Plos One*, vol. 9, no. 4, p. e90303, 2014.
- [2] P. van Mieghem, D. Stevanović, F. Kuipers, C. Li, R. van de Bovenkamp, D. Liu, and H. Wang, "Decreasing the spectral radius of a graph by link removals," *Physical Review E*, vol. 84, no. 016101, 2011.
- [3] A. Koh and G. Vinnicombe, "On the propagation of instability in interconnected network," in *Proceeding of the 51st IEEE Conference on Decision and Control*, Maui, Hawaii, USA, 2012, pp. 3898–3903.
- [4] S. Pahwa, C. Scoglio, and A. Scala, "Abruptness of cascading failures in power grids," *Scientific Reports*, vol. 4, pp. 1–9, 2014.
- [5] A. Bishop and I. Shames, "Link operations for slowing the spread of disease in complex networks," *Europhysics Letters*, vol. 95, no. 18005, 2011.
- [6] V. Preciado, M. Zargham, C. Enyioha, A. Jadbabaie, and G. Pappas, "Optimal resource allocation for network protection against spreading processes," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 99–109, 2014.
- [7] S. Saha, A. Adiga, B. Prakash, and A. Vullikanti, "Approximation algorithms for reducing the spectral radius to control epidemic spread," in *Proceeding of the 2015 SIAM International Conference on Data Mining*, 2015, pp. 568–576.
- [8] C. Dong, Q. Yin, W. Liu, Z. Yan, and T. Shi, "Can rewiring strategy control the epidemic spreading?" *Physica A*, vol. 438, pp. 169–177, 2015.
- [9] Y. Xu, T. Han, K. Cai, Z. Lin, G. Yan, and M. Fu, "A distributed algorithm for resource allocation over dynamic digraphs," *IEEE Transactions on Signal Processing*, vol. 65, no. 10, pp. 2600–2612, 2017.
- [10] H. Tong, B. Prakash, and T. Eliassi-Rad, "Gelling, and melting, large graph by edge manipulation." Proceedings of the 21st ACM international conference on Information and knowledge management, 2012, pp. 245–254.
- [11] H. Chan and L. Akoglu, "Optimizing network robustness by edge rewiring: A general framework," *Data Mining and Knowledge Discovery*, vol. 30, pp. 1395–1425, 2016.
- [12] H. Bhuiyan, M. Khan, J. Chen, and M. Marathe, "Parallel algorithms for switching edges in heterogeneous graphs," *Journal of Parallel and Distributed Computing*, vol. 104, pp. 19–35, 2017.
- [13] G. Demirel, E. Barter, and T. Gross, "Dynamics of epidemic disease on a growing adaptive networks," vol. 7, 2017.
- [14] T. Gross and B. Blasius, "Adaptive coevolutionary networks: A review," *Journal of The Royal Society Interface*, vol. 5, pp. 259–271, 2008.
- [15] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Proceeding of the 45th IEEE Conference on Decision and Control*, San Diego, CA, USA, 2006, pp. 6605–6611.
- [16] A. Bertrand and M. Moonen, "Distributed computation of the Fiedler vector with application to topology inference in ad hoc networks," *Signal Processing*, vol. 93, no. 5, pp. 1106–1117, 2013.
- [17] D. Xue, A. Gusrialdi, and S. Hirche, "A distributed strategy for near-optimal network topology design," in *Proceeding of the 21th International Symposium on Mathematical Theory of Networks and Systems*, Groningen, The Netherlands, 2014, pp. 7–14.
- [18] H. Liu, X. Cao, J. He, P. Cheng, C. Li, J. Chen, and Y. Sun, "Distributed identification of the most critical node for average consensus," *IEEE Transactions on Signal Processing*, vol. 63, no. 16, pp. 4315–4328, 2015.
- [19] S. Liu, P. Chen, and A. Hero, "Accelerated distributed dual averaging over evolving networks of growing connectivity," vol. 66, no. 7, pp. 1845–1859, 2018.
- [20] A. Nedić, "Convergence rate of distributed averaging dynamic and optimization in networks," *Foundations and Trends in System and Control*, vol. 2, no. 1, pp. 1–100, 2015.
- [21] R. Horn and C. Johnson, *Matrix analysis*. New York: Cambridge University Press, 2009.
- [22] D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos, "Epidemic thresholds in real networks," *ACM Transactions on Information and Systems Security*, vol. 10, no. 4, 2008.
- [23] C. Nowzari, V. Preciado, and G. Pappas, "Analysis and control of epidemics: A survey of spreading process on complex networks," *IEEE Control Systems Magazine*, vol. 36, no. 1, pp. 26–46, 2016.
- [24] A. Brouwer and W. Haemers, *Spectra of graphs*. Springer-Verlag New York, 2012.
- [25] P. Chen and O. H. I. A, "Deep community detection," *IEEE Transactions on Signal Processing*, vol. 63, no. 21, pp. 5706–5719, 2015.
- [26] J. G. Restrepo, E. Ott, and B. Hunt, "Onset of synchronization in large networks of coupled oscillators," *Physical Review E*, vol. 71, no. 036151, 2005.
- [27] A. Sydney, C. Scoglio, and D. Gruenbacher, "Optimizing algebraic connectivity by edge rewiring," *Applied Mathematics and Computation*, vol. 219, pp. 5465–5479, 2013.

- [28] Y. Nesterov and A. Nemirovskii, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.
- [29] G. Golub and C. van Loan, *Matrix computations*, 4th ed. Johns Hopkins University Press, 2012.
- [30] P. van Mieghem, *Graph spectra for complex networks*. Cambridge University Press, 2011.
- [31] G. Stewart and J. Sun, *Matrix perturbation theory*. Boston: Academic Press, 1990.
- [32] W. Heemels, K. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *51st Annual Conference on Decision and Control (CDC)*, 2012, pp. 3270–3285.
- [33] B. Oreshkin, M. Coates, and M. Rabbat, "Optimization and analysis of distributed averaging with short node memory," vol. 58, no. 5, pp. 2850–2865, 2014.
- [34] Q. Liu, Z. Wang, X. He, and D.-H. Zhou, "Event-triggered resilient filtering with measurement quantization and random sensor failures: Monotonicity and convergence," vol. 94, pp. 458–464, 2018.
- [35] Q. Liu, Z. Wang, X. He, and D. Zhou, "Event-based distributed filtering over Markovian switching topologies," 2018. [Online]. Available: DOI:10.1109/TAC.2018.2853570
- [36] F. Garin, D. Varagnolo, and K. Johansson, "Distributed estimation of diameter, radius and eccentricities in anonymous networks," in *Proceeding of 3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems*, Santa Barbara, CA, USA, 2012, pp. 13–18.
- [37] J. Hendrickx, R. Jungers, A. Olshevsky, and G. Vankeerberghen, "Graph diameter, eigenvalue and minimum-time consensus," *Automatica*, vol. 50, pp. 635–640, 2014.
- [38] R. Wood and M. O'Neill, "An always convergent method for finding the spectral radius of an irreducible non-negative matrix," *ANZIAM Journal*, vol. 45, pp. 474–485, 2003.
- [39] C. Castellano and R. Pastor-Satorras, "Relating topological determinants of complex networks to their spectral properties: Structural and dynamical effects," *Physical Review X*, vol. 7, p. 041024, 2017.
- [40] P. Jia, N. Friedkin, and F. Bullo, "Opinion dynamics and social power evolution over reducible influence networks," *SIAM Journal on Control and Optimization*, vol. 55, no. 2, pp. 1280–1301, 2017.
- [41] C. Godsil and G. Royle, *Algebraic graph theory*. Springer, 2001.
- [42] S. Sardellitti, S. Barbarossa, and A. Swami, "Optimal topology control and power allocation for minimum energy consumption in consensus networks," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 383–399, 2012.
- [43] E. Estrada, N. Hatano, and M. Benzi, "The physics of communicability in complex networks," *Physics Reports*, vol. 514, pp. 89–119, 2012.



Dong (Tony) Xue received the M.Sc. degree in control theory and engineering from the Tongji University, Shanghai, China, in 2011, and Dr.-Ing degree at the Chair of Information-oriented Control, Technische Universität München, Munich, Germany, in 2018. Since 2016, he is a research scholar with the Engineering and Technology Institute (ENTEG) at the University of Groningen, the Netherlands.

His current research interests include opinion dynamics and social networks, consensus-based applications in distributed control, estimation and optimization, and stochastic analysis and statistical physics in network systems.



Sandra Hirche holds the TUM Liesel Beckmann Distinguished Professorship and heads the Chair of Information-oriented Control in the Department of Electrical and Computer Engineering at Technical University of Munich, Germany (since 2013). She received the diploma engineer degree in Aeronautical and Aerospace Engineering in 2002 from the Technical University Berlin, Germany, and the Doctor of Engineering degree in Electrical and Computer Engineering in 2005 from Technical University of Munich, Germany. From 2005–2007 she has been a PostDoc fellow of the Japanese Society for the Promotion of Science at the Fujita Laboratory at Tokyo Institute of Technology, Japan. Prior to her present appointment she has been an associate professor at Technical University of Munich. Her main research interests include networked dynamical systems, cooperative and distributed control, and event-triggered control with applications in human-in-the-loop systems, robotics, and infrastructure systems. She has published more than 150 papers in international journals, books, and refereed conferences. Dr. Hirche has served or is serving on the Editorial Boards of the IEEE Transactions on Control Systems Technology and the IEEE Transactions on Haptics. She has received multiple awards such as the Rohde & Schwarz Award for her Ph.D. thesis in 2005, the IFAC World Congress Best Poster Award in 2005 and together with students Best Paper Awards of IEEE Worldhaptics and IFAC Conference of Manoeuvring and Control of Marine Craft in 2009.