A Subspace Consensus Approach for Distributed Connectivity Assessment of Asymmetric Networks

Mohammad Mehdi Asadi, Mohammad Khosravi, Stephane Blouin, and Amir G. Aghdam

***Abstract*— The problem of connectivity assessment of an asymmetric network represented by a weighted directed graph is investigated in this paper. The notion of generalized algebraic connectivity is formulated for this type of network in the context of distributed parameter estimation algorithms. The proposed connectivity measure is then defined in terms of the eigenvalues of the Laplacian matrix of the graph representing the network. A novel distributed algorithm based on the subspace consensus approach is developed to compute the generalized algebraic connectivity from the viewpoint of each node. The Laplacian matrix of the network is properly transformed such that the problem of finding the connectivity measure is reduced to the problem of finding the dominant eigenvalue of an asymmetric matrix. Two sequences of one-dimensional and two-dimensional subspaces are generated iteratively by each node such that either of them converges to the desired subspace spanned by the eigenvectors associated with the desired eigenvalues representing the network connectivity. The effectiveness of the developed algorithm is subsequently demonstrated by simulations.**

I. Introduction

Ad-hoc networks are composed of a collection of sensors capable of exchanging data through communication channels without the support of a pre-existing infrastructure [1], [2]. The convergence speed of cooperative algorithms used for various objectives such as consensus, target localization and parameter estimation over ad-hoc networks strongly depends on the connectivity degree of the network [3]. A network with a higher degree of connectivity is able to diffuse information more effectively throughout the network [4]. In parameter estimation algorithms, in particular, a set of unknown parameters are estimated iteratively using the data collected by the sensors which are corrupted by the measurement noise [5], [6]. This task can be done either in a centralized manner by relying on a network with a fusion center or in a distributed fashion using ad-hoc networks. Estimation algorithms performing distributed computation have significant advantages compared to the methods based on a fusion center such as scalability and resilience to node failure [7].

Traditionally, the algebraic connectivity of a symmetric network is defined as the second smallest eigenvalue of the Laplacian matrix of the network undirected graph [8]. A decentralized orthogonal iteration algorithm is introduced in [9] for computing the k dominant eigenvalues of a symmetric network. A distributed procedure is developed in [10] to estimate and control the algebraic connectivity of symmetric ad-hoc networks. Unlike symmetric networks, the notion of algebraic connectivity is not well-defined for asymmetric networks due to the dependency of consensus convergence rate on the initial state vector of such networks. A simple extension of algebraic connectivity to directed graphs is proposed in [11], where the magnitude of the smallest nonzero eigenvalue of the Laplacian matrix is introduced as a measure of network connectivity. However, this notion fails to represent any operational characteristic of the asymmetric network such as the convergence rate of cooperative algorithms running over it. The generalized algebraic connectivity notion is introduced in [12] and it captures the expected asymptotic convergence rate of continuous-time consensus algorithms running over an asymmetric network.

Motivated by applications in underwater acoustic sensors networks subject to asymmetric communication links [13], [14], the problem of connectivity assessment of asymmetric sensor networks is investigated in this work. The generalized algebraic connectivity is chosen as a novel measure of connectivity for networks represented by weighted directed graphs [12]. To justify the definition of generalized algebraic connectivity, the expected asymptotic convergence rate of a discrete-time parameter estimation algorithm with noise-corrupted measurements over an asymmetric network is investigated. A novel algorithm based on the power iteration is introduced to compute the generalized algebraic connectivity in a distributed manner. The Laplacian matrix of the network is transformed such that the original problem is reduced to finding the dominant eigenvalues of an asymmetric matrix. Since an asymmetric matrix can either have a real or complex dominant eigenvalue, every node generates sequences of one-dimensional and two-dimensional subspaces corresponding to the cases that the dominant eigenvalue is real and complex, respectively. Then, either of the subspace sequences converges to a desired subspace associated with the generalized algebraic connectivity of the network. The efficacy of the proposed algorithm is subsequently demonstrated by simulations.

The remainder of the paper is organized as follows. Some background and definitions are given in Section II. The network connectivity is formulated in terms of the convergence rate of a distributed estimation algorithm in Section III. Section IV introduces an iterative algorithm to compute the proposed connectivity measure in a distributed fashion. The

M. M. Asadi, M. Khosravi, and A. G. Aghdam are with the Department of Electrical and Computer Engineering, Concordia University, Montréal, Québec, Canada, {m_asadim, m_khosr, aghdam}@ece.concordia.ca.

S. Blouin is with the Defence Research and Development Canada, Dartmouth, Nova Scotia, Canada, Stephane.Blouin@drdc-rddc.gc.ca.

This work has been supported by PWGSC contract no. W7707-145674/001/HAL funded by Defence Research and Development Canada.

simulation results are subsequently presented in Section V, and conclusions are summarized in Section VI.

II. Preliminaries and Notation

Throughout this work, the set of nonnegative integers and nonnegative real numbers are denoted by $\mathbb{Z}_{\geq 0}$ and $\mathbb{R}_{\geq 0}$, respectively. Also $\mathbb{N}_n$ is the finite set of natural numbers $\{1, 2, \ldots, n\}$. The transpose and conjugate transpose of a complex vector $\mathrm{v} \in \mathbb{C}^n$ are represented by v^T and v^H, respectively. The inner product of two complex vectors $\mathrm{v}, \mathrm{w} \in \mathbb{C}^n$ is denoted by $\langle \mathrm{v}, \mathrm{w} \rangle = \mathrm{w}^\mathsf{H}\mathrm{v}$. The real part, imaginary part, and magnitude of a complex number c are represented by $\Re(c)$, $\Im(c)$, and $|c|$, respectively. The $n \times n$ identity matrix is denoted by $\mathbf{I}_n$, and the all-one column vector of length n is represented by 1_n. The matrix $\mathrm{Diag}(\mathrm{v}) \in \mathbb{C}^{n\times n}$ is defined as a diagonal matrix where the elements of the vector $\mathrm{v} \in \mathbb{C}^n$ appear on its main diagonal. Moreover, $\mathrm{e}_i \in \mathbb{R}^n$ denotes a column vector whose elements are all zero except for its i^{th} element which is one. The trace and determinant of a square matrix $\mathbf{A}$ are represented by $\mathrm{tr}(\mathbf{A})$ and $\det(\mathbf{A})$, respectively.

Let $\mathbf{A} : \mathcal{X} \rightarrow \mathcal{Y}$ be a linear transformation that maps the elements of domain $\mathcal{X}$ to codomain $\mathcal{Y}$. The image and kernel of $\mathbf{A}$ are defined as $\mathrm{Im}(\mathbf{A}) = \{\mathbf{A}\mathrm{x} \,|\, \mathrm{x} \in \mathcal{X}\}$ and $\mathrm{Ker}(\mathbf{A}) = \{\mathrm{x} \in \mathcal{X} \,|\, \mathbf{A}\mathrm{x} = 0\}$, respectively. A linear transformation $\mathbf{P} : \mathbb{C}^n \rightarrow \mathbb{C}^n$ is called a projector if $\mathbf{P}^2 = \mathbf{P}$. An orthogonal projection is defined as a projection map whose image and kernel are orthogonal subspaces, i.e., $\mathrm{Im}(\mathbf{P}) = \mathrm{Ker}(\mathbf{P})^\perp$ [15]. Let $\mathbf{B} \in \mathbb{C}^{n\times k}$ be a matrix whose column vectors represent a basis for the k-dimensional subspace $\mathcal{S}$ of $\mathbb{C}^n$. Then, the orthogonal projection onto subspace $\mathcal{S}$ is defined as $\mathbf{P} = \mathcal{P}(\mathbf{B})$, where the function $\mathcal{P}(\cdot) : \mathbb{C}^{n\times k} \rightarrow \mathbb{C}^{n\times n}$ is defined as $\mathcal{P}(\mathbf{B}) = \mathbf{B}(\mathbf{B}^\mathsf{H}\mathbf{B})^{-1}\mathbf{B}^\mathsf{H}$. Consider two distinct subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$ of $\mathbb{C}^n$. The distance between $\mathcal{S}_1$ and $\mathcal{S}_2$ is defined as $\mathcal{D} = \|\mathbf{P}_1 - \mathbf{P}_2\|$, where $\mathbf{P}_1$ and $\mathbf{P}_2$ are the orthogonal projectors onto subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively [16].

III. Convergence Rate of Distributed Estimation Algorithms

The notion of *generalized algebraic connectivity* introduced in [12] as a measure of connectivity for asymmetric networks will now be formulated in the context of parameter estimation problem. To this end, the distributed estimation of unknown parameters in the discrete-time case is considered as the desired cooperative goal of the network. Note that connectivity has a significant impact on data diffusion across a network where each node exchanges information only with its neighbors. As a result, communication between the nodes is more efficient in a more connected network, in general, and the unknown parameters are identified in a faster and more reliable fashion in such networks. Given its importance, the convergence rate of a parameter estimation algorithm over an asymmetric network will be investigated next for the discrete-time case.

Consider the problem of distributed estimation of a vector of unknown parameters $\theta \in \mathbb{R}^m$ using an asymmetric network of n nodes and represented by a weighted digraph $\mathscr{G} = (\mathscr{V}, \mathscr{E}, \mathbf{W})$. Let $\mathrm{y}_i \in \mathbb{R}^{m_i}$ denote the observation vector of the i^{th} node which is corrupted by the additive noise $\mathrm{u}_i \in \mathbb{R}^{m_i}$, i.e.

$$\mathrm{y}_i = \mathbf{A}_i\theta + \mathrm{u}_i, \tag{1}$$

for any node $i \in \mathbb{N}_n$, where the known observation matrix $\mathbf{A}_i \in \mathbb{R}^{m_i\times m}$ relates the unknown parameters to the measurements of the i^{th} node, and u_i represents the vector of measurement noise. It is assumed that the noise vectors $\{\mathrm{u}_i\}_{i\in\mathbb{N}_n}$ are independent and identically distributed (i.i.d.) with zero mean and bounded second moment, where $\{\mathbf{\Sigma}_i\}_{i\in\mathbb{N}_n}$ represents the corresponding set of covariance matrices. As a result, the i^{th} node observes m_i linear combinations of the elements of θ, where $m_i \ll m$ holds by assumption. The aggregate measurement of all nodes is then given by

$$\mathbf{y} = \mathbf{A}\theta + \mathbf{u}, \tag{2}$$

where $\mathbf{y} = [\mathrm{y}_1^\mathsf{T} \cdots \mathrm{y}_n^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{\bar{m}}$, $\mathbf{A} = [\mathbf{A}_1^\mathsf{T} \cdots \mathbf{A}_n^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{\bar{m}\times m}$, and $\mathbf{u} = [\mathrm{u}_1^\mathsf{T} \cdots \mathrm{u}_n^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{\bar{m}}$ with augmented covariance matrix $\mathbf{\Sigma} = \mathrm{Diag}(\mathbf{\Sigma}_1, \ldots, \mathbf{\Sigma}_n)$ and $\bar{m} = \sum_{i=1}^n m_i$. It is assumed that $m \leq \bar{m}$, and that the augmented observation matrix $\mathbf{A}$ is full-rank. Let $\hat{\theta} \in \mathbb{R}^m$ denote the minimum-variance unbiased estimate of the unknown vector θ when the augmented measurement vector $\mathbf{y}$ along with matrices $\mathbf{A}$ and $\mathbf{\Sigma}$ are given. Then, $\hat{\theta}$ is obtained in a centralized fashion by [5]

$$\begin{aligned}\hat{\theta} &= (\mathbf{A}^\mathsf{T}\mathbf{\Sigma}^{-1}\mathbf{A})^{-1}\mathbf{A}^\mathsf{T}\mathbf{\Sigma}^{-1}\mathbf{y} \\ &= \Big(\sum_{i=1}^n \mathbf{A}_i^\mathsf{T}\mathbf{\Sigma}_i^{-1}\mathbf{A}_i\Big)^{-1}\sum_{i=1}^n \mathbf{A}_i^\mathsf{T}\mathbf{\Sigma}_i^{-1}\mathrm{y}_i. \end{aligned}\tag{3}$$

For the special case when the noises are jointly Gaussian, $\hat{\theta}$ also provides the maximum likelihood estimate of θ [5]. One can find $\hat{\theta}$, given in (3), in a distributed manner by reducing the problem into two average consensus problems. Two distributed discrete-time consensus procedures are used to compute the terms $\frac{1}{n}\sum_{i=1}^n \mathbf{A}_i^\mathsf{T}\mathbf{\Sigma}_i^{-1}\mathrm{y}_i$ and $\frac{1}{n}\sum_{i=1}^n \mathbf{A}_i^\mathsf{T}\mathbf{\Sigma}_i^{-1}\mathbf{A}_i$, respectively, by solving the averaging problems from the standpoint of each node. Let $\hat{\theta}_i(k) \in \mathbb{R}^m$ represent the estimate of $\hat{\theta}$ obtained by the i^{th} node at the k^{th} iteration of the algorithm, such that

$$\hat{\theta}_i(k) = \big(\mathbf{Q}_i(k)\big)^{-1}\mathrm{q}_i(k), \tag{4}$$

where $\mathrm{q}_i(k) \in \mathbb{R}^m$ and $\mathbf{Q}_i(k) \in \mathbb{R}^{m\times m}$ are, respectively, the estimates of the terms $\sum_{i=1}^n \mathbf{A}_i^\mathsf{T}\mathbf{\Sigma}_i^{-1}\mathrm{y}_i$ and $\sum_{i=1}^n \mathbf{A}_i^\mathsf{T}\mathbf{\Sigma}_i^{-1}\mathbf{A}_i$ [5]. Then, $\mathrm{q}_i(k)$ and $\mathbf{Q}_i(k)$ are obtained by the i^{th} node using the information it receives from the neighbors as well as its own information based on the following update procedures

$$\mathrm{q}_i(k) = \mathrm{q}_i(k-1) - \delta(k-1)\sum_{j\in\mathcal{N}_i} w_{ij}(\mathrm{q}_i(k-1) - \mathrm{q}_j(k-1)), \tag{5a}$$

$$\mathbf{Q}_i(k) = \mathbf{Q}_i(k-1) - \delta(k-1)\sum_{j\in\mathcal{N}_i} w_{ij}(\mathbf{Q}_i(k-1) - \mathbf{Q}_j(k-1)). \tag{5b}$$

The set of diminishing step sizes $\{\delta(j)\}_{j\in\mathbb{Z}_{\geq 0}}$ used in (5) satisfy the following three conditions according to [10], [17]: (i) $\lim_{k\rightarrow\infty}\sum_{j=0}^{k-1}\delta(j) = \infty$,

(ii) $\lim_{k\to\infty}\sum_{j=0}^{k-1}\delta^2(j)<\infty$, and
(iii) $0<\delta(j)<\Delta^{-1}$ for any $j\in\mathbb{Z}_{\geq 0}$, where $\Delta=\max_{i\in\mathbb{N}_n}\sum_{j\in\mathscr{N}_i}w_{ij}$ and $\lim_{j\to\infty}\delta(j)=0$.

By considering a quasi-strongly connected (QSC) weighted digraph $\mathscr{G}$ and using the above consensus update laws, the final state vector q^* and final state matrix $\mathbf{Q}^*$ that all nodes will converge to are given by

$$\mathrm{q}^*=\tfrac{1}{\langle \mathrm{w}_1(\mathbf{L}),1_n\rangle}\sum_{i=1}^{n}\langle \mathrm{w}_1(\mathbf{L}),\mathrm{e}_i\rangle \mathrm{q}_i(0), \tag{6a}$$

$$\mathbf{Q}^*=\tfrac{1}{\langle \mathrm{w}_1(\mathbf{L}),1_n\rangle}\sum_{i=1}^{n}\langle \mathrm{w}_1(\mathbf{L}),\mathrm{e}_i\rangle \mathbf{Q}_i(0), \tag{6b}$$

where $\mathrm{w}_1(\mathbf{L})$ denotes the left eigenvector of $\mathbf{L}$ corresponding to its zero eigenvalue. Assume that $\mathrm{w}_1(\mathbf{L})$ is available to all nodes, and the update procedures (5) are initialized by setting

$$\mathrm{q}_i(0)=\frac{\langle \mathrm{w}_1(\mathbf{L}),1_n\rangle}{\langle \mathrm{w}_1(\mathbf{L}),\mathrm{e}_i\rangle}\mathbf{A}_i^{\mathsf{T}}\mathbf{\Sigma}_i^{-1}\mathrm{y}_i, \tag{7a}$$

$$\mathbf{Q}_i(0)=\frac{\langle \mathrm{w}_1(\mathbf{L}),1_n\rangle}{\langle \mathrm{w}_1(\mathbf{L}),\mathrm{e}_i\rangle}\mathbf{A}_i^{\mathsf{T}}\mathbf{\Sigma}_i^{-1}\mathbf{A}_i, \tag{7b}$$

for every node $i\in\mathbb{N}_n$. Under these conditions, it is guaranteed that the average consensus over the weighted digraph $\mathscr{G}$ will be achieved as k goes to infinity. By aggregating the state vector of all nodes, one can form the augmented state vector $\mathbf{q}(k)=[\mathrm{q}_1^{\mathsf{T}}(k)\ \cdots\ \mathrm{q}_n^{\mathsf{T}}(k)]^{\mathsf{T}}\in\mathbb{R}^{mn}$ with the following dynamics

$$\begin{aligned}\mathbf{q}(k)&=((\mathbf{I}_n-\delta(k-1)\mathbf{L})\otimes\mathbf{I}_m)\mathbf{q}(k-1)\\&=(\prod_{j=0}^{k-1}(\mathbf{I}_n-\delta(j)\mathbf{L})\otimes\mathbf{I}_m)\mathbf{q}(0).\end{aligned} \tag{8}$$

The augmented state matrix $\mathbf{Q}(k)=[\mathbf{Q}_1^{\mathsf{T}}(k)\ \cdots\ \mathbf{Q}_n^{\mathsf{T}}(k)]^{\mathsf{T}}\in\mathbb{R}^{mn\times m}$ is also given by the following update procedure

$$\mathbf{Q}(k)=(\prod_{j=0}^{k-1}(\mathbf{I}_n-\delta(j)\mathbf{L})\otimes\mathbf{I}_m)\mathbf{Q}(0). \tag{9}$$

In order to investigate the rate of convergence to consensus, and on noting that (8) and (9) follow similar dynamics, it suffices to examine the convergence rate to consensus using the update law given by (8). To this end, let $\tilde{\mathbf{q}}(k)$ denote the error vector of the network at the k^{th} iteration defined as

$$\tilde{\mathbf{q}}(k)=\mathbf{q}(k)-1_n\otimes\mathrm{q}^*=\mathbf{q}(k)-\Big(\frac{1_n\,\mathrm{w}_1^{\mathsf{T}}(\mathbf{L})}{\mathrm{w}_1^{\mathsf{T}}(\mathbf{L})\,1_n}\otimes\mathbf{I}_m\Big)\mathbf{q}(0). \tag{10}$$

By defining $\mathbf{J}=\frac{1_n\,\mathrm{w}_1^{\mathsf{T}}(\mathbf{L})}{\mathrm{w}_1^{\mathsf{T}}(\mathbf{L})\,1_n}\otimes\mathbf{I}_m$ and since 1_n is the right eigenvector of $\mathbf{L}$ corresponding to its zero eigenvalue, one has $((\mathbf{I}_n-\delta(j)\mathbf{L})\otimes\mathbf{I}_m)\,\mathbf{J}=\mathbf{J}$ for any $j\in\{0,1,\ldots,k-1\}$. As a result, $(\prod_{j=0}^{k-1}(\mathbf{I}_n-\delta(j)\mathbf{L})\otimes\mathbf{I}_m)\,\mathbf{J}=\mathbf{J}$, and it results from (10) that

$$\begin{aligned}\tilde{\mathbf{q}}(k)&=(\prod_{j=0}^{k-1}(\mathbf{I}_n-\delta(j)\mathbf{L})\otimes\mathbf{I}_m)\mathbf{q}(0)-\mathbf{J}\,\mathbf{q}(0)\\&=(\prod_{j=0}^{k-1}(\mathbf{I}_n-\delta(j)\mathbf{L})\otimes\mathbf{I}_m)\tilde{\mathbf{q}}(0).\end{aligned} \tag{11}$$

Define $\mathfrak{d}(k)=\|\tilde{\mathbf{q}}(k)\|$ as the disagreement function representing the Euclidean norm of the difference between the augmented state vector of the network at the k^{th} iteration and the final agreement state. Since $\mathscr{G}$ is QSC, it is guaranteed that $\mathbf{q}(k)$ converges to $1_n\otimes\mathrm{q}^*$ as k goes to infinity, i.e., $\lim_{k\to\infty}\mathfrak{d}(k)=0$. The *asymptotic convergence rate* is defined as $-\lim_{k\to\infty}\log(\mathfrak{d}(k))(\sum_{j=0}^{k-1}\delta(j))^{-1}$, which represents the speed of decreasing the disagreement function to zero in an asymptotic manner. The asymptotic convergence rate of the distributed parameter estimation algorithm is characterized in the next theorem.

Theorem 1: Consider a QSC weighted digraph $\mathscr{G}$ with Laplacian matrix $\mathbf{L}$. Let $\mathbf{q}(k)$ denote the augmented state vector of a network represented by $\mathscr{G}$ at the k^{th} iteration whose dynamics is given by (8). Let also the augmented initial state $\mathbf{q}(0)$ be a unit random vector with a continuous probability distribution, which is obtained from the noise-corrupted measurements (2). By defining $\tilde{\lambda}(\mathbf{L})=\min_{\lambda_i(\mathbf{L})\neq 0,\ \lambda_i(\mathbf{L})\in\Lambda(\mathbf{L})}\Re(\lambda_i(\mathbf{L}))$, it follows that the asymptotic rate of convergence to consensus is almost surely equal to $\tilde{\lambda}(\mathbf{L})$, i.e., $\mathbb{P}[-\lim_{k\to\infty}\log(\mathfrak{d}(k))(\sum_{j=0}^{k-1}\delta(j))^{-1}=\tilde{\lambda}(\mathbf{L})]=1$.

Proof: The proof is omitted due to space limitations and can be found in [18]. ■

Theorem 1 shows that the expected asymptotic convergence rate of distributed estimation algorithm in the discrete-time case is well described by $\tilde{\lambda}(\mathbf{L})$, as defined in the statement of the theorem, which is equal to the smallest nonzero real part of the eigenvalues of $\mathbf{L}$. This motivates the introduction of a new connectivity measure for weighted digraphs.

Definition 1: Given a QSC weighted digraph $\mathscr{G}$ with Laplacian matrix $\mathbf{L}$, the *generalized algebraic connectivity* (GAC) of $\mathscr{G}$, denoted by $\tilde{\lambda}(\mathbf{L})$, is defined as

$$\tilde{\lambda}(\mathbf{L})=\min_{\lambda_i(\mathbf{L})\neq 0,\ \lambda_i(\mathbf{L})\in\Lambda(\mathbf{L})}\Re(\lambda_i(\mathbf{L})). \tag{12}$$

IV. Distributed Computation of the Generalized Algebraic Connectivity

A distributed algorithm is presented in this section to compute the GAC of a weighted digraph $\mathscr{G}$ from the viewpoint of each node. To this end, a matrix transformation, borrowed from [12], is provided in the next lemma.

Lemma 1: Let $\mathbf{L}$ be the Laplacian matrix of a weighted digraph $\mathscr{G}$ composed of n nodes. Assume that the zero eigenvalue of $\mathbf{L}$ has multiplicity one, and define the modified Laplacian matrix of the digraph as

$$\tilde{\mathbf{L}}=\mathrm{e}^{\mathbf{I}_n-\delta\mathbf{L}}-\exp(1)\,\mathrm{w}_1(\mathbf{L})\,\mathrm{w}_1^{\mathsf{T}}(\mathbf{L}), \tag{13}$$

where $\delta<\Delta^{-1}$ for $\Delta=\max_{i\in\mathbb{N}_n}\sum_{j\in\mathscr{N}_i}w_{ij}$. It then follows that

$$\tilde{\lambda}(\mathbf{L})=\frac{1}{\delta}\big(1-\log(\max_{\lambda_i(\tilde{\mathbf{L}})\in\Lambda(\tilde{\mathbf{L}})}|\lambda_i(\tilde{\mathbf{L}})|)\big). \tag{14}$$

The following two assumptions are required.

Assumption 1: The weighted digraph $\mathscr{G}$ representing the network is strongly connected.

Assumption 2: The modified Laplacian matrix $\tilde{\mathbf{L}}$ of the network digraph has either a real dominant eigenvalue or a pair of complex conjugate dominant eigenvalues.

The algorithm given in the next subsection is an extension of the well-known power iteration method which has been used extensively in the literature to compute the algebraic connectivity of symmetric networks in both centralized and distributed fashions [10], [11]. The power iteration algorithm computes the dominant eigenvalue of a matrix (i.e., an eigenvalue with maximum magnitude) under the assumption that the matrix has one dominant eigenvalue whose magnitude is strictly greater than the magnitude of the other eigenvalues [19]. As a result, two main challenges in using the power iteration method to compute the GAC, as noted in [12], are as follows:

1) the power iteration method computes the eigenvalue with maximum (not minimum) magnitude (not real part), and

2) the convergence of the power iteration procedure is not guaranteed when there are two eigenvalues with largest magnitude (i.e., a pair of complex conjugate dominant eigenvalues).

To address the first challenge, the Laplacian matrix $\mathbf{L}$ is transformed into the modified Laplacian matrix $\tilde{\mathbf{L}}$ according to Lemma 1. This transformation converts the problem of finding $\tilde{\lambda}(\mathbf{L})$ into the problem of finding the dominant eigenvalue of matrix $\tilde{\mathbf{L}}$. Since the asymmetric network, considered in this work, is represented by a weighted digraph $\mathscr{G}$ with a real weight matrix $\mathbf{W}$, the dominant eigenvalue(s) of $\tilde{\mathbf{L}}$ can appear as a real number or a complex conjugate pair under Assumption 2. Therefore, $\tilde{\mathbf{L}}$ could have one or two dominant eigenvalue(s), and the power-iteration-based algorithms are not able to address the problem of finding the dominant eigenvalue(s) of $\tilde{\mathbf{L}}$.

To address the second challenge, note that it is not known *a priori* whether the dominant eigenvalue of $\tilde{\mathbf{L}}$ is real or complex. Thus, for any $i \in \mathbb{N}_n$, node i constructs a one-dimensional subspace $\mathcal{W}_k^i$ and a two-dimensional subspace $\mathcal{V}_k^i$ at the k^{th} iteration of the algorithm corresponding to the cases that the dominant eigenvalue of $\tilde{\mathbf{L}}$ is real and complex, respectively. Note that the subspaces $\mathcal{W}_k^i$ and $\mathcal{V}_k^i$ are spanned by the state vectors which are obtained in a distributed manner from the viewpoint of the i^{th} node. Therefore, every node i generates a sequence of one-dimensional subspaces $\{\mathcal{W}_k^i\}_{k\in\mathbb{N}}$ and a sequence of two-dimensional subspaces $\{\mathcal{V}_k^i\}_{k\in\mathbb{N}}$. Under Assumptions 1 and 2, it can be shown that if the dominant eigenvalue of $\tilde{\mathbf{L}}$ is a real number, the subspace sequence $\{\mathcal{W}_k^i\}_{k\in\mathbb{N}}$ converges to the one-dimensional subspace $\mathcal{W}$ spanned by the right eigenvector of $\tilde{\mathbf{L}}$ associated with its real dominant eigenvalue for every node $i \in \mathbb{N}_n$. However, if $\tilde{\mathbf{L}}$ has a pair of complex conjugate dominant eigenvalues, only the subspace sequence $\{\mathcal{V}_k^i\}_{k\in\mathbb{N}}$ converges to the two-dimensional subspace $\mathcal{V}$ spanned by two right eigenvectors associated with the pair of complex conjugate dominant eigenvalues of $\tilde{\mathbf{L}}$, for every node $i \in \mathbb{N}_n$. Once any of the two subspace sequences converges to consensus from the standpoint of every node, the restriction of $\tilde{\mathbf{L}}$ to the desired subspace $\mathcal{W}$ or $\mathcal{V}$ can be obtained in a distributed manner, giving the magnitude of the dominant eigenvalue of $\tilde{\mathbf{L}}$. Then, every node can compute GAC using equation (14).

A. Description of the Proposed Distributed Algorithm

Throughout the algorithm, the modified Laplacian matrix $\tilde{\mathbf{L}}$ is approximated by $\tilde{\mathbf{L}}_k$ at the k^{th} iteration, where the matrix exponential $\mathrm{e}^{\mathbf{I}_n-\delta\mathbf{L}}$ used in the definition of $\tilde{\mathbf{L}}$ in (13) is substituted by the sum of the first k terms in its Taylor expansion, i.e.,

$$\tilde{\mathbf{L}}_k = \sum_{j=0}^{k} \frac{1}{j!}(\mathbf{I}_n - \delta\mathbf{L})^j - \exp(1)\mathrm{w}_1(\mathbf{L})\,\mathrm{w}_1^{\mathsf{T}}(\mathbf{L}). \qquad (15)$$

In the centralized implementation of the algorithm, one can define $\mathrm{x}_k \in \mathbb{R}^n$ as the state vector of network at the k^{th} iteration, which is updated using the following procedure

$$\mathrm{x}_k = \frac{\tilde{\mathbf{L}}_k\,\mathrm{x}_{k-1}}{\|\tilde{\mathbf{L}}_k\,\mathrm{x}_{k-1}\|} = \frac{(\tilde{\mathbf{L}}_k)^k\,\mathrm{x}_0}{\|(\tilde{\mathbf{L}}_k)^k\,\mathrm{x}_0\|}. \qquad (16)$$

The initial state vector x_0 is also chosen such that $\mathrm{x}_0 \not\perp \operatorname{span}\{\mathrm{v}_{n-1}(\tilde{\mathbf{L}}), \mathrm{v}_n(\tilde{\mathbf{L}})\}$. In order to implement the update procedure (16) in a distributed fashion, let $\mathrm{x}_k^i \in \mathbb{R}^n$ denote the state vector from the viewpoint of the i^{th} node at the k^{th} iteration of the distributed algorithm. Note that node i can only directly update the i^{th} element of its state vector at each iteration due to the distributed nature of the update procedure, for any $i \in \mathbb{N}_n$. The procedure employed by the i^{th} node to obtain the updated state vector x_k^i in an iterative manner using the state vectors x_{k-1}^j for $j \in \mathcal{N}_i \cup \{i\}$ is elaborated in the next three steps.

(1) Let the real scalar $y_{k-1}^i(0) = \langle \mathrm{x}_{k-1}^i, \mathrm{e}_i\rangle$ denote the i^{th} element of the state vector x_{k-1}^i obtained at the $(k-1)^{\text{th}}$ iteration by the i^{th} node. Define $\mathrm{y}_{k-1}(0) = [y_{k-1}^1(0)\ \cdots\ y_{k-1}^n(0)]^{\mathsf{T}} \in \mathbb{R}^n$ as the augmented initial vector formed by aggregating the scalars $y_{k-1}^i(0)$ for every $i \in \mathbb{N}_n$. In this step, the i^{th} node is required to compute the i^{th} element of the vector $\hat{\mathrm{y}}_k = \tilde{\mathbf{L}}_k\ \mathrm{y}_{k-1}(0)$. To this end, define $y_{k-1}^i(l)$ as the i^{th} element of the vector $\mathrm{y}_{k-1}(l) \in \mathbb{R}^n$ which is obtained as $\mathrm{y}_{k-1}(l) = (\mathbf{I}_n - \delta\mathbf{L})^l\mathrm{y}_{k-1}(0)$ for any $l \in \mathbb{N}_k$ and $k \in \mathbb{N}$. Then, $y_{k-1}^i(l)$ is computed using the following update law

$$y_{k-1}^i(l) = y_{k-1}^i(l-1) - \delta\sum_{j=1}^{n} w_{ij}\,(y_{k-1}^i(l-1) - y_{k-1}^j(l-1)), \qquad (17)$$

for any $i \in \mathbb{N}_n$ and $l \in \mathbb{N}_k$ at the k^{th} iteration. After repeating the above procedure k times, the i^{th} element of the vector $\hat{\mathrm{y}}_k$, denoted by $\hat{y}_k^i$, is obtained as

$$\hat{y}_k^i = \sum_{l=0}^{k} \frac{1}{l!}\hat{y}_{k-1}^i(l) - \exp(1)\langle \mathrm{w}_1(\mathbf{L}), \mathrm{e}_i\rangle\langle \mathrm{w}_1(\mathbf{L}), \mathrm{x}_{k-1}^i\rangle. \qquad (18)$$

(2) According to the update procedures in (17) and (18), merely the i^{th} element of the updated state vector at the k^{th} iteration of the distributed algorithm is available to the i^{th} node. To ensure that every node has access to all elements of the updated state vector obtained by the other nodes, the notion of consensus observer from

[11] is used in this step. The consensus observer is used to iteratively diffuse the updated elements of the state vector perceived by each node throughout the network. Let $\mathrm{z}_k^i(m) \in \mathbb{R}^n$ denote the state vector of the consensus observer at its m^{th} step from the standpoint of the i^{th} node at the k^{th} iteration. By assuming that the consensus observer is executed k times at the k^{th} iteration of the distributed algorithm, one has $m \in \mathbb{N}_k$. The state vector of the consensus observer is initialized as $\mathrm{z}_k^i(0) = \hat{y}_k^i\, \mathrm{e}_i$ for any $i \in \mathbb{N}_n$ and $k \in \mathbb{N}$. Then, $\mathrm{z}_k^i(m)$ is updated as follows

$$\mathrm{z}_k^i(m)=\mathrm{z}_k^i(m-1)-\delta\sum_{j=1}^{n}w_{ij}(\mathrm{z}_k^i(m-1)-\mathrm{z}_k^j(m-1)), \quad (19)$$

for any $i \in \mathbb{N}_n$, $m \in \mathbb{N}_k$, and $k \in \mathbb{N}$.

(3) By considering $\mathrm{z}_k^i(k)$ as the output of the consensus observer after k steps from the viewpoint of node i, the updated state vector x_k^i computed by the i^{th} node at the k^{th} iteration is given by

$$\mathrm{x}_k^i = \frac{\mathbf{\Xi}^{-1}\, \mathrm{z}_k^i(k)}{\|\mathbf{\Xi}^{-1}\, \mathrm{z}_k^i(k)\|}, \quad (20)$$

where $\mathbf{\Xi} = \frac{1}{\langle \mathrm{w}_1(\mathbf{L}), 1_n\rangle}\mathrm{Diag}(\mathrm{w}_1(\mathbf{L}))$. This completes the required steps to obtain x_k^i for any $i \in \mathbb{N}_n$ and $k \in \mathbb{N}$.

Let $\mathcal{W}_k^i = \mathrm{span}\{\mathrm{x}_k^i\}$ and $\mathcal{V}_k^i = \mathrm{span}\{\mathrm{x}_{k-1}^i, \mathrm{x}_k^i\}$ define the one-dimensional and two-dimensional subspaces formed to obtain, respectively, the real and complex dominant eigenvalues of matrix $\tilde{\mathbf{L}}_k$ at the k^{th} iteration of the distributed algorithm by the i^{th} node. Since every subspace is uniquely characterized by its projection operator, define $\bar{\mathbf{P}}_k^i$ and $\mathbf{P}_k^i$ as the projectors corresponding to the subspaces $\mathcal{W}_k^i$ and $\mathcal{V}_k^i$, respectively, at the k^{th} iteration of the algorithm from the viewpoint of the i^{th} node. Moreover, $\bar{\mathbf{P}}_k^i = \mathcal{P}(\bar{\mathbf{B}}_k^i)$ and $\mathbf{P}_k^i = \mathcal{P}(\mathbf{B}_k^i)$ by definition, where $\bar{\mathbf{B}}_k^i = \mathrm{x}_k^i$ and $\mathbf{B}_k^i = [\mathrm{x}_{k-1}^i\ \mathrm{x}_k^i]$. Once the subspaces $\mathcal{W}_k^i$ and $\mathcal{V}_k^i$ associated with the dominant eigenvalue(s) of $\tilde{\mathbf{L}}_k$ are identified, the restriction of $\tilde{\mathbf{L}}_k$ onto $\mathcal{W}_k^i$ and $\mathcal{V}_k^i$, denoted by $\bar{\mathbf{R}}_k^i$ and $\mathbf{R}_k^i$, respectively, are computed by $\bar{\mathbf{R}}_k^i = (\bar{\mathbf{B}}_k^{i,\mathsf{T}}\bar{\mathbf{B}}_k^i)^{-1}\bar{\mathbf{B}}_k^{i,\mathsf{T}}\tilde{\mathbf{L}}_k\bar{\mathbf{B}}_k^i$ and $\mathbf{R}_k^i = (\mathbf{B}_k^{i,\mathsf{T}}\mathbf{B}_k^i)^{-1}\mathbf{B}_k^{i,\mathsf{T}}\tilde{\mathbf{L}}_k\mathbf{B}_k^i$. On noting that no node has access to the approximate modified Laplacian matrix $\tilde{\mathbf{L}}_k$ during the distributed implementation, the i^{th} node can use $\mathbf{\Xi}^{-1}\, \mathrm{z}_k^i(k)$ and $\mathbf{\Xi}^{-1}\, \mathrm{z}_{k+1}^i(k+1)$ as approximations for the terms $\tilde{\mathbf{L}}_k\, \mathrm{x}_{k-1}^i$ and $\tilde{\mathbf{L}}_k\, \mathrm{x}_k^i$, respectively. This results in

$$\bar{\mathbf{R}}_k^i = (\bar{\mathbf{B}}_k^{i,\mathsf{T}}\bar{\mathbf{B}}_k^i)^{-1}\bar{\mathbf{B}}_k^{i,\mathsf{T}}\mathbf{\Xi}^{-1}\mathrm{z}_{k+1}^i(k+1), \quad (21a)$$

$$\mathbf{R}_k^i = (\mathbf{B}_k^{i,\mathsf{T}}\mathbf{B}_k^i)^{-1}\mathbf{B}_k^{i,\mathsf{T}}\mathbf{\Xi}^{-1}[\mathrm{z}_k^i(k)\ \mathrm{z}_{k+1}^i(k+1)]. \quad (21b)$$

Define $\hat{\bar{\lambda}}_k^i$ and $\hat{\lambda}_k^i$ as the real and complex dominant eigenvalues of matrix $\tilde{\mathbf{L}}_k$ obtained at the k^{th} iteration of the distributed algorithm by the i^{th} node, such that

$$\hat{\bar{\lambda}}_k^i = \bar{\mathbf{R}}_k^i, \quad (22a)$$

$$\hat{\lambda}_k^i = \tfrac{1}{2}\mathrm{tr}(\mathbf{R}_k^i) + \sqrt{(\tfrac{1}{2}\mathrm{tr}(\mathbf{R}_k^i))^2 - \det(\mathbf{R}_k^i)}. \quad (22b)$$

By inverse transformations of (13), $\tilde{\bar{\lambda}}_k^i$ and $\tilde{\lambda}_k^i$ representing the estimates of the GAC from the viewpoint of node i at the k^{th} iteration of the algorithm are obtained as $\tilde{\bar{\lambda}}_k^i = \frac{1}{\delta}[1 - \log(\hat{\bar{\lambda}}_k^i)]$ and $\tilde{\lambda}_k^i = \frac{1}{\delta}[1 - \log(|\hat{\lambda}_k^i|)]$. To examine the convergence of the algorithm, $\bar{\mathcal{D}}_{k+1}^i$ and $\mathcal{D}_{k+1}^i$ are computed as follows

$$\bar{\mathcal{D}}_{k+1}^i = \|\bar{\mathbf{P}}_k^i - \bar{\mathbf{P}}_{k-1}^i\|, \quad (23a)$$

$$\mathcal{D}_{k+1}^i = \|\mathbf{P}_k^i - \mathbf{P}_{k-1}^i\|, \quad (23b)$$

which represent the distance between the last consecutive pair of estimated one-dimensional and two-dimensional subspaces, respectively, from the viewpoint of node i. If either $\bar{\mathcal{D}}_{k+1}^i$ or $\mathcal{D}_{k+1}^i$ becomes less than a prescribed convergence threshold ϵ, it is concluded that the algorithm has converged. Otherwise, it moves to the $(k+1)^{\text{th}}$ step. If the inequality $\bar{\mathcal{D}}_{k+1}^i < \epsilon$ holds, it follows that the dominant eigenvalue of $\tilde{\mathbf{L}}_k$ is real and $\tilde{\bar{\lambda}}_k^i$ represents the estimated GAC of the network from the standpoint of node i. However, if $\mathcal{D}_{k+1}^i < \epsilon$, the dominant eigenvalue of $\tilde{\mathbf{L}}_k$ is a complex conjugate pair and the estimated GAC of the network is given by $\tilde{\lambda}_k^i$. The convergence of the proposed algorithm is addressed by the next theorem.

Theorem 2: Consider an asymmetric network composed of n nodes which is represented by weighted digraph $\mathscr{G}$. Let Assumptions 1 and 2 hold. By applying the proposed distributed algorithm to this network. It follows that either

(i) $\lim_{k\to\infty} \mathcal{V}_k^i = \mathcal{V}$ and $\lim_{k\to\infty} \tilde{\lambda}_k^i = \tilde{\lambda}(\mathbf{L})$ for every node $i \in \mathbb{N}_n$ or,

(ii) $\lim_{k\to\infty} \mathcal{W}_k^i = \mathcal{W}$ and $\lim_{k\to\infty} \tilde{\bar{\lambda}}_k^i = \tilde{\lambda}(\mathbf{L})$ for every node $i \in \mathbb{N}_n$.

Proof: The proof is omitted due to space limitations and can be found in [18]. ■

V. SIMULATION RESULTS

Example 1: Consider an asymmetric network composed of three nodes represented by a strongly connected weighted digraph $\mathscr{G}_1 = (\mathscr{V}, \mathscr{E}_1, \mathbf{W}_1)$ whose weight matrix is given by

$$\mathbf{W}_1 = \begin{bmatrix} 0 & 0.09 & 0.59 \\ 0.28 & 0 & 0.17 \\ 0.07 & 0.4 & 0 \end{bmatrix}. \quad (24)$$

For this example, $\tilde{\lambda}(\mathbf{L}_1) = 0.8$ is the GAC of the network which corresponds to a pair of complex conjugate eigenvalues $0.8 \pm \mathrm{j}\,0.2505$ of $\mathbf{L}_1$. The performance of the proposed algorithm in a distributed implementation is evaluated in this example by choosing $\delta = \frac{1}{3}$ and $\epsilon = 10^{-4}$. The GAC of $\mathscr{G}_1$ is computed from the standpoint of all nodes in Fig. 1. In this example, the sequence of two-dimensional subspaces $\{\mathcal{V}_k^i\}$ for $i \in \{1, 2, 3\}$ converges to the desired two-dimensional subspace $\mathcal{V} = \mathrm{span}\{\mathrm{v}_2(\mathbf{L}_1), \mathrm{v}_3(\mathbf{L}_1)\}$ as the iteration index k increases.

Example 2: Let $\mathscr{G}_2 = (\mathscr{V}, \mathscr{E}_2, \mathbf{W}_2)$ denote a strongly connected weighted digraph representing an asymmetric network composed of three nodes. Let also the weight matrix $\mathbf{W}_2$ be given by

$$\mathbf{W}_2 = \begin{bmatrix} 0 & 0 & 0.97 \\ 0 & 0 & 0.75 \\ 0.78 & 0.52 & 0 \end{bmatrix}. \quad (25)$$

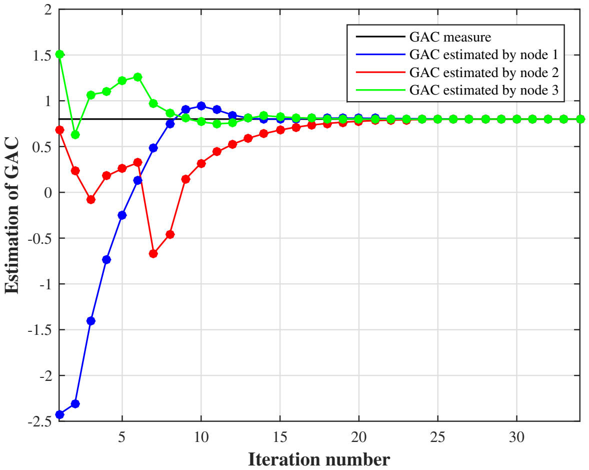

Fig. 1: Evolution of the estimated GAC from the viewpoint of all nodes in Example 1.

The GAC of the network in this example is given by $\tilde{\lambda}(\mathbf{L}_2) = 0.8294$ which corresponds to a real eigenvalue of $\mathbf{L}_2$. By considering $\delta = \frac{1}{3}$, $\epsilon = 10^{-4}$, and applying the proposed distributed algorithm to $\mathscr{G}_2$, the estimated GAC of the network from the standpoint of all nodes is computed as depicted in Fig. 2. The sequence of one-dimensional subspaces, denoted by $\{\mathcal{W}_k^i\}$ for $i \in \{1, 2, 3\}$, converges to the desired one-dimensional subspace $\mathcal{W} = \text{span}\{\mathrm{v}_2(\mathbf{L}_2)\}$ as the iteration index k increases.

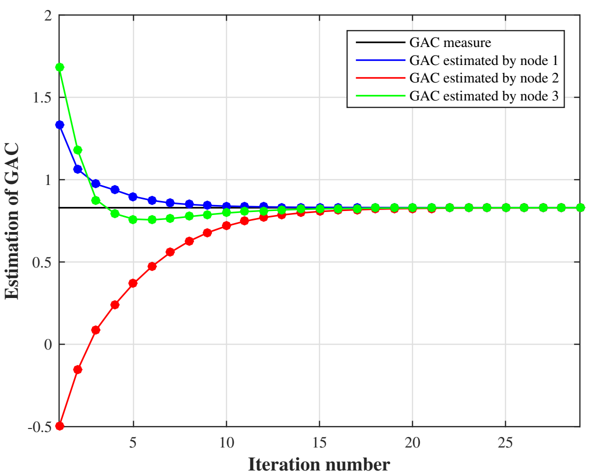

Fig. 2: Evolution of the estimated GAC from the viewpoint of all nodes in Example 2.

VI. Conclusions

The notion of generalized algebraic connectivity for asymmetric weighted networks is formulated as the expected asymptotic convergence rate of distributed parameter estimation algorithms in this work. This connectivity measure is described in terms of the eigenvalues of the Laplacian matrix of the network through an analytical discussion. The subspace consensus procedure is introduced to compute the connectivity measure by generating a sequence of one-dimensional and a sequence of two-dimensional subspaces in a distributed manner by each node based on the information exchange with the neighboring nodes. It then follows that one of the subspace sequences converges to a subspace associated with the desired eigenvalue of the Laplacian matrix which determines the connectivity measure of the network. The efficacy of the proposed algorithm is subsequently verified by simulations.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, March 2002.

[2] P. Santi, "Topology control in wireless ad hoc and sensor networks," *ACM Computing Surveys*, vol. 37, no. 2, pp. 164–194, June 2005.

[3] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1035–1048, March 2010.

[4] P. Di Lorenzo, S. Barbarossa, and A. H. Sayed, "Bio-inspired decentralized radio access based on swarming mechanisms over adaptive networks," *IEEE Transactions on Signal Processing*, vol. 61, no. 12, pp. 3183–3197, June 2013.

[5] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proceedings of International Symposium on Information Processing in Sensor Networks*, April 2005, pp. 63–70.

[6] I. D. Schizas, G. B. Giannakis, S. I. Roumeliotis, and A. Ribeiro, "Consensus in ad hoc WSNs with noisy links-Part II: Distributed estimation and smoothing of random signals," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1650–1666, April 2008.

[7] J.-J. Xiao, A. Ribeiro, Z.-Q. Luo, and G. B. Giannakis, "Distributed compression-estimation using wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 27–41, July 2006.

[8] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 2, pp. 298–305, 1973.

[9] D. Kempe and F. McSherry, "A decentralized algorithm for spectral analysis," in *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, June 2004, pp. 561–568.

[10] P. Di Lorenzo and S. Barbarossa, "Distributed estimation and control of algebraic connectivity over random graphs," *IEEE Transactions on Signal Processing*, vol. 62, no. 21, pp. 5615–5628, November 2014.

[11] C. Li and Z. Qu, "Distributed estimation of algebraic connectivity of directed networks," *Systems & Control Letters*, vol. 62, no. 6, pp. 517–524, June 2013.

[12] M. M. Asadi, M. Khosravi, A. G. Aghdam, and S. Blouin, "Generalized algebraic connectivity for asymmetric networks," in *Proceedings of the American Control Conference*, July 2016, pp. 5531–5536.

[13] ——, "Joint power optimization and connectivity control problem over underwater random sensor networks," in *Proceedings of the American Control Conference*, July 2015, pp. 2709–2714.

[14] M. Stojanovic, "On the relationship between capacity and distance in an underwater acoustic communication channel," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 34–43, October 2007.

[15] R. S. Varga, *Matrix Iterative Analysis*. Springer Series in Computational Mathematics, 2002.

[16] I. Gohberg, P. Lancaster, and L. Rodman, *Invariant Subspaces of Matrices with Applications*. SIAM, 1986.

[17] M. B. Nevelson and R. Z. Hasminskii, *Stochastic Approximation and Recursive Estimation*. Providence: American Mathematical Society, 1973.

[18] M. M. Asadi, M. Khosravi, A. G. Aghdam, and S. Blouin, *Distributed connectivity assessment of asymmetric networks*. Concordia University Technical Report, 2016, available online at www.ece.concordia.ca/~aghdam/TechnicalReports/techrep2016_1.pdf.

[19] Y. Saad, *Numerical Methods for Large Eigenvalue Problems*. Manchester University Press, 1992.