# Distributed Estimation of All the Eigenvalues and Eigenvectors of Matrices Associated With Strongly Connected Digraphs

Azwirman Gusrialdi, *Member, IEEE*, and Zhihua Qu, *Fellow, IEEE*

***Abstract*—This letter considers the problem of estimating all the eigenvalues and eigenvectors of an irreducible matrix, corresponding to a strongly connected digraph, in the absence of knowledge on the global network topology. To this end, we propose a unified distributed strategy performed by each node in the network and relies only on the local information. The key idea is to transform the nonlinear problem of computing both the eigenvalues and eigenvectors of an irreducible matrix into a linear one. Specifically, we first transform distributively the irreducible matrix into a nonsingular irreducible matrix. Each node in the network then estimates in a distributed fashion the inverse of the nonsingular matrix by solving a set of linear equations based on a consensus-type algorithm. The eigenvalues and the corresponding eigenvectors are finally computed by exploiting the relations between the eigenvalues and eigenvectors of both the inverse and the original irreducible matrices. A numerical example is provided to demonstrate the effectiveness of the proposed distributed strategy.**

***Index Terms*—Estimation, distributed algorithm, irreducible matrix, network analysis and control.**

## I. Introduction

THE INTERACTION between individual nodes (subsystems) in a network (such as metabolic, transportation, social, power and robotic networks) can be modeled as a graph which can be further represented (for the purpose of analysis) by the so-called *Laplacian* or *adjacency* matrices. The eigenvalues and eigenvectors associated with those matrices contain important information related to the network's performance and robustness. For example, the information on the eigenvalues of a graph has been used in chemistry and quantum mechanics [1]. Moreover, the Laplacian eigenvalues can also be used to assess the graph robustness through the Kirchhoff index [2] or Laplacian energy [3]. One of the most studied eigenvalue, namely the second smallest eigenvalue of the Laplacian matrix is closely related to the convergence speed of a consensus algorithm [4]. All the eigenvalues of the Laplacian matrix can also be used for designing consensus matrices so that average consensus can be achieved in a finite number of steps [5]. Note that the consensus algorithm has wide applications, ranging from robotics [6], smart grids [7], to transportation systems [8], [9]. The eigenvalues of the adjacency matrix are also related to the natural connectivity which is a measure of structural robustness in complex networks [10]. Furthermore, the largest eigenvalue of the adjacency matrix is a key to the spreading of disease under various types of epidemic models [11]. Similarly, the eigenvectors of the adjacency matrix have applications in community discovery [12].

In practice, the global topology of a network is typically not available or unknown. As a result, the eigenvalues and eigenvectors corresponding to the network cannot be directly computed. This motivates the work on estimating the eigenvalues of the Laplacian matrix or recovering the overall network topology *distributively*, i.e., by using only local information of each node in the network. Various distributed algorithms have been proposed to estimate distributively all the eigenvalues of the Laplacian matrix associated with an *undirected* graph based on the Fast Fourier Transform method [13], observability of the network [14] and by solving a constrained consensus optimization problem [15]. Yang and Tang [16] propose a two-step distributed algorithm to compute the eigenvalues of not only the Laplacian matrix, but also any matrix induced by the graph. However, the result is only limited to undirected graph. The work [17]–[19] focus on distributed estimation of the second smallest eigenvalue of the Laplacian matrix associated with a strongly connected directed graph (*digraph*) together with the corresponding left and right eigenvectors based on the power iteration method. In addition, Charalambous *et al.* [20] propose a distributed algorithm to compute all the eigenvalues of the Laplacian matrix for strongly connected digraphs in finite-time. However, the approach in [20] is only applicable to row stochastic matrix. Furthermore, it is not clear if the approach can also be used to estimate distributively all the left eigenvectors of the Laplacian matrix. To the best of our knowledge, there still exists no work on the distributed estimation of all the eigenvalues together with both the left and

Manuscript received March 6, 2017; revised May 10, 2017; accepted May 31, 2017. Date of publication June 20, 2017; date of current version July 3, 2017. This work was supported in part by the U.S. National Science Foundation under Grant ECCS-1308928 and Grant CCF-0956501, in part by the U.S. Department of Energy under Award DE-EE0006340, Award DE-EE0007327, and Award DE-EE0007998, in part by the U.S. Department of Transportation under Award DTRT13-G-UTC51, in part by L-3 Communication under Contract 11013I2034, in part by Leidos under Contract P010161530, and in part by Texas Instruments' Award. Recommended by Senior Editor M. Arcak.
*(Corresponding author: Azwirman Gusrialdi.)*

The authors are with the Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816 USA (e-mail: azwirman.gusrialdi@ucf.edu; qu@ucf.edu).
Digital Object Identifier 10.1109/LCSYS.2017.2717799

right eigenvectors of any irreducible matrix corresponding to strongly connected digraphs, including both the Laplacian and adjacency matrices as special cases.

The main contribution of this letter is the development of a unified strategy to estimate all the eigenvalues and eigenvectors of any irreducible matrix in a distributed manner. The key idea is to transform the nonlinear problem of computing both the eigenvalues and eigenvectors of a matrix into a linear one. Specifically, we first transform distributively the irreducible matrix into a nonsingular irreducible matrix by utilizing the maximum consensus protocol. Each node in the network then estimates in a distributed fashion the inverse of the nonsingular irreducible matrix by solving a set of linear equations based on a consensus-type algorithm proposed in [21]. The eigenvalues and the corresponding eigenvectors are finally computed by exploiting the relations between both the eigenvalues and eigenvectors of the inverse and the original irreducible matrices. It is worth mentioning that, based on distributed algorithm of solving linear equations such as [21], the proposed method successfully solves the nonlinear problem of distributively calculating eigenvalues/eigenvectors and our contribution lies in the non-trivial idea of recasting the original nonlinear problem into a linear one, namely the problem of solving a set of linear equations, which further facilitates us to adopt the results in [21].

The organization of this letter is as follows: preliminary results on graph theory and the problem formulation are presented in Section II. After providing motivating applications of the problem in Section III, distributed algorithm to estimate all the eigenvalues and eigenvectors of any irreducible matrix is presented in Section IV. Finally, the proposed strategy is demonstrated via a numerical example in Section V.

## II. Problem Statement

In this section, we first provide a brief overview of graph theory and followed by the problem formulation.

### A. Notation and Preliminaries

In this letter, vectors are considered as column vectors. Let $\mathbb{R}$ be the set of real numbers; vector $\mathbf{1}_n \in \mathbb{R}^n$ denotes the vector of all ones. Furthermore, $\mathrm{diag}(a) \in \mathbb{R}^{n\times n}$ represents the diagonal matrix with the vector $a \in \mathbb{R}^n$ on its diagonal. The identity matrix $I_n \in \mathbb{R}^{n\times n}$ is given by $I_n = \mathrm{diag}(\mathbf{1}_n)$. For a given set $\mathcal{N}$, the number of its elements is denoted by $|\mathcal{N}|$. For a matrix $Q = [q_{ij}] \in \mathbb{R}^{n\times n}$, let $[Q]_{i*}$ and $[Q]_{*i}$ represent vectors whose elements are equal to the $i$-th row and column of $Q$ respectively. Without loss of any generality we assume that the eigenvalues of $Q$, denoted by $\lambda_i(Q)$, are ordered as

$$\mathrm{Re}(\lambda_1(Q)) \leq \mathrm{Re}(\lambda_2(Q)) \leq \cdots \leq \mathrm{Re}(\lambda_n(Q)).$$

Moreover, the left and right eigenvectors corresponding to $\lambda_i(Q)$ are denoted by $v_i(Q), w_i(Q)$ respectively. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a directed graph (digraph) with a set of nodes $\mathcal{V} = \{1, 2, \ldots, n\}$ and a set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. An edge $(i,j) \in \mathcal{E}$ denotes that node $i$ can obtain information from node $j$. The set of in-neighbors of node $i$ is denoted by $\mathcal{N}_i^{\mathrm{in}} = \{j|(i,j) \in \mathcal{E}\}$. The directed graph $\mathcal{G}$ is strongly connected if every node can be reached from any other nodes by following a set of directed edges. Matrix $Q \in \mathbb{R}^{n\times n}$ is irreducible if and only if its associated graph $\mathcal{G}$ is strongly connected. The weighted adjacency matrix $A = [a_{ij}] \in \mathbb{R}^{n\times n}$ associated with digraph $\mathcal{G}$ is defined as

$$A = [a_{ij}], \begin{cases} a_{ij} > 0 & \text{if } i \neq j \text{ and } (i,j) \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases}$$

In addition, the weighted Laplacian matrix $L \in \mathbb{R}^{n\times n}$ is defined as

$$L \triangleq D - A$$

where degree matrix $D = \mathrm{diag}(d)$ with $d = [d_1, \ldots, d_n]^T$ and $d_i = \sum_{j\in\mathcal{N}_i^{\mathrm{in}}} a_{ij}$. If the graph $\mathcal{G}$ is strongly connected, then we have $\lambda_1(L) = 0$ and $\mathrm{Re}(\lambda_i(L)) \geq 0$ for $i = \{2, \ldots, n\}$. Note that both the Laplacian and adjacency matrices associated with a strongly connected digraph are the special cases of the irreducible matrix.

### B. Problem Statement

Consider a network consisting of $n$ number of nodes whose (communication) topology is represented by a strongly connected digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Associated with the graph $\mathcal{G}$, consider an irreducible matrix $Q = [q_{ij}] \in \mathbb{R}^{n\times n}$ (can be singular) which has a similar sparsity structure as the graph $\mathcal{G}$, i.e., $q_{ij} = 0$ if $(i,j) \notin \mathcal{E}$ and $q_{ij} \neq 0$ otherwise. Hence, node $i$ knows $q_{ii}$ and $q_{ij}$ with $j \in \mathcal{N}_i^{\mathrm{in}}$ which can be obtained via the communication network. The problem considered in this letter can then be formally stated as follows.

*Problem 1:* Given an irreducible matrix $Q \in \mathbb{R}^{n\times n}$, compute $\lambda_i(Q)$ for $i = \{1, \ldots, n\}$ together with the corresponding eigenvectors distributively, i.e., the $i$-th node in the network collaboratively determines all eigen information of the entire network by applying its local information of the $i$-th row of matrix $Q$. For simplicity and practicality, the information exchange is done through the same local communication network, i.e., graph $\mathcal{G}$ has similar structure to that of matrix $Q$, and all the nodes of the network are numbered apriori. Moreover, it is assumed that each node knows the size of the network (or its upper-bound).

Note that the above settings are standard in the related literature on distributed algorithms, see [16], [20], [22]–[24]. In practice, the overall network topology $\mathcal{G}$ and its associated matrix $Q$ are typically not available (unknown). The unavailability of the global network topology may be due to (i) the geographical constraint [25]; or (ii) the fact that the topology might change over the time.

## III. Motivating Applications

In this section, the importance of estimating distributively the eigenvalues and eigenvectors of an irreducible matrix is illustrated through several applications.

### A. Distributed Stability Test for Interconnected System

Consider an LTI system consisting of $n$ scalar subsystems. The dynamics of each subsystem is given by

$$x_i(k+1) = q_{ii}x_i(k) + \sum_{j\in\mathcal{N}_i^{\mathrm{in}}} q_{ij}x_j(k).$$

where $x_i \in \mathbb{R}$ denotes the state of subsystem $i$. The settings can also be extended to non-scalar case where $x_i \in \mathbb{R}^n$. The overall interconnected system can be written in a compact form as

$$x(k+1) = Qx(k), \tag{1}$$

where $x(k) = [x_1(k), \ldots, x_n(k)]^T$ and the structure of matrix $Q$ represents the interconnection between the individual subsystems. We assume that matrix $Q$ is irreducible. Recently, there has been a growing interest to develop algorithms to test the stability of interconnected system (1) in a distributed manner. For example, Deroo *et al.* [26] propose two strategies based on distributed optimization techniques to distributively test the stability of the continuous-time version of (1). However, the proposed approach only provides sufficient conditions for the stability and in some cases can be conservative. Moreover, there is also a restriction on the communication topology in order to solve the proposed distributed optimization. Therefore, distributed estimation of the spectrum of matrix $Q$ will allow us to develop a distributed stability test which provides necessary and sufficient condition for determining the stability of (1).

### B. Natural Connectivity and Laplacian Energy

Let $L$ and $A$ denote the Laplacian and adjacency matrices of an undirected graph consisting of $n$ nodes respectively. The natural connectivity of the graph is defined as [10]

$$nc = \ln\left[\frac{1}{n}\sum_{j=1}^{n} e^{\lambda_j(A)}\right].$$

In addition, the Laplacian energy of the graph is given by

$$E_L = \sum_{i=1}^{n} \lambda_i^2(L).$$

Both the natural connectivity and Laplacian energy are measures of structural robustness of complex networks and have been used, for example, as metrics to improve the robustness of air transportation network [3]. Note that the above definitions can also be extended to the case of digraphs. Hence, we can then compute both measures distributively by estimating the eigenvalues in a distributed manner.

### C. Graph Clustering and Community Structure Identification

Identifying community structure is important to reveal the structure-functionality relationship in complex networks. It is shown in [12] that communicability between a pair of vertices in a complex network can be rewritten in term of the eigenspaces corresponding to the adjacency matrix of the networks. In other words, the eigenspaces of a network are directly connected to the communities in complex networks. Furthermore, it is demonstrated in [27] that the eigenvalues and the corresponding eigenvectors of the Laplacian matrix can be used to perform hierarchical spectral clustering for a graph with application to power grids. The proposed method in this letter will then allow us to identify the community structure and cluster a network in a distributed fashion.

### D. Cooperative Control of Networked Systems

Consider a network of $n$ heterogeneous systems whose individual dynamics is given by

$$x_i(k+1) = f_i(x_i(k)) + g_i(x_i(k))u_i(k),\ y_i = h_i(x_i(k)) \quad (2)$$

where $x_i \in \mathbb{R}^{n_i}$ is the state, $u_i$ is the control input to be designed, and $y_i \in \mathbb{R}^m$ is the output. It is assumed that the dynamics (2) is passivity short with impact coefficient $\kappa_i \in [0, \bar{\kappa}]$, see [28] for the details. The sensing/communication matrix which represents the network topology is given by matrix $S$ defined as $S = I_n + A$ where $A$ is the adjacency matrix. Moreover, we assume that the network topology is given by a strongly connected digraph. The goal is to design $u_i$ so that consensus is reached, i.e., $\lim_{t\to+\infty} y_i(t) = \bar{y}$ for all $i = \{1, \ldots, n\}$. The cooperative control $u_i$ is given by

$$u_i(k) = k_{y_i}\sum_{j=1}^{n}(y_j - y_i)s_{ij}.$$

It is shown in [29] that consensus is ensured if $k_{y_i} \le \bar{k}_y$ with

$$\bar{k}_y = \frac{\lambda_2(\Gamma L + L^T\Gamma)}{\bar{\kappa}\lambda_{max}(L^T\Gamma L)}$$

where $L$ is the Laplacian matrix and $\Gamma = \text{diag}(v_1(L))$. Hence, the gain $k_{y_i}$ can be designed distributively if we can estimate $L$ in a distributed manner.

### E. Epidemic Propagation in a Network

Consider a network of $n$ nodes whose interconnection is given by a strongly connected digraph. Each node has two possible states at each time, namely healthy and susceptible. Initially, it is assumed that a certain percentage of nodes in the network are infected. The infected node tries to infect its neighbors with rate $\beta$ and the infected node may be cured with rate $\delta$. Then, there is an epidemic threshold $\tau_c$ which separates two different phases, namely if the effective infection rate $\tau = \frac{\beta}{\delta}$ is above the threshold, the epidemic will spread through the network and become persistent. On the other hand, if the rate $\tau < \tau_c$, then the infection dies out. It is known [30] that the threshold $\tau_c$ is given by $\tau_c = \frac{1}{\lambda_n(A)}$ where $A$ is the adjacency matrix of the network. Hence, if $\lambda_n(A)$ can be estimated distributively, we can then analyze the epidemic propagation in the network in the absence of information on the global network topology [31].

## IV. Main Result

In this section, we propose a distributed algorithm to solve Problem 1. First, note that given an irreducible matrix $Q$, computing both the eigenvalues and eigenvectors of $Q$ is a nonlinear problem as can be observed from

$$Qw_i = \lambda_i w_i,\ Q^T v_i = \lambda_i v_i \quad (3)$$

which is challenging to solve in a distributed manner. Hence, the key idea of our strategy is to transform the nonlinear problem (3) into a linear one. To this end, let us define a matrix $\overline{Q} \in \mathbb{R}^{n\times n}$ given by

$$\overline{Q} = Q + cI_n, \quad (4)$$

where the constant $c \in \mathbb{R}$ is chosen such that the matrix $\overline{Q}$ is nonsingular. Since $\overline{Q}$ is nonsingular, we then have the following linear problem of finding $\overline{Q}^{-1}$:

$$\overline{Q}\,\overline{Q}^{-1} = I_n. \quad (5)$$

The following lemma provides relations between the eigenvalues and eigenvectors of the matrices $\overline{Q}^{-1}$ and $Q$.

*Lemma 1:* The eigenvalues and eigenvectors of the matrices $Q$ and $\overline{Q}^{-1}$ in (4), (5) for $i = \{1, \ldots, n\}$ satisfy

$$\lambda_i(Q) = \frac{1}{\lambda_i(\overline{Q}^{-1})} - c,$$
$$v_i(Q) = v_i(\overline{Q}^{-1}),\ w_i(Q) = w_i(\overline{Q}^{-1}). \tag{6}$$

*Proof:* First, from (4) it is known that $\lambda_i(\overline{Q}) = \lambda_i(Q) + c$, $v_i(Q) = v_i(\overline{Q})$ and $w_i(Q) = w_i(\overline{Q})$ for $i = \{1, \ldots, n\}$. Moreover, from (5), we have $\lambda_i(\overline{Q}) = \frac{1}{\lambda_i(\overline{Q}^{-1})}$ and it is also known [32] that $v_i(\overline{Q}) = v_i(\overline{Q}^{-1})$ and $w_i(\overline{Q}) = w_i(\overline{Q}^{-1})$. Hence, we have (6) which completes the proof. ■

Therefore, if each node can compute matrix $\overline{Q}^{-1}$ distributively from (5), then the eigenvalues together with the corresponding eigenvectors of $Q$ can be obtained by simply computing the eigenvalues and the associated eigenvectors of the matrix $\overline{Q}^{-1}$ as shown in Lemma 1. Based on the above discussions, the high-level distributed algorithm for solving Problem 1 can then be summarized as follows.

1) Transform the irreducible matrix $Q$ into a nonsingular matrix $\overline{Q}$ by computing distributively the constant $c$ in (4).
2) Compute $\overline{Q}^{-1}$ by solving (5) distributively.
3) Given matrix $\overline{Q}^{-1}$, compute $\lambda_i(Q)$ and the corresponding eigenvectors using Lemma 1.

The details of each step are described below.

## A. Distributed Computation of Nonsingular Matrix $\overline{Q}$

The first step is to choose the constant $c \in \mathbb{R}$ in (4) distributively so that the matrix $\overline{Q}$ is nonsingular. Based on the Gershgorin theorem, it is clear that the matrix $\overline{Q}$ is nonsingular if the following condition is satisfied:

$$|\overline{q}_{ii}| > \sum_{j \neq i} |\overline{q}_{ij}|,\ i = \{1, \ldots, n\}. \tag{7}$$

The constant $c$ can then be distributively computed to satisfy (7) as follows.

1) Each node first independently computes
$$c_i(0) = \epsilon_i + \sum_{j \in \{i\} \cup \mathcal{N}_i^{\text{in}}} |q_{ij}| \tag{8}$$
for any $\epsilon_i > 0$.
2) Each node updates $c_i(k)$ according to the following maximum consensus protocol
$$c_i(k+1) = \max_{j \in \mathcal{N}_i^{\text{in}} \cup \{i\}} c_j(k).$$
It is shown in [33] that the consensus reached in *finite* time (no more than $n$ steps) is $c = \max_i c_i(0)$.

We then have the following result.

*Lemma 2:* Let matrix $Q \in \mathbb{R}^{n \times n}$ be irreducible and the constant $c = \max_i c_i(0)$ with $c_i(0)$ is defined in (8). Then, the matrix $\overline{Q}$ in (4) is nonsingular.

*Proof:* For all $i = \{1, \ldots, n\}$ we have $|q_{ii} + c_i| > \sum_{j \neq i} |q_{ij}|$. From the definition of $c$, we then have $|q_{ii} + c| > \sum_{j \neq i} |q_{ij}|$. Hence, it can be concluded that matrix $\overline{Q}$ is nonsingular. ■

*Remark 1:* For the following cases, the constant $c$ in (4) can be chosen without the need of performing the maximum consensus protocol.

- If matrix $Q$ is given by the Laplacian matrix and since $\text{Re}(\lambda_i(L)) \geq 0$, we can then choose $c = \epsilon > 0$.
- If matrix $Q$ is given by the unweighted adjacency matrix, we can then simply choose $c = n$.
- If matrix $Q$ is given by the weighted adjacency matrix with each row summing to one, we can then set $c = 1$.

## B. Distributed Computation of Matrix $\overline{Q}^{-1}$

After computing matrix $\overline{Q}$, in the following we compute its inverse $\overline{Q}^{-1}$ by solving distributively a set of linear equations given in (5). To this end, we adopt and extend the consensus-based approach, such as the one originally proposed in [21], to compute $\overline{Q}^{-1}$ from (5) by solving distributively the linear equations. Let us assign a state variable $Z_i(k) \in \mathbb{R}^{n \times n}$ to each node. Each node then estimates $\overline{Q}^{-1}$ according to the following update rule:

$$Z_i(k+1) = Z_i(k) - \frac{1}{|\mathcal{N}_i^{\text{in}}|} P_i \left( |\mathcal{N}_i^{\text{in}}| Z_i(k) - \sum_{j \in \mathcal{N}_i^{\text{in}}} Z_j(k) \right) \tag{9}$$

where $P_i = P_i^T \in \mathbb{R}^{n \times n}$ is the orthogonal projection on the kernel of vector $[\overline{Q}]_{i*}$, namely

$$P_i = I_n - \frac{1}{[\overline{Q}]_{i*}^T [\overline{Q}]_{i*}} [\overline{Q}]_{i*} [\overline{Q}]_{i*}^T. \tag{10}$$

The following is the main result of this letter.

*Theorem 1:* Set $Z_i(0)$ to satisfy $[\overline{Q}]_{i*}^T Z_i(0) = [I_n]_{i*}^T$. Then, under distributed algorithm (9) the state $Z_i(k)$ converges exponentially to $\overline{Q}^{-1}$. Accordingly, we have

$$\lambda_i(Q) = \frac{1}{\lambda_i(Z_i^e)} - c,$$
$$v_i(Q) = v_i(Z_i^e),\ w_i(Q) = w_i(Z_i^e) \tag{11}$$

where $Z_i^e$ denotes the steady-state of $Z_i(k)$.

*Proof:* It is known [21] that under the update rule (9) with $Z_i(0)$ satisfying $[\overline{Q}]_{i*}^T Z_i(0) = [I_n]_{i*}^T$, at the steady state we have $\overline{Q}[Z_i^e]_{*i} = [I_n]_{*i}$, for $i = \{1, \ldots, n\}$. Putting together all the vectors of $[Z_i^e]_{*i}$ yields $\overline{Q} Z_i^e = I_n$. Since the inverse of the matrix $\overline{Q}$ is unique, it can then be concluded that $Z_i^e = \overline{Q}^{-1}$. Finally, from Lemma 1 we obtain (11) which completes the proof. ■

Recall that algorithm (9) solves distributively $\overline{Q}Z = I_n$ which consists of $n$ linear equations $\overline{Q}[Z]_{*j} = [I_n]_{*j}$. From (5), each linear equation has a unique solution $[Z^*]_{*j} = [\overline{Q}^{-1}]_{*j}$. Now, consider the $j$-th linear equation and define the error vector $e_j(k) = [[Z_1(k)]_{*j}^T - [Z^*]_{*j}^T, \ldots, [Z_n(k)]_{*j}^T - [Z^*]_{*j}^T]^T$, algorithm (9) can then be compactly written as [21]

$$e_j(k+1) = (P(F \otimes I_n)P) e_j(k) = H e_j(k), \tag{12}$$

where block diagonal matrix $P = \text{diag}(P_1, \ldots, P_n)$, and $F = D_g^{-1} A_g^T$ with $D_g, A_g$ are unweighted degree and adjacency matrices corresponding to graph $\mathcal{G}$ respectively. It is shown in [21] that the algorithm is convergent and hence $\rho(H) = \max |\lambda_i(H)| < 1$. Thus, we can conclude from the classical control theory the following lemma, in which the settling time is defined as the time needed for the error to be of no larger than two percent.

*Lemma 3:* The settling time of (12) is upper bounded as

$$k_{st} \leq \overline{k}_{st} = 4 \ln^{-1}(\rho^{-1}(H)). \tag{13}$$

*Proof:* The convergence rate of (12) is equal to $\rho(H)$. The settling time is then given by the solution to $\rho^{k_{st}} = (e^{-\ln(\rho^{-1})})^{k_{st}} \leq 0.02$ which results in (13). ■

In the following we provide some discussions regarding the proposed method.

1) Since the update rule (9) estimates matrix $\overline{Q}^{-1} \in \mathbb{R}^{n\times n}$, each node is then required to store $n^2$ values. Moreover, as can be observed from (9) each node also needs to send $n^2$ values to its neighbors. Even though the memory requirement is larger than the other approaches (e.g., [16] and [20]), the proposed strategy is simple and allows us to estimate all the eigenvalues and eigenvectors of any irreducible matrix. Note that the storage and communication costs of (9) are similar to that of the method in [34] developed to re-construct the unweighted undirected network. One may argue that if the nodes use $O(n^2)$ memory, then why the nodes do not just flood the rows of matrix $Q$ across the network so that each node knows $Q$. It is worth noting that the flooding strategy is not locally adaptable when the topology changes since all the nodes in the network need to be informed (which requires a global coordinator) about the change and update their information accordingly in order to estimate the eigen information corresponding to the new topology.
2) In comparison to (distributed) power iteration method (see [17]) which is an alternative approach to solve, e.g., the problem described in Section III-E, the proposed algorithm is able to estimate the eigenvectors even if the corresponding eigenvalue is not simple. Note that the power iteration method may not converge when the associated eigenvalue is not strictly greater in magnitude than the rest of the eigenvalues. Moreover, update rule (9) converges exponentially fast and its convergence is guaranteed under asynchronous setting and time-delay [35].
3) Using the proposed method, each node still needs to compute (independently) the eigenvalues of a full matrix. However, since each node has a copy of the matrix $Q$, the proposed method allows us to compute the eigenvalues of a matrix involving the multiplication of the Laplacian or adjacency matrix, e.g., the one described in Section III-D.

## C. Discussion on Switching Topology Case

Consider the problem where graph $\mathcal{G}$ together with its associated matrix $Q$ switches between several strongly connected digraphs and the nodes need to distributively estimate the eigen information of the irreducible matrix associated with each topology. We have the following result.

*Proposition 1:* Consider a sequence of time varying strongly connected digraphs $\mathcal{G}(k_{s_i})$, with $\mathcal{G}(k) = \mathcal{G}(k_{s_i})$ for $k \in [k_{s_i}, k_{s_{i+1}})$, for $i \in \aleph$. Let $Q(k_{s_i})$ be the unweighted Laplacian matrix corresponding to $\mathcal{G}(k_{s_i})$ and let $\mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_i}))$ denote the set of in-neighbors of node $j$ of the graph $\mathcal{G}(k_{s_i})$. Then, at time $k_{s_{i+1}}$ and for all $i$, only nodes $j$ with $\mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_{i+1}})) \neq \mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_i}))$ need to recompute $P_j$ in (10) using $[\overline{Q}(k_{s_{i+1}})]_{j*}$ and reset their state $Z_j(k_{s_{i+1}})$ to satisfy $[\overline{Q}(k_{s_{i+1}})]_{j*}^T Z_j(k_{s_{i+1}}) = [I_n]_{j*}^T$ when executing (9). Moreover, as long as $(k_{s_{i+1}} - k_{s_i}) > \overline{k}_{st}$ holds for all $i$, each node can use (9) to estimate all the eigenvalues and eigenvectors of matrix $Q(k_{s_i})$.

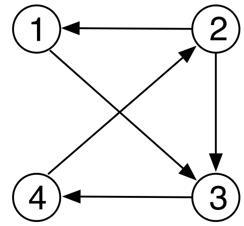

Fig. 1. Strongly connected digraph for the matrix in (14).

*Proof:* First, observe that matrix $\overline{Q}(k_{s_i})$ are nonsingular for all $i$ if $c$ is chosen as in Remark 1. Next, for matrix $\overline{Q}(k_{s_i})$ and under (9) we have $[\overline{Q}(k_{s_i})]_{j*}^T Z_j(k) = [I_n]_{j*}^T$ for $k \in [k_{s_i}, k_{s_{i+1}})$ if $Z_j(k_{s_i})$ is chosen to satisfy $[\overline{Q}(k_{s_i})]_{j*}^T Z_j(k_{s_i}) = [I_n]_{j*}^T$. Moreover, if $\mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_{i+1}})) = \mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_i}))$, that is $[\overline{Q}(k_{s_{i+1}})]_{j*}^T = [\overline{Q}(k_{s_i})]_{j*}^T$, and $[\overline{Q}(k_{s_i})]_{j*}^T Z_j(k_{s_i}) = [I_n]_{j*}^T$, under update law (9) we also have $[\overline{Q}(k_{s_{i+1}})]_{j*}^T Z_j(k) = [I_n]_{j*}^T$ for $k \in [k_{s_{i+1}}, k_{s_{i+2}})$. Hence, from the above observations it can be concluded that at time $k_{s_{i+1}}$ only nodes $j$ with $\mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_{i+1}})) \neq \mathcal{N}_j^{\text{in}}(\mathcal{G}(k_{s_i}))$ need to recompute $P_j$ in (10) using $[\overline{Q}(k_{s_{i+1}})]_{j*}$ and reset $Z_j(k_{s_{i+1}})$ such that $[\overline{Q}(k_{s_{i+1}})]_{j*}^T Z_j(k_{s_{i+1}}) = [I_n]_{j*}^T$ in executing (9). Finally, if $(k_{s_{i+1}} - k_{s_i}) > \overline{k}_{st}$ for all $i$ and $\overline{k}_{st}$ is given by (13), each node can use (9) to estimate all the eigen information of matrix $Q(k_{s_i})$. ■

The above result guarantees that, for switching topologies, algorithm (9) remains to be effective in estimating the eigen information and their changes as long as the dwelling time of any topology is longer than the settling time estimated in (13). As is, estimate (13) provides a theoretical guarantee but, since it is topology dependent, further research is needed to determine a topology-independent estimate and its distributed computation so the result can be used in a distributed implementation.

## V. A Numerical Example

Let $Q$ be given by the unweighted Laplacian matrix

$$Q = L = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & -1 & 2 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} \tag{14}$$

whose associated strongly connected digraph is depicted in Fig. 1. The nonsingular matrix $\overline{Q}$ can then be computed from (4) by setting the constant $c$ to be any positive number, for example $c = 1$. Next, each node estimates $\overline{Q}^{-1}$ distributively according to the update rule (9) whose initial conditions $Z_i(0)$ are chosen as

$$Z_1(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, Z_2(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix},$$

$$Z_3(0) = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, Z_4(0) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

The state $Z_i(k)$ then exponentially converges to

$$Z_i^e = \overline{Q}^{-1} = \begin{bmatrix} 0.5238 & 0.2857 & 0.0476 & 0.1429 \\ 0.0476 & 0.5714 & 0.0952 & 0.2857 \\ 0.1905 & 0.2857 & 0.3810 & 0.1429 \\ 0.0952 & 0.1429 & 0.1905 & 0.5714 \end{bmatrix}$$

and the eigenvalues $\lambda_i(Q)$ together with the corresponding eigenvectors can finally be computed from (11).

## VI. Conclusion

We propose a unified strategy to estimate all the eigenvalues and eigenvectors of any irreducible matrix in a distributed manner. The key idea is to transform the nonlinear problem of computing both the eigenvalues and eigenvectors of a matrix into a linear one. Specifically, after transforming the irreducible matrix into a nonsingular one, each node estimates distributively the inverse of the nonsingular matrix by solving a set of linear equations. The eigenvalues and the corresponding eigenvectors are finally computed by exploiting the relations between the eigenvalues and eigenvectors of both the nonsingular and the original irreducible matrices. Future research can be done to consider noisy communication channel and to reduce the memory requirement together with the communication cost of each node. The communication cost can be potentially reduced by either developing a finite-time distributed algorithm to solve the linear equations or by adopting the strategies presented in [36] and [37] where at each time each node sends a block of its estimate vector to its neighbors. It is also worth investigating if the storage and communication costs can be reduced when one is only interested in estimating a subset of the eigenvalues or eigenvectors.

## References

[1] F. R. K. Chung, *Spectral Graph Theory*, vol. 92. Providence, RI, USA: Amer. Math. Soc., 1997.

[2] D. J. Klein and M. Randić, "Resistance distance," *J. Math. Chem.*, vol. 12, no. 1, pp. 81–95, 1993.

[3] C. Yang, J. Mao, and P. Wei, "Air traffic network optimization via Laplacian energy maximization," *Aerosp. Sci. Technol.*, vol. 49, pp. 26–33, Feb. 2016.

[4] Z. Qu, *Cooperative Control of Dynamical Systems*. New York, NY, USA: Springer, 2009.

[5] A. Y. Kibangou, "Graph Laplacian based matrix design for finite-time distributed average consensus," in *Proc. Amer. Control Conf.*, Montreal, QC, Canada, Jun. 2012, pp. 1901–1906.

[6] W. Ren and N. Sorensen, "Distributed coordination architecture for multi-robot formation control," *Robot. Auton. Syst.*, vol. 56, no. 4, pp. 324–333, 2008.

[7] H. Xin, Z. Qu, J. Seuss, and A. Maknouninejad, "A self-organizing strategy for power flow control of photovoltaic generators in a distribution network," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1462–1473, Aug. 2011.

[8] A. Gusrialdi, Z. Qu, and M. A. Simaan, "Distributed scheduling and cooperative control for charging of electric vehicles at highway service stations," *IEEE Trans. Intell. Transp. Syst.*, to be published, doi: 10.1109/TITS.2017.2661958.

[9] L. Y. Wang, A. Syed, G. G. Yin, A. Pandya, and H. Zhang, "Control of vehicle platoons for highway safety and efficient utility: Consensus with communications and vehicle dynamics," *J. Syst. Sci. Complexity*, vol. 27, no. 4, pp. 605–631, 2014.

[10] J. Wu, M. Barahona, Y.-J. Tan, and H.-Z. Deng, "Spectral measure of structural robustness in complex networks," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 6, pp. 1244–1252, Nov. 2011.

[11] B. A. Prakash, D. Chakrabarti, N. Valler, M. Faloutsos, and C. Faloutsos, "Threshold conditions for arbitrary cascade models on arbitrary networks," *Knowl. Inf. Syst.*, vol. 33, no. 3, pp. 549–575, 2012.

[12] X. Ma, L. Gao, and X. Yong, "Eigenspaces of networks reveal the overlapping and hierarchical community structure more precisely," *J. Stat. Mech. Theory Exp.*, vol. 2010, no. 8, 2010, Art. no. P08012.

[13] M. Franceschelli, A. Gasparri, A. Giua, and C. Seatzu, "Decentralized estimation of Laplacian eigenvalues in multi-agent systems," *Automatica*, vol. 49, no. 4, pp. 1031–1036, 2013.

[14] A. Y. Kibangou and C. Commault, "Decentralized Laplacian eigenvalues estimation and collaborative network topology identification," in *Proc. 3rd IFAC Workshop Distrib. Estimation Control Netw. Syst.*, 2012, pp. 7–12.

[15] T.-M.-D. Tran and A. Y. Kibangou, "Distributed estimation of Laplacian eigenvalues via constrained consensus optimization problems," *Syst. Control Lett.*, vol. 80, pp. 56–62, Jun. 2015.

[16] M. Yang and C. Y. Tang, "Distributed estimation of graph spectrum," in *Proc. Amer. Control Conf.*, Chicago, IL, USA, Jul. 2015, pp. 2703–2708.

[17] C. Li and Z. Qu, "Distributed estimation of algebraic connectivity of directed networks," *Syst. Control Lett.*, vol. 62, no. 6, pp. 517–524, 2013.

[18] H. A. Poonawala and M. W. Spong, "Decentralized estimation of the algebraic connectivity for strongly connected networks," in *Proc. Amer. Control Conf.*, Chicago, IL, USA, Jul. 2015, pp. 4068–4073.

[19] C. Li and Z. Qu, "Distributed finite-time consensus of nonlinear systems under switching topologies," *Automatica*, vol. 50, no. 6, pp. 1626–1631, 2014.

[20] T. Charalambous, M. G. Rabbat, M. Johansson, and C. N. Hadjicostis, "Distributed finite-time computation of digraph parameters: Left-eigenvector, out-degree and spectrum," *IEEE Trans. Control Netw. Syst.*, vol. 3, no. 2, pp. 137–148, Jun. 2016.

[21] S. Mou, J. Liu, and A. S. Morse, "A distributed algorithm for solving a linear algebraic equation," *IEEE Trans Autom. Control*, vol. 60, no. 11, pp. 2863–2878, Nov. 2015.

[22] R. Aragues *et al.*, "Distributed algebraic connectivity estimation for undirected graphs with upper and lower bounds," *Automatica*, vol. 50, no. 12, pp. 3253–3259, 2014.

[23] A. Priolo, A. Gasparri, E. Montijano, and C. Sagues, "A distributed algorithm for average consensus on strongly connected weighted digraphs," *Automatica*, vol. 50, no. 3, pp. 946–951, 2014.

[24] G. Oliva, R. Setola, L. Glielmo, and C. N. Hadjicostis, "Distributed cycle detection and removal," *IEEE Trans. Control Netw. Syst.*, to be published, doi: 10.1109/TCNS.2016.2593264.

[25] "Enterprise GIS and the smart electric grid," White Paper, ESRI, Redlands, CA, USA, 2012. [Online]. Available: http://content.energycentral.com/download/products/enterprise-gis-smartelectric-grid.pdf

[26] F. Deroo, M. Meinel, M. Ulbrich, and S. Hirche, "Distributed stability tests for large-scale systems with limited model information," *IEEE Trans. Control Netw. Syst.*, vol. 2, no. 3, pp. 298–309, Sep. 2015.

[27] R. J. Sánchez-García *et al.*, "Hierarchical spectral clustering of power grids," *IEEE Trans. Power Syst.*, vol. 29, no. 5, pp. 2229–2237, Sep. 2014.

[28] Z. Qu and M. A. Simaan, "Modularized design for cooperative control and plug-and-play operation of networked heterogeneous systems," *Automatica*, vol. 50, no. 9, pp. 2405–2414, 2014.

[29] Y. Joo, R. Harvey, and Z. Qu, "Cooperative control of heterogeneous multi-agent systems in a sampled-data setting," in *Proc. IEEE Conf. Decis. Control*, Las Vegas, NV, USA, 2016, pp. 2683–2688.

[30] C. Li, H. Wang, and P. Van Mieghem, "Epidemic threshold in directed networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 88, Dec. 2013, Art. no. 062802.

[31] A. Gusrialdi and Z. Qu, "Growing connected networks under privacy constraint: Achieving trade-off between performance and security," in *Proc. IEEE Conf. Decis. Control*, Osaka, Japan, 2015, pp. 312–317.

[32] R. Horn and C. R. Johnson, *Matrix Analysis*. New York, NY, USA: Cambridge Univ. Press, 2009.

[33] B. M. Nejad, S. A. Attia, and J. Raisch, "Max-consensus in a max-plus algebraic setting: The case of fixed communication topologies," in *Proc. Int. Symp. Inf. Commun. Autom. Technol.*, 2009, pp. 1–7.

[34] F. Morbidi and A. Y. Kibangou, "A distributed solution to the network reconstruction problem," *Syst. Control Lett.*, vol. 70, pp. 85–91, Aug. 2014.

[35] J. Liu, S. Mou, and A. S. Morse, "An asynchronous distributed algorithm for solving a linear algebraic equation," in *Proc. IEEE Conf. Decis. Control*, Florence, Italy, 2013, pp. 5409–5414.

[36] S. Mou, Z. Lin, L. Wang, D. Fullmer, and A. S. Morse, "A distributed algorithm for efficiently solving linear equations and its applications," *Syst. Control Lett.*, vol. 91, pp. 21–27, May 2016.

[37] X. Gao, J. Liu, and T. Başar, "Stochastic communication-efficient distributed algorithms for solving linear algebraic equations," in *Proc. IEEE Conf. Control Appl.*, Buenos Aires, Argentina, 2016, pp. 380–385.