Automatica 50 (2014) 3253–3259

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Distributed algebraic connectivity estimation for undirected graphs with upper and lower bounds☆

Rosario Aragues [a,1], Guodong Shi [b], Dimos V. Dimarogonas [c], Carlos Sagüés [d], Karl Henrik Johansson [c], Youcef Mezouar [a]

[a] *Institut Pascal, UMR 6602 CNRS - UBP - IFMA, F-63000 Clermont-Ferrand, France*
[b] *College of Engineering and Computer Science, The Australian National University, ACT 0200, Canberra, Australia*
[c] *Access Linnaeus Centre, School of Electrical Engineering, Royal Institute of Technology (KTH), 10044 Stockholm, Sweden*
[d] *Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, 50018 Zaragoza, Spain*

ARTICLE INFO

Article history:
Received 8 March 2013
Received in revised form
18 March 2014
Accepted 30 July 2014
Available online 13 November 2014

Keywords:
Multi-agent systems
Control of networks
Sensor networks
Algebraic connectivity

ABSTRACT

The algebraic connectivity of the graph Laplacian plays an essential role in various multi-agent control systems. In many cases a lower bound of this algebraic connectivity is necessary in order to achieve a certain performance. Lately, several methods based on distributed Power Iteration have been proposed for computing the algebraic connectivity of a symmetric Laplacian matrix. However, these methods cannot give any lower bound of the algebraic connectivity and their convergence rates are often unclear. In this paper, we present a distributed algorithm for estimating the algebraic connectivity for undirected graphs with symmetric Laplacian matrices. Our method relies on the distributed computation of the powers of the adjacency matrix and its main interest is that, at each iteration, agents obtain both upper and lower bounds for the true algebraic connectivity. Both bounds successively approach the true algebraic connectivity with the convergence speed no slower than $O(1/k)$.

1. Introduction

The diverse applications of multi-agent systems, e.g., sensor fusion, flocking, formation, or rendezvous (Olfati-Saber, Fax, & Murray, 2007), have led to tremendous research interest in the past decade. A typical multi-agent system is a network of cooperative agents targeting a collective aim using the distributed control design and local information exchange. An underlying communication graph is thus naturally associated with any given multi-agent network. The second smallest eigenvalue of the Laplacian matrix of this graph, known as the algebraic connectivity, plays an important role in various multi-agent applications and in many cases serves as a fundamental performance measure (Bullo, Cortés, & Martínez, 2009).

The magnitude of the algebraic connectivity determines the connectivity of the communication graph. We first remark some efforts in the literature on maintaining or computing the connectivity of the graph. Control laws for rendezvous and formation control that keep the initial topology have been proposed in Dimarogonas and Johansson (2010) and Ji and Egerstedt (2007). Then in Zavlanos and Pappas (2005), it was shown how to compute the k-hop connectivity matrix of the graph in a centralized fashion. Several distributed methods were then proposed on computing spanning subgraphs (Zavlanos & Pappas, 2008), specifying Laplacian eigenvectors (Qu, Li, & Lewis, 2011), estimating moments of the Laplacian eigenvalue spectrum (Preciado, Zavlanos, Jadbabaie, & Pappas, 2010), or maximizing the algebraic connectivity through motion control (Simonetto, Keviczky, & Babuska, 2011).

How to estimate the value of this algebraic connectivity becomes an intriguing problem for the study of multi-agent networks. In Franceschelli, Gasparri, Giua, and Seatzu (2009), the Laplacian eigenvalues were estimated by making the agents execute a local interaction rule that makes their states oscillate at frequencies corresponding to these eigenvalues and then agents use the Fast Fourier Transform (FFT) on their states to identify these

☆ A preliminary version of this work appears in Aragues et al. (2012). The material in this paper was partially presented at the 2012 American Control Conference (ACC2012), June 27–29, 2012, Montreal, Canada. This paper was recommended for publication in revised form by Associate Editor Tamas Keviczky under the direction of Editor Frank Allgöwer.

E-mail addresses: raragues@unizar.es (R. Aragues), guodongs@kth.se (G. Shi), dimos@kth.se (D.V. Dimarogonas), csagues@unizar.es (C. Sagüés), kallej@ee.kth.se (K.H. Johansson), Youcef.MEZOUAR@ubpmes.univ-bpclermont.fr (Y. Mezouar).

[1] Tel.: +34 976 76 24 72; fax: +34 976 76 19 14.

http://dx.doi.org/10.1016/j.automatica.2014.10.051
0005-1098/

Table 1
Notation.

n	Number of agents.
i, j	Agent indices.
k	Iteration, $k \in \mathbb{N}$.
Special matrices and vectors	
$\mathbf{I}$	Identity matrix.
$\mathbf{0}, \mathbf{1}$	Vectors with all entries equal to 0 and 1.
$\mathcal{A}$	Adjacency matrix of the graph.
$\mathcal{L}$	Laplacian matrix, $\mathcal{L} = \text{diag}(\mathcal{A}\mathbf{1}) - \mathcal{A}$.
D	Perron matrix $D = \mathbf{I} - \beta\mathcal{L}$.
C	Deflated matrix, $C = D - \mathbf{1}\mathbf{1}^T/n$.
Matrix operations, eigenvalues and eigenvectors	
A_{ij}, $[A]_{ij}$	(i, j) entry of matrix A.
$\text{diag}(b_1, \ldots, b_r)$	matrix A with $A_{ii} = b_i$ and $A_{ij} = 0$.
$\lambda_i(A)$	ith eigenvalue of A.
$\mathbf{v}_i(A)$	ith eigenvector of A.
$\lambda_\star(\mathcal{L})$	Algebraic connectivity.
$\|A\|_\infty$	Induced ∞-norm, $\max_i \sum_{j=1}^n |A_{ij}|$.
$\|A\|_2$	Spectral norm, $\max_i \sqrt{\lambda_i(A^T A)}$.
$\rho(A)$	Spectral radius, $\max_i |\lambda_i(A)|$.

eigenvalues. A framework for computing the algebraic connectivity was then introduced in Montijano, Montijano, and Sagues (2011) by iteratively bisecting the interval where it is supposed to belong to. Most of the remaining Laplacian spectra estimation solutions relied on the Power Iteration method or variations (De Gennaro & Jadbabaie, 2006; Kempe & McSherry, 2008; Li & Qu, 2013; Oreshkin, Coates, & Rabbat, 2010; Sabattini, Chopra, & Secchi, 2011; Yang et al., 2010). Power Iteration (Householder, 1964) selects an initial vector and then repeatedly multiplies it by a matrix and normalizes it. This vector converges to the eigenvector associated to the leading eigenvalue (the one with the largest absolute value). The original matrix can be previously deflated so that a particular eigenvalue becomes the leading one. The distributed implementations of the power iteration method let each agent maintain one entry of the state vector. The operations that require global knowledge (normalization and deflation) are usually replaced with averaging iterations, as in Sabattini et al. (2011) and Yang et al. (2010) for continuous-time systems, and in De Gennaro and Jadbabaie (2006) and Li and Qu (2013) for discrete-time systems. A brief summary of the power iteration method can be found in Appendix.

Most of these existing algebraic connectivity estimation methods have asymptotic convergence. However, in order to combine in parallel these methods with some other algorithms or control laws that require the knowledge of the algebraic connectivity, it is necessary to have accurate lower and upper bounds as well as the convergence rate of the algebraic connectivity estimation algorithms (see, e.g., Seyboth, Dimarogonas, & Johansson, 2013), which are typically missing in the literature (Oreshkin et al., 2010; Sabattini et al., 2011; Yang et al., 2010).

In this paper, we present an alternative distributed method for computing the algebraic connectivity (Section 3), whose main interest is that it provides upper and lower bounds for the true algebraic connectivity at each iteration. We prove that both bounds converge to the true algebraic connectivity, with a convergence speed no slower than $O(1/k)$.

2. Preliminaries

We use the notation defined in Table 1.

Consider a set of $n \in \mathbb{N}$ agents with indices $i \in \{1, \ldots, n\}$. The agents can exchange information with nearby nodes. This information is represented by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, n\}$ are the agents, and $\mathcal{E}$ are the edges. There is an edge $(i, j) \in \mathcal{E}$ between nodes i and j if they can exchange data. We assume that $\mathcal{G}$ is connected. We use $\mathcal{N}_i$ for the set of neighbors of a node i with whom i can exchange data, $\mathcal{N}_i = \{j \mid (i, j) \in \mathcal{E}\}$, and we let d_i be the *degree* of node i defined as the cardinality of $\mathcal{N}_i$, and $d_{\max} = \max_{i \in \mathcal{V}} d_i$. We say an $n \times n$ matrix C is *compatible* with $\mathcal{G}$ if $C_{ij} = 0$ iff $(i, j) \notin \mathcal{E}$ for $j \neq i$; we let the elements in the diagonal C_{ii} be either equal or different from 0. The adjacency matrix $\mathcal{A} \in \{0, 1\}^{n \times n}$ of $\mathcal{G}$ is

$$\mathcal{A}_{ij} = 1 \quad \text{if } (i, j) \in \mathcal{E}, \qquad \mathcal{A}_{ij} = 0 \quad \text{otherwise}, \ \text{for } i, j \in \mathcal{V}.$$

The Laplacian matrix $\mathcal{L} \in \mathbb{R}^{n \times n}$ of $\mathcal{G}$ is the positive-semidefinite matrix

$$\mathcal{L} = \text{diag}(\mathcal{A}\mathbf{1}) - \mathcal{A}, \tag{1}$$

where $\mathbf{1}$ is as in Table 1. Note that the same Laplacian $\mathcal{L}$ is obtained for graphs with $(i, i) \in \mathcal{E}$ and without $(i, i) \notin \mathcal{E}$ self-loops. Both $\mathcal{A}$ and $\mathcal{L}$ are compatible with the graph. We sort the eigenvalues $\lambda_i(\mathcal{L})$ of $\mathcal{L}$ as follows,

$$\lambda_1(\mathcal{L}) \leq \lambda_2(\mathcal{L}) \leq \cdots \leq \lambda_n(\mathcal{L}).$$

The Laplacian matrix $\mathcal{L}$ has the following well known properties, see e.g., Olfati-Saber et al. (2007): (i) its eigenvalues are upper bounded by $\lambda_n(\mathcal{L}) \leq 2d_{\max}$; (ii) it has an eigenvector $\mathbf{v}_1(\mathcal{L}) = \mathbf{1}/\sqrt{n}$ with associated eigenvalue $\lambda_1(\mathcal{L}) = 0$, $\mathcal{L}\mathbf{1}/\sqrt{n} = \mathbf{0}$; and (iii) when $\mathcal{G}$ is connected, all the other eigenvalues are strictly greater than zero.

The *algebraic connectivity* of $\mathcal{G}$ denoted by $\lambda_\star(\mathcal{L})$ is defined as the second-smallest eigenvalue $\lambda_2(\mathcal{L})$ of the Laplacian $\mathcal{L}$. Usually, the distributed algorithms that estimate the algebraic connectivity have asymptotic convergence, i.e., if we let $\hat{\lambda}_i(k)$ be the estimated algebraic connectivity after k iterations of the algorithm, then $\lim_{k\to\infty} \hat{\lambda}_i(k) = \lambda_\star(\mathcal{L})$, but for a finite k, we have $\hat{\lambda}_i(k) \neq \lambda_\star(\mathcal{L})$. If we do not know how $\hat{\lambda}_i(k)$ approaches $\lambda_\star(\mathcal{L})$, then the selection of the number of steps k and the adjust of a parameter α satisfying $\alpha < \lambda_\star(\mathcal{L})$ are non-trivial. Instead, if we know that our estimate approaches $\lambda_\star(\mathcal{L})$ satisfying $\hat{\lambda}_i(k) \leq \lambda_\star(\mathcal{L})$ for all k, then we can just choose $\alpha < \hat{\lambda}_i(k) \leq \lambda_\star(\mathcal{L})$ at any step k.

Problem 2.1. Our goal is to design distributed algorithms to allow the agents to compute $\lambda_\star(\mathcal{L})$, and/or a lower bound of $\lambda_\star(\mathcal{L})$ in a distributed fashion. ■

From now on, we let C be the following deflated version of the Perron matrix of the Laplacian $\mathcal{L}$, (Aragues, Shi, Dimarogonas, Sagues, & Johansson, 2012; Olfati-Saber et al., 2007; Xiao & Boyd, 2004; Yang et al., 2010)

$$C = \mathbf{I} - \beta\mathcal{L} - \mathbf{1}\mathbf{1}^T/n, \tag{2}$$

where the eigenvalues $\lambda_1(\mathcal{L}), \ldots, \lambda_n(\mathcal{L})$ of the Laplacian and of C are related by

$$\lambda_1(C) = 0, \qquad \lambda_i(C) = 1 - \beta\lambda_i(\mathcal{L}), \quad \text{for } i \in \{2, \ldots, n\},$$

so that the spectral radius $\rho(C)$ of C is associated to the algebraic connectivity $\lambda_\star(\mathcal{L})$ by

$$\lambda_\star(\mathcal{L}) = (1 - \rho(C))/\beta, \quad \text{if } 0 < \beta < 1/\lambda_n(\mathcal{L}). \tag{3}$$

We let D be the not-deflated matrix,

$$D = \mathbf{I} - \beta\mathcal{L}, \quad \text{so that } C = D - \mathbf{1}\mathbf{1}^T/n. \tag{4}$$

3. Distributed computation of the algebraic connectivity

We present a distributed method for estimating the algebraic connectivity $\lambda_\star(\mathcal{L})$ of an undirected graph, which is not only convergent but also provides lower and upper bounds at each step k. We begin with a brief summary of the method, which is then discussed in detail along this section. The method computes the spectral radius of the deflated matrix C, which is related to the Laplacian $\mathcal{L}$ by Eqs. (2), (3). Agents compute the induced ∞-norm $\|\cdot\|_\infty$ of matrix C^k, which is the maximum absolute row sum of