

Distributed Average Consensus in Sensor Networks with Random Link Failures

Soumya Kar
Department of Electrical and
Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213 USA
Email: soumyak@andrew.cmu.edu

José M. F. Moura
Department of Electrical and
Computer Engineering
Carnegie Mellon University
Pittsburgh, PA 15213 USA
Email: moura@ece.cmu.edu

Abstract—We study the impact of the topology of a sensor network on distributed average consensus algorithms when the network links fail at random. We derive convergence results. In particular, we determine a sufficient condition for mean-square convergence of the distributed average consensus algorithm in terms of a moment of the distribution of the norm of a function of the network graph Laplacian matrix L (which is a random matrix, because the network links are random.) Further, because the computation of this moment involves costly simulations, we relate the mean-square convergence to the second eigenvalue of the mean Laplacian matrix, $\lambda_2(\bar{L})$, which is much easier to compute. We derive bounds on the convergence rate of the algorithm, which show that both the expected algebraic connectivity of the network, $E[\lambda_2(L)]$, and $\lambda_2(\bar{L})$ play an important role in determining the actual convergence rate. Specifically, larger values of $E[\lambda_2(L)]$ or $\lambda_2(\bar{L})$ lead to better convergence rates. Finally, we provide numerical studies that verify the analytical results.

Index Terms—Consensus, Topology, Laplacian, Link Failure, Random Matrix.

I. INTRODUCTION

In this paper, we establish convergence properties for arithmetic mean consensus in large sensor networks as a function of the network topology. The algorithm is distributed. Inter-sensor communication at each step is local and determined by the underlying connectivity network. Distributed consensus is a well-studied problem and has many applications, see [1], [2], [3]. These papers assume that the network topology is fixed or evolves deterministically with time, which is unrealistic in many applications. Ref. [4] considers the problem of determining link weights for optimizing convergence speed for a fixed network. Ref. [2] designs both the optimum weights and the topology of the sensor network. The present paper focuses on the design and analysis of a distributed consensus algorithm for networks with random links, as opposed to the deterministic cases discussed above. Such a situation occurs in sensor networks with unreliable communication links. A

This work was partially supported by the DARPA DSO Advanced Computing and Mathematics Program Integrated Sensing and Processing (ISP) Initiative under ARO grant # DAAD 19-02-1-0180, by NSF under grants # ECS-0225449 and # CNS-0428404 and, by an IBM Faculty Award.

link failure is a common scenario in noisy communication channels, where random transmission errors can be treated as a communication failure. A similar situation may arise in networks with limited power budget, where the network may need to shed links at times to meet the power constraint. We model such link failures by a random field. The convergence analysis of any distributed algorithm on such a network is a difficult problem, because the behavior depends on the actual probability distribution of the topology of the network. A similar type of problem is considered in [5], but the network is modeled as a complete graph, with identical link failure probabilities for all the links. In this paper, we establish results for generic network topologies and distributions and show that the convergence properties of the consensus algorithm are related to the moments of the probability distribution of the eigenvalues of related network matrices. These moments are difficult and expensive to compute. To avoid this expense, we relate the convergence properties of the distributed consensus algorithm to the second eigenvalue of the *average* Laplacian matrix. This average Laplacian matrix is much easier to evaluate as will be shown. Numerical studies verify our analytical results.

A brief outline of the rest of the paper follows. Section II explains elementary spectral graph theory concepts. Section III describes the distributed consensus algorithm. Section IV contains the main mean-square convergence theorems, while Section V presents performance bounds. Numerical studies are in Section VI. Finally, Section VII concludes the paper.

II. SPECTRAL GRAPH THEORY

We define a graph $G = (V, E)$ as a 2-tuple, consisting of a set V of N vertices (sensors in our application) and a set E of M edges. We denote an edge between vertices n and l as an unordered pair (n, l) , where the presence of an edge between two vertices indicates they can communicate with each other. A simple graph is a graph without loops and multiple edges. Unless otherwise stated, all the graphs considered in this paper are simple. To each graph we assign an $N \times N$ adjacency matrix A , given by

$$A_{n,l} = \begin{cases} 1 & \text{if } (n, l) \in E \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

The neighborhood of a vertex n is defined as

$$\Omega_n = \{l \in V : (n, l) \in E\}, \forall n \in \{1, \dots, N\} \quad (2)$$

The degree of a node is the number of edges emanating from it and is given by

$$d_n = |\Omega_n|, \forall n \in \{1, \dots, N\} \quad (3)$$

In a similar manner, we define the $N \times N$ Laplacian matrix L of the graph as

$$L = D - A \quad (4)$$

where, $D = \text{diag}(d_1, \dots, d_N)$ is the degree matrix. L is a symmetric positive semidefinite matrix, so all its eigenvalues are non-negative. We can arrange them as follows:

$$0 = \lambda_1(L) \leq \lambda_2(L) \leq \dots \leq \lambda_N(L) \quad (5)$$

The multiplicity of the zero eigenvalue is the number of connected components and $\lambda_2(L)$ is called the algebraic connectivity or the Fiedler value of the network. For connected graphs $\lambda_2(L) > 0$, see [6].

III. PROBLEM FORMULATION AND DISTRIBUTED CONSENSUS ALGORITHM

We model the network of N sensors at an arbitrary time index i by a graph $G(i) = (V, E(i))$, where $E(i)$ denotes the edge set at time i . Since the network is dynamic, $E(i)$ changes over time. Specifically, because of the random link failures, we assume that $E(i)$ is a random subset of $\mathcal{E} \in V \times V$, where \mathcal{E} denotes the set of realizable edges (i.e., $E(i) = \mathcal{E}$ iff there is no link failure.) We model randomness by assuming that an edge in \mathcal{E} may fail or become alive independently of the other edges with some probability. For $(n, l) \in \mathcal{E}$, we denote by $0 \leq P_{nl} \leq 1$ the probability that an edge exists between sensors n and l at any arbitrary time i . We thus define the $N \times N$ probability of edge formation matrix P as

$$P_{nl} = \begin{cases} \text{Probability of edge } (n, l) & \text{if } (n, l) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Thus the edge set $E(i)$ at each time i is the random subset of the set of realizable edges \mathcal{E} formed according to the entries of the P matrix.

Let us assume that the sensors measure or have some initial data at time $i = 0$. We stack them in a vector, which we call the initial state vector $\mathbf{x}(0) \in \mathbb{R}^{N \times 1}$. Let us define the average of the initial state $\mathbf{x}(0)$ as

$$\bar{\mathbf{r}} = \frac{\mathbf{1}^T \mathbf{x}(0)}{N} \quad (7)$$

where $\mathbf{1}$ denotes the vector of all ones. The purpose of the consensus algorithm is to compute the mean $\bar{\mathbf{r}}$ at each sensor, given any initial state vector $\mathbf{x}(0)$, using linear distributed iterations, see [4]. We thus consider the linear update of the state of each sensor n at time i using only local data exchange, given by

$$x_n(i+1) = W_{nn}(i)x_n(i) + \sum_{l \in \Omega_n(i)} W_{nl}(i)x_l(i) \quad (8)$$

Writing in a matrix-vector format, the above update equation becomes

$$\mathbf{x}(i+1) = W(i)\mathbf{x}(i) \quad (9)$$

where $W(i)$ gives the weight matrix at time i . The sparsity pattern of $W(i)$ is determined by the underlying network connectivity, i.e., for $n \neq l$, $W_{nl}(i) = 0$ if $(n, l) \notin E(i)$. There can be different choices for the weight matrix $W(i)$, see [4] for the deterministic case. We consider the following scheme for choosing the matrix $W(i)$

$$W(i) = I - \alpha L(i) \quad (10)$$

where α is a constant independent of time i . This means that, at each stage i of the iteration, we give an equal weight α to every available link. Also, from an implementation point of view, it follows that such a weight assignment is easiest to implement, because no processing is required for computing the weights.

From eqn.(10), we note that $W(i)$ is a random matrix for all i , since the network and hence the Laplacian $L(i)$ is random. It also follows from our previous discussions that the $L(i)$'s, and so the $W(i)$'s, are independent and identically distributed (in the sequel when we refer to the probability distributions of $W(i)$ and $L(i)$, we drop the indices and use W and L respectively.) This makes the state $\mathbf{x}(i)$ a random vector, and hence the convergence properties of the algorithm must be interpreted in probabilistic terms.

IV. MEAN SQUARE CONVERGENCE

Let us define the vector of averages as

$$\mathbf{x}_{\text{avg}} = \bar{\mathbf{r}} \mathbf{1} \quad (11)$$

where $\bar{\mathbf{r}}$ is given in eqn.(7). Starting from an initial state $\mathbf{x}(0)$, we say that the state vector converges to the desired mean vector \mathbf{x}_{avg} if

$$\lim_{i \rightarrow \infty} \|\mathbf{x}(i) - \mathbf{x}_{\text{avg}}\|_2 = 0 \quad (12)$$

where $\|\cdot\|_2$ denotes the standard Euclidean norm. However, as shown in the last section, $\{\mathbf{x}(i)\}_{i \in \mathcal{I}}$, $\mathcal{I} = \{1, 2, \dots\}$ is a random process, and hence sample-wise convergence (see eqn.(12)) is too restrictive and not interesting to deal with. Hence, we consider mean-square convergence of the process. Thus, we say that, given any initial state $\mathbf{x}(0)$, the algorithm converges in the mean-square sense if

$$\lim_{i \rightarrow \infty} E \|\mathbf{x}(i) - \mathbf{x}_{\text{avg}}\|_2 = 0 \quad (13)$$

We now establish a sufficient condition for the mean-square convergence of the distributed algorithm. This will be possible because we assume noiseless communications. First, we prove a lemma.

Lemma 1 For any $\mathbf{x}(0) \in \mathbb{R}^{N \times 1}$, we have,

$$\frac{\|\mathbf{x}(i+1) - \mathbf{x}_{\text{avg}}\|_2}{\|\mathbf{x}(0) - \mathbf{x}_{\text{avg}}\|_2} \leq \prod_{j=0}^i \rho \left(W(i-j) - \frac{1}{N} J \right) \quad (14)$$

where $J = \mathbf{1}\mathbf{1}^T$ and $\rho(\cdot)$ denotes the spectral radius.

Proof: Since $W(i)$ is a symmetric matrix, we can decompose it using orthonormal eigenvectors and get

$$W(i) = U\Lambda U^T \quad (15)$$

where $U = [\mathbf{u}_1 \dots \mathbf{u}_N]$ is the matrix of orthonormal eigenvectors and Λ is the diagonal matrix of eigenvalues. From eqn.(10), it follows that the eigenvalues of $W(i)$ are

$$\lambda_k(W(i)) = 1 - \alpha \lambda_k(L(i)), \quad \forall k \in \{1, \dots, N\} \quad (16)$$

Using eqn.(5), we note that $\lambda_1(W(i)) = 1$ with the corresponding eigenvector $\mathbf{u}_1 = \frac{1}{\sqrt{N}}\mathbf{1}$. Also (see [2]),

$$\rho\left(W(i) - \frac{1}{N}J\right) = \max(|\lambda_2(W(i))|, \dots, |\lambda_N(W(i))|) \quad (17)$$

It can be shown from eqns.(16,17) that

$$\|\mathbf{x}(i+1) - \mathbf{x}_{\text{avg}}\|_2 \leq \rho\left(W(i) - \frac{1}{N}J\right) \|\mathbf{x}(i) - \mathbf{x}_{\text{avg}}\|_2 \quad (18)$$

where we use the fact that $\frac{J}{N}W(j) = \frac{J}{N}$, $\forall j$ (see [2].) The proof follows by using eqn.(18) repeatedly. Recall that $W(i)$'s are i.i.d. and we refer generically to them as the random matrix W . ■

Theorem 2 The consensus algorithm converges in the mean square sense for any $\mathbf{x}(0) \in \mathbb{R}^{N \times 1}$ if $\mathbb{E}[\rho(W - \frac{J}{N})] < 1$.

Proof: From eqn.(14) it follows that

$$\mathbb{E} \|\mathbf{x}_i - \mathbf{x}_{\text{avg}}\|_2 \leq \left(\mathbb{E} \left[\rho\left(W - \frac{1}{N}J\right)\right]\right)^{i-1} \|\mathbf{x}_0 - \mathbf{x}_{\text{avg}}\|_2 \quad (19)$$

where we use the fact that the $W(i)$'s are i.i.d. Hence, if $\mathbb{E}[\rho(W - \frac{J}{N})] < 1$,

$$\lim_{i \rightarrow \infty} \mathbb{E} \|\mathbf{x}(i) - \mathbf{x}_{\text{avg}}\|_2 = 0 \quad (20)$$

Theorem 3 Let $\bar{L} = \mathbb{E}[L]$. Then a sufficient condition for mean-square convergence is $\lambda_2(\bar{L}) > 0$. In other words, we have mean-square convergence if the average network, \bar{L} , is connected.

Proof: We give a constructive proof. We show that, if $\lambda_2(\bar{L}) > 0$, we can find an α for which $\mathbb{E}[\rho(W - \frac{J}{N})] < 1$. Convergence then follows from theorem 2.

We recall a result from spectral graph theory, which states that a graph is connected, (i.e., $\lambda_2(L) > 0$), iff the corresponding adjacency matrix A is irreducible, see [7]. We note that the expected or mean adjacency matrix $\bar{A} = \mathbb{E}[A] = P$, where P is given in eqn.(6). It follows that,

$$\bar{L} = \bar{D} - \bar{A} \quad (21)$$

where $\bar{D} = \mathbb{E}[D]$. Since \bar{A} assumes the form of a weighted adjacency matrix (entries are not necessarily 0 or 1), applying

the same result (see [7]), we get,

$$\lambda_2(\bar{L}) > 0 \Leftrightarrow \bar{A} \text{ is irreducible} \quad (22)$$

The irreducibility of \bar{A} suggests that with non-zero probability we have graph realizations for which $\lambda_2(L) > 0$. In particular, we can have a realization for which the edge set $E = \mathcal{E}$ and clearly this network is connected (the adjacency matrix in this case has the same sparsity pattern of \bar{A} , with non-zero entries of \bar{A} replaced by ones.) This shows that, with non-zero probability, $\lambda_2(L) > 0$, which makes $\mathbb{E}[\lambda_2(L)] > 0$. Thus, we have

$$\lambda_2(\bar{L}) > 0 \Rightarrow \mathbb{E}[\lambda_2(L)] > 0 \quad (23)$$

We now use another spectral graph theory result, which states that, for any graph G ,

$$\lambda_N(L) \leq 2d_{\max}(G) \quad (24)$$

where $d_{\max}(G)$ denotes the maximum vertex degree of G , see [7]. For our case, let l_{\max} be the maximum degree of the graph with edge set $E = \mathcal{E}$. Then, from eqn.(24), it follows that, for all realizable networks,

$$\lambda_N(L) \leq 2l_{\max} \quad (25)$$

We now claim that the algorithm converges in the mean-square sense for the following choice of α

$$\alpha_{\text{ms}} = \frac{1}{2l_{\max}} \quad (26)$$

Then, using eqns.(16 and 17), we get

$$\rho\left(W - \frac{J}{N}\right) = 1 - \alpha_{\text{ms}} \lambda_2(L) \quad (27)$$

From this it follows that

$$\mathbb{E}[\rho(W - \frac{J}{N})] = 1 - \frac{1}{2l_{\max}} \mathbb{E}[\lambda_2(L)] < 1 \quad (28)$$

and mean square convergence follows from theorem 2. Thus, if $\lambda_2(\bar{L}) > 0$, we can choose an α for which the algorithm converges. ■

Theorem 3 is very significant in the sense that it relates the convergence properties directly to the probability distribution of the graph Laplacian L . In particular, it shows that for mss convergence connectivity on the average is sufficient. The choice of α_{ms} in eqn.(26) is easily obtained from \mathcal{E} .

V. PERFORMANCE BOUNDS

Theorem 2 suggests that a smaller value of $\mathbb{E}[\rho(W - \frac{J}{N})]$ results in a faster convergence. The value of $\mathbb{E}[\rho(W - \frac{J}{N})]$ depends on the choice of α . Thus for the fastest convergence rate we must choose α that minimizes the function

$$C(\alpha) = \mathbb{E}[\rho(W - \frac{J}{N})] \quad (29)$$

Let α^* be the minimizing α . Also, let

$$C^* = \inf_{\alpha} \mathbb{E}[\rho(W - \frac{1}{N}J)] \quad (30)$$

We call $S_c = 1/C^*$ the best achievable convergence rate. We note from Theorem 3 that

$$S_c \geq \frac{1/C(\alpha_{ms})}{1 - E[\lambda_2(L)]/2l_{\max}} \quad (31)$$

Eqn.(31) shows that a higher value of $E[\lambda_2(L)]$ gives a better convergence rate. We now try to relate S_c with $\lambda_2(\bar{L})$. From the properties of Laplacians (see [6]), it can be shown that $\lambda_2(L)$ is a concave function of L . For completeness, we include the following proof.

Let us take any $0 \leq t \leq 1$ and Laplacians L_1 and L_2 . It follows from the Courant-Fischer theorem

$$\begin{aligned} \lambda_2(tL_1 + (1-t)L_2) &= \min_{\mathbf{z} \perp \mathbf{1}} \mathbf{z}^T (tL_1 + (1-t)L_2) \mathbf{z} \quad (32) \\ &\geq t \min_{\mathbf{z} \perp \mathbf{1}} \mathbf{z}^T L_1 \mathbf{z} + (1-t) \min_{\mathbf{z} \perp \mathbf{1}} \mathbf{z}^T L_2 \mathbf{z} \\ &= t\lambda_2(L_1) + (1-t)\lambda_2(L_2) \end{aligned}$$

which proves the concavity of $\lambda_2(L)$. Then, using Jensen's inequality, we get

$$E[\lambda_2(L)] \leq \lambda_2(\bar{L}) \quad (33)$$

This together with eqn.(31) suggests that an increase in $\lambda_2(\bar{L})$ should lead to better convergence rates.

VI. NUMERICAL STUDIES

Eqn.(31) suggests that mss convergence rate increases with increasing $E[\lambda_2(L)]$, which is difficult to compute. Eqn.(33) suggests that, for $E[\lambda_2(L)]$ to be large, $\lambda_2(\bar{L})$ (much easier to compute) must be large. In this section we do simulation studies which show that larger $\lambda_2(\bar{L})$ in fact leads to higher S_c . Fig. 1 (on the top) shows a plot of S_c and $E[\lambda_2(L)]$. In Fig. 1 (on the bottom), we study the relationship between S_c and $\lambda_2(\bar{L})$. We fixed the number of sensors at $N = 500$ and generated 200 different edge-set probability distributions, P , to get the plots. For each P , the computation of $\lambda_2(\bar{L})$ was very easy (see eqn.(21)). However to compute $E[\lambda_2(L)]$ for each P , we generated 450 random L 's, evaluated $\lambda_2(L)$ from each of them and then averaged to get the result (which required the generation of about 200×450 random matrices.) As suggested in the last section, the convergence speed S_c increases with $\lambda_2(\bar{L})$. This relation between S_c and $\lambda_2(\bar{L})$ is indeed very significant, as computation of $\lambda_2(\bar{L})$ for arbitrary network distributions is very easy, whereas computing $E[\lambda_2(L)]$ involves costly numerical simulations in general.

VII. CONCLUSION

In this paper, we study the problem of distributed average consensus in sensor networks with unreliable communication links, which we model as random networks with some probability distribution. We show that, in order to guarantee mean-square convergence, the network need not be connected with probability 1; connectivity on average is enough. We formalize these results, first by establishing a sufficient condition for mean-square convergence in terms of the quantity $E[\rho(W - \frac{1}{N}J)]$. But evaluating $E[\rho(W - \frac{1}{N}J)]$ involves computing a

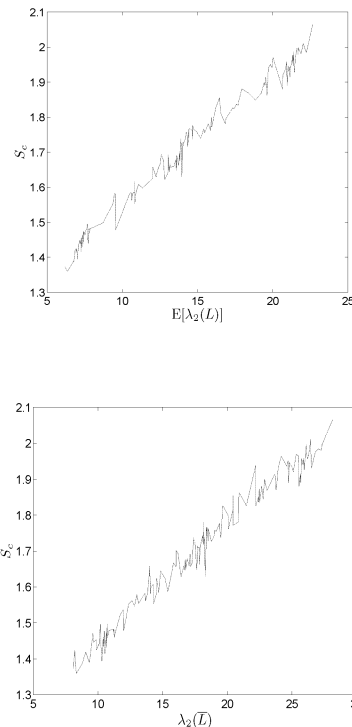


Fig. 1. Top: Figure showing the relation between S_c and $E[\lambda_2(L)]$. Bottom: Figure showing the relation between S_c and $\lambda_2(\bar{L})$.

moment of the spectral norm of a random matrix, and since $E[\rho(W - \frac{1}{N}J)]$ is a function of α , we have to repeat the computation for different values of α to check for mean-square convergence. On the other hand, theorem 3 establishes a sufficient condition in terms of $\lambda_2(\bar{L})$, which is much easier to compute as shown in Section VI. Finally, we derive bounds for the convergence rate of the distributed consensus algorithm and show that larger values of $E[\lambda_2(L)]$ or $\lambda_2(\bar{L})$ lead to better convergence rates.

REFERENCES

- [1] S. A. Aldosari and J. M. F. Moura, "Distributed detection in sensor networks: connectivity graph and small-world networks," *Thirty-Ninth Asilomar Conference on Signals, Systems, and Computers*, pp. 230 – 234, Oct. 2005, Pacific Grove, CA.
- [2] S. Kar, S. Aldosari, and J. M. F. Moura, "Topology for distributed inference on graphs," submitted, 30 pages, June 2006. [Online]. Available: <http://arxiv.org/PS.cache/cs/pdf/0606/0606052.pdf>
- [3] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automat. Contr.*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.
- [4] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Contr. Lett.*, vol. 53, pp. 65–78, 2004.
- [5] Y. Hatano and M. Mesbahi, "Agreement over random networks," *43rd IEEE Conference on Decision and Control, 2004. CDC.*, vol. 2, pp. 2010–2015, Dec. 2004, Atlantis, Paradise Island, Bahamas.
- [6] F. R. K. Chung, *Spectral Graph Theory*. Providence, RI : American Mathematical Society, 1997.
- [7] S. Camiz and S. Stefani, *Matrices and Graphs: Theory and Applications*. Singapore: World Scientific Publishing, 1996.