Broadcast and Gossip Stochastic Average Consensus Algorithms in Directed Topologies

Daniel Silvestre, João P. Hespanha, *Fellow, IEEE*, and Carlos Silvestre

***Abstract*—We address the problem of a set of agents reaching consensus by computing the average of their initial states. We propose two randomized algorithms over a directed communication graph where either a random node broadcast its value or a randomly selected pair of nodes communicate in a distributed fashion. The proposed algorithms guarantee convergence in three important definitions, namely: almost surely, in expectation, and in the mean-square sense. We show how the parameters of the algorithm can be optimized to improve the rate of convergence and compare its rates of convergence for directed and undirected graphs.**

***Index Terms*—Communication networks, consensus, networks of autonomous agents, stability.**

I. Introduction

CONSENSUS refers to the problem where a group of agents needs to agree on a function of their initial state by means of a distributed algorithm, in which the communication between agents is constrained by a network topology. Such problem is of prime importance and examples of application range from distributed optimization [1], [2]; motion coordination tasks like flocking, leader following [3]; rendezvous problems [4]; and resource allocation in computer networks [5].

Many practical applications have been reformulated as a consensus problem and in other cases as a building block for algorithms to address more complicated challenges. In [6], a distributed Kalman Filter is developed based on two consensus systems that compute averages. The filter is than applied to the estimation of the motion of a target in the plane. An experimental evaluation of distributed Kalman filters based on consensus can also be found in [7] where the estimation of the motion of a real robot was performed. Computing the average through a consensus system in a distributed environment can leverage the use of algorithms with mild assumptions on the communication model between the local observers, which motivates the study of this paper (i.e., considering a gossip and asymmetric communication at each time step with improved convergence).

In the context of a smart grid, the authors of [8] have used consensus to implement an incremental cost consensus algorithm that solves the economic dispatch problem, which is the problem of allocating electrical power and generation to the different buses and generators. The work in [9] also discusses how thermal energy storage in a smart building, autonomous space satellites, and frequency control of power systems can be seen as a consensus dynamics. The interested reader is directed to [10] and the vast number of references therein that describe various works where consensus has been used for many different practical applications, namely: biology, sociology, parallel computation, power networks, robotics, etc. Apart from all the references more traditionally related with control and automation, emphasizes is place on: [11] where Hegselmann and Krause introduced a consensus algorithm to sociology, describing the dynamics of opinion formation among different persons; [12] with Evans and Patterson investigating the consensus behavior in flocks of estrildine finches; in [13], Mirollo and Strogatz studied the synchronization of pulse-coupled biological oscillators; and Monus and Barta researched the degree of synchronization of sparrows under different predation risk in [14]. All such examples motivate the study of consensus algorithms.

The average consensus problem has been solved using linear distributed algorithms with each agent computing a weighted average of its state and the values received from its neighbors (see, e.g., [15] and [16]). Several instances of this problem have been proposed such as considering stochastic packet drops and link failures [17], [18], quantized data transmissions [19], and time-varying communication connectivity [15].

An important class of solutions capable of dealing with a varying network topology caused by nodes joining and leaving the network was introduced in [20] as a randomized gossip algorithm. The main feature of this algorithm is that each agent communicates with a randomly selected neighbor at each transmission time. In [20], each pair of communicating nodes

Manuscript received December 25, 2017; revised March 12, 2018; accepted May 1, 2018. Date of publication May 21, 2018; date of current version May 28, 2019. This work was supported in part by the project MYRG2016- 00097-FST of the University of Macau; in part by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) through Institute for Systems and Robotics (ISR), under the Laboratory for Robotics and Engineering Systems (LARSyS) project UID/EEA/50009/2013; and in part by the FCT PhD grant SFRH/BD/71206/2010. The work of J. Hespanha was supported by the National Science Foundation under Grant CNS-1329650. Recommended by Associate Editor Y. Hong. *(Corresponding author: Daniel Silvestre.)*

D. Silvestre is with the Department of Electrical and Computer Engineering, Faculty of Science and Technology, University of Macau, Macau, China, and also with the Institute for Systems and Robotics, Instituto Superior Técnico, Universidade de Lisboa, Lisboa 1649-004, Portugal (e-mail: dsilvestre@isr.ist.utl.pt).

J. P. Hespanha is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106-9560 USA (e-mail: hespanha@ece.ucsb.edu).

C. Silvestre is with the Department of Electrical and Computer Engineering, Faculty of Science and Technology, University of Macau, Macau, China, on leave from the Instituto Superior Técnico, Technical University of Lisbon, Lisbon 1049-001, Portugal (e-mail: csilvestre@umac.mo).

Digital Object Identifier 10.1109/TCNS.2018.2839341

exchange their state information, which assumes bidirectionality in the communication. In contrast, we do not assume that communication is bidirectional in the same transmission time.

In this study, we consider the average consensus problem in scenarios where communication is unidirectional at each time slot, i.e., at each transmission time, a single agent transmits data to one or several agents, but does not receive data. Note that at a different time slot receiver and sender agents may invert their roles, i.e., the word unidirectional refers only to communication at a given transmission time. For concreteness, we consider the two following scenarios: 1) randomized gossip algorithms in wireless networks, where each agent becomes active at randomly chosen times, transmitting its data to a single neighbor; and 2) broadcast wireless networks, where each agent transmits to all the other agents, access to the network occurs with the same probability for every agent, and the intervals between transmissions are independent and identically distributed. As we shall see, the unidirectionality communication constraint precludes in general the existence of a linear distributed algorithm where associated to each agent there is a single scalar state. The state of a node is updated based on the values of the other agents, as in related problems where the communication topology of the network is also time varying, but satisfies different assumptions (see [15] and [16]).

The state-of-the-art related to our study of a consensus algorithm based on the gossip communication includes the works in [20] and [21]. As supra cited, [20] considers bidirectional communications but provides upper and lower bounds on the convergence to the average consensus. More recently, [21] proposes a linear algorithm converging almost surely and in the mean square sense to consensus. The study presented in this paper can be seen as a generalization of the one in [21] since a careful selection of the parameters returns the algorithm in [21] as a worst-case performance. We illustrate via simulations for a directed topology example and a bidirectional one how the convergence rate can vary. Additionally, by employing a technique based on a Lyapunov function, it is possible in this paper to provide a proof of the exponential mean square stability as opposed to asymptotic convergence in [21].

To that extent, we focus on symmetric communication topologies, meaning that if an agent a can communicate with an agent b, then the reverse is also possible, although this does not take place at the same transmission time, i.e., at each transmission time, the graphs modeling communications are in general asymmetric. Note that this is typically the case in wireless networks, and therefore, this assumption is reasonable to assume in both scenarios (1) and (2).

The main contributions of this paper are twofolded.

1) We introduce a new algorithm based on state augmentation to deal with the case that communication is unidirectional in each time slot. We consider two scenarios, namely the *gossip*—where each node communicates with one neighbor; and *broadcast*—where each node transmits to the whole network but does not receive information at that time slot. We show that such an algorithm is faster than the ones in the state-of-the-art and provide results regarding three different stochastic convergence definitions (including the exponential mean square stability) and present the necessary and sufficient conditions for convergence. Results regarding convergence rates in discrete time are presented for both scenarios.
2) We address the problem of finding the fastest converging directed algorithm showing that such an optimization can be separated into optimizing the probabilities matrix using standard techniques from convex optimization and using a brute force strategy (or other form) to select the algorithm's parameters.

The body of research focusing on consensus algorithms is quite extensive. For example, in [22], a technique using a scaling variable is employed and the network model consists of all nodes communicating to its neighbors with the correspondent communication graph being strongly connected. In [23], a gossip algorithm is presented using an asynchronous communication between the pairs of nodes. The average consensus is achieved using a state augmentation technique and a nonlinear operation based on the received state and the node's own state. The method does not assume a symmetric communication topology, but it is only proved to converge almost surely and not in the mean square sense. Our algorithm is the directed linear parallel of the standard gossip algorithm presented in [20] and relates to the linear distributed algorithms [16].

The study of convergence using ergodic infinite sequences of stochastic matrices has also been applied to study the consensus problem. In [24], the underlying network is generated by a random graph process and convergence is shown to be equivalent to the spectral radius of the expected value matrix having the second largest eigenvalue inside the unit circle. The chain product of stochastic matrices is studied in [25] for balanced and strongly aperiodic chains. In [26], the concept of ergodicity is explored to prove that a weighted gossip algorithm, which uses a variable to estimate the sum of all initial states and a weight variable to count the number of nodes, converges to the average consensus. These proposals using the ergodicity concept require each matrix in the chain to have strict positive diagonal, which differs from the class of algorithms studied in this paper. The same concept of a variable to track the sum and another for the number of nodes is used in [27], even though, the main focus is on bounds for the time of convergence. In [28], multiple dynamic weight assignment techniques are proposed and the algorithm is showed to converge if the underlying graph is strongly connected. In essence, all these proposals that require strongly connected graphs as the support graph for each update matrix differ from our work in the sense that in each iteration more than a pair of nodes need to communicate.

This paper extends the work done in [29] by providing a proof for converge in the mean square sense and almost surely and also by showing how to use the structure of the expected value matrix to simplify the nonconvex optimization of the average of the nonsymmetric transmission matrices.

The remainder of this paper is organized as follows. In Section II, we introduce the average consensus problem and the network model. We proceed to the proposed solution in Section III, and state convergence and respective rates results in Sections IV and V. Concluding remarks and directions for future work are given in Section VI.

Notation: The transpose and the spectral radius of a matrix A are denoted by $A^\intercal$ and $r_\sigma(A)$, respectively. We let $\mathsf{1}_n := [1\dots 1]^\intercal$ and $\mathsf{0}_n := [0\dots 0]^\intercal$ indicate n-dimension vector of ones and zeros, and I_n denotes the identity matrix of dimension n . Dimensions are omitted when no confusion arises. The vector e_i denotes the canonical vector whose components equal zero, except component i that equals one. The notation $\mathrm{diag}([A_1 \dots A_n])$ indicates a block diagonal matrix with blocks A_i. The Kronecker product is denoted by $\otimes$. The operation $\mathrm{vec}(A)$ returns a vector resulting from stacking all columns of matrix A.

II. Problem Description

We consider a set of n agents with scalar state $x_i(k)$, $1 \leq i \leq n$, and our goal is to construct a distributed iterative algorithm that guarantees convergence of the state to its initial average value, i.e.,

$$\lim_{k\to\infty} x_i(k) = x_{\text{av}} := \frac{1}{n}\sum_{i=1}^{n} x_i(0). \tag{1}$$

We refer to this problem as the *average consensus problem*.

In gossip algorithms, each node has a clock which at random times chooses one of its neighbors to communicate its own state. The time a communication is attempted is called a transmission time k and assumed that each node has the same probability of being the node that initiated the communication. Such node, denoted by i, chooses a random out-neighbor j according to the probability distributions $w_{i1}, w_{i2}, \dots, w_{\text{in}}$, $\sum_{j=1}^{n} w_{ij} = 1$ $\forall i$. The set of all out-neighbors of i is denoted by $\mathcal{N}_{\text{out}}(i)$, with the number of elements in the set being given by n_{out}, and equivalently, the set of all in-neighbors of i is denoted by $\mathcal{N}_{\text{in}}(i)$.

The communication topology is modeled by a directed graph $G = (\mathcal{V}, E)$, where $\mathcal{V}$ represents the set of n agents, also called nodes, and $E \subseteq \mathcal{V}\times\mathcal{V}$ is the set of communication links, also called edges. The node i can send a message to the node j, if $(i,j) \in E$. If there exists at least one $i \in \mathcal{V}$ such that $(i,i) \in E$, we say that the graph has self-loops, which can model, for example, packet drops since node i only has access to its own value at that transmission time. We associate to the graph G a *weighted adjacency matrix* W with entries

$$W_{ij} := \begin{cases} w_{ij}, \text{if } (i,j) \in E \\ 0, \text{otherwise} \end{cases} ; w_{ij} \in [0,1].$$

Our goal is to solve this problem using a *linear randomized gossip algorithm* defined by the iteration

$$x(k+1) = U_k x(k) \tag{2}$$

where U_k is selected randomly from a set $\{Q_{ij}, 1 \leq i \leq n, 1 \leq j \leq n\}$. The matrices Q_{ij} implement the update on state variables x_i and x_j caused by a transmission from node i to node j and represent a set of *column stochastic* matrices (i.e., $\mathsf{1}^\intercal Q_{ij} = \mathsf{1}^\intercal$) to keep the average between iterations.

Since matrices U_k in (2) are randomly chosen, the state in (2) is a stochastic process and we need to specify how to interpret the convergence in (1).

Definition 1 (Stochastic Convergence): We say that the state of (2):

1) *converges almost surely to average consensus* if

$$\lim_{k\to\infty} x_i(k) = x_{\text{av}} := \frac{1}{n}\sum_{i=1}^{n} x_i(0) \quad \forall_{i\in\{1,\dots,n\}}$$

almost surely;

2) *converges in expectation to average consensus* if

$$\lim_{k\to\infty} \mathbb{E}[x_i(k)] = x_{\text{av}} \quad \forall_{i\in\{1,\dots,n\}}$$

3) *converges in the mean square sense to average consensus* if

$$\lim_{k\to\infty} \mathbb{E}[(x_i(k) - x_{\text{av}})^2] \to 0 \quad \forall_{i\in\{1,\dots,n\}}$$

4) *converges exponentially in the mean square sense to average consensus* if for all k

$$\mathbb{E}[(x_i(k) - x_{\text{av}})^2] \leq \lambda^k (x_i(0) - x_{\text{av}})^2 \;\; \forall i \leq n, \quad 0 < \lambda < 1.$$

III. Proposed Solution

Our original proposed solution [29] to the randomized gossip case is presented next. In [30], it has been showed that the asymmetric randomized gossip algorithm does not achieve average consensus even if the expected value transmission matrix is a symmetric matrix.

We start by augmenting the original state $x(k)$ with an auxiliary vector $y(k) \in \mathbb{R}^n$, and define

$$z = (x, y). \tag{3}$$

We consider a linear distributed algorithm taking the form

$$z(k+1) = U_k z(k) \tag{4}$$

where $z(0) = (x(0), y(0)), y(0) = 0$. Intuitively, the purpose of y it to assure that at each iteration the total state average is kept constant, i.e., that

$$\frac{\sum_1^n x_i(k+1) + \sum_1^n y_i(k+1)}{2n} = \frac{\sum_1^n x_i(k) + \sum_1^n y_i(k)}{2n}. \tag{5}$$

If we initialize y to zero and guarantee that $y(k)$ goes to zero, then average consensus is achieved. More specifically, the proposed algorithm can be described as follows.

At time k, a given node i sends a message containing $x_i(k)$ and $y_i(k)$ to one out-neighbor. The node i does not change its state, i.e.,

$$x_i(k+1) = x_i(k) \tag{6}$$

and resets the auxiliary state to zero

$$y_i(k+1) = 0. \tag{7}$$

A node j receiving this message, updates its state $x_j(k)$ according to

$$x_j(k+1) = (1-\alpha)x_j(k) + \alpha x_i(k) + \beta y_j(k) + \gamma y_i(k) \tag{8}$$

Algorithm 1: Gossip algorithm $\mathcal{G}$.

Require: Set initial state $x(0)$.
Ensure: Computation of the average of $x(0)$.
/* *Initialize* z */
$z(0) := \begin{bmatrix} x(0) \\ 0_n \end{bmatrix}$
for each k **do**
  /* *Randomly select a pair* (i, j) */
  $i = \text{rand}()$
  $j = \text{rand}()$
  /* *Update the state of node* i */
  $z_i(k+1)$ *according to (6) and (7)*
  /* *Update the state of node* j */
  $z_j(k+1)$ *according to (8) and (9)*
end for

Algorithm 2: Broadcast algorithm $\mathcal{B}$.

Require: Set initial state $x(0)$.
Ensure: Computation of the average of $x(0)$.
/* *Initialize* z */
$z(0) := \begin{bmatrix} x(0) \\ 0_n \end{bmatrix}$
for each k **do**
  /* *Randomly select an updating node* i */
  $i = \text{rand}()$
  /* *Update the state of node* i */
  $z_i(k+1)$ *according to (6) and (7)*
  for each $j \in \mathcal{N}_{\text{out}}(i)$ **do**
    /* *Update the state of node* j */
    $z_j(k+1)$ *according to (8) and (9)*
  end for
end for

and updates its variable $y_j(k)$ according to

$$y_j(k+1) = \frac{y_i(k)}{n_{\text{out}}(i,k)} + y_j(k) + x_j(k) - x_j(k+1) \tag{9}$$

so that the total state average is kept constant, i.e., (5) holds. In the following sections, we present the details for each of the considered scenarios.

A. Gossip Algorithm $\mathcal{G}$

The matrices U_k are taken from the set $\{Q_{ij}, 1 \leq i, j \leq n\}$, where each Q_{ij} corresponds to a transmission (see the pseudocode of the algorithm for each transmission in Algorithm 1) from node i to an out-neighbor node j, and these matrices are described as follows. Let $\Lambda_i := \text{diag}(\mathsf{e}_i)$ and $\Omega_{ij} := I - (\Lambda_i + \Lambda_j)$. Then

$$Q_{ij} = \begin{bmatrix} A_{ij} & B_{ij} \\ C_{ij} & D_{ij} \end{bmatrix} \tag{10}$$

where

$$\begin{aligned} A_{ij} &:= I - \alpha\Lambda_j + \alpha \mathsf{e}_j \mathsf{e}_i^{\intercal} \\ B_{ij} &:= \beta\Lambda_j + \gamma \mathsf{e}_j \mathsf{e}_i^{\intercal} \\ C_{ij} &:= \Lambda_j(I - A_{ij}) \\ D_{ij} &:= \Omega_{ij} + \Lambda_j(I + \mathsf{e}_j \mathsf{e}_i^{\intercal} - B_{ij}). \end{aligned} \tag{11}$$

The matrices defined in (10) also model the case where a node i picks itself when there is a clock tick (with probability w_{ii}).

The matrices U_k are by construction independent and identically distributed, and satisfy

$$\text{Prob}[U_k = Q_{ij}] = \frac{1}{n} w_{ij}$$

($\frac{1}{n}$ is the probability that node i is the one whose clock ticks at k and w_{ij} the probability that i picks its out-neighbor node j).

B. Broadcast Algorithm $\mathcal{B}$

The matrices U_k are taken from the set $\{R_i, 1 \leq i \leq n\}$, where each R_i corresponds to a transmission (see Algorithm 2) from node i to every other node. Let $\Lambda_i := \text{diag}(\mathsf{e}_i)$, $\Omega_i = (I - \Lambda_i)$. Then

$$R_i = \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix} \tag{12}$$

$$\begin{aligned} A_i &= (1-\alpha)I + \alpha \mathbf{1}_n \mathsf{e}_i^{\intercal} \\ B_i &= \Omega_i(\beta I + \gamma \mathbf{1}_n \mathsf{e}_i^{\intercal}) \\ C_i &= \Omega_i(I - A_i) \\ D_i &= \Omega_i \left(I + \frac{\mathbf{1}_n \mathsf{e}_i^{\intercal}}{n-1} - B_i \right). \end{aligned}$$

The matrices U_k are independent and identically distributed due to our assumption that nodes access the network with the same probability, i.e.,

$$\text{Prob}[U_k = R_i] = \frac{1}{n}.$$

Hereafter, we denote by *gossip algorithm* $\mathcal{G}$, the linear distributed algorithm modeled by (4) and (10), and denote by *broadcast algorithm* $\mathcal{B}$, the linear distributed algorithm modeled by (4) and (12). Note that, by construction, for both gossip and broadcast algorithms, the matrices $\{U_k, k \geq 0\}$ are such that

$$\begin{bmatrix} \mathbf{1}_n^{\intercal} & \mathbf{1}_n^{\intercal} \end{bmatrix} U_k = \begin{bmatrix} \mathbf{1}_n^{\intercal} & \mathbf{1}_n^{\intercal} \end{bmatrix} \tag{13}$$

which means that the total average is preserved at each iteration, i.e., $\mathbf{1}_{2n}^{\intercal} z(k+1) = \mathbf{1}_{2n}^{\intercal} z(k)$, and

$$U_k \begin{bmatrix} \mathbf{1}_n \\ 0_n \end{bmatrix} = \begin{bmatrix} \mathbf{1}_n \\ 0_n \end{bmatrix} \tag{14}$$

which means that if consensus is achieved at iteration k, i.e., if $x(k) = c\mathbf{1}_n$ and $y(k) = 0_n$, the state remains unchanged at iteration $k+1$, i.e., $x(k+1) = c\mathbf{1}_n$ and $y(k+1) = 0_n$.

IV. Convergence

In this section, we provide results regarding the convergence for the two considered scenarios. We start by providing necessary and sufficient conditions to test the convergence of the algorithms for a particular network topology, which can be seen as a generalization of the bidirectional case for a unidirectional case with state augmentation.

The next theorem provides necessary and sufficient conditions for convergence of any of the algorithms with state augmentation.

Theorem 1: Consider a linear distributed algorithm (4) where $\{U_k, k \geq 0\}$ are characterized by (13) and (14), and are randomly chosen from a set $\mathcal{M} := \{B_i, 1 \leq i \leq n_p\}$, according to

$$\text{Prob}[U_k = B_i] = p_i, \sum_{i=1}^{n_p} p_i = 1.$$

Then, the linear distributed algorithm converges in expectation to average consensus if and only if

$$r_\sigma\left(\sum_{i=1}^{n_p} p_i B_i - \frac{1}{n}\begin{bmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{bmatrix}\begin{bmatrix}\mathsf{1}_n^\intercal & \mathsf{1}_n^\intercal\end{bmatrix}\right) < 1 \tag{15}$$

and converges in the mean square sense to average consensus if and only if

$$r_\sigma\left(\sum_{i=1}^{n_p} p_i B_i \otimes B_i - S\right) < 1 \tag{16}$$

where

$$S := \frac{1}{n^2}\left(\begin{bmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{bmatrix} \otimes \begin{bmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{bmatrix}\right)\left(\begin{bmatrix}\mathsf{1}_n^\intercal & \mathsf{1}_n^\intercal\end{bmatrix} \otimes \begin{bmatrix}\mathsf{1}_n^\intercal & \mathsf{1}_n^\intercal\end{bmatrix}\right).$$

Proof: We start by proving (15). Let $R := \mathbb{E}[U_k] = \sum_{i=1}^{n_p} B_i$. Since U_k are i.i.d matrices, we have $\mathbb{E}[z(k+1)] = R\mathbb{E}[z(k)]$. By conditioning k times, we have $\mathbb{E}[z(k+1)] = R^k z(0)$, from which we must have $R^k \to \frac{1}{n}\left[\begin{smallmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{smallmatrix}\right][\mathsf{1}_n^\intercal \;\; \mathsf{1}_n^\intercal]$ since we want $z(k)$ to converge to consensus. By linearity of the expected value operator and since we defined B_i as in (13) and (14), we have an eigenvalue of 1 corresponding to the left eigenvector $[\mathsf{1}_n^\intercal \;\; \mathsf{1}_n^\intercal]$ and to the right eigenvector $\left[\begin{smallmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{smallmatrix}\right]$. We need to have the remaining eigenvalues to have magnitude strictly less than 1, which gives (15).

To prove (16), we calculate the $\mathbb{E}[z(k+1)z(k+1)^\intercal] = \mathbb{E}[U_k z(k) z(k)^\intercal U_k^\intercal]$. Let us define $Z(k) = z(k)z(k)^\intercal$, then

$$\begin{aligned} Z(k+1)_{ij} &= ((U_k z(k))(U_k z(k))^\intercal)_{ij} \\ &= (U_k z(k))_i (U_k z(k))_j. \end{aligned}$$

Defining $\tilde{Z}(k) = \text{vec}(Z(k))$, we have $\tilde{Z}(k+1) = (U_k \otimes U_k)\tilde{Z}(k)$. Let $R_2 := \mathbb{E}[U_k \otimes U_k] = \sum_{i=1}^{n_p} B_i \otimes B_i$. Since U_k are i.i.d, $\mathbb{E}[\tilde{Z}(k+1)] = R_2\mathbb{E}[\tilde{Z}(k)]$, which by repeating the conditioning, we get $\mathbb{E}[\tilde{Z}(k+1)] = R_2^k \tilde{Z}(0)$. Thus, $R_2^k \to S$ for the system to go to consensus. Take $v = \left[\begin{smallmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{smallmatrix}\right]$ and $w = \begin{bmatrix}\mathsf{1}_n^\intercal & \mathsf{1}_n^\intercal\end{bmatrix}$ and again due to linearity of the expected value operator, R_2 has eigenvalue 1 for the right eigenvector $v \otimes v$ and the left eigenvector $w \otimes w$. To have convergence all the remaining eigenvalues must have magnitude strictly less than 1, which gives (16). ∎

The previous theorem related the convergence with the spectral radius of a matrix for a given network topology and probabilities of communication. The result can be used to test if a given configuration produces an exponential convergent solution in some stochastic sense.

Remark 1: We remark that the algorithm in [21] is a particular instance of $\mathcal{G}$, where $\alpha = w$, $\beta = \epsilon w$, and $\gamma = 0$ (where w and ϵ are mixing parameters of the algorithm in [21]). Therefore, by setting $\gamma = 0$ and picking sufficiently small α and β, we recover the same result that the algorithm $\mathcal{G}$ converges asymptotically in the mean square sense.

In the following theorem, we show that algorithm $\mathcal{G}$ converges exponentially in the mean square sense for any strongly connected graph with symmetric communication probabilities.

Theorem 2 (Convergence of $\mathcal{G}$): For any graph G, which is strongly connected and admits a symmetric weighted adjacency matrix W, the algorithm $\mathcal{G}$ with parameters $\alpha = \beta = \gamma = 1/2$ converges to consensus:

1) *almost surely*;
2) in expectation;
3) in mean square sense.

Proof: We start by proving convergence (2). Let

$$R := \mathbb{E}[U_k] = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_{\text{out}}(i)} w_{ij} Q_{ij}.$$

Since $\mathbb{E}[z(k+1)] = R\mathbb{E}[z(k)]$ from the fact that U_k are independent, we have that

$$\mathbb{E}[z(k+1)] = \mathbb{E}\left[\begin{bmatrix}x(k+1) \\ y(k+1)\end{bmatrix}\right] = R^k z(0) = R^k \begin{bmatrix}x(0) \\ 0\end{bmatrix}$$

and therefore, it suffices to prove that

$$\lim_{k\to\infty} R^k = \frac{1}{n}\begin{bmatrix}\mathsf{1}_n \\ \mathsf{0}_n\end{bmatrix}\begin{bmatrix}\mathsf{1}_n^\intercal & \mathsf{1}_n^\intercal\end{bmatrix} \tag{17}$$

from which we conclude that $\lim_{k\to\infty} \mathbb{E}[x(k+1)] = \mathsf{1}_n x_{\text{av}}$, $x_{\text{av}} = \mathsf{1}_n^\intercal x(0)$.

From (10) and (11), we notice that we can partition R into blocks $R = \left[\begin{smallmatrix}R_1 & R_2 \\ R_3 & R_4\end{smallmatrix}\right]$ where each block is a linear combination of the following three matrices:

$$X = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_{\text{out}}(i)} w_{ij}\Lambda_j, \quad Y = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_{\text{out}}(i)} w_{ij}\Lambda_i$$

$$Z = \sum_{i=1}^{n} \sum_{j \in \mathcal{N}_{\text{out}}(i)} w_{ij}\mathsf{e}_j\mathsf{e}_i^\intercal.$$

It is easy to see that $Z = W^\intercal = W$ (since we assume that matrix W is symmetric) and $Y = I$. Moreover

$$X = \sum_{j=1}^{n} \sum_{i \in \mathcal{N}_{\text{in}}(j)} w_{ij}\Lambda_j = \sum_{j=1}^{n} \Lambda_j = I$$

where we used the fact that $\sum_{i \in \mathcal{N}_{\text{in}}(j)} w_{ij} = 1$, i.e., the sum of weights for the in-neighbors of i equals to one, due to the key assumption that $W : W_{ij} = w_{ij}$ is a doubly stochastic matrix. Therefore, each R_i is a linear combination of the matrices W and I and we can write

$$R = P_1 \otimes I_m + P_2 \otimes W$$

where for $\alpha = \beta = \gamma = \frac{1}{2}$,

$$P_1 = \begin{bmatrix} 1-\frac{1}{2n} & \frac{1}{2n} \\ \frac{1}{2n} & 1-\frac{3}{2n} \end{bmatrix}, \quad P_2 = \begin{bmatrix} \frac{1}{2n} & \frac{1}{2n} \\ -\frac{1}{2n} & \frac{1}{2n} \end{bmatrix}.$$

We denote an eigenvalue of a matrix A by $\lambda_i(A)$ and the set of eigenvalues by $\{\lambda_i(A)\}$. Let $P_S(\delta) := P_1 + \delta P_2$. Then, one can obtain that

$$\lambda_i(P_S(\delta)) = 1 + \frac{\delta - 2 \pm \sqrt{2-\delta^2}}{2n}, \quad i \in \{1, 2\}. \tag{18}$$

Let w_{Pi} be the two eigenvector of $P_S(\delta)$, and v_{Pj} denote the n eigenvectors of W (note that W is symmetric, and therefore, it has n eigenvectors). Then, R has $2n$ eigenvectors $w_{Pi} \otimes v_{Pj}$, since one can show that

$$R(w_{Pi} \otimes v_{Pj}) = \lambda_\ell(R) w_{P_i} \otimes v_{Pj}$$

where the set of eigenvalues of R is given by

$$\{\lambda_\ell(R), 1 \leq \ell \leq 2n\} = \{\lambda_i(P_S(\eta_j)) : \eta_j \in \{\lambda_j(W)\} \\ 1 \leq i \leq 2, 1 \leq j \leq n\}$$

Since W is symmetric and doubly stochastic, and it is a weighted adjacency matrix of a strongly connected and aperiodic graph, the eigenvalues of W are real, W has a simple eigenvalue at 1, and all the remaining eigenvalues belong to the set $(-1, 1)$. Corresponding to the simple eigenvalue 1 of W, R has two eigenvalues at $\{\lambda_i(P_1 + P_2)\} = \{1, 1 - 1/n\}$. Corresponding to the eigenvalues of W that belong to the set $(-1, 1)$, the eigenvalues of R are inside the unit circle. This can be shown by noticing that (18) is a strictly increasing function when $-1 < \delta < 1$ for each i, and using this fact, it is easy to conclude that $r_\sigma(P_1 + \delta P_2) < 1$ for $-1 < \delta < 1$. Thus, R has a single eigenvalue at 1, all the remaining eigenvalues are inside the unit disk, and the vectors $v_R := \left[\begin{smallmatrix} \mathbf{1}_n \\ \mathbf{0}_n \end{smallmatrix}\right]$ and $w_R := \begin{bmatrix} \mathbf{1}_n^\intercal & \mathbf{1}_n^\intercal \end{bmatrix}$ are left and right eigenvalues of R, respectively, associated with this eigenvalue 1. This implies that

$$\lim_{k\to\infty} R^k = \frac{1}{w_R v_R} v_R w_R$$

which is (17).

To prove convergence (3), let us introduce the shorter notation for the minimum and maximum as

$$x_{\text{min}}(k) := \min_\ell x_\ell(k), \;\; x_{\text{max}}(k) := \max_\ell x_\ell(k)$$

and a Lyapunov function

$$V(x(k)) = x_{\text{max}}(k) - x_{\text{min}}(k).$$

Then, we have that $\forall k \geq 0$

$$\begin{aligned} \|x(k) - x_{\text{av}}\mathbf{1}_n\|^2 &= \sum_{\ell=1}^n (x_\ell(k) - x_{\text{av}})^2 \\ &\leq (n-1)V(x(0)) \sum_{\ell=1}^n x_{\max}(k) - x_{\min}(k) \end{aligned} \tag{19}$$

where the inequality in (19) comes from the fact that, given the iteration defined in (8) and (9), any product of matrices Q_{ij} have a constant sum of entries equal to $2n$ and any entry is not larger than 1. Combining these two facts, the maximum difference between two nodes is obtained when the row in the product of matrices Q_{ij} corresponding to $\ell_{\text{max}} := \arg\max_\ell x_\ell(k)$ is

$$\mathbf{e}_{\ell_{\text{max}}}\begin{bmatrix} -(n-1) & \mathbf{1}_{2n-1}^\intercal \end{bmatrix} + \begin{bmatrix} \mathbf{1}_n \\ \mathbf{0}_n \end{bmatrix} * \mathbf{e}_1^\intercal$$

i.e., the $x_{\text{max}}(k) \leq (n-1)(x_{\text{max}}(0) - x_{\text{min}}(0)) \wedge x_{\text{min}}(k) \geq x_{\text{min}}(0)$ (and following the same reasoning, $x_{\text{max}}(k) \leq x_{\text{max}}(0) \wedge x_{\text{min}}(k) \geq x_{\text{min}}(0) - (n-2)(x_{\text{max}}(0) - x_{\text{min}}(0))$ for the case of selecting the row in the product of matrices Q_{ij} corresponding to the $x_{\text{min}}(k)$). In both cases, $V(x(k)) \leq (n-1)V(x(0))$.

Using (19), it follows

$$\mathbb{E}[\|x(k) - x_{\text{av}}\mathbf{1}_n\|^2 | x(0)] \leq (n-1)V(x(0))\mathbb{E}[V(x(k))|x(0)].$$

We shall prove that

$$\mathbb{E}[V(x(k))|x(0)] \leq c\bar{\gamma}^k V(x(0)) \tag{20}$$

for a constant c from which stability in the mean square sense follows, because

$$\mathbb{E}[\|x(k) - x_{\text{av}}\mathbf{1}_n\|^2 | x(0)] \leq (n-1)c\bar{\gamma}^k V(x(0))^2$$

for some positive constant c and $\bar{\gamma} < 1$.

To prove (20), it is sufficient to show that

$$\mathbb{E}[V(x(k+\tau))|x(k)] - \gamma V(x(k)) \leq 0 \tag{21}$$

for time interval of size τ, constant $\gamma < 1$, which relates to $\bar{\gamma}$ through $\gamma^{\frac{k}{\tau}} = \bar{\gamma}^k$, and where $\mathbb{E}[\cdot|\cdot]$ is the conditional expected value operator.

In order to upper bound the expected value in (21), we can define a finite sequence θ, of size τ, such that $\theta_1 = U_{k+1}, \ldots, \theta_\tau = U_{k+\tau}$. Since by assumption the graph G is strongly connected and symmetric, there exists a path of nodes of at most $n-1$ links that go from the maximum to the minimum-value nodes. Let us assume the longest path possible of $n-1$ links and define the random variables $\pi_1, \ldots, \pi_n$ such that $\pi_1(k) = x_{\text{max}}(k)$ and $\pi_n(k) = x_{\text{min}}(k)$ with each $\pi_i(k)$ being the ith node in the path from the maximum and minimum-value nodes at time k.

With the objective of writing $x_{\text{min}}(k+\tau)$ and $x_{\text{max}}(k+\tau)$ with terms that include both $x_{\text{min}}(k)$ and $x_{\text{max}}(k)$, we consider a finite sequence, for the time instant k, $\theta^\star$. This sequence is constructed as follows: $\theta_1^\star = Q_{\pi_1\pi_2}, \theta_2^\star = Q_{\pi_2\pi_1}, \ldots, \theta_{\tau-1}^\star = Q_{\pi_{n-1},\pi_n}, \theta_\tau^\star = Q_{\pi_n,\pi_{n-1}}$, where we omitted the dependence of π on k to improve the readability. Therefore, each $\theta_i^\star$ is also a random variable as it depends on the path given by π. This sequence of updates, of size $\tau = 2(n-1)$ occurs with nonzero probability

$$p_{\text{good}} = \frac{1}{n^{2(n-1)}} \prod_{\ell=1}^{n-1} ([W]_{\pi_\ell,\pi_{\ell+1}})^2$$

as all weights $[W]_{\pi_\ell,\pi_{\ell+1}}$ are nonnegative and $[W]_{\pi_\ell,\pi_{\ell+1}} = [W]_{\pi_{\ell+1},\pi_\ell}$. Computing the product $Q_{\pi_1\pi_2}Q_{\pi_2\pi_1}\ldots Q_{\pi_{n-1},\pi_n}Q_{\pi_n,\pi_{n-1}}x(k)$, the expected value of function $V(\cdot)$ subject to the chosen sequence $\theta^\star$ to occur from time k to $k+\tau$ becomes

$$\begin{aligned}\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^\star] &= \frac{1}{2}x_{\pi_n}(k)+\frac{1}{2}x_{\pi_{n-1}}(k)\\ &\quad -\sum_{\ell=1}^{n-1}\left[\frac{1}{2^\ell}x_{\pi_\ell}(k)\right]\\ &\quad -\frac{1}{2^{n-1}}x_{\pi_n}(k)\end{aligned} \tag{22}$$

where we draw attention for the fact that conditioning on $x(k)$ means that the variable π becomes deterministic. We can upper bound (22) and get

$$\begin{aligned}\mathbb{E}[V(x(k+\tau)|x(k),\theta=\theta^\star] &\le x_{\pi_n}(k)\\ &\quad -\left[\left(1-\frac{1}{2^{n-1}}\right)x_{\pi_1}(k)+\frac{1}{2^{n-1}}x_{\pi_n}(k)\right]\\ &\le \left(1-\frac{1}{2^{n-1}}\right)(x_{\pi_n}(k)-x_{\pi_1}(k))\\ &\le \left(1-\frac{1}{2^{n-1}}\right)V(x(k))\end{aligned}$$

where all the $x_{\pi_\ell}(k)$ inside the summation in (22) were replaced by $x_{\pi_1}(k)$. Let us introduce the notation $\Theta := \{\theta^\star\}\bigcup\Theta^n\bigcup\Theta^b\bigcup\Theta^c$, where Θ is the set of all finite sequences of updates of size τ, Θ^n is a subset of the sequences that do not increase the expected value, Θ^b is the subset of sequences increasing the expected value in at most ϑ for some constant ϑ, and Θ^c is the subset of sequences that decrease of at least ϑ. Sets $\{\theta^\star\},\Theta^n,\Theta^b$, and Θ^c are chosen to be mutually disjoint. Thus, the expected value in (21) can be written as

$$\begin{aligned}\mathbb{E}[V(x(k+\tau))|x(k)] &= \sum_{\theta\in\Theta}p_\theta\mathbb{E}[V(x(k+\tau))|x(k),\theta]\\ &= p_{\text{good}}\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^\star]\\ &\quad +\sum_{\theta^n\in\Theta^n}p_{\theta^n}\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^n]\\ &\quad +\sum_{\theta^b\in\Theta^b}p_{\theta^b}\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^b]\\ &\quad +\sum_{\theta^c\in\Theta^c}p_{\theta^c}\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^c]\end{aligned} \tag{23}$$

where p_θ is the probability of occurring the finite sequence θ out of all possible finite sequences of size τ.

Let us define the random variables $\pi_i^s(k)$ of length ϱ as each representing a node in a sorted path of nodes. All sequences θ^b, of size $\tau=\varrho+1$, are characterized by $\theta_1^b=Q_{\pi_1^s\pi_2^s},\ldots,\theta_{\tau-1}^b=Q_{\pi_{\varrho-1}^s\pi_\varrho^s},\theta_\tau^b=Q_{\varkappa\pi_\varrho^s}$, for some node $\varkappa$ (once again to improve the readability, we omitted the dependence of π^s on k).

We focus on showing that there is an equivalent sequence θ^c with a greater or equal probability and decrease of the function $V(\cdot)$ as that of θ^b. Since matrix W is symmetric, the probability $W_{ij}=W_{ji}$, which means we can reverse paths and maintain the same probability. Also, the selection of matrices Q_{ij} is independent that makes probability of $Q_{i_1j_1}Q_{i_2j_2}$ equal to $Q_{i_2j_2}Q_{i_1j_1}$. We must consider following three cases.

1) $\varkappa=\pi_\varrho^s$—i.e., failed transmission of the last node, which must be the minimum or the maximum.
2) $\varkappa=\pi_{\varrho-1}^s$—i.e., a sequence that ends in the minimum or the maximum.
3) $\varkappa\neq\pi_\varrho^s$—i.e., a communication from a node different than the minimum and maximum.

Let us construct a sequence θ^c, of size $\tau=\varrho+1$ for case 1). Intuitively, the problem with case 1) is that the failed transmission forces the sum of the accumulated y variable with x. For 1), we have $\theta_1^c=Q_{\pi_\varrho^s\pi_\varrho^s},\theta_2^c=Q_{\pi_1^s\pi_2^s},\ldots,\theta_\tau^c=Q_{\pi_{\varrho-1}^s\pi_\varrho^s}$, where we changed the place of the failed transmission. In this case, we are in the same conditions, then 2), which we address next, but for sequences of size $\tau=\varrho$.

For case 2), if $\pi_1^s\in\{x_{\min}(k),x_{\max}(k)\}$, we can construct $\theta_1^c=Q_{\pi_\varrho^s\pi_{\varrho-1}^s},\ldots,\theta_\tau^c=Q_{\pi_2^s\pi_1^s}$, where we reversed the path. In doing so, $p_{\theta^c}=p_{\theta^b}$ and the variation ϑ for θ^c is greater or equal than the variation for θ^b since $\pi_1^s-x_{\text{av}}\ge\pi_\rho^s-x_{\text{av}}$. Intuitively, the bad case was due to nodes above the average contacting the minimum node, which was closer to the average than the maximum, or vice versa. If $\pi_1^s\notin\{x_{\min}(k),x_{\max}(k)\}$, we will have to consider all the sequences π_s of the same length entering π_ϱ^s. Since W is symmetric, all the in-communications links sum to one, and therefore, the probabilities of all sequences π^s for $x_{\min}(k)$ have the same probability as the sequences π^s ending in $x_{\max}(k)$ and the total variation is negative by the same reasoning.

Finally, the construction for case 3) follows $\theta_1^c=Q_{\pi_\varrho^s\pi_{\varrho-1}^s},\ldots,\theta_{\tau-1}^c=Q_{\pi_2^s\pi_1^s},\theta_\tau^c=Q_{\pi_\varrho^s\varkappa}$. The sequence θ^c uses the same communicating pairs of nodes, so it happens with the same probability.

The main consequence is that

$$\sum_{\theta^b\in\Theta^b}p_{\theta^b}\le\sum_{\theta^c\in\Theta^c}p_{\theta^c}.$$

Given that

$$\forall\theta^n\in\Theta^n:\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^n]\le V(x(k))$$
$$\forall\theta^b\in\Theta^b:\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^b]\le V(x(k))+\vartheta$$

and

$$\forall\theta^c\in\Theta^c:\mathbb{E}[V(x(k+\tau))|x(k),\theta=\theta^c]\le V(x(k))-\vartheta$$

it is possible to overbound the terms in θ^n, θ^b, and θ^c in (23) as $(1-p_{\text{good}})V(x(k))$ and get

$$\begin{aligned}\mathbb{E}[V(x(k+\tau))|x(k)] &\le p_{\text{good}}\left(1-\frac{1}{2^{n-1}}\right)V(x(k))\\ &\quad +(1-p_{\text{good}})V(x(k)).\end{aligned} \tag{24}$$

By simplifying (24), we get

$$\mathbb{E}[V(x(k+\tau))|x(k)]\le\left[1-p_{\text{good}}\frac{1}{2^{n-1}}\right]V(x(k))$$

which satisfies (21) for $\gamma=1-p_{\text{good}}\frac{1}{2^{n-1}}$, getting convergence in the mean square sense.

To prove 1) notice that we verified 2) and 3), which means convergence for both the expected value and the expected value of the square occur with an exponential rate. Using the Borel–Cantelli first lemma [31], [32], the sequence converges almost surely.

Remark 2: In Theorem 2, the parameters α, β, and γ were made $\frac{1}{2}$ to alleviate the proofs since the main objective is to show that the algorithm has a possible selection of parameters that achieves the exponential mean square stability for undirected graphs. The study for a general value is a laborious task but follows the same steps.

Let us recall the definition of *disagreement* $\delta(x)$ [33], which is interesting for proving convergence for the broadcast algorithm.

Definition 2 (Disagreement): For any vector $x \in \mathbb{R}^n$, let us define its *disagreement* δ with respect to some norm $||.||$ as

$$\delta(x) = ||x - \mathbf{1}x_{\text{av}}||.$$

In particular, if using the $||.||_\infty$ and introducing the notation $\overline{x} = \max_{i=1,\dots,n} x_i$ and $\underline{x} = \min_{i=1,\dots,n} x_i$, we get

$$\delta(x) = \frac{\overline{x} - \underline{x}}{2}.$$

Definition 2 is particularly important to give properties about the evolution of the state in each iteration, which we introduce in the following definition.

Definition 3 (Nonexpansive and pseudocontraction): A matrix $A \in \mathbb{R}^n$ is said to be *nonexpansive* if

$$||Ax - \mathbf{1}x_{\text{av}}||_\infty \leq ||x - \mathbf{1}x_{\text{av}}||_\infty$$

which is equivalent to say that

$$\delta(Ax) \leq \delta(x)$$

and if the strict inequality holds then the matrix is a *pseudocontraction*.

Definition 4: A *phase* corresponds to an interval of time $[\tilde{k}_\tau, \tilde{k}_{\tau+1})$ such that $\exists k_i^\star \in [\tilde{k}_\tau, \tilde{k}_{\tau+1}) \forall i : 1 \leq i \leq n$, node i transmits at time $k_i^\star$.

The following lemma gives the nonexpansive behavior using the time scale of phases for the algorithm $\mathcal{B}$.

Lemma 1: For $\lambda > 0$, define $S_\lambda = \{z \in \mathbb{R}^n : \delta(z) < \lambda\}$, where z is defined in (3) and satisfy the algorithm specified by (4) and (6)–(9). If $z(0) \in S_\lambda$, then $z(\tilde{k}_\tau) \in S_\lambda \forall \tilde{k}_\tau > 0$ with probability 1. Equivalently

$$\text{Prob}\left[\sup_{0 \leq \tilde{k}_\tau < \infty} \delta(z(\tilde{k}_\tau)) \geq \lambda\right] = 0.$$

Proof: From (6)–(9) and taking all the parameters to be $1/2$, we get that $\forall i \in \mathcal{V}, \forall k > 0 : R_i^k = R_i$. For the base case of a two-node network, and assuming without the loss of generality that we label as node 1 the node that transmitted first, we will get $z(\tilde{k}_1) = R_1^k R_2 z(0) = R_1 R_2 z(0) = \frac{1}{2}\begin{bmatrix}\mathbf{1}_2 \\ \mathbf{0}_2\end{bmatrix}\begin{bmatrix}\mathbf{1}_2^\intercal & \mathbf{1}_2^\intercal\end{bmatrix} z(0)$. This implies $\overline{z}(\tilde{k}_0) \geq \overline{z}(\tilde{k}_1)$, and conversely, $\underline{z}(\tilde{k}_0) \leq \underline{z}(\tilde{k}_1)$, which gives $\delta(z(\tilde{k}_1)) \leq \delta(z(\tilde{k}_0))$. The same reasoning is valid for subsequent $\tilde{k}_\tau$, thus meaning that for a two-node network, we have $\delta(z(\tilde{k}_{\tau+1})) \leq \delta(z(\tilde{k}_\tau))$.

If we assume that $\delta(z(\tilde{k}_{\tau+1})) \leq \delta(z(\tilde{k}_\tau))$ for any τ and a network of n nodes, then let us prove the statement for a network of $n+1$ nodes. Let us label node $n+1$ as the last to transmit for the first time since $\tilde{k}_\tau$. By assumption, all the remaining nodes will have $\delta(z_{-(n+1)}(\tilde{k}_{\tau+1})) \leq \delta(z_{-(n+1)}(\tilde{k}_\tau))$, where the variable $z_{-(n+1)}$ represents all the states except for the one of node $n+1$. Prior to time $\tilde{k}_{\tau+1}$, node $n+1$ state is denoted by $z_{n+1}(\tilde{k}_{\tau+1}^-)$ and in its x component, it has $x_{n+1}(\tilde{k}_{\tau+1}^-) \leq x_{n+1}(\tilde{k}_\tau)$, and in the y component, a value η, which is the difference changed in the x variable to keep the sum of the states constant. See that $\eta < 0$ if $x(0) < x_{\text{av}}$ and nonnegative otherwise. When node $n+1$ had $x_{n+1}(0) < x_{\text{av}}$, this implies that it will decrease the state variable of the remaining nodes on a proportion $\frac{\eta}{n+1}$. Therefore, the quantity $\sum_{i=1}^n x_i - x_{\text{av}} + y_i$ decreases (as the sum of deviation above the average are greater than the deviations below the average when excluding the node $n+1$, which directly implies that $\delta(z(\tilde{k}_{\tau+1})) \leq \delta(z(\tilde{k}_\tau))$. Conversely, it also holds when the node $n+1$ has $x(0) \geq x_{\text{av}}$. Then, by induction, we have the property $\delta(z(\tilde{k}_{\tau+1})) \leq \delta(z(\tilde{k}_\tau))$ for all n, which proves the lemma. ■

Based on Lemma 1, it is possible to state the following theorem regarding the convergence of $\mathcal{B}$.

Theorem 3 (Convergence of $\mathcal{B}$): For any complete graph G, the algorithm $\mathcal{B}$ with parameters $\alpha = \beta = \gamma = 1/2$ converges to consensus:

1) *almost surely*;
2) in expectation;
3) in mean square sense.

Proof: We start by proving convergence in 2) by showing that $r_\sigma(R) \leq 1$. We start by noticing that matrix R in this case can be rewritten as

$$R = P_1 \otimes I_n + P_2 \otimes \frac{\mathbf{1}_n \mathbf{1}_n^\intercal}{n}$$

where

$$P_1 = \begin{bmatrix} 1-\alpha & \frac{-\gamma+(n-1)\beta}{n} \\ \alpha & \frac{n-2}{n-1} + \frac{\gamma-(n-1)\beta}{n} \end{bmatrix}, P_2 = \begin{bmatrix} \alpha & \gamma \\ -\alpha & \frac{1}{n-1} - \gamma \end{bmatrix}.$$

Then, R has two simple eigenvalues in 1 and $1/n$ and two eigenvalues with multiplicity $n-1$ corresponding to

$$\lambda_i(R) = \frac{n - n^2 + 1 \pm \sqrt{n^4 - 4n^3 + 5n^2 - 2n + 1}}{2n(1-n)}. \quad (25)$$

Using the derivatives of this expression for the eigenvalues in (25), we have $\lambda_1 \in [\frac{1}{2}; 1]$ and $\lambda_2 \in [0; \frac{5-\sqrt{13}}{12}]$. Therefore, $r_\sigma(R) \leq 1$, which concludes the proof of convergence in expectation.

To establish 3), let us select time instances as in the Definition 4 of phase and, by Lemma 1, the variable x is pseudocontracting meaning that $x(\tilde{k}) \in S_\lambda, \lambda > 0$ and that the derivative is negative over phase intervals (also see [34] and references therein). An equivalent formulation is that with probability one, we have $\forall \tilde{k} : V(x(\tilde{k})) \leq V(x(0))$.

Given the definition of the function $V(x(k)) := x_{\max}(k) - x_{\min}(k)$, it holds that $\forall k \geq 0$

$$\begin{aligned}\|x(k) - \mathbf{1}_n x_{\text{av}}\|^2 &= \sum_{\ell=1}^{n}(x_\ell(k) - x_{\text{av}})^2 \\ &\leq V(x(0))\sum_{\ell=1}^{n}|x_\ell(k) - x_{\text{av}}| \\ &\leq V(x(0))\sum_{\ell=1}^{n}x_{\max}(k) - x_{\min}(k) \quad (26)\end{aligned}$$

where the aforementioned inequalities come from the definition of maximum and minimum and $\forall \ell \leq n, k \geq 0 : |x_\ell(k) - x_{\text{av}}| \leq x_{\max}(k) - x_{\min}(k)$.

Using (26), it follows that

$$\mathbb{E}[\|x(k) - \mathbf{1}_n x_{\text{av}}\|^2|x(0)] \leq nV(x(0))\mathbb{E}[V(x(k))|x(0)]. \quad (27)$$

However, given the result in 2), we have that

$$\mathbb{E}[V(x(k+1))|x(k)] \leq \zeta V(x(k))$$

for $0 < \zeta < 1$. Therefore, (27) becomes

$$\mathbb{E}[\|x(k) - \mathbf{1}_n x_{\text{av}}\|^2|x(0)] \leq n\zeta^k V(x(0))^2$$

from which the conclusion follows.

The result in 1) is given by the exponential convergence in the mean square sense in 3). In more detail, the Markon's inequality states for a random variable X that

$$\mathbb{P}[X \geq a] \leq \frac{\mathbb{E}[X]}{a}.$$

If we define the error as $e(k) = x(k) - \mathbf{1}x_{\text{av}}$, we can compute

$$\begin{aligned}\lim_{k\to\infty}\mathbb{P}\left[\frac{\|x(k) - \mathbf{1}_n x_{\text{av}}\|}{\|x(0) - \mathbf{1}_n x_{\text{av}}\|} \geq \epsilon\right] &= \lim_{k\to\infty}\mathbb{P}\left[\frac{e(k)^\intercal e(k)}{e(0)^\intercal e(0)} \geq \epsilon^2\right] \\ &\leq \lim_{k\to\infty}\epsilon^{-2}\frac{\mathbb{E}[e(k)^\intercal e(k)]}{e(0)^\intercal e(0)} \\ &= \lim_{k\to\infty}\epsilon^{-2}\zeta^k \\ &= 0\end{aligned}$$

for the $0 < \zeta < 1$ constant found for the convergence in the mean square sense. ∎

V. Convergence Rates

The interesting problem of finding the fastest distributed linear algorithm is addressed and the convergence rates are provided in discrete time. We show that the rates relate to the second largest eigenvalue of the linear combination of the transmission matrices. We start by providing a result available in the literature and showing how both algorithms $\mathcal{G}$ and $\mathcal{B}$ can be seen in that framework.

Definition 5 (ϵ-averaging time): For any $0 < \epsilon < 1$, the ϵ-averaging time, denoted by $t_{\text{avg}}(\epsilon, p)$, of a linear distributed algorithm (4), where $\{U_k, k \geq 0\}$ are characterized by (13) and (14), and randomly chosen from a set $\mathcal{M} := \{B_i, 1 \leq i \leq n_p\}$, where

$$\text{Prob}[U_k = B_i] = p_i, \sum_{i=1}^{n_p} p_i = 1$$

is defined as

$$\sup_{z(0)}\inf\left\{t : \text{Prob}\left[\frac{||z(k) - z_{\text{av}}\mathbf{1}||}{||z(0)||} \geq \epsilon\right] \leq \epsilon\right\}$$

where $||v||$ denotes the l_2 norm of the vector v.

Using the aforementioned definition, we provide the unidirectional version of the bounds found in [20].

Theorem 4 (Convergence in discrete time): The averaging time $t_{\text{avg}}(\epsilon, p)$ (measured in terms of clock ticks) of the linear distributed algorithm, as defined in Definition 5 is bounded by

$$t_{\text{avg}}(\epsilon, p) \leq \frac{3\log\epsilon^{-1}}{\log\lambda_2(R_2)^{-1}}$$

and

$$t_{\text{avg}}(\epsilon, p) \geq \frac{0.5\log\epsilon^{-1}}{\log\lambda_2(R_2)^{-1}}$$

where

$$R_2 = \sum_{i=1}^{n_p} p_i B_i \otimes B_i.$$

Proof: The proof follows from the fact that both algorithms $\mathcal{G}$ and $\mathcal{B}$ can be casted into the formulation of Definition 5, which is the same as [20, Th. 3].x ∎

A. Distributed Optimization

In the previous section, we presented convergence results for the directed gossip algorithm, assuming the second largest eigenvalue of the expected value matrix. In practical applications, either we postulate that a network designer had access to the entire topology and was able to compute the second largest eigenvalue of the expected value (if the network is large such assumption becomes unreasonable and defeats the purpose of a distributed solution) or that the nodes at some point in time optimize their convergence rate. The current section focusses on addressing the issue for the latter case.

The question of optimizing the second largest eigenvalue can be reformulated to a semidefinite program in the symmetric case, which can be solved in a distributed setting by many off-the-shelf solvers. The nonsymmetric matrices Q_{ij} render the problem not convex. We now describe how the objective of minimizing the second largest eigenvalue can be separated into two problems, i.e., minimizing the eigenvalues of the probability matrix, and then, selecting the optimal values for the parameters using a nonconvex technique.

Theorem 5 (Distributed optimization): The directed gossip algorithm $\mathcal{G}$ for a system of the form (4) with the linear iteration as in (10) can be cast as a distributed convex optimization for the communication probabilities in matrix W and as a nonconvex for parameters α, β, and γ.

Proof: When optimizing for matrix W, we are interested in solving the following optimization problem:

$$\begin{aligned} \text{minimize} \quad & \lambda_2(R) \\ \text{subject to} \quad & R = \sum_{i,j=1}^{n} \frac{1}{n} W_{ij} Q_{ij} \\ & W_{ij} \geq 0,\ W_{ij} = 0,\ \text{if } \{i,j\} \notin E \\ & W\mathbf{1}_n = \mathbf{1}_n. \end{aligned}$$

However, notice that we used the fact that $\lambda_i(P_S(\delta))$ is a monotonically increasing function with $\lambda_2(W)$ to prove the convergence, which allows us to rewrite the problem as

$$\begin{aligned} \text{minimize} \quad & t \\ \text{subject to} \quad & W - \mathbf{1}_n\mathbf{1}_n^\intercal \preceq tI_n \\ & W_{ij} \geq 0,\ W_{ij} = 0,\ \text{if } \{i,j\} \notin E \\ & W\mathbf{1}_n = \mathbf{1}_n. \end{aligned}$$

Let us introduce for each directed link (the optimization can be carried out for nonsymmetric matrices W), a new variable $\eta_k, 1 \leq k \leq |E|$ and a correspondent flow matrix $F_k = -(e_i - e_j)(e_i - e_j)^\intercal$ where the pair $\{i,j\}$ is our kth link. The optimization can be written in the distributed form

$$\begin{aligned} \text{minimize} \quad & \lambda_2\left(I_n + \sum_{k=1}^{|E|} \eta_k F_k\right) \\ \text{subject to} \quad & L\mathbf{1}_n = \mathbf{1}_n \\ & \eta_k \geq 0,\ 1 \leq k \leq |E| \end{aligned}$$

where the matrix L is just to short the notation and has $L_{ij} = \eta_k$ for the corresponding k to the vertex $\{i,j\}$ and zeros elsewhere. Using the standard epigraph variable techniques and due to the fact that $\lambda(I_n + A) = 1 + \lambda(A)$

$$\begin{aligned} \text{minimize} \quad & t \\ \text{subject to} \quad & \sum_{k=1}^{|E|} \eta_k F_k - \mathbf{1}_n\mathbf{1}_n^\intercal \preceq tI_n \\ & L\mathbf{1}_n = \mathbf{1}_n \\ & \eta_k \geq 0,\ 1 \leq k \leq |E|. \end{aligned}$$

The formulation of the problem has separated optimization variables, which can be performed in a distributed fashion using techniques such as alternating direction method of multipliers [35] or other techniques (see [36] and [37] and references therein).

Regarding parameters α, β, and γ, the optimization is nonconvex and can be carried out using the brute force both for the eigenvalues of the expectation and mean square matrices using the sufficient and necessary conditions presented in Theorem 1. ■

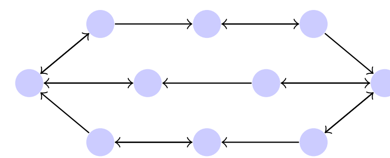


Fig. 1. Communication graph with different out-neighbor degrees.

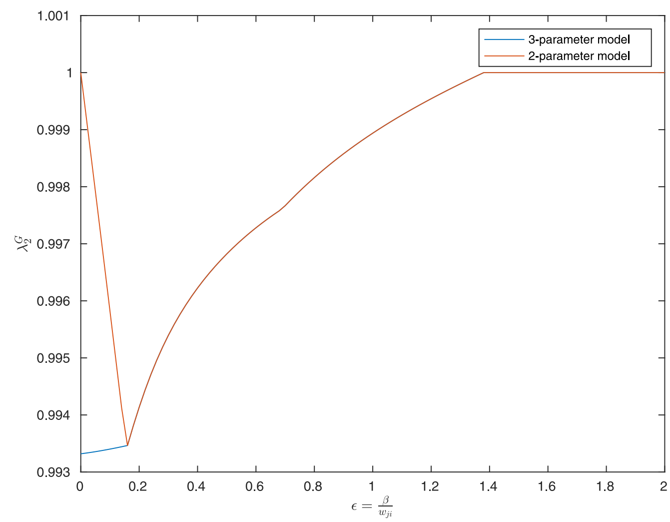


Fig. 2. Evolution of the second largest eigenvalue of the mean square matrix as a function of ϵ and β for the algorithm in [21] (two-parameter) and the algorithm $\mathcal{G}$ (three-parameter).

B. Comparison Between Unidirectional and Bidirectional Cases

In the previous section, we showed how to optimize the probabilities and parameters of the gossip algorithm and how to distribute that computation over the nodes of the network. Using the previous optimization, it is possible to show the difference between the convergence rates of this study and that in [21]. The initial example in [21] is recovered next.

In Fig. 1 is depicted the ten node digraph used in [21] with the selected mixing parameter of $w = \frac{1}{2}$. All edges are assumed to have equal probability of being activated with a constant distribution over time. Algorithm $\mathcal{G}$ is also simulated setting $\alpha = w$ and $\beta = \epsilon w$ and the optimized γ value. The second largest eigenvalue of the mean square expected value is presented for both cases in Fig. 2. From this example, it shows that for larger values of ϵ, the two algorithms become the same for this directed graph and choice of w.

In order to assess the convergence rate for other examples, an undirected topology obtained by making all the edges in Fig. 1 bidirectional is simulated. The correspondent evolution of the second largest eigenvalue is depicted in Fig. 3. Similarly to the previous example, the same trends emerge where the algorithm $\mathcal{G}$ outperforms (in this case, the difference is higher) the proposal in [21]. Figs. 2 and 3 also illustrate the fact that the current proposal has as worst-case convergence rate the values for the algorithm in [21].

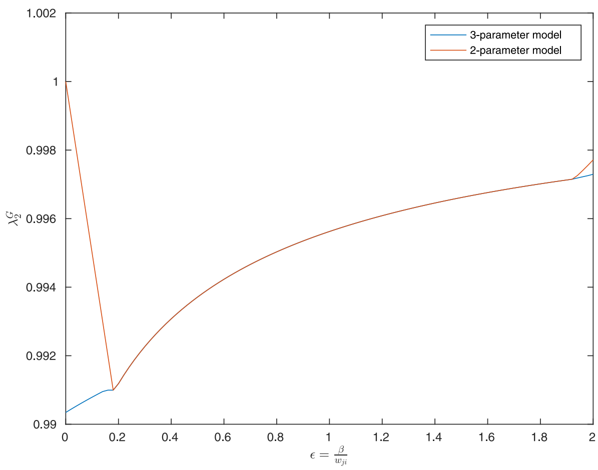


Fig. 3. Evolution of the second largest eigenvalue of the mean square matrix as a function of ϵ and β for the algorithm in [21] (two-parameter) and the algorithm $\mathcal{G}$ (three-parameter).

TABLE I
SECOND LARGEST EIGENVALUE FOR THE BIDIRECTIONAL (B_λ_2) AND THE PRESENTED UNIDIRECTIONAL (U_λ_2) ALGORITHMS FOR THE THREE STUDIED CASES

	Expectation		Second Moment	
Case	b_λ_2	u_λ_2	b_λ_2	u_λ_2
Best	0.75	0.76	0.75	0.9153
Average	0.9401	0.9625	0.9401	0.97
Worst	0.9618	0.9805	0.9618	0.983

TABLE II
UPPER AND LOWER BOUNDS FOR THE MEAN SQUARE ON THE NUMBER OF TICKS FOR THE ALGORITHMS TO REACH A NEIGHBORHOOD OF THE SOLUTION OF $\epsilon = 10^{-2}$ FOR THE BIDIRECTIONAL CASE (B_TICKS) AND THE PRESENTED UNIDIRECTIONAL (U_TICKS) ALGORITHMS

	Lower bound		Upper bound	
Case	b_ticks	u_ticks	b_ticks	u_ticks
Best	8	26.02	48.02	156.1
Average	37.28	75.6	223.66	453.57
Worst	59.12	134.29	354.71	805.75

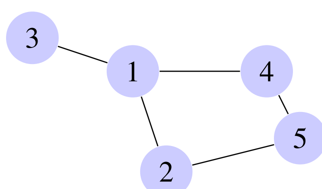


Fig. 4. Communication graph with different out-neighbor degrees.

It is also interesting to compare how the convergence rate is affected when going from bidirectional to unidirectional gossip randomized algorithms. In the sequel, we present results about three communication graphs with different out-neighbor degree to give a general overview.

In selecting different cases to illustrate how the second largest eigenvalue of the matrix of the expected value and second moment varies with parameters choice, we took into consideration what should be the best- and worst-case scenarios and an *average case*, where we are by no means stating that the chosen case is an average case since our aim is to give an example with different out-neighbor degrees. Fig. 4 presents the graph, which we called as the average case example. The best-case scenario is when connectivity is at its maximum (i.e., each node can communicate to every other node) and the worst case is when node i is connected to nodes $i-1$ and $i+1$ except for node 1 and m, which connect to only one neighbor. For all the examples, we take the number of nodes $n = 5$.

Provided that the nodes optimize the matrix of probabilities W, we get the following results:

$$W_{\text{best}} = \frac{1}{n-1}(\mathbf{1}_n \mathbf{1}_n^\intercal - I_n)$$

$$W_{\text{worst}} = \frac{1}{2}(\text{tri}(n) - I_n + \mathsf{e}_1 \mathsf{e}_1^\intercal + \mathsf{e}_n \mathsf{e}_n^\intercal)$$

$$W_{\text{average}} = \begin{bmatrix} 0 & 0.2835 & 0.433 & 0.2835 & 0 \\ 0.2835 & 0.2165 & 0 & 0 & 0.5 \\ 0.4330 & 0 & 0.567 & 0 & 0 \\ 0.2835 & 0 & 0 & 0.2165 & 0.5 \\ 0 & 0.5 & 0 & 0.5 & 0.0000 \end{bmatrix}$$

where $\text{tri}(m)$ is a tridiagonal matrix of size m with the elements in the three main diagonals all equal to 1.

Using the computed matrix W for the probabilities, we calculate the second largest eigenvalue for both expectation and second moment, which are presented in Table I for the three considered cases. Those values were obtained by searching in a brute force fashion for $\alpha, \beta, \gamma \in [0, 1]$, which minimized λ_2. Regard, however that the minimum for the expectation and second moment were not obtained jointly since one may wish to optimize for one or the other.

In order to give a better perspective about the values in Table I, let us compute the upper and lower bound of clock ticks so that the system is in a neighborhood ϵ of the solution x_{av}. Such bounds were provided in [20], although see references therein for additional information. The convergence rate in continuous time is provided in [29, Th. 9 and Corollary 10]. It is important to notice that, in reality, a bidirectional algorithm is using two communication steps in each transmission so the values presented in Table II for the bidirectional case should be seen in a unit of measure, which is double from the unidirectional case.

VI. CONCLUSION

In this chapter, the problem of studying the convergence of the state of an average consensus algorithm with unidirectional communications is tackled. The motivation behind constructing an asynchronous and unidirectional algorithm was to better map the characteristics of wireless networks. In doing so, the algorithm can progress to the average of the initial values even in a realistic scenario with a high packet drop rate as it can use the received information instead of having to wait for a successful two-way communication.

We first provide results to test the convergence for a specific instance of the connectivity graph for a generic algorithm obeying the definitions for the interactions. These relate to determining if the spectral radius of the matrix defining the expected value and the second moment is the unit circle. It is then shown that convergence holds for any connectivity graph that is symmetric.

Selecting the fastest converging algorithm for the average consensus problem is also presented in this chapter. By noticing that the spectral radius depends monotonically with the second largest eigenvalue of the expected value matrix, allowed us to first rewrite that optimization as a semidefinite program, and then, optimize in a brute force fashion for the parameters of the algorithm. The convergence rate is compared to both the unidirectional and bidirectional case.

Acknowledgment

The authors would like to thank D. Antunes for his contribution during the initial steps that led to this study.

References

[1] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. 31, no. 9, pp. 803–812, Sep. 1986.

[2] B. Johansson, T. Keviczky, M. Johansson, and K. Johansson, "Subgradient methods and consensus algorithms for solving convex optimization problems," in *Proc. 47th IEEE Conf. Decision Control*, Dec. 2008, pp. 4185–4190.

[3] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, vol. 48, no. 6, pp. 988–1001, Jun. 2003.

[4] J. Cortes, S. Martinez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Trans. Autom. Control*, vol. 51, no. 8, pp. 1289–1298, Aug. 2006.

[5] M. Chiang, S. Low, A. Calderbank, and J. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[6] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *Proc. 46th IEEE Conf. Decision Control*, Dec. 2007, pp. 5492–5498.

[7] P. Alriksson and A. Rantzer, "Experimental evaluation of a distributed Kalman filter algorithm," in *Proc. 46th IEEE Conf. Decision Control*, Dec. 2007, pp. 5499–5504.

[8] Z. Zhang and M. Y. Chow, "Convergence analysis of the incremental cost consensus algorithm under different communication network topologies in a smart grid," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 1761–1768, Nov. 2012.

[9] M. Andreasson, D. V. Dimarogonas, H. Sandberg, and K. H. Johansson, "Distributed control of networked dynamical systems: Static feedback, integral action and consensus," *IEEE Trans. Autom. Control*, vol. 59, no. 7, pp. 1750–1764, Jul. 2014.

[10] Y. Chen, J. Lu, X. Yu, and D. J. Hill, "Multi-agent systems with dynamical topologies: Consensus and applications," *IEEE Circuits Syst. Mag.*, vol. 13, no. 3, pp. 21–34, Thirdquarter 2013.

[11] R. Hegselmann and U. Krause, "Opinion dynamics and bounded confidence: Models, analysis and simulation," *J. Artif. Soc. Social Simul.*, vol. 5, no. 3, Jun. 2002.

[12] S. Evans and G. Patterson, "The synchronization of behaviour in flocks of estrildine finches," *Animal Behav.*, vol. 19, no. 3, pp. 429–438, 1971.

[13] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, 1990.

[14] F. Mónus and Z. Barta, "Degree of synchronization of tree sparrows in flocks under different predation risk," *Behaviour*, vol. 148, no. 5, pp. 733–744, 2011.

[15] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[16] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks* (Applied Mathematics Series). Princeton, NJ, USA: Princeton Univ. Press, 2009. [Online]. Available: http://coordinationbook.info

[17] S. Patterson, B. Bamieh, and A. El Abbadi, "Convergence rates of distributed average consensus with stochastic link failures," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 880–892, Apr. 2010.

[18] F. Fagnani and S. Zampieri, "Average consensus with packet drop communication," *SIAM J. Control Optim.*, no. 48, pp. 102–133, 2009.

[19] R. Carli, F. Bullo, and S. Zampieri, "Quantized average consensus via dynamic coding/decoding schemes," *Int. J. Robust Nonlinear Control*, vol. 20, no. 2, pp. 156–175, 2010.

[20] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[21] K. Cai and H. Ishii, "Average consensus on general strongly connected digraphs," *Automatica*, vol. 48, no. 11, pp. 2750–2761, 2012.

[22] J. Liu and A. Morse, "Asynchronous distributed averaging using double linear iterations," in *Proc. Amer. Control Conf.*, Jun. 2012, pp. 6620–6625.

[23] K. Cai and H. Ishii, "Quantized consensus and averaging on gossip digraphs," *IEEE Trans. Autom. Control*, vol. 56, no. 9, pp. 2087–2100, Sep. 2011.

[24] A. Tahbaz-Salehi and A. Jadbabaie, "A necessary and sufficient condition for consensus over random networks," *IEEE Trans. Autom. Control*, vol. 53, no. 3, pp. 791–795, Apr. 2008.

[25] B. Touri and A. Nedic, "Product of random stochastic matrices," *IEEE Trans. Autom. Control*, vol. 59, no. 2, pp. 437–448, Feb. 2014.

[26] F. Benezit, V. Blondel, P. Thiran, J. Tsitsiklis, and M. Vetterli, "Weighted gossip: Distributed averaging using non-doubly stochastic matrices," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2010, pp. 1753–1757.

[27] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. 44th IEEE Symp. Found. Comput. Sci.*, Oct. 2003, pp. 482–491.

[28] A. Dominguez-Garcia and C. Hadjicostis, "Distributed strategies for average consensus in directed graphs," in *Proc. 50th IEEE 50th Conf. Decision Control Eur. Control Conf.*, Dec. 2011, pp. 2124–2129.

[29] D. Antunes, D. Silvestre, and C. Silvestre, "Average consensus and gossip algorithms in networks with stochastic asymmetric communications," in *Proc. 50th IEEE Conf. Decision Control*, Dec. 2011, pp. 2088–2093.

[30] F. Fagnani and S. Zampieri, "Randomized consensus algorithms over large scale networks," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 4, pp. 634–649, May 2008.

[31] E. Borel, "Les probabilités dénombrables et leurs applications arithmetiques," *Rend. Circ. Mat. Palermo*, vol. 27, pp. pp. 247–271, 1909.

[32] F. P. Cantelli, "Sulla probabilità come limite della frequenza," *Atti Accad. Naz. Lincei*, vol. 26, no. 1, pp. 39–45, 1917.

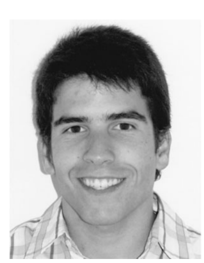
[33] M. Porfiri and D. Stilwell, "Consensus seeking over random weighted directed graphs," *IEEE Trans. Autom. Control*, vol. 52, no. 9, pp. 1767–1773, Sep. 2007.

[34] L. Moreau, "Stability of continuous-time distributed consensus algorithms," in *Proc. IEEE 43rd Conf. Decision Control*, Dec. 2004, vol. 4, pp. 3998–4003.

[35] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multiplier," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[36] S. Boyd, P. Diaconis, and L. Xiao, "Fastest mixing Markov chain on a graph," *SIAM Rev.*, vol. 46, pp. 667–689, 2003.

[37] A. S. Lewis, "Convex analysis on the Hermitian matrices," *SIAM J. Optimization*, vol. 6, pp. 164–177, 1996.



Daniel Silvestre received the B.Sc. degree in computer networks and the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico, Lisbon, Portugal, in 2008 and 2017, respectively, and the M.Sc. degree in advanced computing from the Imperial College London, London, U. K., in 2009.

He is currently a Postdoc with the University of Macau, Macau, China. His research interests include the fields of fault detection and isolation, distributed system, networks, and randomized algorithms.

João P. Hespanha (S'95–M'98–SM'02–F'08) received the Ph.D. degree in electrical engineering and applied science from Yale University, New Haven, CT, USA, in 1998.

From 1999 to 2001, he was an Assistant Professor with the University of Southern California, Los Angeles, CA, USA. He moved to the University of California, Santa Barbara, CA, USA, in 2002, where he is currently a Professor with the Department of Electrical and Computer Engineering. He is the Chair of the Department of Electrical and Computer Engineering and a Member of the Executive Committee for the Institute for Collaborative Biotechnologies. His current research interests include hybrid and switched systems, multiagent control systems, distributed control over communication networks (also known as networked control systems), the use of vision in feedback control, stochastic modeling in biology, and network security.

Dr. Hespanha was the recipient of the Yale University's Henry Prentiss Becton Graduate Prize for exceptional achievement in research in Engineering and Applied Science, a National Science Foundation CAREER Award, the 2005 Best Paper Award at the 2nd International Conference on Intelligent Sensing and Information Processing, the 2005 Automatica Theory/Methodology Best Paper prize, the 2006 George S. Axelby Outstanding Paper Award, and the 2009 Ruberti Young Researcher Prize. He was an IEEE Distinguished Lecturer from 2007 to 2013.

Carlos Silvestre received the Licenciatura degree in electrical engineering, the M.Sc. degree in electrical engineering, the Ph.D. degree in control science, and the Habilitation degree in electrical engineering and computers, all from the Instituto Superior Tecnico (IST), Lisbon, Portugal, in 1987, 1991, 2000, and 2011, respectively.

Since 2000, he has been with the Department of Electrical Engineering, IST, where he is an Associate Professor of control and robotics, currently on leave. Since 2015, he has been a Professor with the Department of Electrical and Computers Engineering, Faculty of Science and Technology, University of Macau, Macau, China. Over the past years, he has conducted research on the subjects of navigation guidance and control of air and underwater robots. His research interests include linear and nonlinear control theory, hybrid control, sensor-based control, coordinated control of multiple vehicles, networked control systems, and fault detection and isolation.