

Asynchronous Coordination of Distributed Energy Resources for the Provisioning of Ancillary Services

Christoforos N. Hadjicostis and Themistoklis Charalambous

Abstract—This paper analyzes the stability of asynchronous coordination protocols and studies their application to energy resource provisioning of ancillary services in power grids. First, we consider and analyze a protocol that can be used to asymptotically reach average consensus in an asynchronous mode of operation. Once we establish the convergence properties of this asynchronous consensus protocol, we demonstrate its effectiveness by applying it towards energy resource provisioning of ancillary services in a power grid.

I. INTRODUCTION

The presence of numerous components that can potentially provide energy and energy-related services opens up exciting possibilities for efficiently managing these resources to provide ancillary services, in order to improve the performance and stability of the grid they are connected to [1]. Ancillary services are support services in the power system, which are essential for transmission capacity support and of vital importance for reliability and security of the grid. Examples of ancillary services include reactive power support and black start capability [2].

The problem of resource coordination and control for ancillary services provision has been previously addressed in a centralized fashion (e.g. [3]). However, a centralized solution requires a coordinator to gather all necessary data and calculate appropriate actions for each component; indirectly, this requires the coordinator to be aware of all relevant activity in the system (which might be changing over time) and also implies the existence of a communication network through which data will be obtained and commands will be disseminated. Thus, a centralized approach is feasible in small networks, but prohibitively restrictive in the case of large-scale power grids, due to computational limitations and connectivity restrictions. Towards this end, an alternative approach has been proposed in [4] that involves distributed strategies for control and coordination of energy resources. One particular strategy of [4] requires that two different iterative algorithms are executed (simultaneously or consecutively) by all nodes, in a way that allows each node to determine in a distributed fashion its contribution. This approach, however, requires synchronization between the components and is susceptible to network changes that

occur before the components have a chance to settle on their contribution. This paper develops techniques to handle delays that might occur in the messages that are exchanged during the process, essentially enabling the asynchronous operation of these strategies. This is important because the energy resources of the power grid do not necessarily share a global clock and communicate over channels in which messages may be delayed. Hence, it is important that a strategy to coordinate energy resources accounts for such realistic effects and facilitates asynchronous operation.

The approach and results presented in this paper are motivated by recent advancements in distributed control for networks of dynamic agents with linear coupling. In particular, the problem of distributed agreement in such systems considers and analyzes appropriate protocols for all agents to reach agreement on a certain quantity of interest. A typical discrete-time consensus control algorithm is provided by [5]. Switching among interaction topologies is investigated in [6], where it is proven that consensus can be achieved asymptotically if the union of the interaction topologies has a *spanning tree*¹ frequently enough as the system evolves. Consensus in the presence of time-delays for such systems has also been studied (see for example [8], [9]) and it has been shown that consensus is reached on *uniformly quasi-strongly connected graphs*.²

The main difference of the work in this paper from the aforementioned work is that the strategies we develop reach consensus to the average of the nodes' initial values, which is important for the coordination of distributed energy resources. We show that by running two iterations simultaneously, the nodes are able to asymptotically reach average consensus (and subsequently, a certain desirable steady-state), whatever the delays (e.g. computation delays, communication delays, memory access delays, software delays) in the multi-agent system, as long as these delays are bounded. Thus, the major difference from previous work is that the consensus value reached does not depend on the structure and nature of the delays.

This result is extremely useful in a wide variety of potential applications and its importance is demonstrated via a realistic scenario for distributed energy resource provisioning for ancillary services in power grids. The advantages of

This material is based upon work supported in part from the European Commission (EC) 7th Framework Programme (FP7/2007-2013), under grant agreements INFOS-ICT-223844 and PIRG02-GA-2007-224877. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of EC. The authors are with the Department of Electrical and Computer Engineering at the University of Cyprus, Nicosia, Cyprus. Corresponding author's address: 75 Kallipoleos Avenue, P.O.Box 20537, 1678 Nicosia, Cyprus. E-mail: {themis, chadjic}@ucy.ac.cy.

¹A graph (representing in this case the interaction topology between agents in a multi-agent system) contains a spanning tree if and only if it contains at least one node with a directed path to all other nodes (more details in [7]).

²A graph is uniformly quasi-strongly connected if the union graph over a bounded period of time contains a spanning tree (more details in [7]).

this strategy over those being suggested in [4] are the following: (i) it does not require synchronization, and (ii) it does not need to restart the protocol when connectivity (topology) changes (as long as each node is aware of the links available in its immediate neighborhood). The suggested strategy requires only local communication among neighbouring nodes and decisions are based exclusively on available information. Thus, the proposed strategy facilitates the Smart Grid initiative [10] by enabling active participation by consumers in demand response.

The remainder of the paper is organised as follows. In Section II, we provide necessary notation and background on graph properties. In Section III, we present our main results and propose an algorithm, whose properties are illustrated via simple examples. The derived theoretical results are applied toward distributed energy resource provisioning for ancillary services in Section IV. Finally, Section V presents concluding remarks and future directions.

II. NOTATION AND PRELIMINARIES

The sets of real, integer and natural numbers are denoted by \mathbb{R} , \mathbb{Z} and \mathbb{N} , respectively; their nonnegative counterparts are denoted by the subscript $+$ (e.g. \mathbb{R}_+). Vectors are denoted by small letters whereas matrices are denoted by capital letters. A^T denotes the transpose of matrix A . By $\mathbf{1}$ we denote the all-ones vector and by I we denote the identity matrix (of appropriate dimensions). A matrix whose elements are nonnegative, called nonnegative matrix, is denoted by $A \geq 0$ and a matrix whose elements are positive, called positive matrix, is denoted by $A > 0$.

Let the exchange of information between nodes (where resources are located) be modeled by a weighted directed graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, P)$ of order n ($n \geq 2$), where $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$ is the set of nodes, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges, and $P = [p_{ji}] \in \mathbb{R}_+^{n \times n}$ is a weighted adjacency matrix where p_{ji} are nonnegative elements. A directed edge from node v_i to node v_j is denoted by $\varepsilon_{ji} = (v_j, v_i) \in \mathcal{E}$ and represents that node v_j can receive information from node v_i . By convention and for notational purposes, we assume that the graph does not include any self-loops (i.e., $\varepsilon_{jj} \notin \mathcal{E}$ for all $v_j \in \mathcal{V}$) although each node v_j has obviously access to its own information. If $p_{ji} > 0$, $j \neq i$, then the directed edge $\varepsilon_{ji} \in \mathcal{E}$ (i.e., the adjacency matrix has zero entries at locations that do not correspond to directed links in the graph). The graphs we will be exploring will have adjacency matrices with positive weights on the diagonal and at each entry that corresponds to a link in the graph (so that the zero-nonzero structure of the adjacency matrix matches exactly the given set of links in the graph). A graph is said to be undirected if and only if $\varepsilon_{ji} \in \mathcal{E}$ implies $\varepsilon_{ij} \in \mathcal{E}$.

All nodes that can transmit information to node v_j directly are said to be in-neighbors of node v_j and belong to the set $\mathcal{N}_j^- = \{v_i \in \mathcal{V} : \varepsilon_{ji} \in \mathcal{E}\}$. The cardinality of \mathcal{N}_j^- , $|\mathcal{N}_j^-|$, is called the *in-degree* of v_j and is denoted by \mathcal{D}_j^- . The nodes that receive information from node v_j belong to the set of out-neighbors of node v_j , denoted by $\mathcal{N}_j^+ = \{v_i \in \mathcal{V} : \varepsilon_{ij} \in$

$\mathcal{E}\}$. The cardinality of \mathcal{N}_j^+ , $|\mathcal{N}_j^+|$, is called the *out-degree* of v_j and it is denoted by \mathcal{D}_j^+ .

A directed graph is called *strongly* connected if there exists a path from each vertex v_i in the graph to each vertex v_j . In other words, for any $v_j, v_i \in \mathcal{V}$, one can find a sequence of nodes $v_i = v_{j_1}, v_{j_2}, v_{j_3}, \dots, v_{j_t} = v_j$ such that link $(v_{j_{k+1}}, v_{j_k}) \in \mathcal{E}$ for all $k = 1, 2, \dots, t - 1$. A *union graph* $\mathcal{G}_{1, \dots, m}$ of a collection of graphs $\mathcal{G}_1, \dots, \mathcal{G}_m$, with the same vertex set \mathcal{V} for some $m \geq 1$ is defined as the graph with vertex set \mathcal{V} and edge set equalling the union of the edge sets of all of the graphs in the collection. The collection of graphs is said to be *jointly strongly connected* if its corresponding union graph forms a strongly connected graph. A strongly connected graph certainly emerges if at least one of the graphs in the collection is strongly connected, but it could also emerge even if none of the graphs forming the union is strongly connected.

Each node v_j updates and sends its information to its neighbors at discrete times t_0, t_1, t_2, \dots . We index nodes' information states and any other information at time t_k by k . We use $x_j[k] \in \mathbb{R}$ to denote the information state of node j at time t_k . When there exist no delays in the communication links, each node updates its information state $x_j[k]$ by combining the available information received by its neighbors $x_i[k]$ ($v_i \in \mathcal{N}_j^-$) using the positive weights $p_{ji}[k]$, that capture the weight of the information inflow from agent v_i to agent v_j at time k . In this work, we assume that each node v_j can choose its self-weight and the weights on its out-going links \mathcal{N}_j^+ only. Hence, in its general form, each node updates its information state according to the following relation:

$$x_j[k+1] = p_{jj}[k]x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p_{ji}[k]x_i[k]. \quad (1)$$

for $k \geq 0$, where $x_j[0] \in \mathbb{R}$ is the initial state of node v_j . If we let $x[k] = (x_1[k] \ x_2[k] \ \dots \ x_n[k])^T$ and $P[k] = [p_{ji}[k]] \in \mathbb{R}_+^{n \times n}$, then (1) can be written in matrix form as

$$x[k+1] = P[k]x[k], \quad (2)$$

where $x[0] = (x_1[0] \ x_2[0] \ \dots \ x_n[0])^T \equiv x_0^T$. (Note that unspecified weights in $P[k]$ are set to zero and correspond to pairs of nodes (v_j, v_i) that are not connected, i.e., $p_{ji}[k] = 0$, $\forall \varepsilon_{ji} \notin \mathcal{E}$.)

In this work, we consider a static network (as it is usually the case for distributed resources in applications such as the power grid) and hence the graph remains invariant. In this case, the weights can be chosen to be constant for all times k (i.e., $p_{ji}[k] = p_{ji} \ \forall k$), and equation (2) can be expressed as $x[k+1] = Px[k]$, where $P[k] = P \in \mathbb{R}_+^{n \times n}$. Note that, with the exception of the diagonal entries, we have $p_{ji} = 0$, $j \neq i$, if and only if $(v_j, v_i) \notin \mathcal{E}$. We say that the nodes asymptotically reach average consensus if $\lim_{k \rightarrow \infty} x_j[k] = \frac{\sum_i x_i[0]}{n}$ for all $v_j \in \mathcal{V}$. The necessary and sufficient conditions for (2) (with $P[k] = P$) to reach average consensus are the following: (a) P has a simple eigenvalue $\lambda_1(P) = 1$ with left eigenvector $\mathbf{1}^T$ and right eigenvector

1, and (b) all other eigenvalues of P ($\lambda_j(P), j \neq i$) have magnitude less than 1 ($|\lambda_j(P)| < 1$). If $P \geq 0$ (as in our case), the necessary and sufficient condition is that P is a primitive doubly stochastic matrix. In an undirected graph, assuming each node knows n (or an upper bound n') and the graph is connected, each node v_j can distributively choose the weights on its outgoing links to be $\frac{1}{n'}$ and set its diagonal to be $1 - \frac{D_j^+}{n'}$, so that the resulting P is primitive doubly stochastic. However, in a directed graph, this choice does not necessarily lead to a doubly stochastic weight matrix.

In [4], an algorithm is suggested that solves the average consensus problem in a directed graph in which each node v_j distributively sets the weights on its self-link and outgoing-links to be $\frac{1}{1+D_j^+}$ so that the resulting weight matrix P is column stochastic, but not necessarily row stochastic. More specifically, average consensus is reached by running two iterations with appropriately chosen initial conditions. The algorithm is stated below.

Lemma 1. [4] Consider a strongly connected graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, P)$ with a weight matrix P that is nonnegative, column stochastic, and has zeros only at entries (j, i) for which we have $(v_j, v_i) \notin \mathcal{E}$. Let $\hat{x}_j[k]$ and $\check{x}_j[k]$ be the result of the iteration

$$x_j[k+1] = p_{jj}x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p_{ji}x_i[k], \quad \forall v_j \in \mathcal{V}, \quad (3)$$

with initial conditions $\hat{x}[0] = x_0$ and $\check{x}[0] = \mathbf{1}$. Then, the solution to the average consensus problem can be asymptotically obtained as the $\lim_{k \rightarrow \infty} \mu_j[k]$, where

$$\mu_j[k] = \frac{\hat{x}_j[k]}{\check{x}_j[k]}, \quad \forall v_j \in \mathcal{V}. \quad (4)$$

A generalization of the iteration suggested in [4] in which the weight matrices are time-varying is presented in [11]. In this paper, we address the average consensus problem over directed graphs in the presence of bounded delays in the communication links. We employ a protocol where each node updates its information state $x_j[k+1]$ by combining the available (possibly delayed) information received by its neighbors $x_i[s]$ ($s \in \mathbb{Z}, s \leq k, v_i \in \mathcal{N}_j^-$) using constant positive weights p_{ji} .

We use the integer $\tau_{ji}[k] \geq 0$ to represent the delay of a message sent from node v_i to node v_j at time instant k . We require that $0 \leq \tau_{ji}[k] \leq \bar{\tau}_{ji} \leq \bar{\tau}$ for all $k \geq 0$ for some finite $\bar{\tau} = \max\{\bar{\tau}_{ji}\}, \bar{\tau} \in \mathbb{Z}_+$. We make the reasonable assumption that $\tau_{jj}[k] = 0, \forall v_j \in \mathcal{V}$, at all time instances k (i.e., the own value of a node is always available without delay). With this notation at hand, each node updates its information state according to the following relation:

$$x_j[k+1] = p_{jj}x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} \sum_{r=0}^{\bar{\tau}} p_{ji}x_i[k-r]I_{k-r,ji}[r], \quad (5)$$

for $k \geq 0$, where $x_j[0] \in \mathbb{R}$ is the initial state of node j , $p_{ji} > 0$ and

$$I_{k,ji}(\tau) = \begin{cases} 1, & \text{if } \tau_{ji}[k] = \tau, \\ 0, & \text{otherwise.} \end{cases}$$

In the absence of delay, we have $\tau_{ji}[k] = 0$ and the update relation (5) reduces to (1) with constant weights. In what follows, we describe an augmented representation of such systems with delays, which we then use to establish that the distributed coordination algorithm will lead to asymptotic average consensus, regardless of the nature and order of the delays, as long as they are bounded.

III. MODELING DELAYS

When there exist no delays, a network of multi-agent systems can be easily captured by a graph where nodes represent the agents and links represent the communication links between them. The interaction among agents is characterized by a positive weight on their links. In the literature, many researchers (e.g. [8]) have considered a link removed at some time k , in case there is a communication failure at that time slot/instant. In this work, we add extra states, as “virtual” nodes and use them to model the delays. The maximum number of extra stages for each node is bounded by the maximum delay $\bar{\tau}$. Hence, for each node $v_j \in \mathcal{V}$ we introduce $\bar{\tau}$ “virtual” nodes $(v_j^{(1)}, v_j^{(2)}, \dots, v_j^{(\bar{\tau})})$. At each time step k , virtual node $v_j^{(\tau)}$ holds the sum of the values that are destined for node v_j and will arrive at it in τ steps. As a result, the augmented graph has $(\bar{\tau} + 1)|\mathcal{V}|$ nodes and $(1 + 2\bar{\tau})|\mathcal{E}|$ edges.

Before presenting the general case, we present the structure of our model via an illustrative example. Consider the network of two agents exchanging information as shown in Figure 1. Suppose the agents experience delays that are bounded by 2 ($\bar{\tau} = 2$). Therefore, two extra states will be added for each node depicting the states at which the delayed messages reside before reaching their destination.

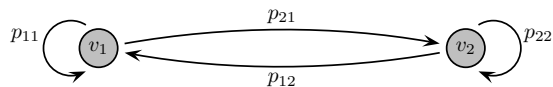


Fig. 1. A simple example with two nodes when the links do not experience any delays.

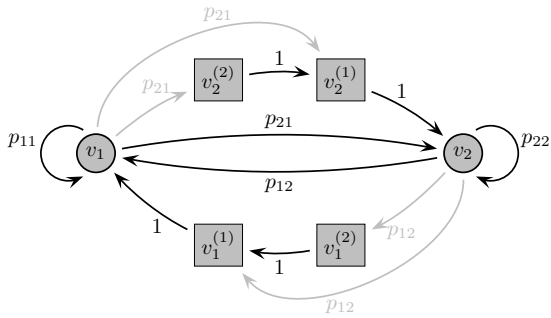
In the augmented graph the iteration can be written as

$$\bar{x}[k+1] = A[k]\bar{x}[k], \quad (6)$$

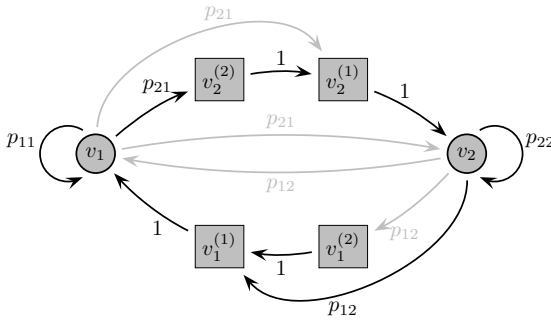
where

$$\bar{x}[k] = (x_1[k] \ x_2[k] \ x_1^{(1)}[k] \ x_2^{(1)}[k] \ x_1^{(2)}[k] \ x_2^{(2)}[k])^T.$$

Suppose there are no delays in the network at time instant $k = q_1$, so that the network can be represented by Figure 2(a). In this case, the weight matrix is



(a) Graph representation of the network at a time instant $k = q_1$ when there exist no delays. As a result, each node uses the value sent by its neighbor directly, plus the delayed information (sent in previous time instances) that arrives at time instant $k = q_1$.



(b) Graph representation of the network at a time instant $k = q_2$ for which node v_2 sends information to node v_1 with delay $\tau_{12} = 1$, while node v_1 sends information to node v_2 with delay $\tau_{21} = 2$.

Fig. 2. A simple example with two nodes to illustrate the modeling of delays using the proposed augmented graph. The maximum allowable delay ($\bar{\tau}$) is 2. In Figure (a) there exist no delays between communication links at that time instant ($k = q_1$), whereas in Figure (b) both nodes experience delays.

$$A[q_1] = \begin{pmatrix} p_{11} & p_{21} & 1 & 0 & 0 & 0 \\ p_{12} & p_{22} & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Likewise, if we assume that at a time instant $k = q_2$ (the network is shown in Figure 2(b)), the matrix representation is given by

$$A[q_2] = \begin{pmatrix} p_{11} & 0 & 1 & 0 & 0 & 0 \\ 0 & p_{22} & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & p_{21} & 0 & 0 & 0 & 1 \\ p_{12} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

In the general case, in a network of $n = |\mathcal{V}|$ nodes, we introduce $\bar{\tau}n$ nodes (for a total of $(\bar{\tau} + 1)n$ nodes) so that

$$\bar{x}[k+1] = A[k]\bar{x}[k], \quad (7)$$

where

$$A[k] \triangleq \begin{pmatrix} P_0[k] & I_{n \times n} & 0 & \cdots & 0 \\ P_1[k] & 0 & I_{n \times n} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_{\bar{\tau}-1}[k] & 0 & 0 & \cdots & I_{n \times n} \\ P_{\bar{\tau}}[k] & 0 & 0 & \cdots & 0 \end{pmatrix}, \quad (8)$$

with

$$\bar{x}[k] = \left(x^T[k] \ x^{(1)}[k] \ \cdots \ x^{(\bar{\tau})}[k] \right)^T,$$

$$x^{(r)}[k] = \left(x_1^{(r)}[k] \ \cdots \ x_n^{(r)}[k] \right), \quad r \in \mathbb{N}, r \leq \bar{\tau}.$$

Note that $P_0[k], P_1[k], \dots, P_{\bar{\tau}}[k]$ are appropriately defined nonnegative matrices, such that

$$P = \sum_{r=0}^{\bar{\tau}} P_r[k].$$

At any given time step k , matrix $A[k]$ may take any of $(\bar{\tau} + 1)^{|\mathcal{E}|}$ matrix values, where $(\bar{\tau} + 1)$ is the total number of states (“virtual” and real nodes) for each node v_j . Specifically, if there exists an edge (v_j, v_i) in the original graph, then that edge also exists in the augmented graph along with edges $(v_j^{(1)}, v_i), (v_j^{(2)}, v_i), \dots, (v_j^{(\bar{\tau})}, v_i)$. However, among the corresponding entries of $A[k]$ only one of these $\bar{\tau} + 1$ entries could be nonzero (and equal to p_{ji}); the others will be zero. Note that in the sequel we do not require that the matrix $A[k]$ is known at each time step k ; what we argue is that $A[k]$ will be a matrix from a finite set of possible matrices \mathcal{A} .

Assumptions 1. For the analysis below we are given a graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, P)$ (that represents the information exchange between agents in a multi-agent system). Each node $v_j \in \mathcal{V}$ has an initial value $x_j[0]$ and runs the iteration in (5). We make the following assumptions:

(A1) The graph is strongly connected, and the (nonnegative) weights p_{ji} are nonzero for $j = i$ and $(v_j, v_i) \in \mathcal{E}$, and satisfy $\sum_{j=1}^n p_{ji} = 1$ (so that they form a column stochastic matrix P).

(A2) There exists a finite $\bar{\tau}$ that uniformly bounds the delay terms; i.e. $\tau_{ji}[k] \leq \bar{\tau} < \infty$ for all links $(v_j, v_i) \in \mathcal{E}$ at time instant k . In addition $\tau_{jj}[k] = 0$ for all $v_j \in \mathcal{V}$ and all k .

Note that assumption (A1) is necessary for the successful operation of any distributed algorithm seeking consensus. Assumption (A2) implies that no message is lost in the network, the total delay in any communication link is uniformly bounded, and every agent updates its value using values from its in-neighbors at least once every $\bar{\tau}$ consecutive updates.

Let A_1, A_2, \dots, A_k be any square matrices of the same order. By a word (in the A 's) of length $\ell \in \mathbb{N}$ we mean the product of ℓ A 's (repetitions permitted). A Stochastic, Indecomposable, and Aperiodic (SIA) matrix is a column stochastic matrix B such that $\lim_{k \rightarrow \infty} B^k$ exists and has all columns the same.

Proposition 1. Let $A[k]$ be defined as in (8). Also, let $\mathcal{A} = \{A_1, A_2, \dots, A_{(\bar{\tau}+1)|\varepsilon|}\}$, such that $A[k] \in \mathcal{A}$. Then, any word B given by $B = A[k]A[k+1] \dots A[k+\ell]$ for some nonnegative integer ℓ is SIA.

Proof: In order to prove that B is SIA, we have to show that it is (i) column stochastic, (ii) indecomposable, and (iii) aperiodic.

(i) This is easy to see as it is equivalent to proving that the product of two column stochastic matrices of the same order is also a column stochastic matrix. (ii) Since all links receive nonnegative weights and the diagonals are strictly positive, indecomposability depends on the structure of the union graph that corresponds to the collection of graphs corresponding to the matrices involved in the word B . Since any link (v_j, v_i) in the original graph is maintained as a path from node v_i to node v_j (it is possible that this path has length greater than unity), the real nodes in the union graph form a strongly connected component. The remaining (virtual) nodes could be part of this strongly connected components (e.g., if matrices that correspond to delays are introduced in the product B) but cannot form another strongly connected component because all of their paths have to eventually end to a real node. The existence of a single strongly connected component in the union graph implies that the matrix product B is indecomposable. (iii) Aperiodicity of the graph is established due to the fact that the diagonal entries that correspond to real nodes are nonzero.

For the derivation of the main results we will need the theorem by Wolfowitz [12] below.

Theorem 1. [12] Let $\mathcal{A} = A_1, A_2, \dots, A_{(\bar{\tau}+1)|\varepsilon|}$ be a collection of stochastic matrices of the same order such that any word in the A 's is stochastic, indecomposable, and aperiodic (SIA). For any $\epsilon > 0$ there exists an integer $\nu_4(\epsilon)$ such that any word B (in the A 's) of length $\ell \geq \nu_4(\epsilon)$ satisfies $\delta(B) < \epsilon$, where $\delta(B) = \max_j \max_{i_1, i_2} |B_{ji_1} - B_{ji_2}|$.

In words, Theorem 1 states that for large enough ℓ , the product of ℓ matrices from the collection \mathcal{A} has all of its columns approximately the same. Note that the result does not mean that the matrix product converges to a single matrix of the form $c\mathbf{1}^T$; however, for large enough ℓ , each word B will take the form $c_B\mathbf{1}^T$ for some column vector c_B . The following lemma makes direct use of the theorem by Wolfowitz and extends Lemma 1 to include the case for which there exist delays in the system between the communication links.

Lemma 2. Let $y_j[k]$, $\forall v_j \in \mathcal{V}$, be the result of iteration (5) with initial conditions $y[0] = x_0$, and let $z_j[k]$, $\forall v_j \in \mathcal{V}$, be the result of iteration (5) with initial condition $z[0] = \mathbf{1}$. Then, the solution to the average consensus problem can be obtained as the $\lim_{k \rightarrow \infty} \mu_j[k]$, where

$$\mu_j[k] = \frac{y_j[k]}{z_j[k]}, \quad \forall v_j \in \mathcal{V}.$$

Proof: If we use the augmented graph representation

with initial conditions $\bar{y}_0 = [x_0^T \ 0 \ 0 \ \dots \ 0]^T$ and $\bar{z}_0 = [\mathbf{1}^T \ 0 \ 0 \ \dots \ 0]^T$, after ℓ steps ($\ell \geq \nu(\epsilon)$), the resulting word B can be represented by

$$\begin{aligned} A_{i_\ell} \dots A_{i_2} A_{i_1} &\rightarrow c_{B_\ell} \mathbf{1}^T, \\ A_{i_{\ell+1}} A_{i_\ell} \dots A_{i_2} A_{i_1} &\rightarrow A_{i_{\ell+1}} c_{B_\ell} \mathbf{1}^T = c_{B_{\ell+1}} \mathbf{1}^T, \end{aligned}$$

where $A_{i_k} \in \mathcal{A}$ and $i_k \in \{1, 2, \dots, (\bar{\tau}+1)|\varepsilon|\}$ at time instant k . Hence, for each iteration at time $k \geq \ell$ we have

$$\begin{aligned} \bar{y}_k &= A_{i_{(k-1)}} \dots A_{i_2} A_{i_1} \bar{y}_0 \rightarrow c_{B_k} \mathbf{1}^T \bar{y}_0, \\ \bar{z}_k &= A_{i_{(k-1)}} \dots A_{i_2} A_{i_1} \bar{z}_0 \rightarrow c_{B_k} \mathbf{1}^T \bar{z}_0. \end{aligned}$$

As a result, the ratio of the values of the two iterations at real node v_j is given by

$$\begin{aligned} \mu_k(j) &= \frac{\bar{y}_k(j)}{\bar{z}_k(j)} = \frac{c_{B_k}(j)(\mathbf{1}^T \bar{y}_0)}{c_{B_k}(j)(\mathbf{1}^T \bar{z}_0)} \\ &= \frac{\sum_{v_j \in \mathcal{V}} \bar{y}_0(j)}{\sum_{v_j \in \mathcal{V}} \bar{z}_0(j)} = \frac{\sum_{v_j \in \mathcal{V}} x_j[0]}{|\mathcal{V}|}. \end{aligned}$$

Note that $\mu_k(j)$, $v_j \in \mathcal{V}$, is identical to $\mu_j[k]$ defined in the lemma and always a finite quantity for all $k \geq 0$ (because at each real nodes $v_j \in \mathcal{V}$, we have $\bar{z}_k(j) > 0$, $\forall k \geq 0$). This is not necessarily true at the virtual nodes.

A. Illustrative example

Consider the directed network in Figure 3 where each node v_j chooses its weight and the weight of its outgoing links to be $(1 + \mathcal{D}_j^+)^{-1}$ (such that the sum of all weight assigned by each node v_j to its out-going links is equal to 1).

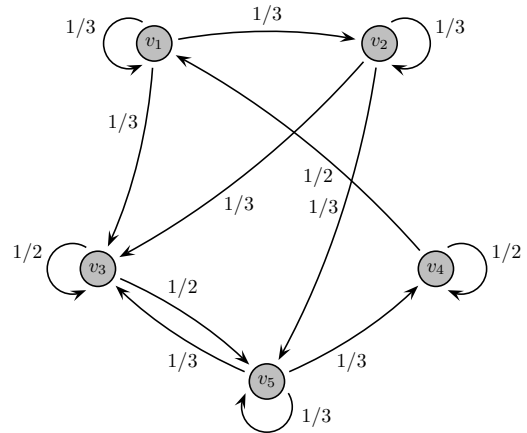


Fig. 3. A simple example of five nodes when the links do not experience any delays.

When each node updates its information state $x_j[k]$ using equation (3), the information state for the whole network is given by $x[k+1] = Px[k]$, where

$$P = \begin{pmatrix} 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 1/3 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/2 & 0 & 1/3 \\ 0 & 0 & 0 & 1/2 & 1/3 \\ 0 & 1/3 & 1/2 & 0 & 1/3 \end{pmatrix}.$$

We first use the update formula (1) with

$$x[0] = (-1 \ 2 \ 3 \ 4 \ 2)^T$$

and with no delays ($\bar{\tau} = 0$). Since the update matrix is column stochastic, the iteration (1) for this network converges, but not necessarily to the average (as shown in Figure 4).

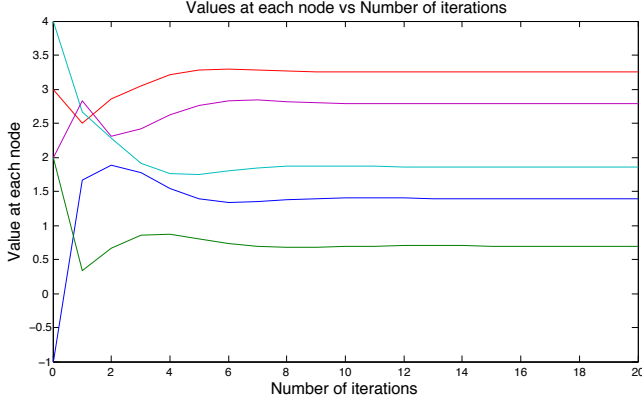


Fig. 4. The iteration of (1) for the network in Figure 3 does not lead to average consensus for the directed graph, since P is not a doubly stochastic matrix.

As suggested in [4], by simultaneously running two iterations $y[k]$ and $z[k]$ with initial conditions $y[0] = x_0$ and $z[0] = \mathbf{1}$, respectively, then average consensus is asymptotically reached for the ratio $y_j[k]/z_j[k]$ (see Figure 5), without requiring matrix P to be doubly stochastic.

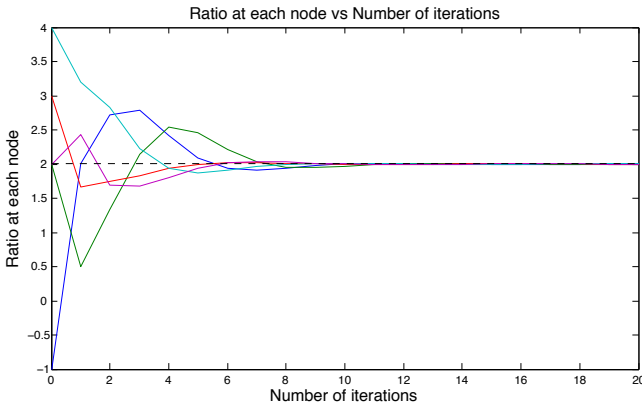


Fig. 5. By running two iterations of (1) with different initial conditions, we reach average consensus (no delays are introduced yet).

In [8], [13], the following update formula is suggested

$$x_j[k+1] = p_{jj}x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p_{ji}x_i[k - d_{ji}[k]], \quad (9)$$

where $d_{ji}[k]$ is chosen so that node v_j uses in its update the most recently seen value from node v_i (i.e., $d_{ji}[k] = \min_{\tau_{ji}[k-t]=t, 0 \leq t \leq \bar{\tau}} \{t\}$). It was shown that this type of iteration asymptotically converges regardless of the delays introduced, as long as they are bounded. However, the update

formula does not necessarily lead to average consensus and the convergence point depends on the nature and structure of the delays. This can be seen if we apply this strategy to our example by allowing the delay to vary with maximum delay $\bar{\tau} = 5$. Then, the information states of the agents in the system do not necessarily converge to the average (see Figure 6).

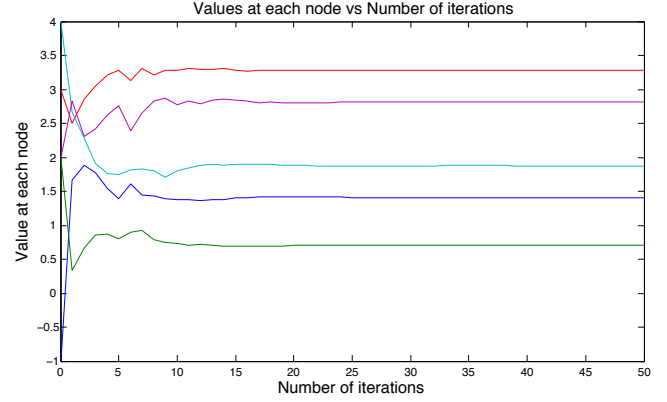


Fig. 6. The iteration suggested in [8], [13] converges for the directed graph in the presence of delays, but it does not converge to the average consensus, as shown in the aforementioned references.

Also, if we apply the same update rule twice, as in [4], then the ratio converges to a single value, but not necessarily to the average consensus (as shown in Figure 7), as someone would expect.

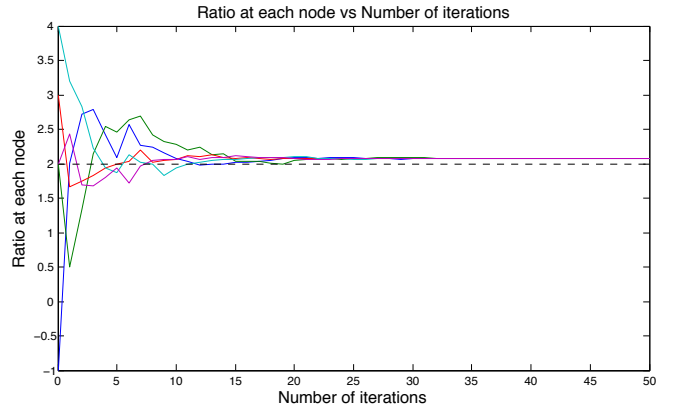


Fig. 7. The iteration converges for the directed graph in the presence of delays to a consensus value that is generally different from the average and depends on the nature and structure of the delays.

If we run our update formula (as in (5)) for the same network (Figure 3) with $x[0] = x_0$, $x[0] \in \mathbb{R}^n$, the algorithm does not converge (Figure 8). However, if we use our update formula (5) twice, then, their ratios reach average consensus (Figure 9), thus demonstrating the validity of our theoretical results.

Summarizing, by simultaneously running two iterations $y[k]$ and $z[k]$ as in (5), with initial condition $y[0] = x_0$ and $z[0] = \mathbf{1}^T$, respectively, then average consensus is

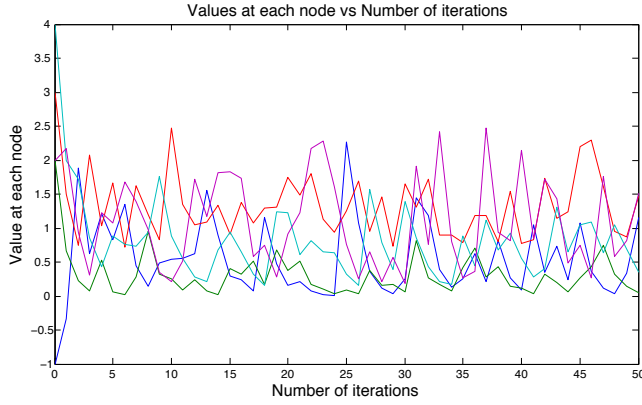


Fig. 8. The update rule (5) does not converge at all for the directed graph, due to the existence of delays.

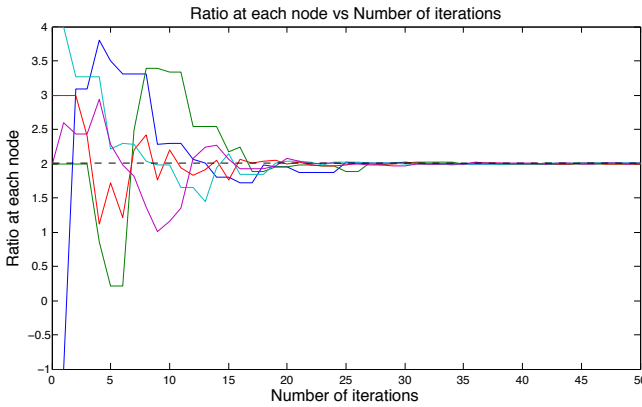


Fig. 9. By running two iterations of the update rule (5) with different initial conditions, we can reach average consensus even in the presence of delays.

asymptotically reached for the ratio $y_j[k]/z_j[k]$, provided the graph characterizing the network is strongly connected, even in the presence of delays in the communication between nodes, as long as these delays are bounded.

IV. DISTRIBUTED ENERGY RESOURCE PROVISIONING FOR ANCILLARY SERVICES

In this section, we apply our results to a realistic scenario for distributed energy resource provisioning for ancillary services, where communication between resources can be delayed for various reasons, such as computational processing time and communication delays. We use $\pi_j[k] \in \mathbb{R}_+$ to be the amount of active (or reactive) power demanded from the distributed resource located at node v_j at time t_k of information exchange between nodes. Let $\pi_j^{\max} \forall j \in \{1, 2, \dots, n\}$, $n = |\mathcal{V}|$, be the maximum active (or reactive) power that node v_j can provide. Also, let $\pi[k] \triangleq (\pi_1[k] \ \pi_2[k] \ \dots \ \pi_n[k])^T$ and $\pi^{\max} \triangleq (\pi_1^{\max} \ \pi_2^{\max} \ \dots \ \pi_n^{\max})^T$, where $\pi[k], \pi^{\max} \in \mathbb{R}_+^n$. The problem being targeted is to find a distributed algorithm with which each resource v_j (as soon as consensus is “almost” reached, say after m steps) chooses its state $\pi_j[m]$,

$0 \leq \pi_j[m] \leq \pi_j^{\max}$, based on locally available information, such that the total power demanded ρ_d is equal to the total amount of power offered in the network, provided the total capacity of power available is greater than the demanded one, i.e.,

$$\rho_d = \sum_{v_j \in \mathcal{V}} \pi_j[m], \quad (10a)$$

such that

$$0 \leq \pi_j[m] \leq \pi_j^{\max}, \quad \forall v_j \in \mathcal{V}, \quad (10b)$$

$$0 \leq \rho_d \leq \sum_{v_j \in \mathcal{V}} \pi_j^{\max} \triangleq \chi^{\max}. \quad (10c)$$

A. Centralised Feasible Splitting

It is noted in [4] that a simple, feasible solution π^\dagger to the problem is given by

$$\pi_j^\dagger = \frac{\rho_d}{\chi^{\max}} \pi_j^{\max}, \quad \forall v_j \in \mathcal{V}, \quad (11)$$

which is always less than the capacity π_j^{\max} of power resource at v_j . However, for each node v_j to be assigned π_j^\dagger , knowledge of χ^{\max} and ρ_d is required at each node v_j and these values have to be computed in a distributed fashion.

B. Distributed Strategies for Splitting

We propose an algorithm and we present its limitations when communication between resources is captured by a directed graph. The algorithm is fully distributed and operates by running a double iteration, one for the power demanded ($\pi_j[k]$) and one for the capacity ($\pi_j^{\max}[k]$). The algorithm can operate in an asynchronous mode and no synchronization is required between the resources. Each node updates its demanded amount $\pi_j[k]$ (or $\pi_j^{\max}[k]$) by combining it with the available (possibly delayed) demanded amount of its neighbors, $\pi_i[s]$ (or $\pi_i^{\max}[k]$) $s \in \mathbb{Z}, s \leq k, v_i \in \mathcal{N}_j^-$, according to the following relation:

$$\pi_j[k+1] = \frac{1}{1 + \mathcal{D}_j^+} \pi_j[k] + \sum_{v_i \in \mathcal{N}_j^-} \sum_{r=0}^{\bar{\tau}} \frac{1}{1 + \mathcal{D}_i^+} \pi_i[k-r] I_{k-r, j_i}[r], \quad (12)$$

for $k \geq 0$, where $\pi_j[0] \in \mathbb{R}_+$ (or $\pi_j^{\max}[0] \in \mathbb{R}_+$) is the initial state of node v_i . In the absence of a delay, we have $\tau_{ji}[k] = 0$ and the update relation (5) reduces to an algorithm for the consensus problem without delays, as in [4]. By defining

$$\mu_j[k] = \frac{\pi_j[k]}{\pi_j^{\max}[k]},$$

after we run the two iterations, we have (according to Lemma 2)

$$\lim_{k \rightarrow \infty} \mu_j[k] = \lim_{k \rightarrow \infty} \frac{\pi_j[k]}{\pi_j^{\max}[k]} = \frac{\sum_{v_j \in \mathcal{V}} \pi_j[0]}{\sum_{v_j \in \mathcal{V}} \pi_j^{\max}[0]} = \frac{\rho_d}{\chi^{\max}}.$$

As a result, the amount of power offered in the network from node v_j is given by

$$\lim_{k \rightarrow \infty} \mu_j[k] \pi_j^{\max} = \frac{\rho_d}{\chi_{\max}} \pi_j^{\max}. \quad (13)$$

The power offered is found in a distributed fashion and it is equal to the feasible solution noted in (11).

C. Example

Consider again the network in Figure 3, with demanded amount, given by $\pi[0] = (1 \ 2 \ 3 \ 4 \ 5)^T$ and capacity $\pi^{\max}[0] = (3 \ 3 \ 3 \ 4 \ 4)^T$, where $\pi[0], \pi^{\max}[0] \in \mathbb{R}_+^n$.

By running the two iterations according to equation (12) when the maximum delay $\bar{\tau} = 5$, the network of resources reaches a consensus which is given by $\rho_d/\chi^{\max} = 15/17$.

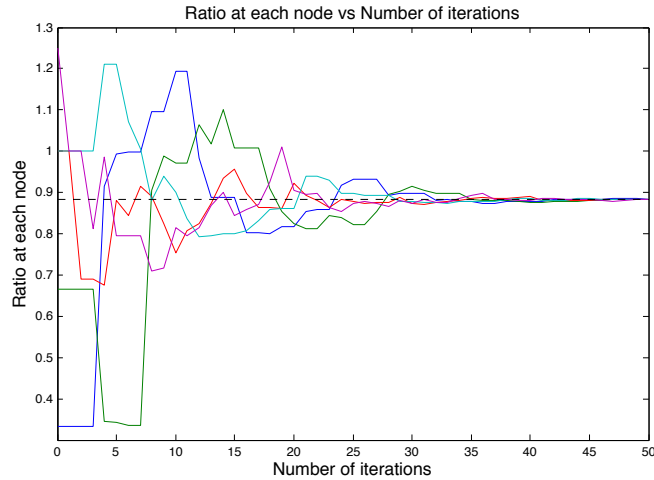


Fig. 10. The ratio of the two iterations converges to the ratio found in (10b).

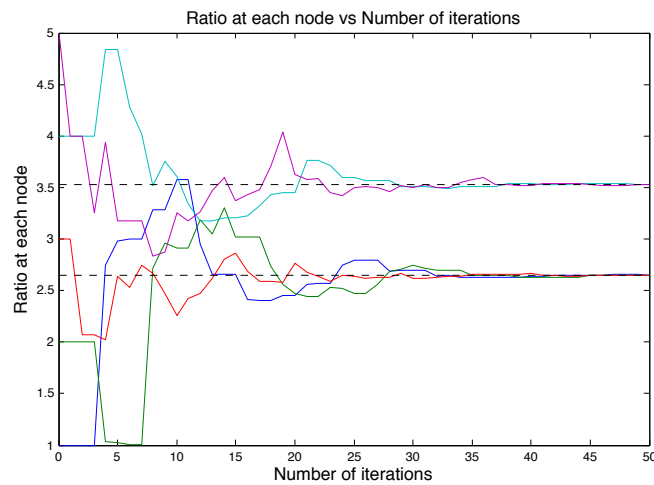


Fig. 11. The amount of power offered in the network by each node according to (13).

The amount of power offered in the network by each node is proportional to the capacity of each resource. In this case,

we have resources with two different capacities: three of them have capacity equal to 3, and two of them have capacity equal to 4. As a result, two of the values converge to $60/17$ and three of them to $45/17$, as shown in Figure 11.

V. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, by modeling the delays in a multi-agent system using an augmented graphical model, we have shown that a proposed asynchronous discrete-time distributed strategy reaches asymptotic average consensus in a distributed fashion, whatever the nature and structure of the delays, as long as they are bounded. This result can find application in a variety of areas, including power grids. Making use of the derived theoretical results for discrete-time average consensus algorithms, we have developed asynchronous discrete-time control strategies that determine, in a distributed fashion, the amount of active (or reactive) power that has to be provided by active (or reactive) power resources with capacity constraints.

Future work includes the study of the behavior of the algorithm in the presence of faults (e.g., broken communication links and nodes not updating their value) and malicious nodes in the multi-agent system.

REFERENCES

- [1] G. Joos, B. Ooi, D. McGillis, F. Galiana, and R. Marceau, "The potential of distributed generation to provide ancillary services," in *IEEE Power Engineering Society Summer Meeting*, vol. 3, 2000, pp. 1762–1767.
- [2] A. G. Isemonger, "The viability of the competitive procurement of Black Start: Lessons from the RTOs," *The Electricity Journal*, vol. 20, no. 8, pp. 60–67, October 2007.
- [3] M. Baran and I. El-Markabi, "A multiagent-based dispatching scheme for distributed generators for voltage support on distribution feeders," *IEEE Transactions on Power Systems*, vol. 22, no. 1, pp. 52–59, 2007.
- [4] A. D. Domínguez-García and C. N. Hadjicostis, "Coordination and control of distributed energy resources for provision of ancillary services," in *First IEEE International Conference on Smart Grid Communications*, 2010, pp. 537–542.
- [5] A. Jadbabaie, J. Lin, and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.
- [6] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, 2005.
- [7] C. Godsil and G. Roy, *Algebraic Graph Theory*. Springer, New York, USA, 2000.
- [8] L. Fang and P. Antsaklis, "Information consensus of asynchronous discrete-time multi-agent systems," in *Proceedings of American Control Conference*, 2005, pp. 1883–1888.
- [9] D. Angeli and P. Bliman, "Stability of leaderless discrete-time multi-agent systems," *Mathematics of Control, Signals, and Systems*, vol. 18, no. 4, pp. 293–322, 2006.
- [10] S. M. Amin and B. Wollenberg, "Toward a smart grid: Power delivery for the 21st century," *IEEE Power and Energy Magazine*, vol. 3, no. 5, pp. 34–41, 2005.
- [11] F. Bénézit, V. Blondel, P. Thiran, J. Tsitsiklis, and M. Vetterli, "Weighted gossip: Distributed averaging using non-doubly stochastic matrices," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, June 2010, pp. 1753–1757.
- [12] J. Wolfowitz, "Products of indecomposable, aperiodic, stochastic matrices," *Proceedings of the American Mathematical Society*, vol. 15, pp. 733–737, 1963.
- [13] B. Lubachevsky and D. Mitra, "A chaotic asynchronous algorithm for computing the fixed point of a nonnegative matrix of unit spectral radius," *J. ACM*, vol. 33, pp. 130–150, January 1986.