Average Consensus in the Presence of Delays in Directed Graph Topologies

Christoforos N. Hadjicostis, *Senior Member, IEEE* and

Themistoklis Charalambous, *Member, IEEE*

Abstract

Classical distributed algorithms for asymptotic average consensus typically assume timely and reliable exchange of information between neighboring components of a given multi-component system. These assumptions are not necessarily valid in practical settings due to varying delays that might affect computations at different nodes and/or transmissions at different links. In this work, we propose a protocol that overcomes this limitation and, unlike existing consensus protocols in the presence of delays, ensures asymptotic consensus to the exact average, despite the presence of arbitrary (but bounded) delays in the communication links. The protocol requires that each component has knowledge of the number of its out-neighbors (i.e., the number of components to which it can send information) and its proof of correctness relies on the weak convergence of a backward product of column stochastic matrices. The proposed algorithm is demonstrated via illustrative examples.

Keywords: Average consensus, digraphs, bounded delays, ratio consensus, weak convergence.

I. INTRODUCTION

A distributed system or network consists of a set of components (nodes) that can share information with neighboring components via connection links (edges), forming a generally

directed interconnection topology (digraph). The objective of a consensus problem is to have all agents agree upon a certain (*a priori* unknown) quantity of interest that is typically a function of some values that the nodes initially posses (initial values). When the agents (asymptotically) reach agreement to the same value, we say that the distributed system (asymptotically) reaches consensus. A special case of (asymptotic) consensus is the case of (asymptotic) average consensus, where the additional challenge is for the nodes to converge to the exact average of their initial values (see, for example, [2], [3]). It has been shown in [4] that, under a fixed interconnection topology, average consensus can be reached by performing a linear iteration in a distributed fashion, i.e., by having each node update its value as a linear combination of its own value and the values of its neighbors. This requires the interconnection topology to be strongly connected and the weights to form a balanced matrix (in continuous time) or a doubly stochastic matrix (in discrete time).

Common challenges that arise in consensus problems include the handling of node failures (e.g., due to the draining of batteries in wireless sensor networks), computational and/or transmission delays on the transfer of data between agents, packet losses in wireless communication networks, and inaccurate sensor measurements. As a result, agreement problems in networks of dynamical agents, possibly with directed information flow, have been successfully developed to operate under disturbances due to delays (e.g., [5]), packet drops (e.g., [6]), changing interconnections (e.g., [7], [8]), or a combination of them (e.g., [9], [10]). In most of these approaches, nodes reach consensus to a value that is *a priori* unknown (and might depend on the disturbance, e.g., the nature and profile of the delays). What is different in this paper is that we devise a protocol that is able to overcome delays while reaching asymptotic consensus to the *exact* average of the values that the nodes initially posses. Among existing algorithms that guarantee convergence to the exact average in a digraph (e.g., [11]–[14]), few of them have addressed delays, and it is unclear how/if these techniques can be modified to overcome delay disturbances while ensuring convergence to the exact average of the initial values.

The methodology developed in this paper is based on an algorithm suggested in [11] that solves the average consensus problem in a digraph using a linear iterative strategy in which each

node v_j distributively sets the weights on its self-link and out-going links to be $\frac{1}{1+\mathcal{D}_j^+}$ (where $\mathcal{D}_j^+$ is the out-degree of node v_j, i.e., the number of nodes to which node v_j can send information). More generally, the set of weights needs to adhere to the graph structure (i.e., be positive on each edge – including self-edges – and zero otherwise), but it is otherwise unrestricted as long as it forms a primitive column stochastic matrix P. Using the weights in matrix P, average consensus is reached in [11] via *ratio consensus*, i.e., two linear iterations (with appropriately chosen initial conditions) that run simultaneously so that the average can be obtained at each node by taking the ratio of the two values it maintains for each of the two iterations.

The idea of ratio consensus can be traced back much earlier (see the discussion on weak convergence at the "Bibliography and Discussion to §§3.1-3.2", pp. 98, in [15]) as it takes advantage of weak convergence of a backward product of column stochastic matrices. This should be contrasted to the strong convergence of a backward product of row stochastic matrices (which behaves equivalently to a forward product of column stochastic matrices) that is typically exploited by consensus protocols under switching communication topologies based on a single iteration (e.g., [2], [7]). The problem with these (single iteration-based) approaches when used in the presence of switching/delays is that, though convergence can be guaranteed regardless of the nature of the switching/delays, the specific value to which the nodes reach consensus depends on the nature of switching/delays (thus, reaching consensus to the exact average cannot be guaranteed unless additional conditions are satisfied).

Using ratio consensus, we address in this paper the problem of discrete-time *average* consensus in a multi-component system under a (fixed) directed interconnection topology and in the presence of bounded delays in the communication links. We devise a protocol, where each node updates its information state (at each iteration) via a linear combination of the (possibly delayed) information state received from its neighbors at that iteration. Unlike other consensus approaches, this robustified version of ratio consensus, henceforth called *robustified ratio consensus*, provably converges to the exact average of the nodes' initial values, despite the presence of arbitrary but bounded time-delays. It is worth pointing out that knowledge of the delay bound is not required.

II. NOTATION AND PRELIMINARIES

A. Notation

The set of real numbers is denoted by $\mathbb{R}$ and the set of nonnegative real numbers is denoted by $\mathbb{R}_+$. Vectors are denoted by small letters whereas matrices are denoted by capital letters. The all-ones vector is denoted by $\mathbb{1}$ and the identity matrix (of appropriate dimensions) is denoted by I. 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 $A > 0$.

In multi-component systems with fixed communication links (edges), the exchange of information between components (nodes) can be conveniently captured by a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ of order n ($n \geq 2$), where $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. A directed edge from node v_i to node v_j is denoted by $\varepsilon_{ji} \triangleq (v_j, v_i) \in \mathcal{E}$ and represents a communication link that allows node v_j to receive information from node v_i. A graph is said to be undirected if and only if $\varepsilon_{ji} \in \mathcal{E}$ implies $\varepsilon_{ij} \in \mathcal{E}$. In this paper, links are not required to be bidirectional, i.e. we deal with directed graphs; for this reason, we use the terms "graph" and "diagraph" interchangeably. By convention and for notational purposes, we assume that the given digraph 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 obviously has a link (access) to its own information. A digraph is called *strongly* connected if there exists a path from each vertex v_i in the graph to each vertex v_j ($v_j \neq v_i$). In other words, for any $v_j, v_i \in \mathcal{V}$, $v_j \neq v_i$, one can find a sequence of nodes $v_i = v_{l_1}, v_{l_2}, v_{l_3}, \ldots, v_{l_t} = v_j$ ($t \geq 2$) such that link $(v_{l_{k+1}}, v_{l_k}) \in \mathcal{E}$ for all $k = 1, 2, \ldots, t-1$.

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 $\mathcal{N}_j^- = \{v_i \in \mathcal{V} \mid \varepsilon_{ji} \in \mathcal{E}\}$. The cardinality of $\mathcal{N}_j^-$ is called the *in-degree* of v_j and is denoted by $\mathcal{D}_j^- = |\mathcal{N}_j^-|$. The nodes that receive information from node v_j belong to the set of out-neighbors of node v_j, denoted by $\mathcal{N}_j^+ = \{v_l \in \mathcal{V} \mid \varepsilon_{lj} \in \mathcal{E}\}$. The cardinality of $\mathcal{N}_j^+$ is called the *out-degree* of v_j and is denoted by $\mathcal{D}_j^+ = |\mathcal{N}_j^+|$.

In the algorithms we will consider, we will associate a positive weight p_{ji} to each edge $\varepsilon_{ji} \in \mathcal{E} \cup \{(v_j, v_j) \mid v_j \in \mathcal{V}\}$. The nonnegative matrix $P = [p_{ji}] \in \mathbb{R}_+^{n \times n}$ (with p_{ji} as the entry

at its jth row, ith column position) is a weighted adjacency matrix (also referred to as weight matrix) that has zero entries at locations that do not correspond to directed edges or self-edges in the graph, and has positive entries otherwise. In other words, apart from the main diagonal, the zero/nonzero structure of the weighted adjacency matrix P matches exactly the set of links in the given graph.

At each time step k, each node v_j updates its information state (a real value $x_j[k]$ it maintains) to $x_j[k+1]$ as a weighted linear combination of its own value $x_j[k]$ and the available information received by its neighbors $\{x_i[k] \mid v_i \in \mathcal{N}_j^-\}$. The positive constant p_{ji} captures the weight of the information inflow from agent v_i to agent v_j. In this work, since we deal with digraphs, we assume that each node v_j chooses its self-weight p_{jj} and the weights p_{lj} on its out-going links $v_l \in \mathcal{N}_j^+$. During the iterations each node updates its information state $x_j[k+1]$ according to

$$x_j[k+1] = p_{jj}x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p_{ji}x_i[k] = p_{jj}x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} x_{j \leftarrow i}[k], \quad k = 0, 1, 2, \ldots \quad (1)$$

where $x_{j \leftarrow i}[k] \triangleq p_{ji}x_i[k]$, $x_i[k] \in \mathbb{R}$, is the value sent to node v_j from node v_i at time step k. Since node v_i chooses the weight p_{ji}, it is more convenient to sent $x_{j \leftarrow i}[k]$ instead of separately sending p_{ji} and $x_i[k]$. If we let $x[k] = (x_1[k] \quad x_2[k] \quad \ldots \quad x_n[k])^T$ and $P = [p_{ji}] \in \mathbb{R}_+^{n \times n}$, then (1) can be written in matrix form as

$$x[k+1] = Px[k]. \quad (2)$$

Notice that we adopt the common assumption that a node can receive several values from different neighboring nodes at the same time instant.

B. Ratio Consensus

In [11], the average consensus problem in a digraph is solved using ratio consensus. Each node v_j distributively sets positive weights on its self-link and out-going links so that the resulting weight matrix P is primitive column stochastic, but not necessarily row stochastic. [Since the graph is strongly connected, it will be sufficient for node v_j to choose $p_{lj} > 0$ for $v_l \in \mathcal{N}_j^+ \cup \{v_j\}$ (zero otherwise) such that $\sum_{v_l \in \mathcal{N}_j^+ \cup \{v_j\}} p_{lj} = 1$.] Average consensus is then reached by using

this weight matrix to run two linear iterations with appropriately chosen initial conditions and by having each node take the ratio of the two values it maintains (one for each iteration). The algorithm is stated below for a specific choice of weights, which assumes that each node knows its out-degree and sets its link weights to $\frac{1}{1+\mathcal{D}_j^+}$ (this has the additional advantage of allowing broadcasts, since the transmissions $x_{l\leftarrow j}[k] \triangleq p_{lj}x_j[k]$ are identical for all $v_l \in \mathcal{N}_j^+$). Note, however, that the algorithm works for any set of weights that adhere to the graph structure and form a primitive column stochastic weight matrix.

Lemma 1. *[11] Consider a strongly connected digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where each node $v_j \in \mathcal{V}$ has some initial value $y_0(j)$. Let $y_j[k]$ and $z_j[k]$ (for all $v_j \in \mathcal{V}$ and $k = 0, 1, 2, \ldots$) be the result of the iterations*

$$y_j[k+1] = p_{jj}y_j[k] + \sum_{v_i \in \mathcal{N}_j^-} y_{j\leftarrow i}[k] \,, \tag{3}$$

$$z_j[k+1] = p_{jj}z_j[k] + \sum_{v_i \in \mathcal{N}_j^-} z_{j\leftarrow i}[k] \,, \tag{4}$$

where $p_{lj} = \frac{1}{1+\mathcal{D}_j^+}$ for $v_l \in \mathcal{N}_j^+ \cup \{v_j\}$ (zeros otherwise), and the initial conditions are $y[0] = (y_0(1) \ y_0(2) \ \ldots \ y_0(|\mathcal{V}|))^T \triangleq y_0$ and $z[0] = \mathbb{1}$. Then, the protocol asymptotically converges to $\lim_{k\to\infty} \mu_j[k] = \frac{\sum_{v_i \in \mathcal{V}} y_0(i)}{|\mathcal{V}|} \,, \forall v_j \in \mathcal{V} \,, \text{ where } \mu_j[k] \triangleq \frac{y_j[k]}{z_j[k]} \,.$

Note that the ratio consensus in [11] is actually a simpler version of more general algorithms that have appeared under various names in the literature (e.g., the push-sum algorithm in [16] and the asynchronous push-sum algorithm in [17]).

C. Products of SIA Matrices

A stochastic matrix P is called in [18] SIA (stochastic, indecomposable, and aperiodic) if the limit $Q = \lim_{k\to\infty} P^k$ exists and has all of its columns identical. Specifically, $Q = \mathbf{c}_P \mathbb{1}^T$ for some nonnegative vector $\mathbf{c}_P$. It can be shown that this definition of a SIA matrix is equivalent

to the standard definitions of indecomposability and aperiodicity for stochastic matrices.[1] Let $A_1, A_2, \ldots, A_m$ 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). For the derivation of our results we make use of the theorem by Wolfowitz [18] below.

Theorem 1. *[18] Let $\overline{\mathcal{P}} = \{\overline{P}_1, \overline{P}_2, \ldots, \overline{P}_m\}$ be a collection of column stochastic matrices of order $n \times n$ such that any word in the $\overline{P}$'s is stochastic, indecomposable, and aperiodic (SIA). For any $\epsilon > 0$ there exists an integer $\nu(\epsilon)$ such that any word $B = [b_{ji}] \in \mathbb{R}_+^{n \times n}$ (in the $\overline{P}$'s) of length $\ell \geq \nu(\epsilon)$ satisfies $\delta(B) < \epsilon$, where $\delta(B) = \max_j \max_{i_1, i_2} |b_{j,i_1} - b_{j,i_2}|$.*

In words, Theorem 1 states that for large enough ℓ, the product of ℓ matrices from the collection $\overline{\mathcal{P}}$ has all of its columns approximately the same. Note that the result does not mean that all matrix products converge to a single matrix of the form $c\mathbb{1}^T$; however, for large enough ℓ, each word B will take the form $c_B \mathbb{1}^T$ for some column vector c_B.

D. Modeling Delays

We assume that the transmission on the link from node v_i to node v_j at time step k undergoes an *a priori unknown* delay $\tau_{ji}[k]$, where $\tau_{ji}[k]$ is an integer that satisfies $0 \leq \tau_{ji}[k] \leq \bar{\tau}_{ji} < \infty$ (i.e., delays are bounded). The maximum delay is denoted by $\bar{\tau} = \max_{(v_j, v_i) \in \mathcal{E}} \bar{\tau}_{ji}$. We also assume 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). Under this model, the information available to node v_j at time step k (and which can be used to update its value to $x_j[k + 1]$) comprises of its own value $x_j[k]$ and all values received by its neighbors by that time, i.e., it is a subset of the values in the set $\{x_{j \leftarrow i}[s] \mid 0 \leq s \leq k, s + \tau_{ji}[s] \leq k, v_i \in \mathcal{N}_j^- \cup \{v_j\}\}$ (recall that, in the digraph setting

---

[1] A stochastic matrix $P \in \mathbb{R}^{n \times n}$ is said to be decomposable if there exists a nonempty proper subset $\mathcal{S} \subset \{1, 2, \ldots, n\}$ such that $p_{ji} = p_{ij} = 0$ whenever $v_i \in \mathcal{S}$ and $v_j \notin \mathcal{S}$; also, P is indecomposable if it is not decomposable. A stochastic matrix P is aperiodic if the Markov chain it describes is aperiodic, that is for every state i there exists k_i such that for all $k' \geq k_i$, the probability of being at state i after k' steps is greater than zero (for all k') or zero (for all k'). Both indecomposability and aperiodicity are properties that can be checked using the structure of the digraph that is induced by the zero/nonzero structure of matrix P. Specifically, indecomposability follows from having a connected digraph with a single strongly connected component; for an indecomposable matrix, aperiodicity is guaranteed as long as at least one component in the strongly connected component has a self loop but this is not a necessary condition.

we consider, node v_i selects the weight of the link (v_j, v_i) and sends to node v_j the value $x_{j \leftarrow i}[s] \triangleq p_{ji} x_i[s]$). The protocol we propose has each node v_j update its information state at time step k by combining (in a linear fashion) its own value $x_j[k]$ and the possibly delayed information received *at time step k* by its in-neighbors. In terms of the notation used above, this information is captured by $\{x_{j \leftarrow i}[s] \mid 0 \leq s \leq k, \ s + \tau_{ji}[s] = k, \ v_i \in \mathcal{N}_j^- \cup \{v_j\}\}$, i.e., the values that arrive at node v_j *exactly* at time k.

III. HANDLING DELAYS IN A DIGRAPH

We consider a digraph where each link transmission can undergo a bounded delay. We assume that each node v_j chooses its self weight p_{jj} and the weights $\{p_{lj} \mid v_l \in \mathcal{N}_j^+\}$ on links to its out-neighbors so that these weights are positive and satisfy $\sum_{v_l \in \mathcal{N}_j^+ \cup \{v_j\}} p_{lj} = 1$ for all $v_j \in \mathcal{V}$ (a simple choice would be to set all of these weights equal to $\frac{1}{1 + \mathcal{D}_j^+}$ as in Lemma 1). In order to handle delays, we employ a strategy where the nodes run a ratio consensus protocol (i.e., two iterations as in Lemma 1) and process information as soon as it arrives. More specifically, each node updates its information state for each iteration according to:

$$x_j[k+1] = p_{jj} x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} \sum_{r=0}^{\bar{\tau}} x_{j \leftarrow i}[k-r] I_{k-r, ji}[r] \ , \ k = 0, 1, 2, \ldots \tag{5}$$

where $x_j[0] \in \mathbb{R}$ is the initial state of node v_j, $x_{j \leftarrow i}[k-r] \triangleq p_{ji} x_i[k-r]$, and

$$I_{k, ji}(\tau) = \begin{cases} 1, & \text{if } \tau_{ji}[k] = \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

Note that the second summation is over all values received from in-neighbor $v_i \in \mathcal{N}_j^-$ at time step k (i.e., the set of values $\{x_{j \leftarrow i}[s] \mid 0 \leq s \leq k, s + \tau_{ji}[s] = k, v_i \in \mathcal{N}_j^- \cup \{v_j\}\}$). Also note that node v_j is oblivious to delays and does not even need to know $\bar{\tau}$; it simply processes (delayed) packets as they arrive. In the absence of delays, we have $\tau_{ji}[k] = 0$ and the update relation (5) reduces to (1) with constant weights. We will show that if (5) is employed in place of (1) to run two iterations as in Lemma 1, the resulting ratio consensus approach can still be used to calculate the exact average, despite arbitrary but bounded delays in the communication

links. Essentially, we establish that the two iterations of (5) result in a ratio consensus protocol tolerant to arbitrary but bounded delays.

Assumptions. For the analysis below we are given a digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (that represents the information exchange between agents in a multi-agent system). Each node $v_j \in \mathcal{V}$ has an initial value $y_0(j)$ and runs ratio consensus, i.e., two versions of the iteration in (5), one with initial value $y_0(j)$ and one with initial value $z_0(j) = 1$. We make the following assumptions:

(A1) The digraph is strongly connected, and the (nonnegative) weights p_{lj} are positive for $l = j$ and $(v_l, v_j) \in \mathcal{E}$ (zero otherwise), and satisfy $\sum_{l=1}^{n} p_{lj} = 1$ for all $v_j \in \mathcal{V}$ (so that they form a primitive column stochastic matrix P). For simplicity, we will assume that each node sets the weights on the links to its out-neighbors (including its self-link) to $p_{lj} = \frac{1}{1+\mathcal{D}_j^+}$ for $v_l \in \mathcal{N}_j^+ \cup \{v_j\}$ (zero otherwise).

(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 $(v_j, v_i) \in \mathcal{E}$ for all time instants 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. The particular choice of weights ensures that the weight matrix P is primitive column stochastic. Assumption (A2) implies that no message is lost in the network (i.e., each message will eventually arrive at its destination), and every agent updates its value, using values from its in-neighbors, at least once every $\bar{\tau}$ consecutive updates. The proof of the theorem below is developed in the remainder of this section.

Theorem 2. *Consider a strongly connected digraph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where each node $v_j \in \mathcal{V}$ has some initial value $y_0(j)$. Let $y_j[k]$ and $z_j[k]$ (for all $v_j \in \mathcal{V}$ and $k = 0, 1, 2, \ldots$) be the result of the iterations*

$$y_j[k+1] = p_{jj} y_j[k] + \sum_{v_i \in \mathcal{N}_j^-} \sum_{r=0}^{\bar{\tau}} y_{j \leftarrow i}[k-r] I_{k-r, ji}[r] , \tag{7}$$

$$z_j[k+1] = p_{jj} z_j[k] + \sum_{v_i \in \mathcal{N}_j^-} \sum_{r=0}^{\bar{\tau}} z_{j \leftarrow i}[k-r] I_{k-r, ji}[r] , \tag{8}$$

where $y[0] = (y_0(1) \ \ y_0(2) \ \ \ldots \ \ y_0(|\mathcal{V}|))^T \triangleq y_0$ and $z[0] = \mathbb{1}$, and $I_{k, ji}$ is an indicator function that captures the bounded delay $\tau_{ji}[k]$ on link (v_j, v_i) at iteration k (as defined in (6), $\tau_{ji}[k] \leq \bar{\tau}$).

Then, under Assumptions (A1) and (A2), we have $\displaystyle\lim_{k\to\infty}\mu_j[k] = \frac{\sum_{v_i\in\mathcal{V}} y_0(i)}{|\mathcal{V}|}$, $\forall v_j \in \mathcal{V}$, *where*

$\mu_j[k] = \dfrac{y_j[k]}{z_j[k]}.$

Notice that the two iterations in the above theorem are coupled via the delays (the indicator functions $I_{k,ji}$ are the same in both iterations). Our proof is based on an augmented representation (digraph) that allows us to establish that (for fixed communication topologies) the distributed ratio consensus algorithm in (7)–(8) will lead to asymptotic average consensus, regardless of the nature and order of the delays, as long as they are bounded. Note that the nodes are not required to know the delay of any packet or any upper bound on the delay at each time step; instead, at each time step k, each node considers all the packets that it receives at time step k, and includes their value in the sum. In the augmented graph representation, we add extra, "virtual" nodes and use them to capture the effect of delays on the various links. This augmented representation is only used for modeling/analysis purposes and does not affect the implementation of the algorithm. The maximum number of "virtual" nodes for each original node is bounded by the maximum delay $\bar{\tau}$. In particular, for each node $v_j \in \mathcal{V}$ we introduce $\bar{\tau}$ "virtual" nodes $v_j^{(1)}, v_j^{(2)}, \ldots, v_j^{(\bar{\tau})}$. At each time step k, virtual node $v_j^{(r)}$ holds the sum of the values that are destined to arrive to node v_j in r steps. The augmented graph has $(1 + 2\bar{\tau})|\mathcal{E}|$ edges; specifically, for each edge (v_j, v_i) in the original graph, that edge also exists in the augmented graph along with edges $(v_j^{(1)}, v_i)$, $(v_j^{(2)}, v_i)$, $\ldots$, $(v_j^{(\bar{\tau})}, v_i)$, and also edges $(v_j, v_j^{(1)})$, $(v_j^{(1)}, v_j^{(2)})$, $\ldots$, $(v_j^{(\bar{\tau}-1)}, v_j^{(\bar{\tau})})$.

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 and $(1 + 2\bar{\tau})|\mathcal{E}|$ edges). If we let $\overline{x}[k] = \left(x^T[k]\ \ x^{(1)}[k]\ \ \ldots\ \ x^{(\bar{\tau})}[k]\right)^T$ and $x^{(r)}[k] = \left(x_1^{(r)}[k]\ \ \ldots\ \ x_n^{(r)}[k]\right)$, $r = 1, 2, \ldots \bar{\tau}$, then we can write $\overline{x}[k+1] = \overline{P}[k]\overline{x}[k]$, where

$$\overline{P}[k] \triangleq \begin{pmatrix} P_0[k] & I_{n\times n} & 0 & \cdots & 0 \\ P_1[k] & 0 & I_{n\times n} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_{\bar{\tau}-1}[k] & 0 & 0 & \cdots & I_{n\times n} \\ P_{\bar{\tau}}[k] & 0 & 0 & \cdots & 0 \end{pmatrix}. \tag{9}$$

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

the link delays that are experienced by messages sent at time k. Specifically, $P_r[k]$ is a matrix associated only with the links of the graph for which the message was delayed by r steps at time step k, and satisfies

$$P_r[k](j,i) = \begin{cases} P(j,i), & \text{if } \tau_{ji}[k] = r, \ (v_j, v_i) \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases}$$

Note that, for each $(v_j, v_i) \in \mathcal{E}$, only one of $P_0[k](j,i)$, $P_1[k](j,i)$, ..., $P_{\bar{\tau}}[k](j,i)$ is nonzero and is equal to $P(j,i)$. Thus, we also have

$$P = \sum_{r=0}^{\bar{\tau}} P_r[k], \quad k = 0, 1, 2, \ldots \tag{10}$$

Matrix $\overline{P}[k]$ may take at most $(\bar{\tau}+1)^{|\mathcal{E}|}$ matrix values, where $(\bar{\tau}+1)$ is the number of different delays for each link $(v_j, v_i) \in \mathcal{E}$. In the sequel we do not require the matrix $\overline{P}[k]$ to be known at each time step k; what we utilize is that $\overline{P}[k]$ will be a matrix from a finite set of possible matrices $\overline{\mathcal{P}}$, which have certain useful properties.

Proposition 1. *Let $\overline{\mathcal{P}} = \{\overline{P}_1, \overline{P}_2, \ldots, \overline{P}_{(\bar{\tau}+1)^{|\mathcal{E}|}}\}$ be the set of all possible $\overline{P}[k]$ as defined in* (9). *Then, for integer ℓ, $\ell \geq \bar{\tau} + 1$, any ℓ-length word $B = \overline{P}[k+\ell]\overline{P}[k+\ell-1]\ldots\overline{P}[k+1]$ is SIA. Moreover, for $\ell \geq n(\bar{\tau}+1)$, the first n rows of matrix B will be positive with minimum entry greater or equal to $c_{\min} \triangleq \left(\frac{1}{1+\mathcal{D}_{\max}^+}\right)^{n(\bar{\tau}+1)}$, where $\mathcal{D}_{\max}^+ = \max_{v_j \in \mathcal{V}} \mathcal{D}_j^+$.*

    Proof of Proposition 1: The proof is included in the Appendix. ∎

    Proof of Theorem 2: The proof is included in the Appendix. ∎

Remark 1. *Our study allows nodes to simultaneously receive/transmit information from/to more than one node; it also assumes that all nodes update at each time step, but the transmitted information might get delayed (due to various reasons) and a node may receive multiple updates from the same in-neighbor at some iterations. These extra features are not covered in [17], even though asynchronous updates can occur (with the main challenge being the fact that nodes may be viewing different values from the same node due to varying delays). Due to the different model used, the approach followed by the authors in [17] is different from our approach.*

Example 1. *Consider the directed network on the left of Figure 1 where each node v_j chooses its self-weight and the weights of its out-going links to be $(1 + \mathcal{D}_j^+)^{-1}$ so that the weight matrix P is primitive column stochastic as shown on the right of the figure. We consider the update formula (7) with $y[0] = (-1 \ \ 2 \ \ 3 \ \ 4 \ \ 2)^T \triangleq y_0$ and suppose that the maximum allowable*

$$P = \begin{pmatrix} 1/3 & 0 & 0 & 1/2 & 0 \\ 1/3 & 1/3 & 0 & 0 & 0 \\ 1/3 & 1/3 & 1/2 & 0 & 1/3 \\ 0 & 0 & 0 & 1/2 & 1/3 \\ 0 & 1/3 & 1/2 & 0 & 1/3 \end{pmatrix}$$
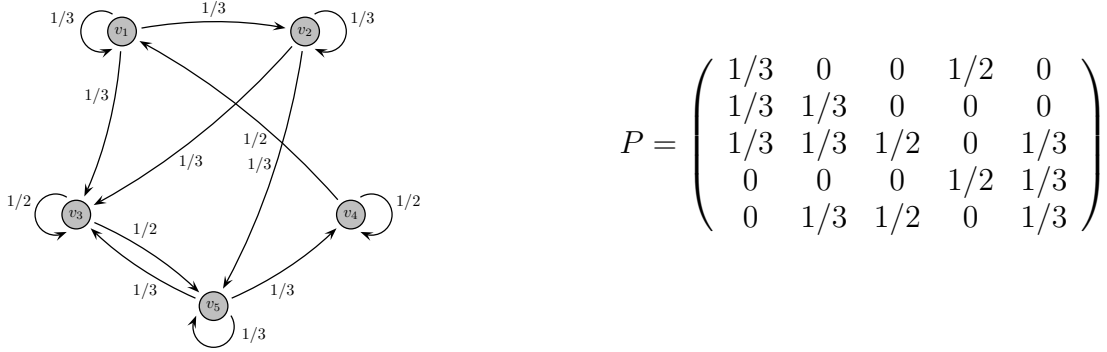
Fig. 1. A simple digraph with five nodes (left) and its associated weight matrix P (right).

delay is $\bar{\tau} = 5$. More specifically and for simplicity, we assume that at each link at each time instant, the delay is an integer in $\{0, 1, 2, ..., 5\}$, each chosen uniformly with probability $1/6$ in our simulations. If we run our update formula as in (7) for the network in Figure 1 with weights P and $y[0] = y_0$, the algorithm does not converge (see Figure 2, left). However, if we run ratio consensus in (7) and (8) with initial conditions $y[0] = y_0$ and $z[0] = \mathbb{1}$ respectively, then average consensus is asymptotically reached for the ratio $y_j[k]/z_j[k]$ (Figure 2, right). This demonstrates the validity of our theoretical analysis, both in the sense that each of the individual iterations does not converge and also in the sense that the ratios converge to the average of the initial values.

To gain additional insight into the problem, we also consider the convergence of node 1 under (i) different upper bounds in delays (see left of Figure 3 where delays are equally likely as before), and (ii) varying network size (see right of Figure 3 where random geometric graphs of different sizes are used and $\bar{\tau} = 5$ with delays being equally likely as before). It is obvious from the simulations that the convergence speed of the algorithm depends on the delays (e.g., longer delays result in slower convergence). Nevertheless, for fixed $\bar{\tau}$ it appears that the size of the network has no effect on the convergence time (at least for geometric graphs).
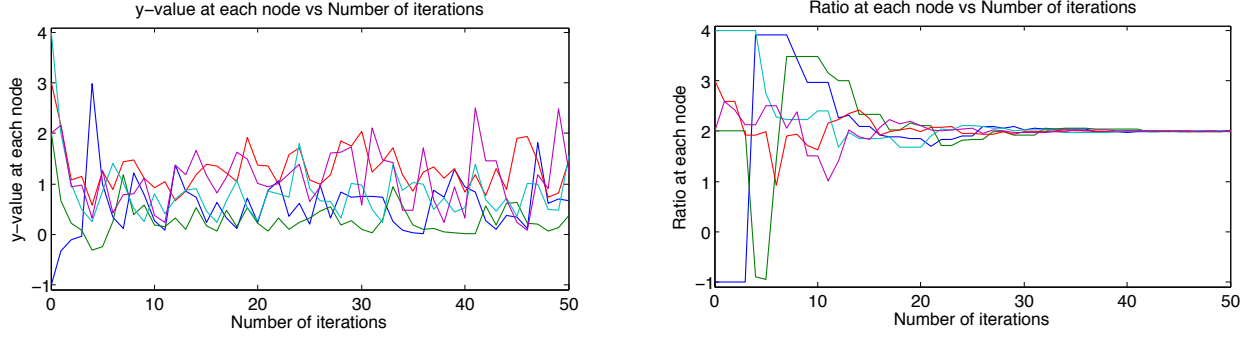
Fig. 2. The update formula in (7) for the network on the left of Figure 1 with weights P and $y[0] = y_0$, does not converge (left); however, if we run ratio consensus in (7) and (8) with initial conditions $y[0] = y_0$ and $z[0] = \mathbb{1}$ respectively, then average consensus is asymptotically reached for the ratio $y_j[k]/z_j[k]$ (right).
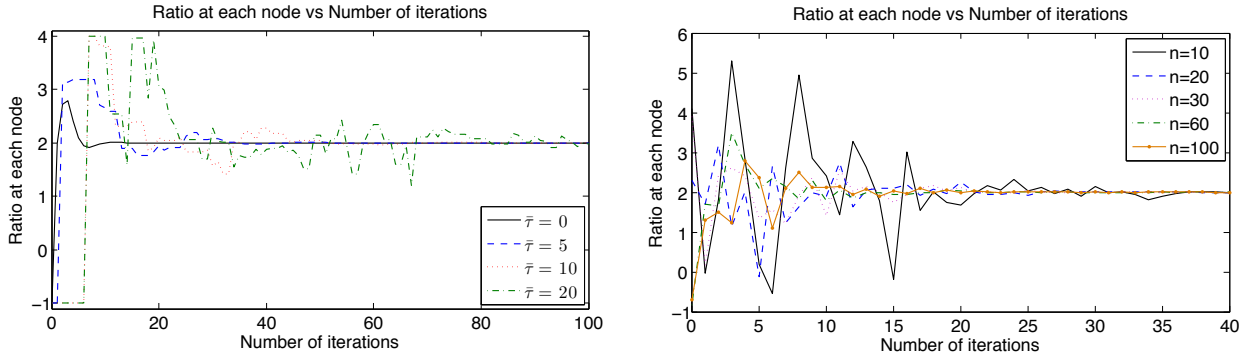

Fig. 3. Convergence of the ratio at node 1 for different upper bound $\bar{\tau}$ on delays (left) and different network size n (right).

Remark 2. *In [19], [20], the following update formula is suggested*

$$x_j[k+1] = p'_{jj}x_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p'_{ji}x_i[k - d_{ji}[k]], \quad k = 0, 1, 2, \ldots \tag{11}$$

where $x[0] = y_0$, the weights p'_{ji} form a primitive doubly stochastic weight matrix $P' = [p'_{ji}]$ and $d_{ji}[k]$ is chosen so that node v_j uses in its update the most recently seen value from node v_i (i.e., $d_{ji}[k] = \min_{\tau_{ji}[k-t]=t, 0 \leq t \leq \bar{\tau}}\{t\}$). Since the weight matrix $P' = [p'_{ji}] \in \mathbb{R}_+^{n \times n}$ is primitive doubly stochastic, we know that in the absence of delays the iteration in (11) would reach asymptotic average consensus [3]. The iteration also reaches consensus in the presence of delays (regardless of the delays introduced, as long as they are bounded [19]), but not necessarily to the exact average of the initial values. The value the nodes converge to depends on the specific delays that are introduced during the execution of the iteration.

Remark 3. *One obvious alternative is to have the nodes wait for $\bar{\tau}$ steps until they collect all the delayed packets and then update. This alternative approach requires knowledge of the upper bound $\bar{\tau}$ of the delays by each node v_j and essentially amounts to delaying each iteration step by the maximum delay, ensuring in this way that all packets are received at each node before processing for the next iteration starts. This is a conservative approach that implies slower convergence compared to our algorithm (see Figure 4 below).*
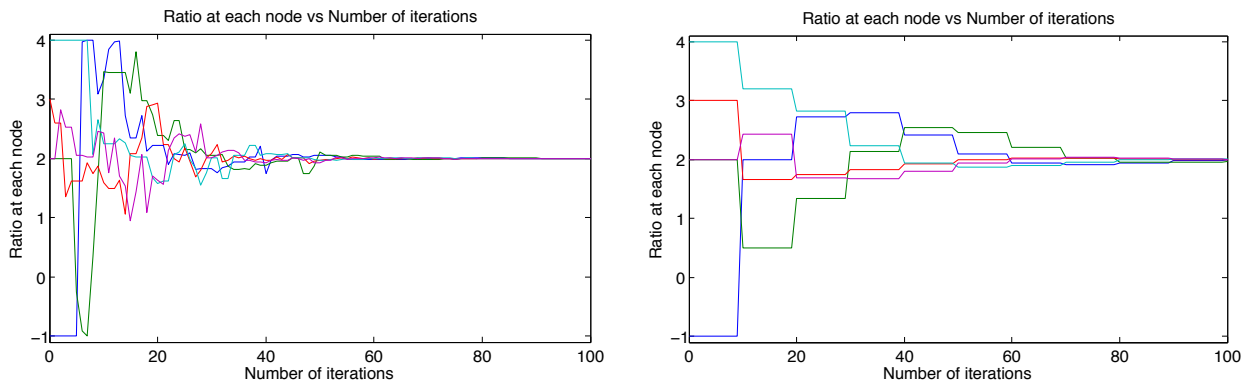

Fig. 4. Comparison of convergence time for running ratio consensus in (7) and (8) for uniformly random delays between 0 and 10 (equiprobable as before), and for updating every $\bar{\tau} = 10$ so that all the delayed information is collected.

IV. CONCLUSIONS

In this paper, we studied distributed strategies for a discrete-time multi-component system to reach asymptotic average consensus in the presence of time-varying delays. By assuming that nodes in the multi-component system have knowledge of their out-degree (i.e., the number of nodes to which they send information) and by modeling the time-delays using an augmented graphical model, we have shown (using weak convergence of backward products of column stochastic matrices) that our proposed discrete-time strategy reaches asymptotic average consensus in a distributed fashion for whatever the realization of delays, as long as they are bounded.

REFERENCES

[1] C. N. Hadjicostis and T. Charalambous, "Asynchronous coordination of distributed energy resources for the provisioning of ancillary services," in *Proc. of 47th Annual Allerton Conference on Communication, Control, and Computing*, Sep. 2011, pp. 1500–1507.

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

[3] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, no. 1, pp. 65–78, Sep. 2004.

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

[5] P. Bliman and G. Ferrari-Trecate, "Average consensus problems in networks of agents with delayed communications," *Automatica*, vol. 44, no. 8, pp. 1985–1995, Aug. 2008.

[6] F. Fagnani and S. Zampieri, "Average consensus with packet drop communication," *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 102–133, Feb. 2009.

[7] W. Ren and R. Beard, "Consensus seeking in multiagent systems under dynamically changing interaction topologies," *IEEE Trans. on Automatic Control*, vol. 50, no. 5, pp. 655–661, May 2005.

[8] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Trans. on Automatic Control*, vol. 50, no. 2, pp. 169–182, Feb. 2005.

[9] F. Xiao and L. Wang, "State consensus for multi-agent systems with switching topologies and time-varying delays," *Inter. Journal of Control*, vol. 79, no. 10, pp. 1277–1284, Oct. 2006.

[10] A. Nedic, A. Ozdaglar, and P. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 922–938, April 2010.

[11] A. D. Domínguez-García and C. N. Hadjicostis, "Coordination and control of distributed energy resources for provision of ancillary services," in *Proc. of the First IEEE International Conference on Smart Grid Communications*, Oct. 2010, pp. 537–542.

[12] M. Franceschelli, A. Giua, and C. Seatzu, "Distributed averaging in sensor networks based on broadcast gossip algorithms," *IEEE Sensors Journal*, vol. 11, no. 3, pp. 808–817, March 2011.

[13] Y. Chen, R. Tron, A. Terzis, and R. Vidal, "Accelerated corrective consensus: Converge to the exact average at a faster rate," in *Proceedings of American Control Conference*, July 2011, pp. 3417–3422.

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

[15] E. Seneta, *Non-negative Matrices and Markov Chains*, 2nd ed. New York, NY: Springer, 2006.

[16] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-based computation of aggregate information," in *Proc. of the 44th Annual IEEE Symp. on Foundations of Computer Science*, 2003, pp. 482–491.

[17] F. Bénézit, V. Blondel, P. Thiran, J. Tsitsiklis, and M. Vetterli, "Weighted gossip: Distributed averaging using non-doubly stochastic matrices," in *Proc. of the IEEE International Symposium on Information Theory*, June 2010, pp. 1753–1757.

[18] J. Wolfowitz, "Products of indecomposable, aperiodic, stochastic matrices," *Proceedings of the American Mathematical Society*, vol. 15, pp. 733–737, Oct. 1963.

[19] B. Lubachevsky and D. Mitra, "A chaotic asynchronous algorithm for computing the fixed point of a nonnegative matrix of unit spectral radius," *Journal of ACM*, vol. 33, no. 1, pp. 130–150, Jan. 1986.

[20] L. Fang and P. Antsaklis, "Information consensus of asynchronous discrete-time multi-agent systems," in *Proc. of American Control Conference*, June 2005, pp. 1883–1888.

V. Appendix

Proof of Proposition 1

In order to prove that $B = \overline{P}[k+\ell]\ldots\overline{P}[k+2]\overline{P}[k+1]$ is SIA, we have to show that it is (α) column stochastic, (β) indecomposable, and (γ) aperiodic.

(α) *Column Stochasticity*: This is easy to see as it is equivalent to proving that the product of two or more column stochastic matrices of the same order is also a column stochastic matrix (the result follows easily by induction and is standard).

(β) *Indecomposability*: We argue indecomposability for $\ell \geq \bar{\tau}+1$ (the result also holds for any $0 \leq \ell < \bar{\tau}+1$ but we do not discuss the proof here due to space limitations). Write matrix B in block form as

$$
B = \begin{pmatrix}
B_{0,0} & B_{0,1} & B_{0,2} & \cdots & B_{0,\bar{\tau}} \\
B_{1,0} & B_{1,1} & B_{1,2} & \cdots & B_{1,\bar{\tau}} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
B_{\bar{\tau}-1,0} & B_{\bar{\tau}-1,1} & B_{\bar{\tau}-1,2} & \cdots & B_{\bar{\tau}-1,\bar{\tau}} \\
B_{\bar{\tau},0} & B_{\bar{\tau},1} & B_{\bar{\tau},2} & \cdots & B_{\bar{\tau},\bar{\tau}}
\end{pmatrix},
$$

where all blocks are nonnegative matrices of size $n \times n$. We will argue that (i) the zero/nonzero structure of $B_{0,0}$ corresponds to a graph that is strongly connected, and (ii) each of $B_{0,0}$, $B_{0,1}$, $B_{0,2}$, ..., $B_{0,\bar{\tau}}$ has strictly positive entries on its diagonal. These two facts establish that the graph that corresponds to the zero/nonzero structure of the overall matrix B has the following property: (i) any pair of non-virtual nodes (i.e., the top n nodes) can be connected via a directed path (that can actually involve only non-virtual nodes); (ii) all other (virtual) nodes have an outgoing link to at least one of the non-virtual nodes. Therefore, the set of non-virtual nodes is part of a strongly connected component; this component could potentially involve other (virtual) nodes in the graph, but no other strongly connected component exists. Thus, matrix B is indecomposable.

For fact (i), we need to explain why $B_{0,0}$ corresponds to a graph of n nodes that is strongly connected. It is not hard to see that one can write

$$
B_{0,0} = (\Pi_{l=2}^{\ell} P_0[k+l]) P_0[k+1] + (\Pi_{l=3}^{\ell} P_0[k+l]) P_1[k+1] + \ldots + (\Pi_{l=\bar{\tau}+2}^{\ell} P_0[k+l]) P_{\bar{\tau}}[k+1] + E_{0,0}
$$

where $\Pi_{l=l_1}^{l_2} A[l] \equiv A[l_2]A[l_2 - 1]...A[l_1]$ ($\Pi_{l=l_1}^{l_2} A[l] \equiv I$ for $l_2 = l_1 - 1$ and zero otherwise) and $E_{0,0}$ is a nonnegative matrix (that can be expressed as the sum of various products of the nonnegative[2] blocks composing the $\overline{P}$ matrices). Since the diagonal elements in matrix $P_0[k + l]$ (for $l = 1, 2, ..., \ell$ are strictly positive, we know that the diagonals of each product $\Pi_{l=m}^{\ell} P_0[k + l]$, $m = 2, 3, ..., \bar{\tau} + 2$, will be strictly positive and thus the elements of each term $(\Pi_{l=m}^{\ell} P_0[k + l])P_{m-2}[k + 1]$ will be positive, at least at the locations where $P_{m-2}[k]$ is positive. Thus, from the expression for $B_{0,0}$ above, the zero/nonzero structure of $B_{0,0}$ corresponds to a graph of n nodes that includes all the edges in $\sum_{r=0}^{\bar{\tau}} P_r[k + 1] = P$ (recall (10)); thus, all edges in the original graph are included and, since the original graph is strongly connected, $B_{0,0}$ corresponds to a graph that is strongly connected.

For fact (ii), we need to explain why each $B_{0,r}$, $r = 0, 1, ..., \bar{\tau}$, has strictly positive diagonal entries. For $r = 0$, this follows for the discussion above. For $r = 1, 2, ..., \bar{\tau}$, we can also write

$$B_{0,r} = (\Pi_{l=r+2}^{\ell} P_0[k + l])P_0[k + 1 + r] + E_{0,r} ,$$

where $E_{0,r}$ is again a nonnegative matrix that can be expressed as the sum of various products of nonnegative matrices. Since the diagonal elements in matrix $P_0[k + l]$ (for $l = 1, 2, ..., \ell$) are strictly positive, we know that the diagonals of each $B_{0,r}$ will be strictly positive.

(γ) *Aperiodicity*: Since the graph corresponding to B is indecomposable, aperiodicity is easily established due to the fact that the diagonal entries that correspond to the original (non-virtual) nodes in the strongly connected component are nonzero (it is sufficient for at least one of them to be nonzero).

To prove the second part of the proposition (i.e., for $\ell \geq n(\bar{\tau} + 1)$, the first n rows of matrix B will be positive with minimum entry greater or equal to $c_{\min} \equiv \left(\frac{1}{\mathcal{D}_{\max}^+}\right)^{n(\bar{\tau}+1)}$), notice that for $\ell = n(\bar{\tau} + 1)$ we can write B as

$$B = \underbrace{B_{i_n} B_{i_{n-1}}...B_{i_2}}_{B'} B_{i_1} ,$$

---

[2]The fact that the blocks are nonnegative is important because it means that nonzero entries created by some products cannot be cancelled by nonzero entries of other products.

where each B_{i_m} is the product of $\bar{\tau}+1$ consecutive $\overline{P}$, i.e.,

$$B_{i_m} = \overline{P}[k+m(\bar{\tau}+1)]\dots\overline{P}[k+(m-1)(\bar{\tau}+1)+2]\overline{P}[k+(m-1)(\bar{\tau}+1)+1]\,.$$

From the discussion on indecomposability, we know that each B_{i_m} has blocks $B_{0,0}^{(i_m)}$, $B_{0,1}^{(i_m)}$, ..., $B_{0,\bar{\tau}}^{(i_m)}$ such that the zero/nonzero structure of $B_{0,0}^{(i_m)}$ corresponds to a graph that includes the original strongly connected graph of size n and has a positive diagonal. Thus, the product of $n-1$ such blocks will result in a strictly positive diagonal block for matrix B'. An additional multiplication by B_{i_1} on the right, will ensure that each of the top $\bar{\tau}+1$ blocks of matrix B will be strictly positive. Since B involves the product of $n(\bar{\tau}+1)$ nonnegative matrices $\overline{P}$ (whose minimum nonzero entry is $\frac{1}{\mathcal{D}_{\max}^+}$), the minimum entry in B will be greater or equal to $c_{\min}$.

For $\ell = n(\bar{\tau}+1)+1$, we have a matrix product of the form $B\overline{P}[k+1]$, where B is the product of $n(\bar{\tau}+1)$ matrices $\overline{P}$ (thus, its top n rows are strictly positive with minimum entry $c_{\min}$). Since $\overline{P}[k+1]$ is a column stochastic matrix, we can easily conclude that matrix $B\overline{P}[k+1]$ will also have its top n rows positive with minimum entry $c_{\min}$. The claim in the second part of the proposition (that, for $\ell \geq n(\bar{\tau}+1)$, the first n rows of matrix B will be positive with minimum entry greater or equal to $c_{\min}$), then follows easily by induction.

Proof of Theorem 2

If we use the augmented graph representation with initial conditions $\bar{y}[0] = [y_0^T\ 0\ 0\ \dots\ 0]^T$ and $\bar{z}[0] = [\mathbb{1}^T\ 0\ 0\ \dots\ 0]^T$, we can write

$$\bar{y}[k] = B_k\bar{y}[0]\,,\quad \bar{z}[k] = B_k\bar{z}[0]\,,$$

where $B_k = \overline{P}[k]\dots\overline{P}[2]\overline{P}[1]$. By Proposition 1 and Wolfowitz's Theorem, we know that for any $\epsilon > 0$, the resulting word B_k satisfies (for $k \geq \nu(\epsilon)$) $B_k = c_{B_k}\mathbb{1}^T + E_k$, where c_{B_k} is an appropriate nonnegative vector, and E_k is an error matrix with entries with absolute value smaller than $\epsilon/2$ (i.e., $|E_k(j,i)| < \epsilon/2$ for all j,i). Taking k to also satisfy $k \geq n(\bar{\tau}+1)$, it follows from Proposition 1 that each of the first n entries of c_{B_k} (i.e., the entries that correspond to non-virtual nodes) will be greater than $c_{\min}$. Without loss of generality, we take $\epsilon < 2c_{\min}$ in

the remainder of this discussion.

With the above notation at hand, we have for $v_j \in \mathcal{V}$

$$\mu_j[k] \triangleq \frac{\bar{y}_j[k]}{\bar{z}_j[k]} = \frac{B_k(j,:)\bar{y}[0]}{B_k(j,:)\bar{z}[0]} = \frac{c_{B_k}(j)(\mathbb{1}^T + e_k^T)\bar{y}[0]}{c_{B_k}(j)(\mathbb{1}^T + e_k^T)\bar{z}[0]} = \frac{(\mathbb{1}^T + e_k^T)\bar{y}[0]}{(\mathbb{1}^T + e_k^T)\bar{z}[0]} \ ,$$

where $c_{B_k}(j)$ is the jth element of vector c_{B_k}, and $e_k^T = E_k(j,:)$ is the jth row of matrix E_k and satisfies $e_{\max}(k) \equiv \max_i\{|e_k(i)|\} < \epsilon/2$.

Since $\bar{z}[0] = \mathbb{1} \geq 0$ (elementwise), the denominator of the above expression can be bounded as

$$n(1 - e_{\max}(k)) \leq (\mathbb{1}^T + e_k^T)\bar{z}[0] \leq n(1 + e_{\max}(k)) \ .$$

Similarly, assuming that $\sum_l \bar{y}_l[0] = \sum_l y_l[0] > 0$ (when $\sum_l \bar{y}_l[0] = \sum_l y_l[0] < 0$ or $\sum_l \bar{y}_l[0] = \sum_l y_l[0] = 0$ we can apply a similar analysis), we can bound the numerator of the above expression as

$$\Sigma_y - e_{\max}(k)\Sigma_{|y|} \leq (\mathbb{1}^T + e_k^T)\bar{y}[0] \leq \Sigma_y + e_{\max}(k)\Sigma_{|y|} \ ,$$

where $\Sigma_y = \sum_l \bar{y}_l[0] = \sum_l y_l[0]$ and $\Sigma_{|y|} = \sum_l |\bar{y}_l[0]| = \sum_l |y_l[0]|$. Putting the above inequalities together, we obtain

$$\frac{\Sigma_y - e_{\max}(k)\Sigma_{|y|}}{n(1 + e_{\max}(k))} \leq \mu_j[k] \triangleq \frac{\bar{y}_j[k]}{\bar{z}_j[k]} \leq \frac{\Sigma_y + e_{\max}(k)\Sigma_{|y|}}{n(1 - e_{\max}(k))} \ ,$$

which can be relaxed (after some algebraic manipulations) to $\mu^* - M_k \leq \mu_j[k] \leq \mu^* + M_k$, where $\mu^* = \frac{\sum_l y_l[0]}{n}$ is the exact average and $M_k = \mu^* \frac{(\Sigma_y + \Sigma_{|y|})e_{\max}(k)}{\Sigma_y(1 - e_{\max}(k))}$. By Wolfowitz theorem, we can take k as large as necessary to make M_k arbitrarily small (by ensuring that ϵ and thus $e_{\max}(k)$) is as small as desired).