Average Consensus in the Presence of Dynamically Changing Directed Topologies and Time Delays

Themistoklis Charalambous and Christoforos N. Hadjicostis

***Abstract*— We have recently proposed a robustified ratio consensus algorithm which achieves asymptotic convergence to the global average in a distributed fashion in static strongly connected digraphs, despite the possible presence of bounded but otherwise arbitrary delays. In this work, we propose a protocol which reaches asymptotic convergence to the global average in a distributed fashion under possible changes in the underlying interconnection topology (e.g., due to component mobility), as well as time-varying delays that might affect transmissions at different times. More specifically, we extend our previous work to also account for the case where, in addition to arbitrary but bounded delays, we may have time varying communication links. The proposed protocol requires that each component has knowledge of the number of its outgoing links, perhaps with some bounded delay, and that the digraphs formed by the switching communication topologies over a finite time window are jointly strongly connected.**

I. INTRODUCTION

Convergence of consensus algorithms can usually be established under relatively weak requirements. Typical applications involve motion of mobile agents (e.g., coordination of unmanned air vehicles, autonomous underwater vehicles, or satellites) and averaging of measurements in wireless sensor networks. Common challenges include the handling of node failures (e.g., due to the draining of batteries in wireless sensor networks), inhomogeneous transmission delays on the transfer of data between agents, packet losses in wireless communication networks, and inaccurate sensor measurements. As a result, it is imperative to address agreement problems that consider networks of dynamical agents, possibly with directed information flow, under changing topologies with/without delays.

One of the most well known consensus problems is the so-called *average consensus* problem in which agents aim to reach the average of their initial values (see, for example, [1]). It has been shown in [2] that, under a fixed interconnection topology, average consensus can be achieved by performing a linear iteration in a distributed fashion if the interconnection topology is both strongly connected and balanced. Even though various approaches have been proposed for forming a balanced matrix (e.g., [3]) and a primitive doubly stochastic matrix (e.g., [4]), which can subsequently be used for reaching average consensus, most existing schemes are not applicable in directed graphs and/or fail in the presence of changing interconnection topology and time-delays. In particular, among the limited existing algorithms that guarantee convergence to the exact average in a directed graph (e.g., [5], [6]), few of them have addressed delays and topology changes. More recently, [7] has proposed an approach that can reach asymptotic average consensus under switching topologies, but it is unclear how/if their techniques can be modified to consider the case where delays are also present.

In this paper, we investigate the problem of discrete-time *average* consensus in a multi-component system under a directed interconnection topology in the presence of changing interconnections (due to communication links being added or removed, as in a mobile network setting) and bounded delays in the communication links. We consider a fixed topology and we devise a protocol, based on ratio consensus [8] and robustified ratio consensus [9], where each node updates its information state by combining the available (possibly delayed) information received by its neighbors using constant positive weights. We establish that, unlike other consensus approaches, this new version of ratio consensus converges to the exact average of the initial values of the nodes, even in the presence of arbitrary changes in the communication links and bounded time-delays.

The remainder of the paper is organized as follows. In Section II, the notation used throughout the paper is provided, along with some background on graph theory that is needed for our subsequent development. This section also outlines our model for changing interconnection topology in the multi-agent system. In Section III we consider a fixed set of nodes and allow changes in the communication links among them, in order to study the behavior of our algorithm in the presence of both interconnection topology changes and delays. Finally, Section IV summarizes the results.

II. NOTATION AND PRELIMINARIES

A. Notation

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. The transpose of matrix A is denoted by A^T. By $\mathbb{1}$ we denote the all-ones vector and by I we denote the identity matrix (of appropriate dimensions). A matrix whose elements

Themistoklis Charalambous is with the Automatic Control Lab, Electrical Engineering Department and ACCESS Linnaeus Center, Royal Institute of Technology (KTH), Stockholm, Sweden. E-mail: `themisc@kth.se`.

Christoforos N. Hadjicostis is with the Department of Electrical and Computer Engineering at the University of Cyprus, Nicosia, Cyprus. E-mail: `chadjic@ucy.ac.cy`.

The work by T. Charalambous was sponsored in part by the Swedish Foundation for Strategic Research (SSF) and the Swedish Research Council (VR). The work by C. N. Hadjicostis was co-funded by the European Regional Development Fund and the Republic of Cyprus through the Research Promotion Foundation (New Infrastructure Project/Strategic/0308/26).

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 directed graph (or 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 “digraph" interchangeably. Note that by convention and for notational purposes, we assume that the given 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 obviously has a link (access) to its own information. A directed graph is called *strongly* connected if there exists a path from each vertex v_i in the graph to every other vertex v_j. 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}, $\ldots$, $v_{l_t} = v_j$ such that link $(v_{l_{k+1}}, v_{l_k}) \in \mathcal{E}$ for all $k = 1, 2, \ldots, t-1$.

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 $\mathcal{N}_j^- = \{v_i \in \mathcal{V} \mid \varepsilon_{ji} \in \mathcal{E}\}$. The cardinality of $\mathcal{N}_j^-$, is called the *in-degree* of v_j and is denoted by $\mathcal{D}_j^- = \left|\mathcal{N}_j^-\right|$. 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 $\mathcal{D}_j^+ = \left|\mathcal{N}_j^+\right|$.

In the type of algorithms we will consider, we will associate a positive weight p_{ji} for 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. In other words, apart from the main diagonal, the zero-nonzero structure of the adjacency matrix P matches exactly the given set of links in the graph.

We use $x_j[k] \in \mathbb{R}$ to denote the information state of node j at discrete time k. To capture dynamically changing topologies we will assume that we are given a *fixed* set of components $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ but the set of edges among them might change at various points in time. This results in a sequence of graphs of the form $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k])$ and means that at each time instant k, each node v_j has possibly different sets of in- and out-neighbors, denoted respectively by $\mathcal{N}_j^-[k]$ and $\mathcal{N}_j^+[k]$. Given a collection of graphs $\mathcal{G}[1], \ldots, \mathcal{G}[m]$ (for some $m \geq 1$) of the form $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k])$, $k = 1, 2, \ldots, m$, the *union graph* is defined as $\mathcal{G}_{1,2,\ldots,m} = (\mathcal{V}, \cup_{k=1}^{m} \mathcal{E}[k])$. The collection of graphs is said to be *jointly strongly connected*, if its corresponding union graph $\mathcal{G}_{1,2,\ldots,m}$ 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.

At each time step k, each node v_j updates its information state 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 $\{x_i[k] \mid v_i \in \mathcal{N}_j^-[k]\}$. Weight $p_{ji}[k]$ is positive if there exists a link at time step k from agent v_i to agent v_j and captures the weight of the information inflow; it is zero, otherwise. In this work, since we deal with directed graphs, we assume that each node v_j chooses its self-weight $p_{jj}[k]$ and the weights $p_{lj}[k]$ on its out-going links at time k, i.e., $v_l \in \mathcal{N}_j^+[k]$. Hence, in the general case, each node updates its information state $x_j[k+1]$, $k = 0, 1, 2, \ldots$, according to:

$$\begin{aligned} x_j[k+1] &= p_{jj}[k]x_j[k] + \textstyle\sum_{v_i \in \mathcal{N}_j^-[k]} p_{ji}[k]x_i[k] \\ &= p_{jj}[k]x_j[k] + \textstyle\sum_{v_i \in \mathcal{N}_j^-[k]} x_{j \leftarrow i}[k], \end{aligned} \tag{1}$$

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

$$x[k+1] = P[k]x[k]. \tag{2}$$

Note that, with the exception of the diagonal entries, we have $p_{ji}[k] = 0$, $j \neq i$, if and only if $(v_j, v_i) \notin \mathcal{E}[k]$. We say that the nodes asymptotically reach average consensus if $\lim_{k \to \infty} x_j[k] = \frac{\sum_l x_l[0]}{n}$ for all $v_j \in \mathcal{V}$.

B. Ratio Consensus

In [8], an algorithm is suggested that solves the average consensus problem in a static digraph setting, where each node v_j distributively sets the weights on its self-link and outgoing-links to be $p_{lj} = \frac{1}{1+\mathcal{D}_j^+}$, $\forall (v_l, v_j) \in \mathcal{E}$, so that the resulting weight matrix P is column stochastic, but not necessarily row stochastic. Average consensus is reached by using this weight matrix to run two iterations with appropriately chosen initial conditions. The algorithm is stated below for the specific choice of weights mentioned above (which assumes that each node knows its out-degree).

Lemma 1: [8] Consider a strongly connected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. 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

$$y_j[k+1] = p_{jj} y_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p_{ji} y_i[k] \,, \tag{3a}$$

$$z_j[k+1] = p_{jj} z_j[k] + \sum_{v_i \in \mathcal{N}_j^-} p_{ji} z_i[k] \,, \tag{3b}$$

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 $y[0] = (y_0(1) \;\; y_0(2) \;\; \ldots \;\; y_0(|\mathcal{V}|))^T \equiv y_0$ and $z[0] = \mathbb{1}$. Then, the solution to the average consensus problem can be asymptotically obtained as

$$\lim_{k\to\infty} \mu_j[k] = \frac{\sum_{v_\ell \in \mathcal{V}} y_0(\ell)}{|\mathcal{V}|} \ , \quad \forall v_j \in \mathcal{V} \ ,$$

where $\mu_j[k] = y_j[k]/z_j[k]$.

C. Modeling Switching

As in the case when there is no change in the interconnection topology, each node v_j is in charge of setting the weights $p_{lj}[k]$, $v_l \in \mathcal{N}_j^+[k]$, on all links to its out-neighbors. Due to the changing topology, the weight matrix will be time-varying and will be denoted by $P[k]$. What is important is for $P[k]$ to be column stochastic and have nonzero diagonal elements. As in Lemma 1, nodes can easily set the weights on the links to their out-neighbors to ensure column stochasticity as long as each node v_j has knowledge of its out-degree $\mathcal{N}_j^+[k]$ at each time step (in such case, each node $v_j \in \mathcal{V}$ sets $p_{lj}[k] = \frac{1}{1+\mathcal{D}_j^+[k]}$ for $v_l \in \mathcal{N}_j^+[k] \cup \{j\}$). There are various ways in which the out-degree information can become available at each node as we describe in more detail later. In our analysis of changing interconnection topology, we consider two cases.

(i) Switching without delays: When we have a time-varying interconnection topology and there exist no delays in the communication links, each node v_j updates its information state at time step k to $x_j[k+1]$ by combining its own state $x_j[k]$ and the available information received by its neighbors $\{x_{j\leftarrow i}[k] \mid v_i \in \mathcal{N}_j^-[k]\}$ (the latter information also includes the positive weights $p_{ji}[k]$ that capture the weight of the information inflow assigned by component v_i to the link (v_j, v_i) at time k). Here, we will consider two sub-cases: (a) each transmitting node knows its out-degree as soon as the change takes place, and (b) each transmitting node knows its out-degree with some delay.

(ii) Switching with delays: In this case, each node v_j updates its information state at time step k to $x_j[k+1]$ by combining its own value $x_j[k]$ (i.e., the own value of a node is always available without delay) and the available (possibly delayed) information $\{x_i[s] | s \in \mathbb{Z}, s \leq k,\ v_i \in \mathcal{N}_j^-[s], s + \tau_{ji}[s] = k\ \}$. Integer $\tau_{ji}[k] \geq 0$ represents 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 $\bar{\tau} = \max\{\bar{\tau}_{ji}\}$, $\bar{\tau} \in \mathbb{Z}_+$. The possibly delayed information also includes the positive weights $p_{ji}[s]$, that component v_i assigns to link (v_j, v_i) at time s. We consider again the cases where each node v_j discovers its out-degree without or with delay, and also consider a third case (c) in which node v_j discovers an established link with some delay.

III. Handling Changing Interconnections

In this section, we extend [9] to include time-varying communication links (in addition to bounded delays on each link). We assume that we have a time-varying graph, in which the set of nodes is fixed but the communication links can change, i.e., at time step k the interconnections between components in the multi-component system are captured by a directed graph $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k])$. For the analysis below, we let $\bar{\mathcal{G}} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\nu\}$, $\nu \in \mathbb{N}$, $\nu \leq 2^{n^2-n}$, be the set of all possible directed graphs[1] defined for a given set of nodes $\mathcal{V}$.

A. Changing interconnection topology without communication delays

We start our analysis by considering the simplest case where we have a changing interconnection topology without delays and instantaneous knowledge at each node of its out-degree (at that particular time instant). We start with the assumptions below (some of these assumptions are relaxed later on).

Assumptions 1:

(B1) At each time instant k, each node v_j knows the number of nodes receiving its message (i.e., the number of its out-neighbors $\mathcal{D}_j^+[k]$).

(B2) There exist no delays in the delivery of messages.

(B3) Given the infinite sequence of graphs $\mathcal{G}[1]$, $\mathcal{G}[2]$, ..., $\mathcal{G}[k]$, ..., we can find a finite window length l and an infinite sequence of times $t_0, t_1, \ldots, t_m$, ..., where $t_0 = 0$, such that for any $m \in \mathbb{Z}_+$, $0 < t_{m+1} - t_m \leq \ell < \infty$ and the union graph $\mathcal{G}_{t_m,\ldots,t_{m+1}-1}$ (comprising of graphs $\mathcal{G}[t_m], \mathcal{G}[t_m+1], \ldots, \mathcal{G}[t_{m+1}-1]$) is strongly connected.

Remark 1: Assumption (B1) requires that the transmitting node knows the number of nodes it transmits messages to at each time instant. In an undirected graph setting, this is not too difficult; in a directed graph setting, this is not as straightforward but there are ways in which knowledge of the out-degree might be possible. For example, there can be an acknowledgement signal (ACK) via a *distress signal* (special tone in a control slot or some separate control channel) sent at higher power than normal so that it is received by transmitters in its vicinity [10]. Knowledge of the out-degree is also possible if the nodes periodically perform checks to determine the number of their out-neighbors (e.g., by periodically transmitting the distress signals mentioned above). As we discuss later, at the cost of additional complexity, the nodes can also handle situations where they learn their out-degree with some delay. Assumption (B2) is made to keep things simple and it is relaxed later. Assumption (B3) stems from the fact that we require that there exists paths between any pair of nodes infinitely often.

In its general form, each node updates its information state according to the following relation:

$$x_j[k+1] = p_{jj}[k] x_j[k] + \sum_{v_i \in \mathcal{N}_j^-[k]} x_{j\leftarrow i}[k] \ , \qquad (4)$$

where $k = 0, 1, \ldots$, $x_{j\leftarrow i}[k] \triangleq p_{ji}[k] x_i[k]$ is the information sent from node v_i to node v_j at time step k, and $x_j[0] \in \mathbb{R}$ is the initial state of node v_j. Since the out-degree is known,

[1] Each of the n nodes may be connected (have an out-going link) with up to $(n-1)$ other nodes. As a result, we have $n(n-1)$ possible links, each of which can be either present or not. Hence, we have $2^{n(n-1)}$ possible graph combinations. Of course, depending on the underlying application, some of these interconnection topologies may be unrealizable.

the transmitting node v_j can easily set the (positive) weights to $p_{lj}[k] = \frac{1}{1+\mathcal{D}_j^+[k]}$ for $l = j$ and $(v_l, v_j) \in \mathcal{E}[k]$ (this choice satisfies $\sum_{l=1}^{n} p_{lj}[k] = 1$ for all $v_j \in \mathcal{V}$). Note that unspecified weights in $P[k]$ are set to zero and correspond to pairs of nodes (v_l, v_j) that are not connected at time step k, i.e., $p_{lj}[k] = 0$, for all $(v_l, v_j) \notin \mathcal{E}$, $j \neq i$. If we let $x[k] = (x_1[k] \;\; x_2[k] \;\; \ldots \;\; x_n[k])^T$ and $P[k] = [p_{ji}[k]] \in \mathbb{R}_+^{n \times n}$ then (1) can be written in matrix form as $x[k+1] = P[k]x[k]$, where $x[0] = (x_1[0] \;\; x_2[0] \;\; \ldots \;\; x_n[0])^T \equiv x_0^T$.

Remark 2: Communication links can be initiated/terminated throughout the operation of the algorithm, from either (a) the receiving or (b) the transmitting node. Possible communication protocols to perform these tasks are described briefly below:

(a) When node v_l wants to receive messages from node v_j (e.g., because it is in the neighborhood of v_j), it can send a distress signal to pass this request to v_j (alternatively, node v_l can send the message to node v_j using some path in the directed graph or using some sort of flooding scheme). When node v_j receives the request from v_l, it sends an acknowledgement packet (directly to node v_l) and the communication link is initiated. In practice, this might not necessarily require node v_j to transmit a separate package to node v_l (e.g., in a wireless broadcast setting) or to transmit at a higher power (e.g., if v_l is already in its range); however, it does imply that node v_j will adjust its self-weight and the weights p_{lj}, $v_l \in \mathcal{N}_j^+$, on the links to its out-neighbors in order to ensure that column stochasticity is preserved. If, on the other hand, node v_l wants to terminate the communication link, it sends (or broadcasts if the message cannot be specifically directed to node v_j) a distress signal destined for node v_j (alternatively, it can use a flooding-like strategy via the paths in the directed graph); as soon as node v_l receives an acknowledgement from node v_j along with the latest message with values for the last update, then the link can be terminated. If node v_l does not receive the acknowledgement message from node v_j, the link remains active.

(b) Note that if the transmitting node v_j wants to terminate a communication link to node v_l, it is enough to simply initiate such a request to node v_l (a direct link is available).

Lemma 2: Consider an infinite sequence of graphs of the form $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k]))$, $k = 0, 1, 2, ...$ such that there exists a finite time window length ℓ and an infinite sequence of time instants $t_0, t_1, \ldots, t_m, \ldots$, where $t_0 = 0$, such that for any $m \in \mathbb{Z}_+$, $0 < t_{m+1} - t_m \leq \ell < \infty$, and the union of graphs $\mathcal{G}[t_m], \mathcal{G}[t_m + 1], \ldots, \mathcal{G}[t_{m+1} - 1]$ is strongly connected. Let $y_j[k]$, $\forall v_j \in \mathcal{V}$, be the result of iteration (4) with $p_{lj}[k] = \frac{1}{1+\mathcal{D}_j^+[k]}$ for $v_l \in \mathcal{N}_j^+[k] \cup \{v_j\}$ (zeros otherwise) and initial conditions $y[0] = y_0$, and let $z_j[k]$, $\forall v_j \in \mathcal{V}$, be the result of iteration (4) with $p_{lj}[k] = \frac{1}{1+\mathcal{D}_j^+[k]}$ for $v_l \in \mathcal{N}_j^+[k] \cup \{v_j\}$ (zeros otherwise) and with initial condition $z[0] = \mathbb{1}$. Then, the solution to the average consensus problem in the presence of dynamically changing topologies can be obtained as $\lim_{k \to \infty} \mu_j[k] = \frac{\sum_{v_l \in \mathcal{V}} y_0(l)}{|\mathcal{V}|}$, $\forall v_j \in \mathcal{V}$, where $\mu_j[k] = \frac{y_j[k]}{z_j[k]}$.

Proof: Let $\overline{P}_{t_{m+1}-t_m} \triangleq P[t_{m+1} - 1]P[t_{m+1} - 2] \ldots P[t_m]$. Since the union of graphs from time instant t_m until $t_{m+1} - 1$, i.e., the union of graphs $\mathcal{G}[t_m], \mathcal{G}[t_m + 1], \ldots, \mathcal{G}[t_{m+1} - 1]$ is strongly connected and each matrix involved in the product has strictly positive elements on the diagonal, matrix $\overline{P}_{t_{m+1}-t_m}$ is SIA[2] for $m \in \mathbb{Z}_+$. Furthermore, products of matrices of the form $\overline{P}_{t_{m+1}-t_m}$ are SIA. Hence, according to Wolfowitz theorem [11], for any $\epsilon > 0$, there exists a finite integer $\nu(\epsilon) \in \mathbb{N}$, such that a finite word W given by the product of a collection of ν stochastic matrices of the form $\overline{P}_{t_{m+1}-t_m}$ has all of its columns approximately the same, i.e., $\overline{P}_{t_{k+\nu}-t_{k+\nu-1}} \ldots \overline{P}_{t_{k+2}-t_{k+1}} \overline{P}_{t_{k+1}-t_k} \to c_{\nu+k,k} \mathbb{1}^T$, where $c_{\nu+k,k}$ is a positive n-dimensional column vector. The proof continues as in Proposition 1 in [9]. ■

B. Changing interconnection topology with communication delays

Assumptions 2: When delays are present, we make the following extra assumption:

(C1) There exists a finite $\bar{\tau}$ that uniformly bounds the delay terms, i.e. $\tau_{ji}[k] \leq \bar{\tau}_{ji} \leq \bar{\tau}$.

In this case, each node updates its information state according to the following iteration:

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

for $k = 0, 1, 2, \ldots$, where

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

and $x_{j \leftarrow i}[k-r] \triangleq p_{ji}[k-r]x_i[k-r]$ is the value sent from node v_i to node v_j at time step $k - r$ that suffers delay r, $x_j[0] \in \mathbb{R}$ is the initial value of node v_j, and the values $p_{ji}[k] \geq 0$ depend on the topology of the graph at time k.

To handle delays 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 we can write

$$\overline{x}[k+1] = \overline{P}[k]\overline{x}[k] \; , \quad (7)$$

where (as in [9])

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

with

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

$$x^{(r)}[k] = \left(x_1^{(r)}[k] \;\; \ldots \;\; x_n^{(r)}[k] \right), \quad r = 1, 2, \ldots \bar{\tau}.$$

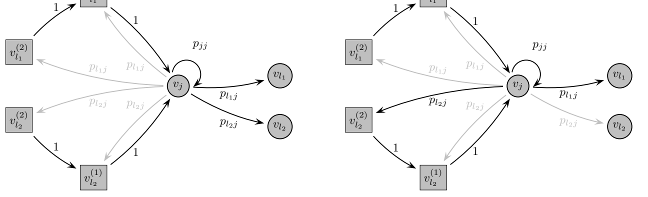
[2] In [11], a stochastic matrix P is called indecomposable and aperiodic (SIA) if $Q = \lim_{k \to \infty} P^k$ existts and all the rows of Q are the same.

As in [9], $P_0[k], P_1[k], \ldots, P_{\bar{\tau}}[k]$ are appropriately defined column stochastic matrices, such that $P[k] = \sum_{r=0}^{\bar{\tau}} P_r[k]$, i.e., the sum $P[k]$ of all the nonnegative matrices $P_r[k]$, $r \in \{0, 1, 2, \ldots, \bar{\tau}\}$, gives the weights of the zero-delay interconnection topology at time instant k. The difference from the case when only delays are present in the network is that the interconnection topology is dynamically changing and the weights at each time instant might differ. The proposed protocol is able to asymptotically reach average consensus, as stated in Lemma 3 below. The proof is similar to the proof for delays with no changes in the interconnection topology and is omitted.

Lemma 3: Consider a sequence of graphs of the form $\mathcal{G}[k] = (\mathcal{V}, \mathcal{E}[k]))$, $k = 0, 1, 2, ...$ such that there exists a finite time window length ℓ and an infinite sequence of time instants $t_0, t_1, \ldots, t_m, \ldots$, where $t_0 = 0$, such that for any $m \in \mathbb{Z}_+$, $0 < t_{m+1} - t_m \leq \ell < \infty$, $m \in \mathbb{Z}_+$, and the union of graphs $\mathcal{G}[t_m], \mathcal{G}[t_m + 1], \ldots, \mathcal{G}[t_{m+1} - 1]$ is strongly connected. Let $y_j[k]$ for all $v_j \in \mathcal{V}$ be the result of iteration (5) with $p_{lj}[k] = \frac{1}{1+\mathcal{D}_j^+[k]}$ for $v_l \in \mathcal{N}_j^+[k] \cup \{v_j\}$ (zeros otherwise) and initial conditions $y[0] = y_0$, and let $z_j[k]$, $\forall v_j \in \mathcal{V}$, be the result of iteration (4) with $p_{lj}[k] = \frac{1}{1+\mathcal{D}_j^+[k]}$ for $v_l \in \mathcal{N}_j^+[k] \cup \{v_j\}$ (zeros otherwise) and with initial condition $z[0] = \mathbb{1}$. The indicator function $I_{k,ji}$ 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, the solution to the average consensus problem can be asymptotically obtained as $\lim_{k \to \infty} \mu_j[k] = \dfrac{\sum_{v_l \in \mathcal{V}} y_0(l)}{|\mathcal{V}|}$, $\forall v_j \in \mathcal{V}$, where $\mu_j[k] = \frac{y_j[k]}{z_j[k]}$.

We now discuss the case in which a node, say node v_j, receives an indication (e.g., an acknowledgement message) that one of its out-neighbors, say $v_l \in \mathcal{N}_j^+[k-1]$, no longer receives its transmissions. In other words, node $v_l \notin \mathcal{N}_j^+[k]$ but node v_j finds out about it with some bounded delay that we denote by $T_{lj}[k]$. Such bounded delays could arise from communication protocols in a variety of ways, e.g., when using periodic acknowledgement signals like the distress signals discussed in Remark 2. Another way for node v_j to discover that its out-degree has changed is by using acknowledgement signals that arrive at node v_j through paths in the network that connect the out-neighbors of node v_j to node v_j (a direct link between an out-neighbor of node v_j and node v_j may not necessarily exist in a directed graph). We use $T_{lj}[k] \leq \overline{T} < \infty$ to denote the time it takes for node v_j to realize that node v_l is no longer in the set $\mathcal{N}_j^+[k]$. The problem that node v_j has to address at time $k + T_{lj}[k]$ when it realizes that, at time steps k, $k+1$, …, $k + T_{lj}[k] - 1$, it erroneously assumed an out-degree $\mathcal{D}_{je}^+$ that (supposing no other changes) was greater than the true out-degree $\mathcal{D}_j^+$, is that the weighted message from node v_j has not been conveyed to the out-neighbor v_l (because the link was terminated). The most straightforward way to reconcile this is to add these values back to node v_j. This can be done easily *as long as node v_j keeps track of the messages it has recently transmitted —within the last $\overline{T}$ steps— to its out-neighbors*. Note, however, that node v_j has to track these messages for each of its perceived out-neighbors, because if more than one out-neighbors terminate their links with v_j (possibly at different time steps), then node v_j must know what needs to be added back to its own value for each such former out-neighbor.

A way to model the tracking of the message is that node v_j adds, for each of its perceived out-neighbors, $\overline{T}$ "virtual" nodes that loop back to itself. These virtual nodes essentially keep track of the values that have been sent to each out-neighbor in the last $\overline{T}$ steps. The following example discusses this issue is more detailed. Consider the model for the part of the network consisting of node v_j and its two ($\mathcal{D}_j^+ = 2$) out-neighbors v_{l_1}, v_{l_2} as shown in Figure 1(a). Suppose that the maximum delay required for an acknowledgement signal from any out-neighbor of node v_j is 2 (i.e., $\overline{T} = 2$). Suppose now that out-neighbor v_{l_2} terminates the link $v_{l_2} \leftarrow v_j$ and node v_j receives an ACK with delay 2. This means that node v_j erroneously considered an out-degree of 2 (instead of 1) for the last 2 updates. In this model, the message is passed through the two extra "virtual" nodes (added in a self-loop), allowing us to loop the weighted message back to node v_j (see Figure 1(b)). Therefore, node v_j is able to recover the lost values (sent to node v_{l_2}).



(a) The maximum delay required for an acknowledgement signal from the out-neighbors of node v_j is 2.

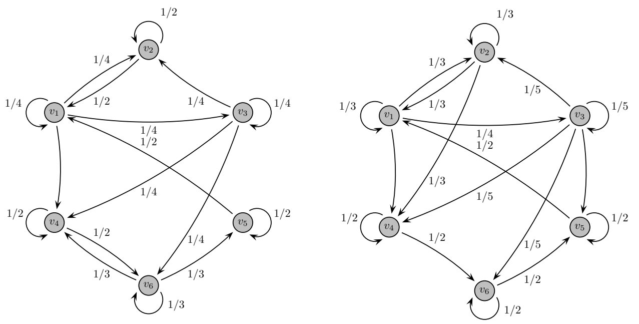
(b) A model in which node v_j directs the weighted messages of the links that no longer exist as delayed information to itself.

Fig. 1. The weighted messages from node v_j that were not conveyed to the out-neighbor v_{l_2} (because the link was terminated) are added back to the value of node v_j.

In the general case, in a network of $n = \mathcal{V}$ nodes, we introduce $\max(\bar{\tau}n, \overline{T}n)$ nodes (for a total of $(\max(\bar{\tau}n, \overline{T}n) + n)$ nodes) and we proceed as in (7).

Remark 3: There are also cases in which the transmitting node v_j may not have knowledge of its out-degree at time instant k. Such situations can also be handled if, at each time instant k, node v_j (i) knows the number of nodes with which it has established a communication link in the past and were not officially terminated, and (ii) is able to multicast a table of values to each of these out-neighbors. One way to do this is to employ the communication protocol proposed in [12] where, at each time instant, each node v_j broadcasts its own state (as updated via the iterations in equation (4)), as well as the sum of all the values, called the *total mass* in [12], that have been broadcasted to each neighboring node $v_l \in \mathcal{N}_j^+$ so far. If, for any reason, some messages are lost (dropped) or the communication link disappears for some time period, the total mass will enable the receiving out-neighbor to retrieve the information of the lost messages, with some time delay. Thus, even though the communication links may not be reliable and can even change, the problem boils down to dealing with delayed information (as in [9]). Note, however, that each node v_j needs to keep track of its own current state, the total mass transmitted to each neighboring node $v_l \in \mathcal{N}_j^+$ (the total mass can be different for each node v_l due to, for example, newly established communication links), and the total mass received from each neighboring node $v_i \in \mathcal{N}_j^-$ that transmits information to node v_j. Since different information might be needed at each node v_l at each time instant k, node v_j is required to broadcast a table of values with entries for each receiving node.

Example 1: We illustrate how the algorithm operates via a small network of six nodes. Each node v_j chooses its self-weight and the weight of its outgoing links at each time instant k to be $(1 + \mathcal{D}_j^+[k])^{-1}$ (such that the sum of all weights $p_{lj}[k]$, $v_l \in \mathcal{N}_j^+[k] \cup \{v_j\}$, assigned by each node v_j to links to its out-neighbors and itself at time step k is equal to 1). First, suppose the nodes experience only changes in interconnection topology but no delays. When each node updates its information state $x_j[k]$ using equation (4), the information state for the whole network is given by $x[k+1] = P[k]x[k]$, where $P[k]$ depends on the links present at time instant k. For example, at time instants $k = k_1$ and $k = k_2$, the interconnection topologies are captured by the graphs in Figures 2(a) and 2(b).



(a) Connections and weights at instant k_1.

(b) Connections and weights at instant k_2.

Fig. 2. A network of six nodes, where each node v_j chooses its self-weight and the weight of the links to its out-neighbors to be $(1 + \mathcal{D}_j^+[k])^{-1}$.

We use twice the update formula (4) with initial conditions $y[0] = (-1 \quad 1 \quad 2 \quad 3 \quad 4 \quad 3)^T$ and $z[0] = \mathbb{1}^T$ respectively. At each iteration, we generate a new random graph with six nodes that includes a directed link (v_j, v_i) from node v_i to node v_j ($v_i, v_j \in V$, $v_i \neq v_j$) with some probability p independently between different links. Note that once the graph is chosen at iteration k, the update matrix $P[k]$ will be column stochastic. A realization of the ratios at each node is shown in Figure 3; in this case, the average is 2. When delays are present with maximum delay $\bar{\tau} = 5$ we use the update formula (5) with the same initial conditions and we observe that the system again converges to the exact average.

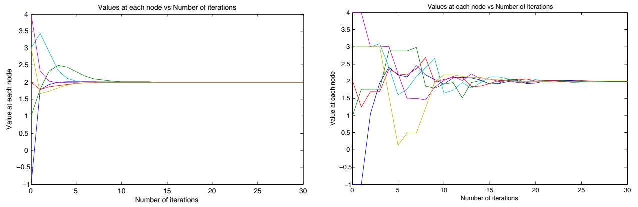


Fig. 3. We use twice the update formula (4) with initial conditions $y[0] = (-1 \quad 1 \quad 2 \quad 3 \quad 4 \quad 3)^T$ and $z[0] = \mathbb{1}^T$ respectively, and plot the ratio $y_j[k]/z_j[k]$ for each node v_j under changing interconnection topology but no delays (left). When delays are present (maximum delay $\bar{\tau} = 5$) we use the update formula (5) with the same initial conditions and observe that the ratios again converge to the average, but with a slower convergence (right).

IV. CONCLUSIONS

In this paper, we studied distributed strategies for a discrete-time networked system to reach asymptotic average consensus in the presence of dynamically changing topologies on top of time-delays. By assuming that nodes in the multi-agent system have knowledge of their out-degree, we have shown that our proposed discrete-time strategy reaches asymptotic average consensus in a distributed fashion, in the presence of dynamically changing interconnection topology for whatever the realization of delays, as long as they are bounded and the union graph of the graph topologies over consecutive time intervals forms a strongly connected graph infinitely often. The results are illustrated via examples.

REFERENCES

[1] 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, June 2003.

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

[3] A. I. Rikos, T. Charalambous, and C. N. Hadjicostis, "Distributed weight balancing over digraphs," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 2, pp. 190–201, June 2014.

[4] A. D. Domínguez-García and C. N. Hadjicostis, "Distributed matrix scaling and application to average consensus in directed graphs," *IEEE Transactions on Automatic Control*, vol. 58, no. 3, pp. 667–681, March 2013.

[5] 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.

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

[7] ——, "Average consensus on arbitrary strongly connected digraphs with time-varying topologies," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 1066–1071, April 2014.

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

[9] C. N. Hadjicostis and T. Charalambous, "Average consensus in the presence of delays in directed graph topologies," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 763–768, March 2014.

[10] N. Bambos, S. C. Chen, and G. J. Pottie, "Channel access algorithms with active link protection for wireless communication networks with power control," *IEEE/ACM Transactions on Networking*, vol. 8, no. 5, pp. 583–597, May 2000.

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

[12] N. H. Vaidya, C. N. Hadjicostis, and A. D. Domínguez-García, "Robust average consensus over packet dropping links: Analysis via coefficients of ergodicity," in *IEEE Conference on Decision and Control (CDC)*, Dec. 2012, pp. 2761–2766.