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 $I$ 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 $A > 0$.

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) $G(\mathcal{V}, \mathcal{E})$ 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 $\varepsilon_{ij} \triangleq (v_j, v_i) \in \mathcal{E}$ 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 $v_i, v_j \in \mathcal{V}$, $v_i \neq v_j$, one can find a sequence of nodes $v_i = v_{i1}, v_{i2}, v_{i3}, \ldots, v_{it} = v_j$ such that link $(v_{ik+1}, v_{ik}) \in \mathcal{E}$ 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 $\mathcal{N}^{-}_j = \{v_i \in \mathcal{V} | \varepsilon_{ij} \in \mathcal{E}\}$. The cardinality of $\mathcal{N}^{-}_j$, is called the in-degree of $v_j$ and is denoted by $D^{-}_j = |\mathcal{N}^{-}_j|$. The nodes that receive information from node $v_j$ belong to the set of out-neighbors of node $v_j$, denoted by $\mathcal{N}^{+}_j = \{v_i \in \mathcal{V} | \varepsilon_{ji} \in \mathcal{E}\}$. The cardinality of $\mathcal{N}^{+}_j$, is called the out-degree of $v_j$ and is denoted by $D^{+}_j = |\mathcal{N}^{+}_j|$. 

In the type of algorithms we will consider, we will associate a positive weight $p_{ij}$ for each edge $\varepsilon_{ij} \in \mathcal{E} \cup \{(v_j, v_i) \in \mathcal{E} \setminus \mathcal{E} \mid v_j \in \mathcal{V}\}$. The nonnegative matrix $P = [p_{ij}] \in \mathbb{R}^{n \times n}_+$ (with $p_{ij}$ as the entry at its $j$th row, $i$th 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 $p_{ij}[k]$ 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 $p_{ij}[k]$ on its out-going links at time $k$, i.e., $v_i \in \mathcal{N}^{+}_j[k]$. Hence, in the general case, each node updates its information state $x_j[k+1]$, $k = 0, 1, 2, \ldots$, according to:

$$x_j[k+1] = p_{jj}[k]x_j[k] + \sum_{v_i \in \mathcal{N}^{-}_j[k]} p_{ji}[k]x_i[k] = p_{jj}[k]x_j[k] + \sum_{v_i \in \mathcal{N}^{-}_j[k]} x_{j-i}[k],$$

(1)

where $x_{j-i}[k] \triangleq p_{ji}[k]x_i[k]$, $x_i[k] \in \mathbb{R}$, 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 send $x_{j-i}[k]$ 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 $P[k] = [p_{ij}[k]] \in \mathbb{R}^{n \times n}_+$, then (1) can be written in matrix form as

$$x[k+1] = P[k]x[k].$$

(2)

Note that, with the exception of the diagonal entries, we have $p_{jj}[k] = 0$, $j \neq i$, if and only if $(v_j, v_i) \notin \mathcal{E}[k]$. We say that the nodes asymptotically reach average consensus if $\lim_{k \to \infty} x_j[k] = \frac{1}{n} \sum_{i=0}^{n} x_i[0]$ 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 $p_{ij} = \frac{1}{1+D^{-}_j[v_j]}$, $\forall (v_i, v_j) \in \mathcal{E}$, 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 $G(\mathcal{V}, \mathcal{E})$. 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

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

(3a)

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

(3b)

where $p_{ij} = \frac{1}{1+D^{-}_j[v_i]}$ for $v_i \in \mathcal{N}^{+}_j \cup \{v_j\}$ (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 $z[0] = 1$. Then, the
solution to the average consensus problem can be asymptotically obtained as
\[
\lim_{k \to \infty} \mu_j[k] = \frac{\sum_{v \in V} y_{0}(\ell)}{|V|}, \quad \forall v_j \in V,
\]
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 \( p_{lj}[k], v_i \in \mathcal{N}_{lj}^+[k] \), 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 \( \mathcal{N}_{lj}^+[k] \) at each time step (in such case, each node \( v_j \in V \) sets \( p_{lj}[k] = \frac{1}{1 + |D^+_j[k]|} \) for \( v_i \in \mathcal{N}_{lj}^+[k] \cup \{j\} \)). 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 \( \{x_{j-i}[k] \mid v_i \in \mathcal{N}_{j-i}^+[k]\} \) (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 \( \{x_i[s] \mid s \leq k, v_i \in \mathcal{N}_{j}^-[s], s + \tau_i[k] = k\} \). Integer \( \tau_i[k] \geq 0 \) represents the delay of a message sent from node \( v_i \) to node \( v_j \) at time \( k \). We require that \( 0 \leq \tau_i[k] \leq \bar{\tau} \leq \bar{\tau} \) for all \( k \geq 0 \) for some finite \( \bar{\tau} = \max\{\tau_i[k]\}, \bar{\tau} \in \mathbb{Z}_+ \). 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 \( \mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\nu\}, \nu \in \mathbb{N}, \nu \leq 2^{n^2-n} \), be the set of all possible directed graphs 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 \( D_j^+[k] \)).

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

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

\[
x_j[k+1] = p_{j}x_j[k] + \sum_{v_i \in \mathcal{N}_{j}^[k]} x_{j-i}[k], \quad (4)
\]

where \( k = 0, 1, \ldots, x_{j-i}[k] \triangleq p_{ji}[k]x_i[k] \) 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, 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 \(p_{lj}[k] = \frac{1}{1+D_j[k]}\) for \(l = j\) and \((v_l,v_j) \in E[k]\) (this choice satisfies \(\sum_{l=1}^{n} p_{lj}[k] = 1\) for all \(v_j \in V\)). Note that unspecified weights in \(P[k]\) are set to zero and correspond to pairs of nodes \((v_i,v_j)\) that are not connected at time step \(k\), i.e., \(p_{lj}[k] = 0\), for all \((v_i,v_j) \notin E, j \neq i\). If we let \(x[k] = (x_1[k] x_2[k] \ldots x_n[k])^T\) and \(P[k] = [p_{lj}[k]]\in \mathbb{R}^{n \times n}\) 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 \(v_t\) 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, \(v_j\) can send the message to node \(v_t\) using some path in the directed graph or using some sort of flooding scheme). When node \(v_j\) receives the request from \(v_t\), it sends an acknowledgement packet (directly to node \(v_j\)) and the communication link is initiated. In practice, this might not necessarily require node \(v_j\) to transmit a separate package to node \(v_t\) (e.g., in a wireless broadcast setting) or to transmit at a higher power (e.g., if \(v_i\) is already in its range); however, it does imply that node \(v_j\) will adjust its self-weight and the weights \(p_{lj}, v_l \in N_j^+,\) on the links to its out-neighbors in order to ensure that column stochasticity is preserved. If, on the other hand, node \(v_t\) 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 \(v_j\) receives an acknowledgement from node \(v_t\) along with the latest message with values for the last update, then the link can be terminated. If node \(v_t\) 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 \(v_t\), it is enough to simply initiate such a request to node \(v_j\) (a direct link is available).

**Lemma 2:** Consider an infinite sequence of graphs of the form \(G[k] = (\mathcal{V},E[k])\), \(k = 0,1,2,\ldots\) such that there exists a finite time window length \(\ell\) and an infinite sequence of time instants \(t_0,t_1,t_2,\ldots\), 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 \(G[t_m],G[t_{m+1}],\ldots,G[t_{m+1}-1]\) is strongly connected. Let \(y_j[k], \forall v_j \in V\), be the result of iteration (4) with \(p_{lj}[k] = \frac{1}{1+D_j[k]}\) for \(v_l \in N_j^+[k] \cup \{v_j\}\) (zeros otherwise) and initial conditions \(y_j[0] = y_0\), and let \(z_j[k], \forall v_j \in V\), be the result of iteration (4) with \(p_{lj}[k] = \frac{1}{1+D_j[k]}\) for \(v_l \in N_j^+[k] \cup \{v_j\}\) (zeros otherwise) and with initial condition \(z_j[0] = 1\). Then, the solution to the average consensus problem in the presence of dynamically changing topologies can be obtained as \(\lim_{k \rightarrow \infty} \mu_j[k] = \frac{\sum_{v_j \in \mathcal{V}} y_j[k]}{|\mathcal{V}|}\), \(\forall v_j \in V\), where \(\mu_j[k] = y_j[k]/z_j[k]\).

**Proof:** Let \(\overline{P}_{tm+1-t_m} \triangleq P[tm+1-1]P[tm+1-2]\ldots P[t_m]\). Since the union of graphs from time instant \(t_m\) until \(t_{m+1}-1\), i.e., the union of graphs \(G[t_m],G[t_{m+1}],\ldots,G[t_{m+1}-1]\) is strongly connected and each matrix involved in the product has strictly positive elements on the diagonal, matrix \(\overline{P}_{tm+1-t_m}\) is SIA\(^2\) for \(m \in \mathbb{Z}_+\). Furthermore, products of matrices of the form \(\overline{P}_{tm+1-t_m}\) 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 \(\nu(\epsilon)\) stochastic matrices of the form \(\overline{P}_{tm+1-t_m}\) has all of its columns approximately the same, i.e., \(\overline{P}_{tk+\nu(\epsilon)-tk+\nu(\epsilon)-1}\ldots \overline{P}_{tk+2-tk+1}\overline{P}_{tk+1-tk} \rightarrow c_{\nu(\epsilon)+1}\), where \(c_{\nu(\epsilon)+1}\) is a positive \(\nu(\epsilon)\)-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:

\[
x_j[k+1] = p_{lj}[k]x_j[k] + \sum_{r=0}^{\bar{\tau}} \sum_{v_r \in \mathcal{N}_j^-[k-r]} x_{j-r}[k-r]I_{k-r,ji}[r],
\]

for \(k = 0, 1, 2, \ldots\), where

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

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

To handle delays in a network of \(n = |V|\) nodes, we introduce \(\bar{\tau}n\) nodes (for a total of \((\bar{\tau} + 1)n\) nodes) so that we can write

\[
\overline{x}[k+1] = \overline{P}[k]\overline{x}[k],
\]

where (as in [9])

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

with

\[
\overline{x}[k] = \begin{pmatrix} x^T[k] & x^{(1)}[k] & \cdots & x^{(\bar{\tau})}[k] \end{pmatrix}^T,
\]

\[
x^{(r)}[k] = \begin{pmatrix} x_1^{(r)}[k] & \cdots & x_n^{(r)}[k] \end{pmatrix}, \quad r = 1, 2, \ldots, \bar{\tau}.
\]

\(^2\)In [11], a stochastic matrix \(P\) is called indecomposable and aperiodic (SIA) if \(Q = \lim_{k \rightarrow \infty} P^k\) exists and all the rows of \(Q\) are the same.
As in [9], $P_0[k], P_1[k], \ldots, P_\tau[k]$ are appropriately defined column stochastic matrices, such that $P[k] = \sum_{r=0}^{\tau} P_r[k]$, i.e., the sum $P[k]$ of all the nonnegative matrices $P_r[k], r \in \{0, 1, 2, \ldots, \tau\}$, 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 $G[k] = (V, E[k]), k = 0, 1, 2, \ldots$ such that there exists a finite time window length $\ell$ 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 $G[t_m], G[t_m+1], \ldots, G[t_{m+1} - 1]$ is strongly connected. Let $y_{lj}[k]$ for all $v_j \in V$ be the result of iteration (5) with $p_{lj}[k] = 1 + \frac{1}{1+D_j^{+}[k]}$ for $v_l \in N_j^+[k] \cup \{v_j\}$ (zeros otherwise) and initial conditions $y_{lj}[0] = y_{0}$, and let $z_{lj}[k], \forall v_l, v_j \in V$, be the result of iteration (4) with $p_{lj}[k] = \frac{1}{1+D_j^{+}[k]}$ for $v_l \in N_j^+[k] \cup \{v_j\}$ (zeros otherwise) and with initial condition $z_{lj}[0] = 1$. The indicator function $I_{k, lj}$ captures the bounded delay $\tau_{lj}[k]$ on link $(v_j, v_l)$ at iteration $k$ (as defined in (6), $\tau_{lj}[k] \leq \tau$). Then, the solution to the average consensus problem can be asymptotically obtained as $\lim_{k \to \infty} \mu_{lj}[k] = \frac{\sum_{v_l \in V} y_{0}(l)}{|V|}, \forall v_j \in V$, where $\mu_{lj}[k] = \frac{y_{lj}[k]}{z_{lj}[k]}$.

We now discuss the case in which a node, say $v_j$, chooses an acknowledgment (e.g., an acknowledgment message) that one of its out-neighbors, say $v_l \in N_j^+[k-1]$, no longer receives its transmissions. In other words, node $v_l \notin N_j^+[k]$ 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 communication protocols in a variety of ways, e.g., when

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 broadcast to each neighboring node $v_j \in N_j^+$ 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 $v_l \in N^+_j$ (the total mass can be different for each node $v_j$ due to, for example, newly established communication links), and the total mass received from each neighboring node $v_l \in N^-_j$ that transmits information to node $v_j$. Since different information might be needed at each node $v_j$ 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 $(1 + D^+_j[k])^{-1}$ (such that the sum of all weights $p_{ij}[k]$, $v_l \in N^+_j[k] \cup \{v_j\}$, 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).

We use twice the update formula (4) with initial conditions $y[0] = (-1 1 2 3 4 3)^T$ and $z[0] = I^3$ 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 $\tau = 5$) 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


