2018 Annual American Control Conference (ACC)
June 27–29, 2018. Wisconsin Center, Milwaukee, USA

Fully Asynchronous Push-Sum With Growing Intercommunication Intervals*

Alex Olshevsky, Ioannis Ch. Paschalidis[†] and Artin Spiridonoff[‡]

Abstract— **We propose an algorithm for average consensus over a directed graph which is both fully asynchronous and robust to unreliable communications. We show its convergence to the average while allowing for slowly growing but potentially unbounded communication failures.**

I. Introduction

Consider a set of agents, whose goal is to reach consensus by exchanging information locally with their neighbors through a directed graph. There is a large body of work on consensus algorithms. Ordinary consensus has been shown to converge asymptotically under various scenarios such as growing intercommunicating intervals [1], presence of delays and/or unbounded intercommunication intervals [2]. Another problem of interest for which extensive research has been carried out is *average* consensus. While most related works study asymptotic convergence, [3] studies average consensus in a finite number of steps. Push-sum is one of the many algorithms for average consensus that was first proposed by [4]. This algorithm has been widely used to develop protocols that reach average consensus, under different assumptions and scenarios; such as the presence of bounded delays [5], time varying graphs [6][7], or asynchronous communication [8].

Since reliable communication is a very restrictive assumption in network applications, or expensive to enforce, recent work has considered algorithms that reach consensus in a setting where communication between agents is unreliable. While in this case, push-sum might not converge to average, exponential convergence still holds and the error between the final value and the true average can be characterized [9]. In [10], Vaidya et al. introduce the technique of running sums (counters) and modify push-sum to overcome possible packet drops and imprecise knowledge of the network in a synchronous communication setting. They prove *almost surely* convergence of their algorithms using weak ergodicity. Inspired by [10], [11] takes this further and develops an asynchronous algorithm for average consensus, which is robust to unreliable communication. This algorithm uses a *broadcast asymmetric* communication protocol; that is, at each iteration only one node is allowed to wake up and transmit information to its neighbors. Exponential convergence of this algorithm is proved under bounded consecutive link failures and nodes' update delays.

Consensus and average consensus have a lot of application in other algorithms as well; they can be used as a building block to develop distributed optimization algorithms [12][13]. For example, in [14] the authors use a robust version of push-sum as a building block to develop an asynchronous Newton-based distributed optimization algorithm, robust to packet losses.

A lot of available works in the literature assume bounded intercommunication intervals; which motivated us to study and explore sufficient connectivity conditions which allow intercommunication intervals to slowly grow and potentially be unbounded. We propose logarithmically growing upper bounds which guarantee convergence.

Distributed synchronous systems require coordination between the agents. Asynchronous systems, in contrast, do not depend on global clock signals. This can save power as agents do not have to perform computation and communication at every iteration. However, it might require more iterations to converge. While existing works on push-sum in the presence of link failures assume synchronous [10] or broadcast asymmetric [11] communication setting, our major contribution in this paper is to develop a *fully asynchronous* robust push-sum algorithm that allows the successive link failures to grow to infinity.

The rest of the paper is organized as follows. In Section II we introduce our notation and define the problem. In Sections III and IV we study ordinary consensus and push-sum algorithms, respectively, and state our convergence results. In Section V, we propose an asynchronous push-sum algorithm which is robust to unreliable communication links, followed by concluding remarks in Section VI.

II. Problem Formulation

A. Notations and Definitions

Suppose $\mathbf{A}$ is a matrix, by A_{ij} we denote its (i,j) entry. A matrix is called *(row) stochastic* if it is non-negative and the sum of the elements of each row equals to one. Similarly, a matrix is *column stochastic* if its transpose is stochastic. A matrix is called *doubly stochastic* if it is both column and row stochastic.

To a non-negative matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ we associate a directed graph $\mathcal{G}_{\mathbf{A}}$ with vertex set $\mathcal{N} = \{1, 2, \ldots, n\}$ and edge set $\mathcal{E}_{\mathbf{A}} = \{(i,j)|A_{ji} > 0\}$. Note that the graph might contain self-loops.

* Research was partially supported by the NSF under grants CNS-1645681, CCF-1527292, IIS-1237022, and IIS-1724990, by the ARO under grant W911NF-12-1-0390, by the NIH under grant 1UL1TR001430, by the Boston University Digital Health Initiative, and by the joint Boston University and Brigham & Women's Hospital program in Engineering and Radiology.

[†] A. Olshevsky and I. Paschalidis are with the ECE Department, Boston University, {alexols,yannisp}@bu.edu.

[‡] A. Spiridonoff is at the Division of Systems Engineering, Boston University, artin@bu.edu.

978-1-5386-5428-6/$31.00 ©2018 AACC

By $[\mathbf{A}]_\alpha$ we denote the *thresholded* matrix obtained by setting every element of $\mathbf{A}$ smaller than α to zero.

Given a sequence of matrices $\mathbf{A}^0, \mathbf{A}^1, \mathbf{A}^2, \ldots$, we denote by $\mathbf{A}^{k_2:k_1}$, $k_2 \geq k_1$, the product of elements k_1 to k_2 of the sequence, inclusive, in the following order:

$$\mathbf{A}^{k_2:k_1} = \mathbf{A}^{k_2}\mathbf{A}^{k_2-1}\cdots\mathbf{A}^{k_1}.$$

Node i is an *in-neighbor* of node j, if there is a directed link from i to j. Hence j would be an *out-neighbor* of node i. We denote the set of in-neighbors and out-neighbors of node i at time k with $N_i^{-,k}$ and $N_i^{+,k}$, respectively. Moreover, we denote the number of in-neighbors and out-neighbors of node i at time k with $d_i^{-,k}$ and $d_i^{+,k}$, as its *in-degree* and *out-degree*, respectively. If the graph is fixed, we will simply drop the index k in the aforementioned notations.

By $x_{\min}$ and $x_{\max}$ we denote $\min_i x_i$ and $\max_i x_i$, respectively, unless mentioned otherwise. We also denote a $n \times 1$ column vector of all ones by $\mathbf{1}_n$, or $\mathbf{1}$ when its size is clear from the context.

We sometimes use the notion of *mass* to denote the value an agent holds, sends or receives. With that in mind, we can think of a value being sent from one node, as a mass being transferred.

B. Problem Formulation

Consider a set of n agents $\mathcal{N} = \{1, 2, \ldots, n\}$, where each agent i holds an initial scalar value x_i^0. These agents communicate with each other through a sequence of directed graphs. Our goal is to develop protocols through which these agents communicate and update their values so that they reach consensus. Throughout this paper we use the terms *agents* and *nodes* interchangeably.

Ordinary consensus and push-sum are two main algorithms proposed for this purpose. In ordinary consensus, each node updates its value by forming a convex combination of the values of its in-neighbors. In push-sum, average consensus is reached by running two parallel iterations in which, each node splits and sends its value to its out-neighbors and updates its own value by forming the sum of the messages that it has received.

III. Ordinary Consensus

Although the main target of this paper is push-sum, in this section we state and prove similar results for ordinary consensus. Comparable results can be found in [1], however the proofs provided here are necessary to understand the methods used in the following sections.

Linear consensus is defined as,

$$\mathbf{x}^{k+1} = \mathbf{A}^k \mathbf{x}^k, \quad k = 0, 1, \ldots, \tag{1}$$

where the matrices $\mathbf{A}^k$ are stochastic and $\mathbf{x}^k$ is constructed by collecting all x_i^k in a column vector. Under the following conditions, the iteration (1) results in consensus, meaning all the x_i^k converge to the same value as $k \to \infty$.

The following assumption ensures sufficient connectivity of the graphs.

Assumption 1. *There exist a sequence $b_1, b_2, \ldots$ of positive integers such that when we partition the sequence of graphs $\mathcal{G}^0, \mathcal{G}^1, \mathcal{G}^2, \ldots$ to consecutive blocks of length b_k, $k = 1, 2, \ldots$, the graph constructed by the union of the edges in each block, is strongly connected. Also each graph $\mathcal{G}^k$ has a self-loop at every node.*

Let us define $\mu_0 = \lambda_0 = 0$, and for $k \geq 1$:

$$\mu_k = \sum_{j=1}^{k} b_j, \tag{2}$$

$$\lambda_k = \sum_{j=(k-1)n+1}^{kn} b_j = \mu_{kn} - \mu_{(k-1)n}. \tag{3}$$

The following proposition states sufficient conditions for the convergence of ordinary consensus with growing inter-communication intervals.

Proposition 1. *Suppose there exist some $\alpha > 0$ such that the sequence of graphs $\mathcal{G}_{[\mathbf{A}^0]_\alpha}, \mathcal{G}_{[\mathbf{A}^1]_\alpha}, \mathcal{G}_{[\mathbf{A}^2]_\alpha}, \ldots$ satisfies Assumption 1. If there exist some $K \geq 1, T \geq 0$, such that $\lambda_k \leq -\frac{\ln(k+T)}{\ln(\alpha)}$ for all $k \geq K$, then x^k converges to a limit in span$\{\boldsymbol{I}\}$.*

Before proving the proposition, we need the following lemmas.

Lemma 1. *Suppose there exists some $\alpha > 0$ such that the sequence of graphs $\mathcal{G}_{[\mathbf{A}^0]_\alpha}, \mathcal{G}_{[\mathbf{A}^1]_\alpha}, \mathcal{G}_{[\mathbf{A}^2]_\alpha}, \ldots$ satisfies Assumption 1. Then for $l \geq 0$, $\mathbf{A}^{\mu_{l+n}-1:\mu_l}$ is a strictly positive matrix, with its elements at least $\alpha^{\mu_{l+n}-\mu_l}$.*

This lemma is proved in [15].

Lemma 2. *Suppose $\mathbf{A}$ is a stochastic matrix with entries at least $\beta > 0$. If $\boldsymbol{v} = \boldsymbol{A}\boldsymbol{u}$ then,*

$$v_{\max} - v_{\min} \leq (1 - n\beta)\,(u_{\max} - u_{\min})\,. \tag{4}$$

This lemma is proved in [16, Theorem 3.1 & Exercise 3.8].

Lemma 3. *Suppose $\mathbf{A}$ is a stochastic matrix and $\boldsymbol{v} = \boldsymbol{A}\boldsymbol{u}$. Then for all i,*

$$u_{\min} \leq v_i \leq u_{\max}. \tag{5}$$

This lemma holds true because each v_i is a convex combination of elements of $\mathbf{u}$.

Lemma 4. *Suppose $0 < \alpha_k < 1$ for $k = 1, \ldots, \infty$, then $\prod_{k=1}^{\infty}(1-\alpha_k) = 0$ if and only if $\sum_{k=1}^{\infty}\alpha_k = \infty$.*

This lemma is proved in [17, Appendix: Theorem 1.9] and we will skip the proof here.

Proof of Proposition 1. By Lemma 1, we have for $k \geq 1$,

$$\left[\mathbf{A}^{\mu_{kn}-1:\mu_{(k\ \ 1)n}}\right]_{ij} \geq \alpha^{\mu_{kn}-\mu_{(k\ \ 1)n}} = \alpha^{\lambda_k}.$$

Applying Lemma 2, we get,

$$x_{\max}^{\mu_{kn}} - x_{\min}^{\mu_{kn}} \leq \left(1 - n\alpha^{\lambda_k}\right)\left(x_{\max}^{\mu_{(k\ \ 1)n}} - x_{\min}^{\mu_{(k\ \ 1)n}}\right). \tag{6}$$

Hence, using (6) for $k = 1, \dots, l$ we obtain,

$$x_{\max}^{\mu_{l_n}} - x_{\min}^{\mu_{l_n}} \leq \prod_{k=1}^{l} \left(1 - n\alpha^{\lambda_k}\right) \left(x_{\max}^{0} - x_{\min}^{0}\right).$$

We have $0 < \alpha < 1$ and $\lambda_k \leq -\frac{\ln(k+T)}{\ln(\alpha)}$ for all $k \geq K$. It follows,

$$\sum_{k=1}^{\infty} n\alpha^{\lambda_k} \geq \sum_{k=K}^{\infty} n\alpha^{\lambda_k} \geq \sum_{k=K}^{\infty} n\alpha^{-\frac{\ln(k+T)}{\ln(\alpha)}}$$
$$= \sum_{k=K}^{\infty} n\left(\alpha^{\frac{1}{\ln(\alpha)}}\right)^{-\ln(k+T)} = \sum_{k=K}^{\infty} \frac{n}{k+T} = \infty.$$

Using Lemmas 3 and 4 and (6) we conclude that Proposition 1 holds. □

Proposition 1 proves the convergence of x_i^k's to a value which is not necessarily the total average and depends on the sequence of matrices. However if the matrices $\mathbf{A}^k$ are doubly stochastic, the sum of the values of all nodes (agents) is preserved and therefore the algorithm converges to *average* consensus.

Slight modifications to Example 1.2, Chapter 7 of [18] shows that if intercommunication intervals grow logarithmically in time, ordinary consensus fails to reach consensus.

IV. Push-Sum

Push-sum is an algorithm that reaches average consensus and does not require *doubly* stochastic matrices, as opposed to ordinary average consensus. Here, we assume each node knows its out-degree at every iteration. Under this assumption, it turns out that average consensus is possible and may be accomplished using the following iteration,

$$\begin{aligned} x_i^{k+1} &= \sum_{j \in N_i^{-,k}} \frac{x_j^k}{d_j^{+,k}}, \\ y_i^{k+1} &= \sum_{j \in N_i^{-,k}} \frac{y_j^k}{d_j^{+,k}}, \\ z_i^{k+1} &= \frac{x_i^{k+1}}{y_i^{k+1}}, \end{aligned} \qquad (7)$$

where the auxiliary variables y_i are initialized as $y_i^0 = 1$ and are collected in a column vector $\mathbf{y}$. This iteration is implemented in a distributed way using two steps. First each node i broadcasts $x_i^k / d_i^{+,k}$ to its out-neighbors. Next, every node sets x_i^{k+1} to be the sum of the incoming messages. Variables y_i^k follow the same evolution. z_i^k may be thought of as node i's estimation of the average.

We define $\mathbf{W}^k$ to be the matrix such that iteration (7) may be written as,

$$\mathbf{x}^{k+1} = \mathbf{W}^k \mathbf{x}^k,$$
$$\mathbf{y}^{k+1} = \mathbf{W}^k \mathbf{y}^k.$$

Next, we will state a proposition regarding the sufficient conditions for the push-sum algorithm to converge.

Proposition 2. *Suppose the sequence of graphs $\mathcal{G}_{W^0}, \mathcal{G}_{W^1}, \mathcal{G}_{W^2}, \dots,$ satisfies Assumption 1. If there exist some $K \geq 1, T \geq 0$, such that $\lambda_k \leq \frac{\ln(k+T)}{2\ln(n)}$ for all $k \geq K$, by implementing the push-sum algorithm (7), it follows*

$$\lim_{k \to \infty} z_i^k = \frac{\sum_{j=1}^{n} x_j^0}{n}.$$

This proposition is proved in [15].

V. Robust Asynchronous Push-Sum

Here we describe and study another algorithm for average consensus, in which the communication system is asynchronous and unreliable. In an unreliable setting, communication links might fail to transmit data packets and information might get lost.

This algorithm is originally inspired by the algorithm proposed by [10], but under asynchronous communication. As the algorithm in [10], this algorithm is also based on the push-sum consensus. [11] has proved exponential convergence of this algorithm for the case when at each iteration only one node wakes up and transmits. Here we modify the algorithm presented by [11] and show that average consensus still holds while allowing for any subset of nodes to perform updates at each iteration.

In this algorithm, as opposed to the previous ones, we assume nodes do not have self-loops.

Algorithm 1 Robust Asynchronous Push-Sum

```
Initialize the algorithm with $\mathbf{y}^0 = \mathbf{1}$, $\sigma_i^{x,0} = \sigma_i^{y,0} = 0$ $\forall i \in \{1, \dots, n\}$ and $\rho_{ji}^{x,0} = \rho_{ji}^{y,0} = 0$, $\forall (i,j) \in \mathcal{E}$.
At every iteration $k$, for every node $i$:
if node $i$ wakes up then
    $\sigma_i^{x,k+1} = \sigma_i^{x,k} + \frac{x_i^k}{d_i^+ + 1}$;
    $\sigma_i^{y,k+1} = \sigma_i^{y,k} + \frac{y_i^k}{d_i^+ + 1}$;
    $x_i^{k+1} = \frac{x_i^k}{d_i^+ + 1}$;
    $y_i^{k+1} = \frac{y_i^k}{d_i^+ + 1}$;
    Node $i$ broadcasts $\sigma_i^{x,k+1}$ and $\sigma_i^{y,k+1}$ to its out-neighbors: $N_i^+$
end if
if node $i$ receives $\sigma_j^{x,k+1}$ and $\sigma_j^{y,k+1}$ from $j \in N_i^-$ then
    $\rho_{ij}^{x,k+1} = \sigma_j^{x,k+1}$;
    $\rho_{ij}^{y,k+1} = \sigma_j^{y,k+1}$;
    $x_i^{k+1} = x_i^{k+1} + \rho_{ij}^{x,k+1} - \rho_{ij}^{x,k}$;
    $y_i^{k+1} = y_i^{k+1} + \rho_{ij}^{y,k+1} - \rho_{ij}^{y,k}$;
end if
Other variables remain unchanged.
```

The impressive idea proposed by [10] that allows us to overcome the issue of unreliable of links, is that of introducing the counters: in particular each node i has a counter $\sigma_i^{x,k}$ ($\sigma_i^{y,k}$ respectively) to keep track of the total x-mass (y-mass) sent by itself to its neighbors from time 0 to time k, and counters $\rho_{ij}^{x,k}$ ($\rho_{ij}^{y,k}$ respectively) $\forall j \in N_i^-$, to take into account the total x-mass (y-mass) received from its neighbor j from time 0 to time k.

While in reality, nodes will perform computations when they wake up; to make the analysis easier, we assume nodes perform computations (but no transmission) when they are not awake.

Next, we state and prove the main theorem of this paper, which shows that the algorithm above reaches average consensus under sufficient connectivity assumptions.

Theorem 1. *Suppose we apply the Robust Asynchronous Push-Sum algorithm to a set of agents communicating with each other through a strongly connected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{E}$ does not have self-loops. Let $\mathcal{G}^0, \mathcal{G}^1, \ldots,$ be the sequence of graphs $\mathcal{G}^i = (\mathcal{N}, \mathcal{E}^i)$, $\mathcal{E}^i \subset \mathcal{E}$, containing only the links which transmit successfully at iteration i. Also, suppose there is another sequence $b_1, b_2, \ldots,$ of positive integers such that, if we split the sequence of $\mathcal{G}^0, \mathcal{G}^1, \ldots,$ to consecutive blocks of length b_i, the union of graphs of each block is equal to $\mathcal{G}$; i.e., $\cup_{i=\mu_k}^{\mu_{k+1}-1} \mathcal{E}^i = \mathcal{E}, \forall k \geq 0$, where μ_k and λ_k are defined in (2) and (3). Suppose that there exists some $K \geq 1, T \geq 0$, such that $\lambda_k \leq \frac{\ln(k+T)}{6\ln(n)}$, $\forall k \geq K$. Then, $z_i^k = x_i^k / y_i^k$ converges to the average of $\mathbf{x}^0$, i.e.,*

$$\lim_{k\to\infty} z_i^k = \frac{\sum_{j=1}^n x_j^0}{n}.$$

Proof. Similar to the proofs of the previous propositions, here we first rewrite the evolution of $\mathbf{x}^k$ and $\mathbf{y}^k$ in a matrix form. We show these matrices are column stochastic. Then we write the evolution of the agents' estimate of the average, $\mathbf{z}^k$, in matrix form. Finally, we exploit the properties of these matrices to show the convergence of z_i^k to one limit which turns out to be the average.

Before we rewrite the iteration in a matrix form, we introduce the indicator variables τ_i^k, for $i = 1, 2, \ldots, n$, and τ_{ij}^k, for $(i,j) \in \mathcal{E}$. τ_i^k is equal to 1 if node i wakes up at time k, and is 0 otherwise. Likewise τ_{ij}^k is 1 whenever node i wakes up at time k, $j \in N_i^+$ and the edge (i,j) is reliable, while it is 0 otherwise.

Let us introduce the following variables:

$$u_{ij}^k = \sigma_i^{x,k} - \rho_{ji}^{x,k}, \qquad \forall (i,j) \in \mathcal{E},$$
$$v_{ij}^k = \sigma_i^{y,k} - \rho_{ji}^{y,k}, \qquad \forall (i,j) \in \mathcal{E},$$

which are, intuitively, the total x-mass and y-mass, respectively, that has been sent by node i but due to link failures has not been delivered to node j yet. The evolution of y-mass is exactly the same as x-mass; hence to avoid repetition, we only analyze the evolution of $\mathbf{x}^k$ and u_{ij}^k. We can write the update equations:

$$u_{ij}^{k+1} = \left(1 - \tau_i^k \tau_{ij}^k\right) \left(u_{ij}^k + \tau_i^k \frac{x_i^k}{d_i^+ + 1}\right), \tag{8}$$

$$x_i^{k+1} = \sum_{j \in N_i} \left(\frac{x_j^k}{d_j^+ + 1} + u_{ji}^k\right) \tau_j^k \tau_{ji}^k + x_i^k \left(1 - \tau_i^k + \frac{\tau_i^k}{d_i^+ + 1}\right). \tag{9}$$

Let us introduce the column vectors $\mathbf{u}^k$ and $\mathbf{v}^k$ which collect all different u_{ij}^k and v_{ij}^k, respectively. Moreover, let us introduce the column vectors $\phi^{(x)}(k) = \left[(\mathbf{x}^k)^T, (\mathbf{u}^k)^T\right]^T$, $\phi^{(y)}(k) = \left[(\mathbf{y}^k)^T, (\mathbf{v}^k)^T\right]^T \in \mathbb{R}^{n+m}$, where $m = |\mathcal{E}|$. Using (8) and (9) we can rewrite the algorithm in the following matrix form:

$$\phi^{(x)}(k+1) = \mathbf{M}^k \phi^{(x)}(k), \tag{10}$$
$$\phi^{(y)}(k+1) = \mathbf{M}^k \phi^{(y)}(k). \tag{11}$$

Lemma 5. *$\mathbf{M}$ is column stochastic and each positive element of it is at least $1/(\max_i\{d_i^+\} + 1)$. Also we have for $1 \leq i \leq n$:*

$$M_{ii}^k = \begin{cases} 1, & \text{if } \tau_i^k = 0, \\ \frac{1}{d_i^+ + 1}, & \text{if } \tau_i^k = 1. \end{cases} \tag{12}$$

Proof of this lemma can be found in [15].

Let us augment the graph $\mathcal{G}^k$ to $\mathcal{H}^k = \mathcal{G}_{\mathbf{M}^k}$ by adding auxiliary nodes b_{ij}, $\forall (i,j) \in \mathcal{E}$. Note that by Lemma 5, node $i \in \{1, \ldots, n\}$ has self-loop all the time and node b_{ij} has self-loop unless the link (i,j) transmits reliably. Let us call nodes b_{ij} *buffers* and assign values u_{ij}^k and v_{ij}^k to them.

The algorithm is equivalent to the following process: Suppose node i wakes up. If the link (i,j) works properly, node i sends some mass ($x_i^k/(d_i^+ + 1)$ and $y_i^k/(d_i^+ + 1)$) to node j and also node b_{ij} sends all of its mass (u_{ij}^k and v_{ij}^k) to node j and becomes zero. Otherwise, the mass is sent from node i to node b_{ij} instead of j. Then all the mass gets accumulated at node b_{ij} because of its self-loop, until the link (i,j) transmits reliably.

Lemma 6. *The first n rows of $\mathbf{M}^{\mu_{l+n}-1:\mu_l}$ are strictly positive, $l \geq 0$. The positive elements of this matrix are at least $(1/n)^{\mu_{l+n}-\mu_l}$.*

This lemma is proved in [15].

Define $\mathbf{W}^k = \mathbf{M}^{\mu_{(k+1)n}-1:\mu_{kn}}, k \geq 0$, which has positive elements of at least $\alpha^{\lambda_{k+1}}$ where $\alpha = 1/n$. Then we have:

$$\begin{bmatrix} \mathbf{x}^{\mu_{(k+1)n}} \\ \mathbf{u}^{\mu_{(k+1)n}} \end{bmatrix} = \mathbf{W}^k \begin{bmatrix} \mathbf{x}^{\mu_{kn}} \\ \mathbf{u}^{\mu_{kn}} \end{bmatrix}, \tag{13}$$

$$\begin{bmatrix} \mathbf{y}^{\mu_{(k+1)n}} \\ \mathbf{v}^{\mu_{(k+1)n}} \end{bmatrix} = \mathbf{W}^k \begin{bmatrix} \mathbf{y}^{\mu_{kn}} \\ \mathbf{v}^{\mu_{kn}} \end{bmatrix}. \tag{14}$$

Let us split the matrix $\mathbf{W}^k$ to four sub-matrices as follows:

$$\mathbf{W}^k = \begin{bmatrix} \mathbf{A}^k & \mathbf{B}^k \\ \mathbf{C}^k & \mathbf{D}^k \end{bmatrix}, \tag{15}$$

where $\mathbf{A}^k \in \mathbb{R}^{n\times n}$, $\mathbf{B}^k \in \mathbb{R}^{n\times m}$, $\mathbf{C}^k \in \mathbb{R}^{m\times n}$ and $\mathbf{D}^k \in \mathbb{R}^{m\times m}$. By Lemma 6 we know that matrices $\mathbf{A}^k$ and $\mathbf{B}^k$ are strictly positive.

For $h = 1, \ldots, m$ define r_h^k as follows:

$$r_h^k = \begin{cases} \frac{u_h^k}{v_h^k}, & \text{if } v_h^k \neq 0, \\ 0, & \text{if } v_h^k = 0. \end{cases}$$

Lemma 7. $u_{ij}^k = 0$ *whenever* $v_{ij}^k = 0$.

Proof of this lemma can be found in [15].

Therefore, the following always holds for $h = 1, \ldots, m$:

$$u_h^k = r_h^k v_h^k, \tag{16}$$

Define $\bar{\mathbf{x}}^k = \mathbf{x}^{\mu_{kn}}$, $\bar{\mathbf{y}}^k = \mathbf{y}^{\mu_{kn}}$, $\bar{\mathbf{u}}^k = \mathbf{u}^{\mu_{kn}}$, $\bar{\mathbf{v}}^k = \mathbf{v}^{\mu_{kn}}$, $\bar{\mathbf{z}}^k = \mathbf{z}^{\mu_{kn}}$ and $\bar{\mathbf{r}}^k = \mathbf{r}^{\mu_{kn}}$. Using (13) and (15) we obtain:

$$\begin{aligned} \bar{z}_i^{k+1}\bar{y}_i^{k+1} = \bar{x}_i^{k+1} &= \sum_{j=1}^{n} A_{ij}^k \bar{x}_j^k + \sum_{j=1}^{m} B_{ij}^k \bar{u}_j^k \\ &= \sum_{j=1}^{n} A_{ij}^k \bar{z}_j^k \bar{y}_j^k + \sum_{j=1}^{m} B_{ij}^k \bar{r}_j^k \bar{v}_j^k. \end{aligned}$$

Hence,

$$\bar{z}_i^{k+1} = \left(\bar{y}_i^{k+1}\right)^{-1} \sum_{j=1}^{n} A_{ij}^k \bar{z}_j^k \bar{y}_j^k + \left(\bar{y}_i^{k+1}\right)^{-1} \sum_{j=1}^{m} B_{ij} \bar{r}_j^k \bar{v}_j^k,$$

$$\mathbf{z}^{k+1} = \left(\mathbf{Y}^{k+1}\right)^{-1} \mathbf{A}^k \mathbf{Y}^k \mathbf{z}^k + \left(\mathbf{Y}^{k+1}\right)^{-1} \mathbf{B}^k \mathbf{V}^k \mathbf{r}^k,$$

where $\mathbf{Y}^k = \operatorname{diag}\left(\bar{\mathbf{y}}^k\right)$ and $\mathbf{V}^k = \operatorname{diag}\left(\bar{\mathbf{v}}^k\right)$. Note that $\bar{\mathbf{y}}^k$ is strictly positive. Similarly, using (14)-(16) we have,

$$\begin{aligned} \bar{r}_i^{k+1}\bar{v}_i^{k+1} = \bar{u}_i^{k+1} &= \sum_{j=1}^{n} C_{ij}^k \bar{x}_j^k + \sum_{j=1}^{m} D_{ij}^k \bar{u}_j^k \\ &= \sum_{j=1}^{n} C_{ij}^k \bar{z}_j^k \bar{y}_j^k + \sum_{j=1}^{m} D_{ij}^k \bar{r}_j^k \bar{v}_j^k. \end{aligned}$$

Here $\bar{\mathbf{v}}^k$, as opposed to $\bar{\mathbf{y}}^k$, is not necessarily strictly positive. Therefore instead of $\left(\mathbf{V}^k\right)^{-1}$, we define the following:

$$\tilde{v}_i^k = \begin{cases} \frac{1}{\bar{v}_i^k}, & \text{if } \bar{v}_i^k \neq 0, \\ 0, & \text{if } \bar{v}_i^k = 0. \end{cases}$$

It follows:

$$\bar{r}_i^{k+1} = \tilde{v}_i^{k+1} \sum_{j=1}^{n} C_{ij}^k \bar{z}_j^k \bar{y}_j^k + \tilde{v}_i^{k+1} \sum_{j=1}^{m} D_{ij}^k \bar{r}_j^k \bar{v}_j^k,$$

$$\mathbf{r}^{k+1} = \tilde{\mathbf{V}}^{k+1} \mathbf{C}^k \mathbf{Y}^k \mathbf{z}^k + \tilde{\mathbf{V}}^{k+1} \mathbf{D}^k \mathbf{V}^k \mathbf{r}^k.$$

where $\tilde{\mathbf{V}}^k = \operatorname{diag}(\tilde{v}^k)$. Thus,

$$\begin{bmatrix} \mathbf{z}^{k+1} \\ \mathbf{r}^{k+1} \end{bmatrix} = \mathbf{P}^k \begin{bmatrix} \mathbf{z}^k \\ \mathbf{r}^k \end{bmatrix}, \tag{17}$$

where,

$$\mathbf{P}^k = \begin{bmatrix} \left(\mathbf{Y}^{k+1}\right)^{-1} \mathbf{A}^k \mathbf{Y}^k & \left(\mathbf{Y}^{k+1}\right)^{-1} \mathbf{B}^k \mathbf{V}^k \\ \tilde{\mathbf{V}}^{k+1} \mathbf{C}^k \mathbf{Y}^k & \tilde{\mathbf{V}}^{k+1} \mathbf{D}^k \mathbf{V}^k \end{bmatrix}. \tag{18}$$

Now we show that the sum of the elements of each row 1 to n of $\mathbf{P}^k$ is equal to 1, but for the rest of the rows they either sum to 1 or they are all zeros.

$$\begin{aligned} \mathbf{P}^k \begin{bmatrix} \mathbf{1}_n \\ \mathbf{1}_m \end{bmatrix} &= \begin{bmatrix} \left(\mathbf{Y}^{k+1}\right)^{-1} \left(\mathbf{A}^k \bar{\mathbf{y}}^k + \mathbf{B}^k \bar{\mathbf{v}}^k\right) \\ \tilde{\mathbf{V}}^{k+1} \left(\mathbf{C}^k \bar{\mathbf{y}}^k + \mathbf{D}^k \bar{\mathbf{v}}^k\right) \end{bmatrix} \\ &= \begin{bmatrix} \left(\mathbf{Y}^{k+1}\right)^{-1} \bar{\mathbf{y}}^{k+1} \\ \tilde{\mathbf{V}}^{k+1} \bar{\mathbf{v}}^{k+1} \end{bmatrix} = \begin{bmatrix} 1_n \\ 1 \text{ or } 0 \\ \vdots \\ 1 \text{ or } 0 \end{bmatrix}. \end{aligned}$$

The $(n+h)^{th}$ row of $\mathbf{P}^k$ is zero if and only if v_h^{k+1} is zero.

Lemma 8. *For $k \geq 0$ and $1 \leq i \leq n$ we have:*

$$\alpha^{\lambda_k} \leq \bar{y}_i^k \leq n. \tag{19}$$

Moreover, for $1 \leq h \leq m$ and $k \geq 1$ we have either $\bar{v}_h^k = 0$ or,

$$\alpha^{\lambda_k+\lambda_{k\;1}} \leq \bar{v}_h^k \leq n. \tag{20}$$

This lemma is proved in [15].

Now we are able to find a lower bound on positive elements of $\mathbf{P}^k$. Let us divide $\mathbf{P}^k$ to four sub-matrices as:

$$\mathbf{P}^k = \begin{bmatrix} \mathbf{E}^k & \mathbf{F}^k \\ \mathbf{G}^k & \mathbf{H}^k \end{bmatrix},$$

where $\mathbf{E}^k \in \mathbb{R}^{n \times n}$, $\mathbf{F}^k \in \mathbb{R}^{n \times m}$, $\mathbf{G}^k \in \mathbb{R}^{m \times n}$ and $\mathbf{H}^k \in \mathbb{R}^{m \times m}$ are defined as in (18).

By construction, positive elements of $\mathbf{E}^k$ and $\mathbf{G}^k$ are at least $\frac{1}{n}\alpha^{\lambda_{k+1}}\alpha^{\lambda_k} = \alpha^{\lambda_{k+1}+\lambda_k+1}$. Similarly, positive elements of $\mathbf{F}^k$ and $\mathbf{H}^k$ are at least $\alpha^{\lambda_{k+1}+\lambda_k+\lambda_{k\;1}+1}$. Hence we can define the following lower bound for all positive elements of $\mathbf{P}^k$:

$$\beta_k = \alpha^{\lambda_{k+1}+\lambda_k+\lambda_{k\;1}+1}. \tag{21}$$

We note the following facts by observing (18):

- $\mathbf{E}^k$ is strictly positive.
- if $\bar{v}_h^k$ is positive, the h^{th} column of $\mathbf{F}^k$ is strictly positive. Otherwise the whole $(n+h)^{th}$ column of $\mathbf{P}^k$ is zero.
- if $\bar{v}_h^{k+1}$ is positive, the h^{th} row of $\mathbf{G}^k$ has at least one positive entry. This is true because during the time μ_{kn} to $\mu_{(k+1)n} - 1$, the corresponding link (i, j), transmits successfully at least once, which sets the values of $\bar{v}_h$ and $\bar{u}_h$ to 0. Therefore since $\bar{v}_h^{k+1}$ is positive, link (i, j) has failed at least once after the last successful transmission. Hence, C_{hi}^k is positive, and therefore G_{hi}^k is also positive.

Define the index set $I^k = \{h | \bar{v}_h^k > 0\}$. If $h \notin I^k$ we have $\bar{r}_h^k = \bar{v}_h^k = 0$, and also the $(n+h)^{th}$ column of $\mathbf{P}^k$ has only zero entries; hence, $\bar{r}_h^k$ does not influence any variable of time $k+1$. We also have for $h \notin I^{k+1}$ the $(n+h)^{th}$ row of $\mathbf{P}^k$ has only zero entries. Thus, $\bar{r}_h^{k+1}$ is formed by the sum of zero numbers. Intuitively, this means that for $h \notin I^k$, $\bar{r}_h^k$ is zero and so are the coefficients related to it in (17). Therefore it gives us no meaningful information and it can be ignored. For the rest of the proof, we assume that all the variables $\bar{r}_h^k$ considered in the equations are the ones with $h \in I^k$.

We obtain:

$$\begin{aligned} \bar{r}_{\max}^{k+1} &\leq \beta^k \bar{z}_{\max}^k + (1-\beta^k)\max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\}, \\ \bar{z}_{\max}^{k+1} &\leq \beta^k \min\{\bar{z}_{\min}^k, \bar{r}_{\min}^k\} + (1-\beta^k)\max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\}. \end{aligned}$$

Then,

$$\max\{\bar{z}_{\max}^{k+1}, \bar{r}_{\max}^{k+1}\} \leq \beta^k \bar{z}_{\max}^k + (1-\beta^k)\max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\}.$$

Similarly,

$$\min\{\bar{z}_{\min}^{k+1}, \bar{r}_{\min}^{k+1}\} \geq \beta^k \bar{z}_{\min}^k + (1-\beta^k)\min\{\bar{z}_{\min}^k, \bar{r}_{\min}^k\}.$$

We also have:

$$\bar{z}_{\max}^{k+1} \leq \beta^k \sum_{i=1}^{n} \bar{z}_i^k + (1 - n\beta^k) \max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\},$$

$$\bar{z}_{\min}^{k+1} \geq \beta^k \sum_{i=1}^{n} \bar{z}_i^k + (1 - n\beta^k) \min\{\bar{z}_{\min}^k, \bar{r}_{\min}^k\}.$$

Thus,

$$\bar{z}_{\max}^{k+1} - \bar{z}_{\min}^{k+1} \leq \\ (1 - n\beta^k) \left(\max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\} - \min\{\bar{z}_{\min}^k, \bar{r}_{\min}^k\} \right).$$

Equivalently,

$$s^{k+1} \leq \beta^k t^k + (1 - \beta^k) s^k, \\ t^{k+1} \leq (1 - n\beta^k) s^k,$$

where $s^k = \max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\} - \min\{\bar{z}_{\min}^k, \bar{r}_{\min}^k\}$ and $t^k = \bar{z}_{\max}^k - \bar{z}_{\min}^k$. Observing that $0 \leq t^k \leq s^k$, we get:

$$\begin{aligned} s^{k+1} &\leq \beta^k (1 - n\beta^{k-1}) s^{k-1} + (1 - \beta^k) s^k \\ &\leq \beta^k (1 - n\beta^{k-1}) s^{k-1} + (1 - \beta^k) s^{k-1} \\ &= (1 - n\beta^k \beta^{k-1}) s^{k-1}. \end{aligned}$$

Hence $\lim_{k\to\infty} s^k = 0$ if $\prod_{k=1}^{\infty} \left(1 - n\beta^{2k}\beta^{2k-1}\right) = 0$, which, by Lemma 4, holds true if and only if $\sum_{k=1}^{\infty} \beta^{2k}\beta^{2k-1} = \infty$. Using (21), we have:

$$\begin{aligned} \sum_{k=1}^{\infty} \beta^{2k}\beta^{2k-1} &= \sum_{k=1}^{\infty} \alpha^{\lambda_{2k+1}+2\lambda_{2k}+2\lambda_{2k\ \ 1}+\lambda_{2k\ \ 2}+2} \\ &\geq \frac{1}{n^2} \sum_{k=K}^{\infty} \alpha^{-\frac{\ln(2k+1+T)}{\ln(\alpha)}} \\ &= \frac{1}{n^2} \sum_{k=K}^{\infty} \frac{1}{2k+1+T} = \infty. \end{aligned}$$

Hence $\max\{\bar{z}_{\max}^k, \bar{r}_{\max}^k\} - \min\{\bar{z}_{\min}^k, \bar{r}_{\min}^k\}$ converges to 0 as k goes to infinity. Combining this with Lemma 3 we obtain,

$$\lim_{k\to\infty} \bar{z}_i^k = \lim_{k\to\infty,\ h\in I^k} \bar{r}_h^k = L. \tag{22}$$

We have:

$$\begin{aligned} L &= L \lim_{k\to\infty} \frac{\sum_{i=1}^{n} \bar{y}_i^k + \sum_{h=1}^{m} \bar{v}_h^k}{\sum_{i=1}^{n} \bar{y}_i^k + \sum_{h=1}^{m} \bar{v}_h^k} \\ &= \lim_{k\to\infty} \left(\frac{\sum_{i=1}^{n} \bar{z}_i^k \bar{y}_i^k + \sum_{h=1}^{m} \bar{r}_h^k \bar{v}_h^k}{n} \right) \\ &+ \lim_{k\to\infty} \left(\frac{\sum_{i=1}^{n} (L - \bar{z}_i^k) \bar{y}_i^k + \sum_{h=1}^{m} (L - \bar{r}_h^k) \bar{v}_h^k}{n} \right) \\ &= \lim_{k\to\infty} \left(\frac{\sum_{i=1}^{n} \bar{x}_i^k + \sum_{h=1}^{m} \bar{u}_h^k}{n} \right) \\ &+ \lim_{k\to\infty} \left(\frac{\sum_{i=1}^{n} (L - \bar{z}_i^k) \bar{y}_i^k + \sum_{h=1}^{m} (L - \bar{r}_h^k) \bar{v}_h^k}{n} \right) \\ &= \frac{\sum_{i=1}^{n} x_i^0}{n}, \end{aligned}$$

where in the last equality, we used (22), and the fact that $\bar{v}_h^k = 0$ for $h \notin I^k$. □

VI. Conclusion

In this paper we established sufficient conditions on connectivity and link failures for consensus algorithms to converge. We started by showing that ordinary consensus and push-sum still work if intercommunication intervals do not grow too fast. Then we moved on to our main result, which is a fully asynchronous push-sum algorithm robust to link failures. We proved its convergence while allowing consecutive link failures to grow to infinity, as long as they remain smaller than a logarithmically growing upper bound.

This work can be extended by improving the upper bounds using ergodicity theory. It is also possible to use our results to develop asynchronous distributed optimization algorithms robust to packet losses.

References

[1] J. Lorenz, "Convergence to consensus in multiagent systems and the lengths of inter-communication intervals," *arXiv preprint arXiv:1101.2926*, 2011.

[2] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*. IEEE, 2005, pp. 2996–3000.

[3] T. Charalambous, Y. Yuan, T. Yang, W. Pan, C. N. Hadjicostis, and M. Johansson, "Distributed finite-time average consensus in digraphs in the presence of time delays," *IEEE Transactions on Control of Network Systems*, vol. 2, no. 4, pp. 370–381, 2015.

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

[5] 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, 2014.

[6] ——, "Average consensus in the presence of delays and dynamically changing directed graph topologies," *arXiv preprint arXiv:1210.4778*, 2012.

[7] P. Rezaeinia, B. Gharesifard, T. Linder, and B. Touri, "Push-sum on random graphs," *arXiv preprint arXiv:1708.00915*, 2017.

[8] F. Bénézit, V. Blondel, P. Thiran, J. Tsitsiklis, and M. Vetterli, "Weighted gossip: Distributed averaging using non-doubly stochastic matrices," in *Information theory proceedings (isit), 2010 ieee international symposium on*. IEEE, 2010, pp. 1753–1757.

[9] B. Gerencsér and J. M. Hendrickx, "Push sum with transmission failures," *CoRR*, vol. abs/1504.08193, 2015. [Online]. Available: http://arxiv.org/abs/1504.08193

[10] C. N. Hadjicostis, N. H. Vaidya, and A. D. Dominguez-Garcia, "Robust distributed average consensus via exchange of running sums," *IEEE Transactions on Automatic Control*, pp. 1492–1507, Jun. 2016.

[11] N. Bof, R. Carli, and L. Schenato, "Average consensus with asynchronous updates and unreliable communication," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 601 – 606, 2017, 20th IFAC World Congress. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S240589631730126X

[12] K. I. Tsianos, S. Lawlor, and M. G. Rabbat, "Push-sum distributed dual averaging for convex optimization," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*. IEEE, 2012, pp. 5453–5458.

[13] D. Varagnolo, F. Zanella, A. Cenedese, G. Pillonetto, and L. Schenato, "Newton-raphson consensus for distributed convex optimization," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, pp. 994–1009, 2016.

[14] N. Bof, R. Carli, G. Notarstefano, L. Schenato, and D. Varagnolo, "Newton-raphson consensus under asynchronous and lossy communications for peer-to-peer networks," *arXiv preprint arXiv:1707.09178*, 2017.

[15] A. Olshevsky, I. C. Paschalidis, and A. Spiridonoff, "Fully asynchronous push-sum with growing intercommunication intervals," *arXiv preprint arXiv:1802.08634*, 2018.

[16] E. Seneta, *Non-negative matrices and Markov chains*. Springer Science & Business Media, 2006.

[17] P. Brémaud, *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*. Springer Science & Business Media, 2013, vol. 31.

[18] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*. Prentice hall Englewood Cliffs, NJ, 1989, vol. 23.