Discrete-time dynamic average consensus [⋆]

Minghui Zhu [a,*], Sonia Martínez [a]

[a] *Department of Mechanical and Aerospace Engineering, University of California, San Diego, 9500 Gilman Dr, La Jolla CA, 92093-0411, USA. Tel: 1-858-8224243. Fax: 1-858-5344387*

Abstract

We propose a class of discrete-time dynamic average consensus algorithms that allow a group of agents to track the average of their reference inputs. The convergence results rely on the input-to-output stability properties of static average consensus algorithms and require that the union of communication graphs over a bounded period of time be strongly connected. The only requirement on the set of reference inputs is that the maximum relative deviation between the n^{th}-order differences of any two reference inputs be bounded for some integer $n \geq 1$.

Key words: Consensus algorithms, multi-agent systems, cooperative control

1 Introduction

We consider the problem in which a set of autonomous agents aims to track the average of individually measured time-varying signals by local communication with neighbors. This problem is referred to as dynamic average consensus in opposition to the more studied static consensus. The dynamic average consensus problem arises in different contexts, such as formation control [30], sensor fusion [16][20][24], distributed estimation [11] and distributed tracking [29]. These tasks require that all agents agree on the average of time-varying signals and thus the consensus on a static average value, e.g., the initial states of the agents, is insufficient.

Literature review. The distributed static consensus problem was introduced in the literature of parallel processors in [26] and has attracted significant attention in the controls community. A necessarily incomplete list of references includes [6][19] for continuous-time consensus, [2][8][14] for discrete-time consensus, [1][12] discuss asynchronous consensus, and [3][9][25][15] treat randomized consensus, quantized consensus, consensus over random graphs and constrained consensus, respectively. The convergence rate of consensus algorithms is e.g., discussed in [21][27], consensus propagation is considered in [13], and conditions on consensus algorithms to achieve different consensus values is discussed in [4]. Consensus algorithms find application in a variety of areas such as load balancing [5][28], formation control [6][30], and, as we have mentioned, sensor fusion [11][16][20][24], distributed tracking [17][29] and consensus-based belief propagation in Bayesian networks [18].

The dynamic average consensus problem in continuous-time is studied in [7][20][22][24]. By using standard frequency-domain techniques, the authors in [24] showed that their algorithm was able to track the average of ramp reference inputs with zero steady-state error. In the context of input-to-state stability, [7] showed that proportional dynamic average consensus algorithm could track with bounded steady-state error the average of bounded reference inputs with bounded derivatives. On the other hand, [7] showed that proportional-integral dynamic average consensus algorithm could track the average of constant reference inputs with sufficiently small steady-state error. The authors in [20] proposed a dynamic consensus algorithm and applied it to the design of consensus filters. The algorithm in [20] can track with some bounded steady-state error the average of a common reference input with a bounded derivative. The problem studied in [22] is similar to that in [20], and consensus of agents is over a common time-varying reference signal. However, the algorithm in [22] assumes that agents know the nonlinear model which generates the time-varying reference function. The problem studied in the present paper is close to those in [7] and [24] and includes those in [20] and [22] as special cases.

[⋆] Research supported in part by NSF Career Award CMS-0643673 and NSF IIS-0712746. This paper was not presented at any IFAC meeting.

[*] Corresponding author.

Email addresses: mizhu@ucsd.edu (Minghui Zhu), soniamd@ucsd.edu (Sonia Martínez).

Statement of contributions. In this paper, we propose a class of discrete-time dynamic average consensus algorithms and analyze their convergence properties. This paper contributes to the problem of dynamic average consensus in the following aspects: The continuous-time communication assumption for dynamic average consensus in [7] and [24] is relaxed, and we consider more realistic discrete-time synchronous communication models. This allows us to obtain a direct relation between the frequency of inter-agent communication and the differences of reference signals. Our dynamic average consensus algorithms are able to track the average of a larger class of time-varying reference inputs than [7] and [24] with zero or sufficiently small steady-state error. This includes polynomials, logarithmic-type functions, periodic functions and other functions whose n^{th}-order differences are bounded, for $n \geq 1$. We can also handle the case where the difference of the common part, that appears in all the individual reference inputs, explodes. Furthermore, the algorithms proposed are robust to the dynamic change of communication topologies as well as the joining and leaving (or failure) of nodes. The convergence analysis for our dynamic average consensus algorithms relies upon the input-to-output stability property of discrete-time static average consensus algorithms in the presence of external disturbances.

Organization of the paper. We now outline the remainder of the paper. In Section 2, we introduce general notation and the statement of the problem we study. In Section 3, we focus on a first-order algorithm for dynamic average consensus. Section 4 generalizes this to a class of n^{th}- order algorithms for dynamic average consensus and analyze their convergence properties. In Section 5, we present some remarks on the extension of the results in Section 3 and 4. In Section 6, we provide some examples to illustrate the effectiveness of our algorithms. Section 7 includes some concluding remarks. The proofs for our main results can be given in Section 8. The enlarged version of this paper can be found on [10].

2 Preliminaries and problem statement

In this section, we introduce the notation to be employed along the paper and state the problem of dynamic average consensus.

The positive real number h is the time discretization unit and the update time instants $t \in \mathbb{R}$ (or s, τ) will be of the form $t = ph$ (or $s = ph$, $\tau = qh$) for $p, q \in \mathbb{Z}$.

We will consider a network of N nodes or agents, labeled by $i \in V = \{1, \ldots, N\}$, interacting over a communication network. The topology of the network at time t will be represented by a directed graph $\mathcal{G}(t) = (V,\ E(t))$ with an edge set $E(t) \subset V \times V$. We consider that $(i, j) \in E(t)$ if and only if node i communicates to node j at time t. The in-neighbors of node i at time t are denoted by $\mathcal{N}_i(t) = \{j \in V : (j, i) \in E(t) \text{ and } j \neq i\}$. The matrix $A(t) = [a_{ij}(t)] \in \mathbb{R}^{N\times N}$ represents the adjacency matrix of $\mathcal{G}(t)$ where $a_{ij}(t) \neq 0$ if edge $(j, i) \in E(t)$. Finally, $\mathbf{1} \in \mathbb{R}^N$ is the column vector whose entries are all ones.

At each time instant t, every node synchronously measures the local continuous physical process $r_i : \mathbb{R} \to \mathbb{R}$, communicates with its neighbors and updates the state of its consensus algorithm. We ignore the delays induced by the communication and computation process. In the remainder of this paper, the sample $r_i(t)$ is referred to as the reference signal (or input) of node i at time t. Denote by $\bar{r}(t) = \frac{1}{N}\sum_{i=1}^{N} r_i(t)$ the average of the reference inputs of the network at time t.

Our objective is to design an n^{th}-order dynamic average consensus algorithm that the nodes can utilize to asymptotically achieve the average of the reference inputs if the maximum relative deviation between the n^{th}-order differences of any two reference inputs is bounded for some integer $n \geq 1$. We denote by $x_i(t) = (x_i^{[1]}(t), \cdots, x_i^{[n]}(t)) \in \mathbb{R}^n$ the consensus state of node i at time t. The quantity of $\max_{i\in V} \limsup_{t\to\infty} |x_i^{[n]}(t) - \bar{r}(t - h)|$ is referred to as the steady-state error of n^{th}-order dynamic average consensus algorithm. This can be interpreted as a measurement of how far the components of the consensus state $(x_1^{[n]}(t), \cdots, x_N^{[n]}(t))$ are from achieving the dynamic average consensus. Our algorithms will reach the dynamic average consensus with either a zero steady-state error or rendering the steady-state error smaller than (or equal to) any given bound.

3 First-order dynamic average consensus algorithm

In this section, we present first-order algorithms to achieve dynamic average consensus. Our presentation mainly follows [1][2], and [21]. We define:

$$\begin{aligned} M(t) &= \max_{i\in V} x_i(t), & m(t) &= \min_{i\in V} x_i(t),\\ D(t) &= M(t) - m(t), & \Delta r_i(t) &= r_i(t) - r_i(t - h),\\ \Delta r_{\max}(t) &= \max_{i\in V} \Delta r_i(t), & \Delta r_{\min}(t) &= \min_{i\in V} \Delta r_i(t). \end{aligned}$$

By induction, the n^{th}-order difference of $r_i(t)$ is $\Delta^{(n)} r_i(t) = \Delta^{(n-1)} r_i(t) - \Delta^{(n-1)} r_i(t - h)$ for $n \geq 2$ where $\Delta^{(1)} r_i(t) = \Delta r_i(t)$. We will use the notations $\Delta^{(n)} r_{\max}(t) = \max_{i\in V} \Delta^{(n)} r_i(t)$ and $\Delta^{(n)} r_{\min}(t) = \min_{i\in V} \Delta^{(n)} r_i(t)$ for $n \geq 2$.

In what follows, we will make use of the following assumption on $\mathcal{G}(t)$ that was proposed in [8] and also used in [2][21].

Assumption 3.1 (Periodical strong connectivity) *There is some positive integer $B \geq 1$ such that, for all time instant $t \geq 0$, the directed graph $(V, E(t) \cup E(t+h) \cup \cdots \cup E(t+(B-1)h))$ is strongly connected.*

Assumption 3.2 (Relatively bounded first-order differences) *For any $h > 0$, there exists a time invariant constant $\theta > 0$ such that*

$$\Delta R(t) := \Delta r_{\max}(t) - \Delta r_{\min}(t) \leq h\theta, \quad \forall t \geq 0. \quad (1)$$

Remark 3.1 Inequality (1) becomes $\max_{i \in V} \dot{r}_i(t) - \min_{i \in V} \dot{r}_i(t) \leq \theta$ as $h \to 0$. Hence, Assumption 3.2 can be viewed as the discretized version of the property $\max_{i \in V} \dot{r}_i(t) - \min_{i \in V} \dot{r}_i(t) \leq \theta$ for some fixed $\theta \geq 0$ and all time instant $t \geq 0$. •

We propose the *First-Order Dynamic Average Consensus algorithm* (the FODAC algorithm for short) below to reach dynamic average consensus:

$$x_i(t+h) = x_i(t) + \sum_{j \neq i} a_{ij}(t)(x_j(t) - x_i(t)) + \Delta r_i(t), \quad (2)$$

when the reference input $r(t)$ satisfies Assumption 3.2.

Remark 3.2 The FODAC algorithm can be rewritten as:

$$\begin{aligned}[x_i(t+h) - x_i(t)]/h &= \delta \sum_{j \neq i} a_{ij}(t)(x_j(t) - x_i(t)) \\ &\quad + [r_i(t) - r_i(t-h)]/h\,, \qquad (3)\end{aligned}$$

where the parameters δ and h satisfy $h\delta = 1$. Observe that (3) is close to the discretized version of the continuous-time dynamic consensus algorithm in [24] but is not exactly the same. If $h \to 0$, then $\delta = \frac{1}{h} \to \infty$, and thus the right-hand side of (3) is not well-defined.•

We will further suppose that the coefficients $a_{ij}(t)$ in the FODAC algorithm satisfy the following two assumptions.

Assumption 3.3 ([2], Non-degeneracy) *There exists a constant $\alpha > 0$ such that $a_{ii}(t) = 1 - \sum_{j \neq i} a_{ij}(t) \geq \alpha$, and $a_{ij}(t)$ $(i \neq j)$ satisfies $a_{ij}(t) \in \{0\} \cup [\alpha,\ 1]$, $\forall t \geq 0$.*

Assumption 3.4 ([19], Balanced Communication) *There hold that $\mathbf{1}^T A(t) = \mathbf{1}^T$ and $A(t)\mathbf{1} = \mathbf{1}$, $\quad \forall t \geq 0$.*

Equivalently, the matrix $A(t)$ is referred to as doubly stochastic, each of whose rows and columns sums to 1. Assumption 3.4 renders the conservation property $\sum_{i=1}^N x_i(t+h) = \sum_{i=1}^N r_i(t)$ which is essential to reach the dynamic average consensus. It plays a similar role in achieving the static average consensus [19].

We now proceed to analyze the FODAC algorithm. Let us fix $k \in V$ for every s and define $\mathcal{D}_0 = \{k\}$. By Assumption 3.1, there is a non-empty set $\mathcal{D}_1 \subset V \setminus \{k\}$ of nodes such that for all $i \in \mathcal{D}_1$, node k communicates to node i at least once during the time frame $[s, s+(B-1)h]$. By induction, a set $\mathcal{D}_{\ell+1} \subset V \setminus (\mathcal{D}_0 \cup \cdots \cup \mathcal{D}_\ell)$ can be defined by considering those nodes j to which some $i \in \mathcal{D}_0 \cup \cdots \cup \mathcal{D}_\ell$ communicates at least once during the time frame $[s + \ell Bh, s + ((\ell+1)B - 1)h]$. By Assumption 3.1, $\mathcal{D}_{\ell+1} \neq \emptyset$ as long as $V \setminus (\mathcal{D}_0 \cup \cdots \cup \mathcal{D}_\ell) \neq \emptyset$. Thus, there exists $\mathcal{L} \leq N - 1$ such that the collection of $\mathcal{D}_0, \ldots, \mathcal{D}_\mathcal{L}$ is a partition of V.

Lemma 3.1 *Consider the* FODAC *algorithm and suppose that Assumptions 3.1 and 3.3 hold. Let $s \geq 0$ and $k \in V$ be fixed and consider the associated $\mathcal{D}_0, \ldots, \mathcal{D}_\mathcal{L}$. Then for every $\ell \in \{1, \ldots, \mathcal{L}\}$, there exists a real number $\eta_\ell > 0$ such that for every integer $p \in [\ell B,\ (\mathcal{L}B + B - 1)]$, and $i \in \mathcal{D}_\ell$, it holds that for $t = s + ph$*

$$x_i(t) \geq m(s) + \sum_{q=0}^{p-1} \Delta r_{\min}(s + qh) + \eta_\ell(x_k(s) - m(s)), \quad (4)$$

$$x_i(t) \leq M(s) + \sum_{q=0}^{p-1} \Delta r_{\max}(s + qh) - \eta_\ell(M(s) - x_k(s)). \quad (5)$$

The following theorem is the main result in this section and shows the convergence properties of the FODAC algorithm.

Theorem 3.1 *Let δ_1 be a positive constant and $h_1 = \frac{\delta_1 \alpha^{\frac{1}{2}N(N+1)B+1}}{4\theta(NB-2)}$. Under Assumptions 3.1, 3.3, 3.4 and 3.2, the implementation of the* FODAC *algorithm with $h \in (0, h_1]$ and initial conditions $x_i(0) = r_i(-h)$, $i \in \{1, \ldots, N\}$, achieves dynamic average consensus with a nonzero steady-state error upper bounded by δ_1.*

The following corollary states an interesting special case of Theorem 3.1 when $\lim_{t \to \infty} \Delta R(t) = 0$ for any $h > 0$.

Corollary 3.1 *Under the Assumptions 3.1, 3.3, 3.4 and $\lim_{t \to \infty} \Delta R(t) = 0$ for any $h > 0$, the implementation of the* FODAC *algorithm with any $h > 0$ and initial state $x_i(0) = r_i(-h)$, $i \in \{1, \ldots, N\}$, achieves the dynamic average consensus with a zero steady-state error.*

4 Higher-order algorithms for dynamic average consensus

In this section, we present n^{th}-order algorithms for dynamic average consensus where $n \geq 2$. First of all, let us consider the case of $n = 2$. We will assume that the reference inputs satisfy the following condition weaker than Assumption 3.2.

Assumption 4.1 (Relatively bounded second-order differences) *For any $h > 0$, there exists a time invariant constant $\theta_2 > 0$ such that*

$$\Delta^{(2)} r_{\max}(t) - \Delta^{(2)} r_{\min}(t) \leq h\theta_2, \quad t \geq 0.$$

Correspondingly, we propose the following *Second-Order Dynamic Average Consensus* algorithm (the SODAC algorithm for short)

$$\begin{aligned} & x_i^{[2]}(t+h) = x_i^{[2]}(t) + \sum_{j \neq i} a_{ij}(t)(x_j^{[2]}(t) - x_i^{[2]}(t)) \\ & + x_i^{[1]}(t+h), \\ & x_i^{[1]}(t+h) = x_i^{[1]}(t) + \sum_{j \neq i} a_{ij}(t)(x_j^{[1]}(t) - x_i^{[1]}(t)) \\ & + \Delta^{(2)} r_i(t), \end{aligned} \tag{6}$$

and its convergence properties are described in the following theorem and corollary.

Theorem 4.1 *Let δ_2 be a positive constant and $h_2 = \frac{\delta_2 \alpha^{N(N+1)B+2}}{16\theta_2 (NB-2)^2}$. Under Assumptions 3.1, 3.3, 3.4 and 4.1, the implementation of the* SODAC *algorithm with $h \in (0, h_2]$, and initial states $x_i^{[1]}(0) = \Delta r_i(-h)$, $x_i^{[2]}(0) = r_i(-h)$, $i \in \{1, \ldots, N\}$, achieves dynamic average consensus with a nonzero steady-state error upper bounded by δ_2.*

Corollary 4.1 *Under Assumptions 3.1, 3.3, 3.4 and* $\lim_{t\to\infty}(\Delta^{(2)} r_{\max}(t) - \Delta^{(2)} r_{\min}(t)) = 0$ *for any $h > 0$, the implementation of the* SODAC *algorithm with any $h > 0$ and initial states $x_i^{[1]}(0) = \Delta r_i(-h)$, $x_i^{[2]}(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$ achieves dynamic average consensus with a zero steady-state error.*

Now, let us consider the following general n^{th}*-Order Dynamic Average Consensus* algorithm (the NODAC algorithm for short).

$$\begin{aligned} & x_i^{[\ell]}(t+h) = x_i^{[\ell]}(t) + \sum_{j \neq i} a_{ij}(t)(x_j^{[\ell]}(t) - x_i^{[\ell]}(t)) \\ & + x_i^{[\ell-1]}(t+h), \\ & x_i^{[1]}(t+h) = x_i^{[1]}(t) + \sum_{j \neq i} a_{ij}(t)(x_j^{[1]}(t) - x_i^{[1]}(t)) \\ & + \Delta^{(n)} r_i(t), \qquad \ell \in \{2, \ldots, n\}. \end{aligned} \tag{7}$$

Remark 4.1 In [23], the authors propose a continuous-time higher-order consensus algorithm to allow higher-order derivatives converge to common values. While related, the problem statement of [23] is different from ours. •

The previous algorithm is the cascade of n FODAC algorithms and can be compactly rewritten in the following vector form

$$\begin{aligned} & x^{[\ell]}(t+h) = A(t)x^{[\ell]}(t) + x^{[\ell-1]}(t+h), \\ & x^{[1]}(t+h) = A(t)x^{[1]}(t) + \Delta^{(n)} r(t), \quad \ell \in \{2, \ldots, n\}. \end{aligned}$$

The above NODAC algorithm is able to track the average of reference inputs which satisfy the following condition under which Theorem 4.2 holds.

Assumption 4.2 (Relatively bounded n^{th}-order differences) *For any $h > 0$, there exists a time invariant constant $\theta_n > 0$ such that*

$$\Delta^{(n)} r_{\max}(t) - \Delta^{(n)} r_{\min}(t) \leq h\theta_n, \qquad \forall\, t \geq 0.$$

The following theorem and corollary show the convergence properties of n^{th}-order dynamic average consensus algorithm.

Theorem 4.2 *Let δ_n be a positive constant and $h_n = \frac{\delta_n \alpha^{n(\frac{1}{2}N(N+1)B+1)}}{2^{2n}\theta_n (NB-2)^n}$. Under the Assumptions 3.1, 3.3, 3.4 and 4.2, the implementation of the* NODAC *algorithm with $h \in (0, h_n]$ and initial states $x_i^{[\ell]}(0) = \Delta^{(n-\ell)} r_i(-h)$ ($\ell = 1, \cdots, n-1$), $x_i^{[n]}(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$, achieves the dynamic average consensus with a nonzero steady-state error upper bounded by δ_n.*

Proof: The case of $n = 1$ has been proven in Theorem 3.1. By using similar arguments in Theorem 4.1, we can finish the proof in an inductive way. ■

Corollary 4.2 *Under the Assumptions 3.1, 3.3, 3.4 and* $\lim_{t\to\infty}(\Delta^{(n)} r_{\max}(t) - \Delta^{(n)} r_{\min}(t)) = 0$ *with any $h > 0$, the implementation of the* NODAC *algorithm for any $h > 0$ and initial states $x_i^{[\ell]}(0) = \Delta^{(n-\ell)} r_i(-h)$ ($\ell =$*

$1, \cdots, n-1$), $x_i^{[n]}(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$, achieves dynamic average consensus with a zero steady-state error.

5 Extensions

This section includes some remarks about the possible extension of the presented results.

5.1 Discussion on the choice of the order for the dynamic average consensus algorithm

If Assumption 4.2 holds, m^{th}-order dynamic consensus algorithm can reach the dynamic average consensus for any $m > n$. However, we need a smaller h than the NODAC algorithm to render the steady-state error smaller than the given bound. Then there is no advantage to use m^{th}-order average dynamic consensus algorithm when Assumption 4.2 is satisfied.

5.2 Discussion on Assumption 4.2

It can be shown that for any n^{th}-order polynomial $f(t) = \sum_{i=0}^{n} a_i t^i$, there holds that $\Delta^{(n)} f(t) = a_n n! h$. Hence, any set of n^{th}-order polynomials satisfies Assumption 4.2 with $\theta_{n+1} = 0$.

If the reference inputs $r_i(t)$ take the form of $r_i(t) = v(t) + \tilde{r}_i(t)$, $i \in V$, and the function $\tilde{r}_i(t)$ is a linear combination of polynomials, the logarithmic function, periodic functions and other functions whose n^{th}-order differences are bounded, then Assumption 4.2 also holds for any common $v(t)$ even when n^{th}-order difference of $v(t)$ explodes, e.g., like the exponential function. It is worth mentioning that it is unnecessary for Assumption 4.2 to hold that $\Delta^{(n)} r_i(t)$ be bounded for all i, $t \geq 0$.

5.3 Discussion on Assumption 3.1

In the case that the communication is symmetric; i.e., when $(i, j) \in E(t)$ if and only if $(j, i) \in E(t)$, then Assumption 3.1 in Corollary 4.2 can be weakened into:

Assumption 5.1 (Eventual strong connectivity) *The directed graph $(V, \cup_{s \geq t} E(s))$ is strongly connected for all time instant $t \geq 0$.*

Corollary 5.1 *Suppose $\mathcal{G}(t)$ is undirected. Under the Assumptions 5.1, 3.3, 3.4 and $\lim_{t\to\infty}(\Delta^{(n)} r_{\max}(t) - \Delta^{(n)} r_{\min}(t)) = 0$ with any $h > 0$, the implementation of the* NODAC *algorithm with any $h > 0$ and initial states $x_i^{[\ell]}(0) = \Delta^{(n-\ell)} r_i(-h)$ ($\ell = 1, \cdots, n-1$), $x_i^{[n]}(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$, achieves dynamic average consensus with a zero steady-state error.*

If the communication is symmetric, Assumption 3.1 in Corollary 4.2 can also be replaced with the assumption in Proposition 2 of [14]; i.e., for any time instant $t \geq 0$, there is a node connected to all other nodes in the undirected graph $(V, \cup_{s \geq t} E(s))$. It is interesting to further think about the weaker assumption in Proposition 1 of [14]; i.e., there exists an integer $B \geq 1$ such that for any time instant $t \geq 0$, there is a node connected to all other nodes in the directed graph $(V, E(t) \cup E(t+h) \cup \cdots \cup E(t + (B-1)h))$.

5.4 The robustness to joining and leaving of nodes

If some nodes join the network at some time $t_0 > 0$ during the implementation of the NODAC algorithm, all the nodes in the new network are able to reach the new dynamic average consensus as long as the joining nodes choose their "initial" states at time t_0 according to the rules in Theorem 4.2 and Assumptions 3.1, 3.3, 3.4 and 4.2 are satisfied for the new network.

To make the NODAC algorithm adaptive to the departure of some nodes, we slightly modify its implementation. Assume node k wants to leave the network at some time t_0 and $(k, i) \in E(t_0)$ for some node i. Node k sends the value of $x_k^{[n]}(t_0) - r_k(t_0)$ to node i, and then node i updates its values according to the NODAC algorithm by replacing the top equation in the NODAC algorithm with the following

$$x_i^{[n]}(t_0 + h) = x_i^{[n]}(t_0) + \sum_{j \neq i} a_{ij}(t_0)(x_j^{[n]}(t_0) - x_i^{[n]}(t_0)) + x_i^{[n-1]}(t_0 + h) + (x_k^{[n]}(t_0) - r_k(t_0)) . \qquad (8)$$

All other remaining nodes update their values according to the NODAC algorithm at time t_0. After time t_0, the remaining nodes in the network update their values according to the NODAC algorithm, and then the dynamic average consensus is reached if Assumptions 3.1, 3.3, 3.4 and 4.2 are satisfied for the new network. The update law (8) ensures the conservation property of $\sum_{j \neq k} x_j^{[n]}(t_0 + h) = \sum_{j \neq k} r_j(t_0)$.

5.5 Asynchronous first-order dynamic average consensus algorithm

In this part, the asynchronism is incorporated into the FODAC algorithm. First, let us define the following notations: a set T_i of time instants when node i measures $r_i(t)$; a variable $\tau_i(t)$ which denotes the latest time instant before time t in T_i. We adopt the partial asynchronism time model adapted from [1]; i.e., there exists a positive integer B_a such that $t - B_a h \leq \tau_i(t) \leq t$ for each $i \in V$ and each $t \geq 0$.

Asynchronous First-Order Dynamic Average Consensus algorithm (asynchronous FODAC algorithm for short) is given by: if $t \in T_i$, node i measures $r_i(t)$ and updates its value according to the following rule:

$$x_i(t+h) = x_i(t) + \sum_{j \neq i} a_{ij}(t)(x_j(t) - x_i(t)) \\ + (r_i(t) - r_i(\tau_i(t))); \qquad (9)$$

otherwise, node i sets $x_i(t+h) = x_i(t)$.

Assumption 5.2 (Bounded first-order differences) *For any $h > 0$, there exist time invariant constants $\rho_1 > 0$ and $\rho_2 < 0$ such that $\Delta r_{\max}(t) \leq h\rho_1$ and $\Delta r_{\min}(t) \geq h\rho_2$ hold for all $t \geq 0$.*

Theorem 5.1 *Let $\tilde{\delta}_1$ be a positive constant and $\tilde{h}_1 = \frac{\tilde{\delta}_1}{4B_a(\rho_1-\rho_2)(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}+\max\{\rho_1,|\rho_2|\}(B_a-1)}$. Under Assumptions 3.1, 3.3, 3.4 and 5.2, the implementation of the asynchronous* NODAC *algorithm with the partial asynchronism time model, $h \in (0, \bar{h}_1]$ and initial conditions $x_i(0) = r_i(-h)$, $i \in \{1, \ldots, N\}$, achieves dynamic average consensus with a nonzero steady-state error upper bounded by $\tilde{\delta}_1$.*

6 Simulations

In this section, we present several examples with their simulations to demonstrate the effectiveness of our theoretical results.

6.1 Example 1

We first illustrate the conclusion of Corollary 3.1 with a simulation. Let us consider a network consisting of four nodes, labeled 1 through 4. Suppose that the graph $\mathcal{G}(t)$ satisfies Assumption 3.1 with $B = 4$. The reference inputs are given by:

$$r_1(t) = 5\sin t + \frac{10}{t+2} + 1, \\ r_2(t) = 5\sin t + \frac{10}{(t+2)^2} + 2, \\ r_3(t) = 5\sin t + \frac{10}{(t+2)^3} + 3, \\ r_4(t) = 5\sin t + 10e^{-t} + 4.$$

Figure 1 shows that the tracking errors of the nodes asymptotically converge to zero.

6.2 Example 2

Now, we provide an example to illustrate the robustness of the NODAC algorithm. Consider a network with five nodes. The graph $\mathcal{G}$ is fixed when no node joins or leaves the network. The reference inputs are given by:

$$r_1(t) = t + 1 + 5\sin t, \quad r_2(t) = t - 1 + 5\sin t, \\ r_3(t) = t + 5\sin t, \quad r_4(t) = t + 50 + 5\sin t, \\ r_5(t) = t - 50 + 5\sin t.$$

It can be readily verified that Assumption 3.2 holds with $\theta = 0$. Thus we choose the FODAC algorithm with $h = 1$. During the simulation, node 5 leaves the network at time 50 and joins the network at time 100 again. Figure 2 provides the consensus states in comparison with the average of the reference inputs.

6.3 Example 3

In this part, we present an example of the asynchronous FODAC algorithm. The network consists of five nodes. The topology is switching and satisfies Assumption 3.1 with $B = 3$. The reference inputs for the nodes are as follows

$$r_1(t) = -t - 2, \quad r_2(t) = -2t - 2, \\ r_3(t) = 5t + 0.5, \quad r_4(t) = 2t + 1.5, \\ r_5(t) = t + 2.$$

Then Assumption 5.2 holds with $\rho_1 = 5$ and $\rho_2 = -2$. For the given $\tilde{\delta}_1 = 1$, we choose $h = 0.01$ for the algorithm. The consensus states with the average of the reference inputs are shown in Figure 3. The oscillation in the evolution is induced by the changes of network topologies. And the steady-state error is upper bounded by 0.9.

7 Conclusions

We have proposed a class of discrete-time dynamic average consensus algorithms and analyze their convergence properties. Due to slow convergence rates of the algorithms, tracking is shown at the expense of frequent communication and higher throughput. Future work will explore the algorithms application for sensor fusion.

Acknowledgements

The second author wishes to thank Sara Susca and Francesco Bullo for early discussions on ISS properties of static average consensus algorithms.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods.* Athena Scientific, 1997.

[2] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis. Convergence in multiagent coordination, consensus, and flocking. In *IEEE Conf. on Decision and Control and European Control Conference*, pages 2996–3000, Seville, Spain, December 2005.

[3] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, 2006.

[4] J. Cortés. Analysis and design of distributed algorithms for χ-consensus. In *IEEE Conf. on Decision and Control*, pages 3363–3368, San Diego, USA, December 2006.

[5] G. Cybenko. Dynamic load balancing for distributed memory multiprocessors. *Journal of Parallel and Distributed Computing*, 7(2):279–301, 1989.

[6] J. A. Fax and R. M. Murray. Information flow and cooperative control of vehicle formations. *IEEE Transactions on Automatic Control*, 49(9):1465–1476, 2004.

[7] R. A. Freeman, P. Yang, and K. M. Lynch. Stability and convergence properties of dynamic average consensus estimators. In *IEEE Conf. on Decision and Control*, pages 398–403, San Diego, CA, December 2006.

[8] A. Jadbabaie, J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6):988–1001, 2003.

[9] A. Kashyap, T. Başar, and R. Srikant. Quantized consensus. *Automatica*, 43(7):1192–1203, 2007.

[10] S. Martínez. `http://faemino.ucsd.edu/~soniamartinez/papers/index.html`.

[11] S. Martínez. Distributed representation of spatial fields through an adaptive interpolation scheme. In *2007 American Control Conference*, pages 2750–2755, 2007.

[12] M. Mehyar, D. Spanos, J. Pongsajapan, S. H. Low, and R. M. Murray. Asynchronous distributed averaging on communication networks. *IEEE/ACM Transactions on Networking*, 15(3):512–520, 2007.

[13] C. C. Moallemi and B. Van Roy. Consensus propagation. *IEEE Transactions on Information Theory*, 52(11):4753–4766, 2006.

[14] L. Moreau. Stability of multiagent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, 2005.

[15] A. Nedic, A. Ozdaglar, and P.A. Parrilo. Constrained consensus and optimization in multi-agent networks. *IEEE Transactions on Automatic Control*, 2009. To appear.

[16] R. Olfati-Saber. Distributed Kalman filter with embedded consensus filters. In *IEEE Conf. on Decision and Control and European Control Conference*, Seville, Spain, 2005. Submitted.

[17] R. Olfati-Saber. Distributed tracking for mobile sensor networks with information-driven mobility. In *American Control Conference*, pages 4606–4612, New York, USA, July 2007.

[18] R. Olfati-Saber, E. Franco, E. Frazzoli, and J. S. Shamma. Belief consensus and distributed hypothesis testing in sensor networks. In P.J. Antsaklis and P. Tabuada, editors, *Network Embedded Sensing and Control. (Proceedings of NESC'05 Worskhop)*, volume 331 of *Lecture Notes in Control and Information Sciences*, pages 169–182. Springer, 2006.

[19] R. Olfati-Saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[20] R. Olfati-Saber and J. S. Shamma. Consensus filters for sensor networks and distributed sensor fusion. In *IEEE Conf. on Decision and Control and European Control Conference*, pages 6698–6703, Seville, Spain, December 2005.

[21] A. Olshevsky and J. N. Tsitsiklis. Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48(1):33–55, 2009.

[22] W. Ren. Consensus seeking in multi-vehicle systems with a time-varying reference state. In *American Control Conference*, pages 717–722, New York, USA, July 2007.

[23] W. Ren, K. Moore, and Y. Chen. High-order and model reference consensus algorithms in cooperative control of multi-vehicle systems. *ASME Journal of Dynamic Systems, Measurement, and Control*, 129(5):678–688, 2007.

[24] D. P. Spanos, R. Olfati-Saber, and R. M. Murray. Dynamic consensus on mobile networks. In *IFAC World Congress*, Prague, Czech Republic, July 2005.

[25] A. Tahbaz-Salehi and A. Jadbabaie. Consensus over random networks. *IEEE Transactions on Automatic Control*, 53(3):791–795, 2008.

[26] J. N. Tsitsiklis. *Problems in Decentralized Decision Making and Computation*. PhD thesis, Massachusetts Institute of Technology,
November 1984. Technical Report LIDS-TH-1424. Available at `http://web.mit.edu/jnt/www/Papers/PhD-84-jnt.pdf`.

[27] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53:65–78, 2004.

[28] C. Xu and F. Lau. *Load balancing in parallel computers: theory and practice*. Kluwer Academic Publishers, 1997.

[29] P. Yang, R. A. Freeman, and K. M. Lynch. Distributed cooperative active sensing using consensus filters. In *IEEE International Conference on Robotics and Automation*, pages 405–410, Roma, Italy, April 2007.

[30] P. Yang, R. A. Freeman, and K. M. Lynch. Multi-agent coordination by decentralized estimation and control. *IEEE Transactions on Automatic Control*, 53(11):2480–2496, 2008.

8 Appendix

8.1 Proof of Lemma 3.1

Proof: Without loss of generality, we only consider the case where $s = 0$, being the proof for a general s identical. Fix some i, it holds that

$$\begin{aligned}
&x_i(h) \\
&= x_i(0) + \sum_{j\neq i} a_{ij}(0)(x_j(0) - x_i(0)) + \Delta r_i(0) \\
&= (1 - \sum_{j\neq i} a_{ij}(0))x_i(0) + \sum_{j\neq i} a_{ij}(0)x_j(0) + \Delta r_i(0) \\
&\geq (1 - \sum_{j\neq i} a_{ij}(0))m(0) + \sum_{j\neq i} a_{ij}(s)m(0) + \Delta r_{\min}(0) \\
&= m(0) + \Delta r_{\min}(0). \qquad (10)
\end{aligned}$$

Since (10) holds for all i, we have

$$m(h) \geq m(0) + \Delta r_{\min}(0). \qquad (11)$$

Repeatedly applying (11) gives that

$$m(t) \geq m(0) + \sum_{p=0}^{\frac{t}{h}-1} \Delta r_{\min}(ph). \tag{12}$$

Since $\sum_{j=1}^{N} a_{kj}(t) = 1$ at every $t \geq 0$, we have that

$$\begin{aligned}
&x_k(t+h) - m(0) - \sum_{p=0}^{\frac{t}{h}} \Delta r_{\min}(ph) \\
&= \sum_{j=1}^{N} a_{kj}(t)(x_j(t) - m(0) - \sum_{p=0}^{\frac{t}{h}-1} \Delta r_{\min}(ph)) \\
&+ \Delta r_k(t) - \Delta r_{\min}(t) \\
&\geq a_{kk}(t)(x_k(t) - m(0) - \sum_{p=0}^{\frac{t}{h}-1} \Delta r_{\min}(ph)) \\
&\geq \alpha(x_k(t) - m(0) - \sum_{p=0}^{\frac{t}{h}-1} \Delta r_{\min}(ph)), \qquad (13)
\end{aligned}$$

where we are using the property of (12) in the last two inequalities. Applying repeatedly (13) we have that, for any integer $p \in [0, (\mathcal{L}B + B - 1)]$, the following holds for $t = ph$

$$\begin{aligned}
&x_k(t) - m(0) - \sum_{q=0}^{p-1} \Delta r_{\min}(qh) \\
&\geq \alpha^{p-1}(x_k(h) - m(0) - \Delta r_{\min}(0)) \\
&\geq \alpha^{p}(x_k(0) - m(0)) \geq \eta_0(x_k(0) - m(0))\,,
\end{aligned}$$

where $\eta_0 = \alpha^{NB-1}$ and we are using the properties of (12) and $x_k(0) - m(0) \geq 0$. This proves inequality (4) for the nodes in $\mathcal{D}_0 = \{k\}$ and for any integer $p \in [0, (\mathcal{L}B + B - 1)]$.

Now we proceed by induction on ℓ. Suppose that (4) holds for some $0 \leq \ell < \mathcal{L}$; then we should show (4) for $i \in \mathcal{D}_{\ell+1}$. It follows from the construction of the sets of $\{\mathcal{D}_0, \cdots, \mathcal{D}_{\mathcal{L}}\}$ that there exists some time $t' \in [\ell Bh,\ (\ell B + B - 1)h]$ such that $a_{ij}(t') \neq 0$ for some $j \in \mathcal{D}_0 \cup \cdots \cup \mathcal{D}_\ell$ and $i \in \mathcal{D}_{\ell+1}$. By the induction hypothesis, we have that for all integers $p \in [\ell B,\ (\mathcal{L}B + B - 1)]$, there exists some $\eta_\ell > 0$ such that the following holds for $t = ph$

$$x_j(t) - m(0) - \sum_{\tau=0}^{p-1} \Delta r_{\min}(qh) \geq \eta_\ell(x_k(0) - m(0)).$$

Consequently, as in (13), we have

$$\begin{aligned}
&x_i(t'+h) - m(0) - \sum_{q=0}^{\frac{t'}{h}} \Delta r_{\min}(qh) \\
&\geq a_{ij}(t')(x_j(t') - m(0) - \sum_{q=0}^{\frac{t'}{h}-1} \Delta r_{\min}(qh)) \\
&\geq \alpha\eta_\ell(x_k(0) - m(0)).
\end{aligned}$$

Following along the same lines as in (13), we have that

$$x_i(t+h) - m(0) - \sum_{q=0}^{p} \Delta r_{\min}(qh) \geq \eta_{\ell+1}(x_k(0) - m(0))\,,$$

holds for all $p \in [(\ell+1)B,\ (\mathcal{L}B + B - 1)]$, where $\eta_{\ell+1} = \alpha^{(N-\ell)B}\eta_\ell$ and $t = ph$. This establishes (4) for $i \in \mathcal{D}_{\ell+1}$. By induction, we have shown that (4) holds. The proof for (5) is analogous. ■

8.2 Proof of Theorem 3.1

Proof: Let $\eta = \alpha^{\frac{1}{2}N(N+1)B-1}$, then $\eta \leq \eta_\ell$ for any $\ell \in \{1, \ldots, N-1\}$. By replacing s and t in (4) with t and $t_1 = t + (\mathcal{L}B + B - 1)h$ respectively, we have that for every $t \geq 0$, there holds that

$$\begin{aligned}
m(t_1) &= \min_{\ell \in \{0, \cdots, \mathcal{L}\}} \min_{i \in \mathcal{D}_\ell} x_i(t_1) \\
&\geq m(t) + \sum_{q=\frac{t}{h}}^{\frac{t_1}{h}-1} \Delta r_{\min}(qh) + \min_{\ell} \eta_\ell(x_k(t) - m(t)) \\
&\geq m(t) + \sum_{q=\frac{t}{h}}^{\frac{t_1}{h}-1} \Delta r_{\min}(qh) + \eta(x_k(t) - m(t)).
\end{aligned}$$

Similarly, we have

$$M(t_1) \leq M(t) + \sum_{q=\frac{t}{h}}^{\frac{t_1}{h}-1} \Delta r_{\max}(qh) - \eta(M(t) - x_k(t)).$$

Combining the above two inequalities gives that

$$D(t_1) \leq (1-\eta)D(t) + \sum_{q=\frac{t}{h}}^{\frac{t_1}{h}-1} \Delta R(qh). \tag{14}$$

Let us denote $T_k = k(NB-1)h$ for an integer $k \geq 1$. From (11), we know that $D(t+h) \leq D(t) + \Delta R(t)$. Thus we have

$$D(T_1) \le (1-\eta)D(t) + \sum_{q=\frac{t}{h}}^{\frac{T_1}{h}-1} \Delta R(qh). \qquad (15)$$

and thus,

$$D(T_n) \le (1-\eta)^n D(0) + \Omega(n)\,,$$

where

$$\Omega(n) = (1-\eta)^{n-1} \sum_{q=0}^{\frac{T_1}{h}-1} \Delta R(qh) + \cdots + \sum_{q=\frac{T_{n-1}}{h}}^{\frac{T_n}{h}-1} \Delta R(qh).$$

For any $t \ge 0$, let ℓ_t to be the largest integer such that $\ell_t(NB-1)h \le t$, and $\bar{\Omega}(t) := \Omega(\ell_t) + \sum_{q=\frac{T_{\ell_t}}{h}}^{\frac{t}{h}-1} \Delta R(qh)$. Thus for all $t \ge 0$ it follows that

$$\begin{aligned} D(t) &\le D(T_{\ell_t}) + \sum_{q=\frac{T_{\ell_t}}{h}}^{\frac{t}{h}-1} \Delta R(qh) \\ &\le (1-\eta)^{\ell_t} D(0) + \bar{\Omega}(t) \\ &\le (1-\eta)^{\frac{t}{(NB-1)h}-1} D(0) + \bar{\Omega}(t). \qquad (16) \end{aligned}$$

Since $\Delta R(t) \le h\theta$, $D(t)$ is input-to-output stable with ultimate bound $\Xi \le 4h\theta(NB-2)\frac{1}{\eta} \le 4h\theta(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}$, i.e., there exist $\Gamma > 0$ and $0 < \lambda < 1$ such that

$$D(t) \le \max\{\Gamma \lambda^{\frac{t}{h}}, \Xi\}\,, \qquad \forall\, t \ge 0. \qquad (17)$$

Choose as initial state $x_i(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$. By Assumption 3.4, the following conservation property of the FODAC algorithm is satisfied for all $t \ge 0$:

$$\begin{aligned} &\sum_{i=1}^{N} x_i(t+h) = \sum_{i=1}^{N} x_i(t) + \sum_{i=1}^{N} \Delta r_i(t) \\ &= \sum_{i=1}^{N} x_i(0) + \sum_{i=1}^{N} \sum_{q=0}^{\frac{t}{h}} \Delta r_i(qh) \\ &= \sum_{i=1}^{N} x_i(0) + \sum_{i=1}^{N} (r_i(t) - r_i(-h)) = \sum_{i=1}^{N} r_i(t), \qquad (18) \end{aligned}$$

where we have used the induction in Line 2 of the above expressions.

It follows from (18) that $m(t+h) \le \frac{1}{N}\sum_{i=1}^{N} r_i(t) \le M(t+h)$ and thus

$$\max_{i \in V} \limsup_{t\to\infty} |x_i(t) - \frac{1}{N}\sum_{i=1}^{N} r_i(t-h)| \le \limsup_{t\to\infty} D(t) \le \Xi.$$

Hence, for any given $\delta_1 > 0$, choosing $h \le h_1$ gives an steady-state error $\Xi \le \delta_1$. In other words, choosing a step of size h induces at least an error of order $4\theta(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}$. ■

8.3 Proof of Theorem 4.1

Proof: Note that the algorithm for $x_i^{[1]}(t)$ in the SODAC algorithm has the same form as the FODAC algorithm, and can be obtained from this by replacing $\Delta r_i(t)$ with $\Delta^{(2)} r_i(t)$. Since Assumption 4.1 holds, it follows from Theorem 3.1 that by choosing the initial state as $x_i^{[1]}(0) = \Delta r_i(-h)$ we can find $\Gamma_1 > 0$ and $0 < \lambda_1 < 1$ such that for all $t \ge 0$ and all $i \in \{1, \ldots, N\}$, there holds that

$$|x_i^{[1]}(t) - \frac{1}{N}\sum_{i=1}^{N} \Delta r_i(t-h)| \le D^{[1]}(t) \le \max\{\Gamma_1 \lambda_1^{\frac{t}{h}}, \Xi_1\}\,,$$

where $D^{[1]}(t) = \max_{i\in V} x_i^{[1]}(t) - \min_{i \in V} x_i^{[1]}(t)$ and $\Xi_1 \le 4h\theta_2(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}$.

Hence, there exists a finite $\bar{t} \ge 0$ such that $\Gamma_1 \lambda_1^t \le \Xi_1$ for all $t \ge \bar{t}$. In this way, $x_i^{[2]}(t)$ in the SODAC algorithm can be written in the following way for $t \ge \bar{t}$

$$x_i^{[2]}(t+h) = x_i^{[2]}(t) + \sum_{j \ne i} a_{ij}(t)(x_j^{[2]}(t) - x_i^{[2]}(t)) + d_i(t), \qquad (19)$$

with a reference input $d_i(t) = \frac{1}{N}\sum_{i=1}^{N} \Delta r_i(t) + \vartheta_i(t)$ and $|\vartheta_i(t)| \le \Xi_1$. Note that for all $t \ge \bar{t}$, there holds that

$$\begin{aligned} &\max_{i\in V} d_i(t) - \min_{i \in V} d_i(t) \le 2\Xi_1 \\ &\le 4h\theta_2(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}. \end{aligned}$$

Hence, (19) has the same form as the FODAC algorithm, and can be obtained from it by replacing $\Delta r_i(t)$ with $d_i(t)$ where $\theta = 4\theta_2(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}$ in Assumption 3.2.

Furthermore, consider as initial states $x_i^{[2]}(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$. Similarly to (18) with $\Delta r_i(t)$ instead of $r_i(t)$, we can obtain the following conservation property of the SODAC algorithm for every $t \geq 0$

$$\sum_{i=1}^{N} x_i^{[1]}(t+h) = \sum_{i=1}^{N} \Delta r_i(t), \ \sum_{i=1}^{N} x_i^{[2]}(t+h) = \sum_{i=1}^{N} r_i(t).$$

By using similar arguments to those employed in Theorem 3.1, we have that there exist $\Gamma_2 > 0$ and $0 < \lambda_2 < 1$ such that for all $t \geq \bar{t}$ and all $i \in V$, there holds

$$|x_i^{[2]}(t) - \frac{1}{N}\sum_{i=1}^{N} r_i(t-h)| \leq D^{[2]}(t) \leq \max\{\Gamma_2 \lambda_2^{\frac{t-\bar{t}}{h}}, \Xi_2\}\,,$$

where $D^{[2]}(t) = \max_{i\in V} x_i^{[2]}(t) - \min_{i \in V} x_i^{[2]}(t)$ and $\Xi_2 = 4h\theta(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1} = 16h\theta_2(NB-2)^2\alpha^{-N(N+1)B+2}$. For any given $\delta_2 > 0$, choosing $h \leq h_2$ leads to $\Xi_2 \leq \delta$. ■

8.4 *The sketch of the proof for Theorem 5.1*

Following the same lines in Lemma 3.1 and Theorem 3.1, we utilize Assumption 5.2 to have that there exist $\Gamma > 0$ and $\lambda > 0$ such that

$$D(t) \leq \max\{\Gamma\lambda^{\frac{t}{h}}, \Xi\}\,, \qquad \forall\, t \geq 0 \qquad (20)$$

where $\Xi \leq 4B_a h(\rho_1 - \rho_2)(NT-2)\frac{1}{\eta} \leq 4B_a h(\rho_1 - \rho_2)(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}$.

Since $x_i(0) = r_i(-h)$ for all $i \in \{1, \ldots, N\}$, the conservation property of $\sum_{i=1}^{N} x_i(t) = \sum_{i=1}^{N} r_i(\tau_i(t))$ holds.

From Assumption 5.2 and the property of $t - \tau_i(t) < B_a h$, it can be shown that

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^{N} r_i(t-h) - (B_a - 1)h\rho_1 &\leq \frac{1}{N}\sum_{i=1}^{N} x_i(t)\\ &\leq \frac{1}{N}\sum_{i=1}^{N} r_i(t-h) - (B_a-1)h\rho_2. \qquad (21)\end{aligned}$$

Combining (20) and (21) gives the following estimate for the steady-state error

$$\begin{aligned}&\max_{i\in V}\limsup_{t\to\infty} |x_i(t) - \frac{1}{N}\sum_{i=1}^{N} r_i(t-h)|\\ &\leq 4B_a h(\rho_1-\rho_2)(NB-2)\alpha^{-\frac{1}{2}N(N+1)B+1}\\ &+ \max\{\rho_1, |\rho_2|\}(B_a-1)h.\end{aligned}$$

■

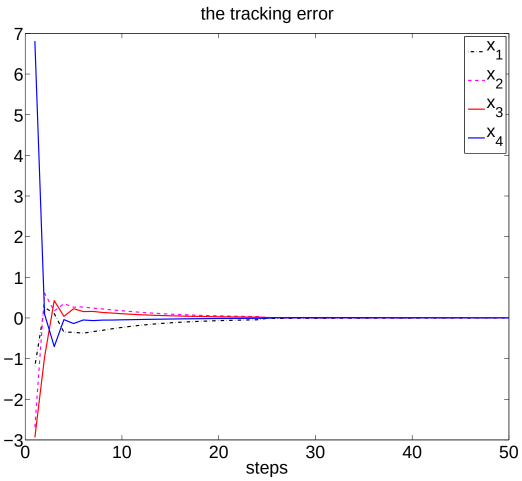

Fig. 1. The tracking errors of the FODAC algorithm

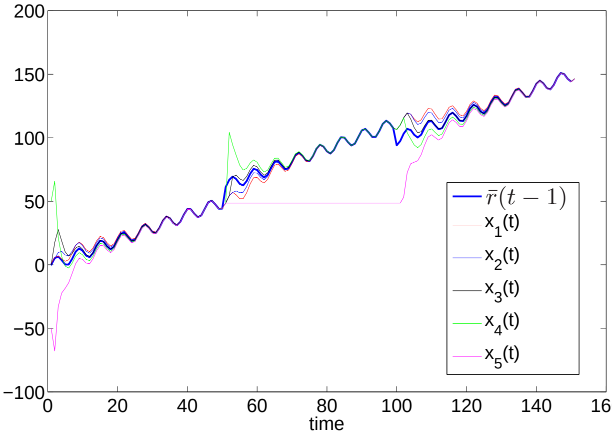

Fig. 2. Evolution of the states of the FODAC algorithm in comparison with the average of the inputs with the joining and leaving nodes

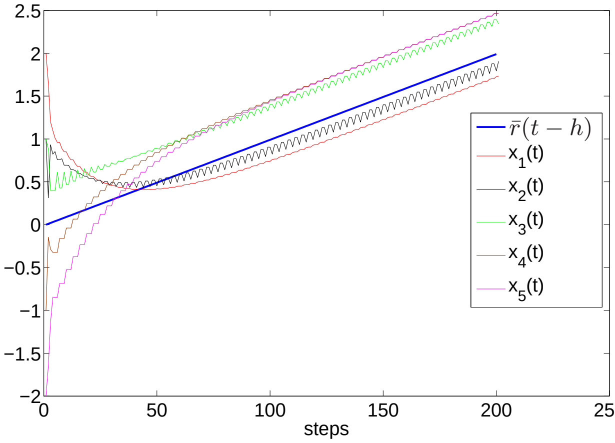

Fig. 3. Evolution of the states of the asynchronous first-order dynamic average consensus algorithm in comparison with the average of the inputs