Multi-Agent Trajectory Tracking with Self-Triggered Cloud Access

Antonio Adaldo, Davide Liuzza, Dimos V. Dimarogonas, *Member, IEEE* and Karl H. Johansson, *Fellow, IEEE*

Abstract— **This paper presents a cloud-supported control algorithm for coordinated trajectory tracking of networked autonomous agents. The motivating application is the coordinated control of Autonomous Underwater Vehicles. The control objective is to have the vehicles track a reference trajectory while keeping an assigned formation. Rather than relying on inter-agent communication, which is interdicted underwater, coordination is achieved by letting the agents intermittently access a shared information repository hosted on a cloud. An event-based law is proposed to schedule the accesses of each agent to the cloud. We show that, with the proposed scheduling of the cloud accesses, the agents achieve the required coordination objective. Numerical simulations corroborate the theoretical results.**

I. INTRODUCTION

Networked vehicle systems have attracted a notable amount of research in the past few decades [1]–[3]. In most applications, employing a team of vehicle agents instead of a single platform has numerous advantages. For example, a group of agents usually provides robustness with respect to the failure of a single agent in the group. When sampling a property in a region of the space, a team of mobile agents will provide a larger number of samples and increased data redundancy. Also, certain tasks may be structurally impossible to perform with a single agent. However, the use of a fleet inevitably brings about the problem of coordinating the vehicles. Multi-agent coordination is particularly challenging in the case of Autonomous Underwater Vehicles (AUVs) because of their limited communication, sensing and localization capabilities [4], [5]. AUVs have numerous applications, including, to name just a few, oceanographic surveys, mine search, inspection of underwater structures, and measurement of chemical properties in a water body [6]. Underwater communication may be implemented by means of acoustic modems, but such modems are notoriously expensive, power-hungry and limited in both radius and bandwidth. Underwater positioning is also difficult, since good inertial sensors are very expensive, and acoustic positioning systems have a limited range. Moreover, GPS is not available underwater, and a vehicle has to surface whenever it needs to get a position fix [7]. To deal with communication constraints, event- and self-triggered control designs [8] can be applied to networked multi-agent systems [9]. In this paper, self-triggered multi-agent control is considered in combination with the support of a shared information repository hosted on a cloud. Namely, the cloud is intermittently accessed by the agents according to a self-triggered protocol. Since direct communication among the agents is interdicted when they are underwater, the agents only exchange data through the cloud repository, which is accessed asynchronously. The motivating application is a leader-following trajectory tracking task for a formation of AUVs subject to disturbances. In the traditional event- and self-triggered networked control, one-to-one communication needs to be established at least when an agent needs to update its control signal. Conversely, with cloud-based approaches [10]–[12], the vehicles exchange information without opening a communication channel between each other. Each vehicle simply leaves its information on the cloud for the others to download later. A cloud-supported control architecture for multi-agent coordination was proposed by the authors in [12], where the problem of driving a team of vehicles to a static formation is addressed. In this paper, the approach is further developed to address multi-agent leader-follower tracking problems, under more general network topologies. Two different coordination objectives, namely practical and asymptotic convergence to a given formation, are formulated mathematically, and graph-theoretical results are used to show that the proposed cloud-supported strategy achieves the desired objectives, despite only using outdated information. Edge-space analysis [13], [14] is used to address directed network topologies, thus allowing for leader-follower coordination.

A. Adaldo, D. V. Dimarogonas and K. H. Johansson are with the department of Automatic Control and ACCESS Linnaeus Center, School of Electrical Engineering, KTH Royal Institute of Technology, Osquldas väg 10, 10044, Stockholm, Sweden; emails: {adaldo,dimos,kallej}@kth.se. D. Liuzza is with University of Sannio in Benevento, Department of Engineering, Piazza Roma 21, 82100, Benevento, Italy; e-mail: davide.liuzza@unisannio.it.

This work has received funding from the European Union Horizon 2020 Research and Innovation Programme under the Grant Agreement No. 644128, AEOROWORKS, from the Swedish Foundation for Strategic Research, from the Swedish Research Council, and from the Knut och Alice Wallenberg foundation.

II. PRELIMINARIES

In this paper, $\|\cdot\|$ denotes the euclidean norm of a vector or the corresponding induced norm of a matrix. Moreover, $\{A\}_{i,k}$ denotes the entry of A in the i-th row and k-th column, while $\{A\}_i^\top$ denotes the row vector corresponding to the i-th row of A. The null vector in $\mathbb{R}^n$ is denoted as 0_n. The set of the positive integers is denoted as $\mathbb{N}$, while $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$.

A digraph is a tuple $(\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = \{1, \dots, N\}$ and $\mathcal{E} \subseteq \{(j, i) : \ i, j \in \mathcal{V}, i \neq j\}$. The elements of $\mathcal{V}$ and $\mathcal{E}$ are called respectively *vertexes* and *edges* of the graph. A *path* from vertex j to vertex i is a sequence of vertexes starting with j and ending with i such that any two consecutive vertexes in the sequence constitute an edge. A *spanning tree* is a digraph

$(\mathcal{V}, \mathcal{T})$ with $N-1$ edges such that there exists a node r with a path to any other node. The node r is called the *root* of the spanning tree. A digraph $(\mathcal{V}, \mathcal{E})$ is said to contain a spanning tree if $(\mathcal{V}, \mathcal{T})$ is a spanning tree for some subset $\mathcal{T}$ of $\mathcal{E}$. Consider a digraph $(\mathcal{V}, \mathcal{E})$, and let the edges be denoted as $\mathcal{E} = \{e_1, \dots, e_M\}$. The *incidence matrix* of the digraph is defined as $B \in \mathbb{R}^{N \times M}$ such that

$$\{B\}_{i,k} = \begin{cases} 1 & \text{if } e_k = (j, i) \text{ for some } j \in \mathcal{V}, \\ -1 & \text{if } e_k = (i, j) \text{ for some } j \in \mathcal{V}, \\ 0 & \text{otherwise.} \end{cases}$$

The in-incidence matrix is defined as $B^{\odot} \in \mathbb{R}^{N \times M}$ such that $\{B^{\odot}\}_{i,k} = \{B\}_{i,k}$ if $\{B\}_{i,k} \in \{0, 1\}$ and $\{B^{\odot}\}_{i,k} = 0$ if $\{B\}_{i,k} = -1$. For a spanning tree, the *edge Laplacian* [14] is defined as $L_e = B^\top B^{\odot}$. For a digraph $(\mathcal{V}, \mathcal{E})$ that contains a spanning tree, but is not itself a spanning tree, let $\mathcal{E} = \mathcal{T} \cup \mathcal{C}$, with $(\mathcal{V}, \mathcal{T})$ being a spanning tree. Without loss of generality, let $\mathcal{T} = \{e_1, \dots, e_{N-1}\}$ and $\mathcal{C} = \{e_N, \dots, e_M\}$. Partition the incidence and in-incidence matrices accordingly as $B = [B_{\mathcal{T}}, B_{\mathcal{C}}]$ and $B^{\odot} = [B_{\mathcal{T}}^{\odot}, B_{\mathcal{C}}^{\odot}]$ respectively. Then, $B_{\mathcal{T}}$ has a left-pseudoinverse $B_{\mathcal{T}}^{\dagger}$ [15], and the *reduced edge Laplacian* is defined as

$$L_r = B_{\mathcal{T}}^\top (B_{\mathcal{T}}^{\odot} + B_{\mathcal{C}}^{\odot} (B_{\mathcal{T}}^{\dagger} B_{\mathcal{C}})^\top). \tag{1}$$

For any spanning tree, the edge Laplacian is positive definite, while for any graph that contains a spanning tree, but is not itself a spanning tree, the reduced edge Laplacian is positive definite [14].

III. Problem Setting

Consider a multi-agent system composed of N agents indexed as $i \in \{1, \dots, N\} = \mathcal{N}$, with kinematics described by

$$\begin{cases} \dot{x}_i(t) = u_i(t) + d_i(t) & \forall t \geq 0, i \in \mathcal{N}, \\ x_i(0) = x_{i,0} & \forall i \in \mathcal{N}. \end{cases} \tag{2}$$

where $x_i(t) \in \mathbb{R}^n$ represents the state of agent i, $x_{i,0}$ is the initial state, $u_i(t)$ represents a control input, and $d_i(t)$ represents a disturbance signal. The control objective is to have all the agents follow a desired reference trajectory $r(t)$ within a given tolerance. Such control objective is referred to as *practical consensus* to the trajectory $r(t)$, and can be formalized as follows.

Definition III.1. *The multi-agent system* (2) *is said to achieve practical consensus to the reference trajectory $r(t)$ with tolerance $\epsilon > 0$ if $\limsup_{t \to \infty} \|x_i(t) - r(t)\| \leq \epsilon$ for all $i \in \mathcal{N}$.*

Remark III.1. *In terms of our motivating application, we have $n = 2$, each agent represents an* AUV*, and $x_i(t) + b_i \in \mathbb{R}^2$ represents a waypoint for vehicle i, where b_i is a constant bias vector. In this way, practical consensus of $x_1(t), \dots, x_N(t)$ corresponds to convergence of the vehicles to a formation about $r(t)$ defined by the bias vectors $b_1, \dots, b_N$. However, the analysis remains valid for any $n \in \mathbb{N}$.*

The reference trajectory can be measured only by a subset $\mathcal{P} \subseteq \mathcal{N}$ of the agents, which are referred to as the *leaders* in the multi-agent system. In this work, we assume that the agents cannot exchange any direct information with each other. This models the scenarios where, as in our AUV setup, communication among the agents is physically interdicted. In order to exchange information, the agents can only upload and download data on a shared repository hosted on a cloud. Namely, when it is connected to the cloud, an agent can deposit some information, and, at the same time, download some information that was previously uploaded by some other agents. On the other hand, when it is not connected to the cloud, an agent cannot exchange information at all. For the purposes of this work, an agent's access to the cloud can be considered an instantaneous event, while communication protocol problems, such as delays and packet losses, are left out of the scope of this work. The cloud is modelled as a shared resource with limited throughput and storage capacity, and thanks to the control algorithm that we are going to define, it is accessed only intermittently and asynchronously, and the amount of data stored therein does not grow over time. For each agent i, we define the sequence $\{t_{i,k}\}_{k \in \mathbb{N}_0}$ of the agent's accesses to the cloud. Namely, $t_{i,k}$ with $k \in \mathbb{N}$ denotes the time when agent i accesses the cloud for the k-th time, while conventionally $t_{i,0} = 0$ for all $i \in \mathcal{N}$. When an agent i accesses the cloud at time $t_{i,k}$, it also triggers a measurement of its current state, which we denote as $x_{i,k}$:

$$x_{i,k} = x(t_{i,k}). \tag{3}$$

If agent i is a leader, it also produces a measurement of the current value of the reference trajectory, which we denote as $r_{i,k}$:

$$r_{i,k} = r(t_{i,k}). \tag{4}$$

The disturbance signals and the reference trajectory satisfy the following assumption.

Assumption III.1. *The disturbance signals $d_i(t)$ in* (2) *and the reference trajectory $r(t)$ satisfy $\|d_i(t)\| \leq \delta_i(t)$ and $\|\dot{r}(t)\| \leq \delta_0(t)$, where*

$$\delta_i(t) = (\delta_{i,0} - \delta_{i,\infty}) \, \mathrm{e}^{-\lambda_\delta} + \delta_{i,\infty}, \; i \in \{0, 1, \dots, N\},$$

and $\delta_{i,0}$, $\delta_{i,\infty}$, λ_δ, are known non-negative constants for all $i \in \{0, 1, \dots, N\}$.

Our goal is to propose a control strategy such that each agent uses the information acquired from the cloud to attain practical consensus as by Definition III.1. To completely specify the control strategy, we need to define: the control signals for each agent, the information that is uploaded and downloaded by each agent when accessing the cloud, and a law for scheduling the cloud accesses.

First, we define the control signals $u_i(t)$. In the proposed control strategy, each signal $u_i(t)$ is piecewise constant, and it is updated upon agent's i cloud accesses, i.e.,

$$u_i(t) = u_{i,k} \;\; \forall t \in [t_{i,k}, t_{i,k+1}). \tag{5}$$

AGENT	TIME	POSITION	CONTROL	NEXT
1	$t_{1,l_1(t)}$	$x_{1,l_1(t)}$	$u_{1,l_1(t)}$	$t_{1,l_1(t)+1}$
2	$t_{2,l_2(t)}$	$x_{2,l_2(t)}$	$u_{2,l_2(t)}$	$t_{2,l_2(t)+1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
N	$t_{N,l_N(t)}$	$x_{N,l_N(t)}$	$u_{N,l_N(t)}$	$t_{N,l_N(t)+1}$

Table III. Schematic representation of the data stored in the cloud at a generic time instant $t \geq 0$.

Namely, the control signals are computed as follows:

$$u_{i,k} = c\left(p_i(r_{i,k} - x_{i,k}) + \sum_{j\in\mathcal{N}_i}(\hat{x}_j^{i,k} - x_{i,k})\right), \tag{6}$$

where $c > 0$ is a control gain, $p_i = 1$ if $i \in \mathcal{P}$ and $p_i = 0$ otherwise, $\mathcal{N}_i \subseteq \mathcal{N} \setminus \{i\}$, and $\hat{x}_j^{i,k}$ is an estimate of the state of agent j done by agent i at time $t_{i,k}$. Such estimate is defined later in this section.

Next, we define the information uploaded and downloaded by each agent when accessing the cloud. When agent i accesses the cloud at time $t_{i,k}$, it uploads: the current time $t_{i,k}$, the measurement of its current state $x_{i,k}$, the control signal $u_{i,k}$ that is going to be applied until the following access, and the scheduled time $t_{i,k+1}$ of the following access. When these values are uploaded, they overwrite those that were uploaded by the same agent upon the previous access. In this way, the amount of data contained in the cloud remains constant. Namely, for each agent, the cloud contains the information that was uploaded upon that agent's most recent access. Denoting as $l_i(t)$ the index of the most recent access of agent i before time t, i.e. $l_i(t) = \max\{k \in \mathbb{N}_0 : \ t_{i,k} \leq t\}$, a tabular representation of the data contained in the cloud at a generic time t is given in Table III.

Before uploading its own information, agent i downloads and stores the information corresponding to the agents $j \in \mathcal{N}_i$. Such information is used by agent i to construct the estimates $\hat{x}_j^{i,k}$ for $j \in \mathcal{N}_i$ that are used for computing the control signal (6), and also to schedule its following access to the cloud. Namely, the estimate $\hat{x}_j^{i,k}$ is computed as follows:

$$\hat{x}_j^{i,k} = x_{j,l_j(t_{i,k})} + u_{j,l_j(t_{i,k})}(t_{i,k} - t_{j,l_j(t_{i,k})}). \tag{7}$$

Note that such estimate coincides with the state that agent j would have at time $t_{i,k}$ if no disturbances were acting on it in the time interval $[t_{j,l_j(t_{i,k})}, t_{i,k})$.

Finally, let us define the rule for scheduling the agents' accesses to the cloud. Each agent schedules its own accesses recursively, according to the following rule:

$$t_{i,k+1} = \inf\{t > t_{i,k} : \Delta_{i,k}(t) \geq \zeta_{i,k}(t) \vee \sigma_{i,k}(t) \geq \varsigma_i(t)\}, \tag{8}$$

$$\Delta_{i,k}(t) = \int_{t_{i,k}}^{t} \delta_i(\tau)d\tau, \tag{9}$$

$$\zeta_{i,k}(t) = \min_{q : i\in\mathcal{N}_q}\left\{\frac{\varsigma_q(t)}{2c|\mathcal{N}_q|}\right\}, \tag{10}$$

$$\begin{aligned}\sigma_{i,k}(t) = c\Bigg(&\Bigg\|(|\mathcal{N}_i| + p_i)u_{i,k}(t - t_{i,k}) - \sum_{j\in\mathcal{N}_i} u_{j,h_j}(\min\{t, t_{j,h_j+1}\} - t_{i,k})\Bigg\| \\ &+ (|\mathcal{N}_i| + p_i)\Delta_{i,k}(t) + p_i\int_{t_{i,k}}^{t}\delta_0(\tau)d\tau \\ &+ \sum_{j\in\mathcal{N}_i}\Delta_{j,h_j}(t) + \sum_{\substack{j\in\mathcal{N}_i \\ t>t_{j,h_j+1}}}\int_{t_{j,h_j+1}}^{t}\mu_j(\tau)d\tau\Bigg),\end{aligned} \tag{11}$$

$$\varsigma_i(t) = (\varsigma_{i,0} - \varsigma_{i,\infty})\,\mathrm{e}^{-\lambda_\varsigma} + \varsigma_{i,\infty}, \tag{12}$$

where $\varsigma_{i,0}$, $\varsigma_{i,\infty}$ and λ_ς are given non-negative constants for all $i \in \mathcal{N}$, $h_j = l_j(t_{i,k})$, $\delta_i(t)$ and $\rho(t)$ are defined in Assumption III.1, and $\mu_j(\tau)$ is a bounded scalar signal to be given later in the paper. The expression of $C_{i,k}(t)$ emerges from the analysis conducted in the following section, and therefore, will be clarified later. Note however that functions (9)–(12) can be computed locally by agent i at time $t_{i,k}$ by using the information acquired from the cloud at that time, and knowing $|\mathcal{N}_q|$ for $q : i \in \mathcal{N}_q$. The functions $\varsigma_i(t)$ with $i \in \mathcal{N}$ are referred to as *threshold functions*, since $\varsigma_i(t)$ a threshold that $\sigma_{i,k}(t)$ must overcome to trigger the cloud access $t_{i,k+1}$ of agent i.

Remark III.2. *In terms of our motivating application, an agent's accesses the cloud correspond to the times when an* AUV *comes to the water surface. A position measurement corresponds to* GPS *fix that a vehicle can obtain while on the water surface. On the other hand, when a vehicle is underwater, it cannot communicate with other vehicles or access* GPS. *Nevertheless, it has to find a control value and the next surfacing instant coping with the fact that in the future other vehicles will surface and update their control input to a yet unknown value.*

Remark III.3. *The fundamental difference between the proposed control strategy and the majority of the existing self-triggered coordination strategies for multi-agent systems is that, in the proposed strategy, an agent does not require other agents to exchange information when it needs to update its control signal. Conversely, when an agent needs to update its control signal, it uses the information that is already available in the cloud, i.e., that was previously uploaded by the other agents upon their own access times.*

IV. Main Result

In this section, we show how the multi-agent system (2), under the control algorithm defined by (5)–(12), can achieve practical consensus to the reference trajectory as by Definition III.1. First, we need to introduce a digraph induced by the sets $\mathcal{P}$ and $\mathcal{N}_i$ with $i \in \mathcal{N}$ that captures the topology of the information exchanges processed through the cloud.

Definition IV.1. *Consider the multi-agent system* (2) *under the control law* (6). *Let* $\mathcal{V} = \mathcal{N} \cup \{0\}$ *and* $\mathcal{E} = \{(j,i) : j \in \mathcal{N}_i, \ i \in \mathcal{N}\} \cup \{(0,i) : \ i \in \mathcal{P}\}$. *We say that the digraph* $(\mathcal{V},\mathcal{E})$ *is the digraph associated with the multi-agent system. Moreover, we denote the edges of the digraph as* $\mathcal{E} = \{e_1, \ldots, e_M\}$.

Note that $(j,i) \in \mathcal{E}$ for some $i,j \in \mathcal{N}$ if and only if agent i downloads the information uploaded by agent

j, while $(0,i) \in \mathcal{E}$ if and only if agent i is a leader, i.e., if it receives information about the reference trajectory. Therefore, the digraph $(\mathcal{V},\mathcal{E})$ represents the topology of the information exchanges that are processed through the cloud. The following assumption ensures that the information about the reference trajectory can reach all the agents in the system.

Assumption IV.1. *The digraph $(\mathcal{V},\mathcal{E})$ associated with the multi-agent system* (2) *contains a spanning tree with root in the vertex* 0*. Namely, we let, without loss of generality, $\mathcal{E} = \mathcal{T} \cup \mathcal{C}$, where $\mathcal{T} = \{e_1,\dots,e_{N-1}\}$, $\mathcal{C} = \{e_N,\dots,e_M\}$, and $(\mathcal{V},\mathcal{T})$ is a spanning tree with root in the vertex* 0*.*

For each edge $e_\ell = (j,i) \in \mathcal{E}$, with $\ell \in \{1,\dots,M\}$, we let $y_\ell(t) = x_j(t)-x_i(t)$ if $j \in \mathcal{N}$, and $y_\ell(t) = r(t)-x_i(t)$ if $j = 0$. In other words, $y_\ell(t)$ is the mismatch between the states of the two agents whose indexes appear in the edge e_ℓ. Let $y_\mathcal{T}(t) = [y_1^\top(t),\dots,y_{N-1}^\top(t)]^\top$, $y_\mathcal{C}(t) = [y_N^\top(t),\dots,y_M^\top(t)]^\top$, and

$$y(t) = [y_\mathcal{T}^\top(t), y_\mathcal{C}^\top(t)]^\top. \tag{13}$$

Let B and $B^\odot$ be respectively the incidence and in-incidence matrices of $(\mathcal{V},\mathcal{E})$, and let them be partitioned as $B = [B_\mathcal{T}, B_\mathcal{C}]$ and $B^\odot = [B_\mathcal{T}^\odot, B_\mathcal{C}^\odot]$ according to how $\mathcal{E}$ is partitioned into $\mathcal{T}$ and $\mathcal{C}$. Note that, letting $x(t) = [r^\top(t), x_1^\top(t),\dots,x_N^\top(t)]^\top$, we have

$$y_\mathcal{T}(t) = -(B_\mathcal{T}^\top \otimes I_n)x(t) \tag{14}$$
$$y_\mathcal{C}(t) = -(B_\mathcal{C}^\top \otimes I_n)x(t). \tag{15}$$

Under Assumption IV.1, $B_\mathcal{T}$ has a left-pseudoinverse $B_\mathcal{T}^\dagger$ (see [15] for further details), therefore, from (14) and (15), it follows that

$$y_\mathcal{C}(t) = ((B_\mathcal{T}^\dagger B_\mathcal{C})^\top \otimes I_n) y_\mathcal{T}(t). \tag{16}$$

Finally, let the reduced edge Laplacian L_r of $(\mathcal{V},\mathcal{E})$ be defined as in Section II. If $(\mathcal{V},\mathcal{E})$ is itself a spanning tree, let $y_\mathcal{T}(t) = y(t)$, $B_\mathcal{T} = B$ and $L_r = L_e$, where L_e is also defined in Section II. Next, let us introduce some signals which shall be used in the convergence analysis. Consider the signals

$$v_i(t) = c\left(p_i(r(t)-x_i(t)) + \sum_{j\in\mathcal{N}_i}(x_j(t)-x_i(t))\right), \tag{17}$$

for all $i \in \mathcal{N}$. Note that $v_i(t)$ can be obtained from (6) by substituting the measurements $r_{i,k}, x_{i,k}$ and the estimates $\hat{x}_j^{i,k}$ respectively with $r_i(t), x_i(t)$ and $x_j(t)$. Let $v(t) = [0_n^\top, v_1(t)^\top,\dots,v_N(t)^\top]^\top$ so that we can rewrite (17) compactly as

$$v(t) = c(B^\odot \otimes I_n)y(t). \tag{18}$$

Let $\tilde{u}_i(t)$ be the mismatch between the actual control input $u_i(t)$ and $v_i(t)$ for each $i \in \mathcal{N}$, i.e.,

$$\tilde{u}_i(t) = u_i(t) - v_i(t), \tag{19}$$

and let $\tilde{u}(t) = [0_n^\top, \tilde{u}_1(t)^\top,\dots,\tilde{u}_N(t)^\top]^\top$. We are now in the position to state our first convergence result.

Theorem IV.1. *Consider the multi-agent system* (2)*, and let Assumptions III.1 and IV.1 hold. If $\|\tilde{u}_i(t)\| \le \varsigma_i(t)$ for all $t \in [0,\bar{t})$ and all $i \in \mathcal{N}$, then there exist $\alpha,\lambda > 0$ such that $\|y_\mathcal{T}(t)\| \le \eta(t)$ for all $t \in [0,\bar{t})$, where*

$$\eta(t) = \alpha\left(\eta_0\, \mathrm{e}^{-c\lambda t} + \|B_\mathcal{T}\| \int_0^t e^{-c\lambda(t-\tau)}\|\delta(\tau)+\varsigma(\tau)\|d\tau\right), \tag{20}$$

$\eta_0 = \|y_\mathcal{T}(0)\|$, $\delta(t) = [\delta_0(t),\delta_1(t),\dots,\delta_N(t)]^\top$ and $\varsigma(t) = [0,\varsigma_1(t),\dots,\varsigma_N(t)]$.

Proof. Substituting (19) into (2), we have

$$\dot{x}_i(t) = v_i(t) + \tilde{u}_i(t) + d_i(t). \tag{21}$$

Letting $d(t) = [\dot{r}^\top(t), d_1^\top(t),\dots,d_N^\top(t)]^\top$, (21) can be rewritten compactly as

$$\dot{x}(t) = v(t) + \tilde{u}(t) + d(t). \tag{22}$$

Left-multiplying both sides of (22) by $-(B_\mathcal{T}^\top \otimes I_n)$, and using (14) and (18) and the properties of the Kronecker product, we have

$$\dot{y}_\mathcal{T}(t) = -c(B_\mathcal{T}^\top B^\odot \otimes I_n)y(t) - (B_\mathcal{T}^\top \otimes I_n)(\tilde{u}(t)+d(t)), \tag{23}$$

Substituting (13) into (23), observing that (16) holds thanks to Assumption IV.1, and using the properties of the Kronecker product, we have

$$\dot{y}_\mathcal{T}(t) = -c(L_r \otimes I_n)y_\mathcal{T}(t) - (B_\mathcal{T}^\top \otimes I_n)(\tilde{u}(t)+d(t)), \tag{24}$$

where L_r is the reduced edge Laplacian of $(\mathcal{V},\mathcal{E})$, as defined in (1). The Laplace solution of (24) reads

$$y_\mathcal{T}(t) = \mathrm{e}^{-cL't}\, y_\mathcal{T}(0) - \int_0^t \mathrm{e}^{-cL'(t-\tau)}\, B'(\tilde{u}(t)+d(t))d\tau, \tag{25}$$

where we have denoted $L' = L_r \otimes I_n$ and $B' = B_\mathcal{T}^\top \otimes I_n$ for brevity. Taking norms of both sides in (25), and using the triangular inequality, the properties of the Kronecker product, Assumption III.1, and the hypothesis $\|u_i(t)\| \le \varsigma_i(t)$ for all $t \in [0,\bar{t})$ and all $i \in \mathcal{N}$, we have

$$\begin{aligned}\|y_\mathcal{T}(t)\| \le& \|\mathrm{e}^{-cL't}\| \cdot \|y_\mathcal{T}(0)\| \\ &+ \|B_\mathcal{T}\| \int_0^t \|\mathrm{e}^{-cL'(t-\tau)}\| \|\delta(\tau)+\varsigma(\tau)\|d\tau,\end{aligned} \tag{26}$$

for all $t \in [0,\bar{t})$, where $\delta(t)$ and $\varsigma(t)$ are defined in the theorem statement. Since L_r is positive definite, $-L' = -L_r \otimes I_n$ is Hurwitz, and therefore there exist $\alpha,\lambda > 0$ such that

$$\|\mathrm{e}^{-cL't}\| \le \alpha\, \mathrm{e}^{-c\lambda t} \quad \forall t \ge 0. \tag{27}$$

The proof is concluded by substituting (27) into (26). □

Remark IV.1. *The positive scalar λ must be smaller than $\min\{\mathrm{eig}(L_r)\}$, but can be chosen as close to that as desired. If L_r is diagonalizable, one can choose $\lambda = \min\{\mathrm{eig}(L_r)\}$ and $\alpha = \|V\|\|V^{-1}\|$, where $L_r = V\Lambda V^{-1}$ and Λ is diagonal [16].*

Corollary IV.1. *Under the same hypotheses as Theorem IV.1, we have $\|u_i(t)\| \le \mu_i(t)$, for all $t \in [0,\bar{t})$ and all $i \in \mathcal{N}$, where*

$$\mu_i(t) = \beta_i\eta(t) + \varsigma_i(t), \tag{28}$$

$\beta_i = c\|\{B_{\mathcal{T}}^{\odot} + B_{C}^{\odot}(B_{\mathcal{T}}^{\dagger}B_C)^{\top}\}_i\|$, *and* $\eta(t)$ *is defined in* (20).

Proof. Substituting (13) into (18), and using (16), we have $\upsilon_i(t) = -c(\{B_{\mathcal{T}}^{\odot} + B_{C}^{\odot}(B_{\mathcal{T}}^{\dagger}B_C)^{\top}\}_i \otimes I_n)y_{\mathcal{T}}(t)$. Taking norms of both sides, and using the Cauchy-Swartz inequality, we have

$$\|\upsilon_i(t)\| \le \beta_i\|y_{\mathcal{T}}(t)\|. \tag{29}$$

From (19), using the triangular inequality, we have

$$\|u_i(t)\| \le \|\upsilon_i(t)\| + \|\tilde{u}_i(t)\|. \tag{30}$$

Using (29) and the hypothesis $\|\tilde{u}_i(t)\| \le \varsigma_i(t)$ into (30) concludes the proof. □

The next step in our analysis is to show that the condition $\|\tilde{u}_i(t)\| \le \varsigma_i(t)$ holds for all $t \ge 0$ and all $i \in \mathcal{N}$ if the control algorithm defined by (5)–(12) is applied with $\eta(t)$ given by (20). This is formalized in the following theorem.

Theorem IV.2. *Consider the multi-agent system* (2)*, let Assumptions III.1 and IV.1 hold, and let the system be controlled by the algorithm defined by* (5)–(12)*,* (20) *and* (28)*. Then the closed-loop system does not exhibit Zeno behavior and* $\|\tilde{u}_i(t)\| \le \varsigma_i(t)$ *holds for all* $t \ge 0$ *and all* $i \in \mathcal{N}$*.*

Proof. Substituting (6) and (17) into (19), we have

$$\begin{aligned}\tilde{u}_i(t) =& c\Bigg(p_i(r_{i,k} - x_{i,k}) + \sum_{j\in\mathcal{N}_i}(\hat{x}_j^{i,k} - x_{i,k}) \\ &- p_i(r(t) - x_i(t)) - \sum_{j\in\mathcal{N}_i}(x_j(t) - x_i(t))\Bigg)\end{aligned} \tag{31}$$

for $t \in [t_{i,k}, t_{i,k+1})$, where $x_{i,k}$, $r_{i,k}$ and $\hat{x}_j^{i,k}$ are defined in (3), (4) and (7) respectively. Reordering the terms in (31), we have

$$\begin{aligned}\tilde{u}_i(t) =& c(|\mathcal{N}_i| + p_i)(x_i(t) - x_{i,k}) - cp_i(r(t) - r_{i,k}) \\ &- c\sum_{j\in\mathcal{N}_i}(x_j(t) - \hat{x}_j^{i,k})\end{aligned} \tag{32}$$

for $t \in [t_{i,k}, t_{i,k+1})$. First consider the term $x_i(t) - x_{i,k}$ in (32). Integrating (2) in $[t_{i,k}, t)$, and using (3), (5) and (6), we have

$$x_i(t) - x_{i,k} = u_{i,k}(t - t_{i,k}) + \int_{t_{i,k}}^{t} d_i(\tau)d\tau. \tag{33}$$

Now consider the term $r(t) - r_{i,k}$ in (32). Using (4), we can write

$$r(t) - r_{i,k} = \int_{t_{i,k}}^{t} \dot{r}(\tau)d\tau. \tag{34}$$

Finally, consider the terms $x_j(t) - \hat{x}_j^{i,k}$ in (32). For these terms we need to distinguish two cases, namely $t \le t_{j,h_j+1}$ and $t > t_{j,h_j+1}$. Notice that the latter case corresponds to the fact that agent j updates its control input to a value unknown to agent i. In the first case, integrating (2) for agent j in $[t_{j,h_j}, t)$, using (3) and (7), and noting that $u_j(t) = u_{j,h_j}$ for $t \in [t_{j,h_j}, t_{j,h_j+1})$, we have

$$\begin{aligned}x_j(t) - \hat{x}_j^{i,k} =& u_{j,h_j}(t - t_{i,k}) \\ &+ \int_{t_{j,h_j}}^{t} d_j(\tau)d\tau,\ t \in [t_{j,h_j}, t_{j,h_j+1}).\end{aligned} \tag{35}$$

In the second case, similar observations lead to

$$\begin{aligned}x_j(t) - \hat{x}_j^{i,k} =& u_{j,h_j}(t_{j,h_j+1} - t_{i,k}) \\ &+ \int_{t_{j,h_j+1}}^{t} u_j(\tau)d\tau + \int_{t_{j,h_j}}^{t} d_j(\tau)d\tau,\ t > t_{j,h_j+1}.\end{aligned} \tag{36}$$

Substituting (33)–(36) into (32) yields

$$\begin{aligned}\tilde{u}_i(t) =& c\Bigg((|\mathcal{N}_i| + p_i)u_{i,k}(t - t_{i,k}) \\ &- \sum_{j\in\mathcal{N}_i} u_{j,h_j}(\min\{t, t_{j,h_j+1}\} - t_{i,k}) \\ &+ \int_{t_{i,k}}^{t}((|\mathcal{N}_i| + p_i)d_i(\tau) - p_i\dot{r}(t))d\tau \\ &- \sum_{j\in\mathcal{N}_i}\int_{t_{j,h_j}}^{t} d_j(\tau)d\tau - \sum_{\substack{j\in\mathcal{N}_i\\ t\ge t_{j,h_j+1}}}\int_{t_{j,h_j+1}}^{t} u_j(\tau)d\tau\Bigg).\end{aligned} \tag{37}$$

Taking norms of both sides in (37), and using the triangular inequality and Assumption III.1, we have

$$\begin{aligned}\|\tilde{u}_i(t)\| \le& c\Bigg(\Big\|(|\mathcal{N}_i| + p_i)u_{i,k}(t - t_{i,k}) \\ &- \sum_{j\in\mathcal{N}_i} u_{j,h_j}(\min\{t, t_{j,h_j+1}\} - t_{i,k})\Big\| \\ &+ (|\mathcal{N}_i| + p_i)\Delta_{i,k}(t) + p_i\int_{t_{i,k}}^{t}\delta_0(\tau)d\tau \\ &+ \sum_{j\in\mathcal{N}_i}\Delta_{j,h_j}(t) + \sum_{\substack{j\in\mathcal{N}_i\\ t\ge t_{j,h_j+1}}}\int_{t_{j,h_j+1}}^{t}\|u_j(\tau)\|d\tau\Bigg),\end{aligned} \tag{38}$$

for $t \in [t_{i,k}, t_{i,k+1})$. Next, suppose by contradiction that some agent i at some time $\bar{t} \in [t_{i,k}, t_{i,k+1})$ attains $\|\tilde{u}_i(\bar{t})\| > \varsigma_i(\bar{t})$, while $\|\tilde{u}_q(t)\| \le \varsigma_q(t)$ for all $t \in [0, \bar{t})$ and all $q \in \mathcal{N}$. Then, using Corollary IV.1, we have

$$\|u_j(\tau)\| \le \beta_j\eta(\tau) + \varsigma_j(\tau)\ \forall t \in [0, \bar{t})\ \forall j \in \mathcal{N}_i. \tag{39}$$

Evaluating (38) for $t = \bar{t}$, and using (39), we have

$$\|\tilde{u}_i(\bar{t})\| \le \sigma_{i,k}(\bar{t}), \tag{40}$$

where $\sigma_{i,k}(t)$ is defined in (11). By (40), $\|\tilde{u}_i(\bar{t})\| > \varsigma_i(\bar{t})$ implies $\sigma_{i,k}(\bar{t}) > \varsigma_i(\bar{t})$. But this is a contradiction, since the scheduling rule (8)–(12) and (20) guarantees that $\sigma_{i,k}(t) \le \varsigma_i(t)$ for all $t \in [t_{i,k}, t_{i,k+1})$ and all $k \in \mathbb{N}_0$.

To exclude that the system exhibits Zeno behavior, consider the conditions (8) that trigger the cloud accesses. From (9), we see that the triggering condition $\Delta_{i,k}(t) \ge \zeta_{i,k}(t)$ requires $t - t_{i,k} \ge \varsigma_{q,\infty}/(2c\delta_{i,0}|\mathcal{N}_q|)$ for some $q \in \mathcal{N}$, and therefore, it cannot generate Zeno behavior. Next, consider the condition $\sigma_{i,k}(t) \geq \varsigma_i(t)$. Evaluating (11) for $t = t_{i,k}$ we have

$$\sigma_{i,k}(t_{i,k}) = c \sum_{j\in\mathcal{N}_i} \Delta_{j,h_j}(t_{i,k}). \tag{41}$$

Recalling that $t_{i,k} \in [t_{j,h_j}, t_{j,h_j+1})$, and noting that (8) guarantees $\Delta_{j,h_j}(t) < \varsigma_i(t)/(2c|\mathcal{N}_i|)$ for all $t \in [t_{j,h_j}, t_{j,h_j+1})$, from (41) we have

$$\sigma_{i,k}(t_{i,k}) \leq \varsigma_i(t_{i,k})/2. \tag{42}$$

Differentiating both sides of (11), and using (9), the continuity of $\sigma_{i,k}(t)$ and the triangular inequality, we have

$$\sigma_{i,k}(t) \leq \sigma_{i,k}(t_{i,k}) + \int_{t_{i,k}}^{t} s_{i,k}(\tau)d\tau, \tag{43}$$

where

$$\begin{aligned} s_{i,k}(t) =& c\Bigg((|\mathcal{N}_i| + p_i)(\|u_{i,k}\| + \delta_i(t)) + p_i\delta_0(t) \\ &+ \sum_{\substack{j\in\mathcal{N} \\ t<t_{j,h_j+1}}} \|u_{j,h_j}\| + \sum_{j\in\mathcal{N}_i} \delta_j(t) + \sum_{\substack{j\in\mathcal{N} \\ t\geq t_{j,h_j+1}}} \mu_j(t)\Bigg) \end{aligned} \tag{44}$$

From Corollary IV.1, we have $\|u_{i,k}\| = \|u_i(t)\| \leq \mu_i(t)$ and $\|u_{j,h_j}\| = \|u_j(t)\| \leq \mu_j(t)$ for all $j \in \mathcal{N}_i$ such that $t < t_{j,h_j+1}$. Since $\mu_j(t) = \beta_j\eta(t)+\varsigma_j(t)$ is upper-bounded, $\delta_j(t) \leq \delta_{j,0}$ and $\rho(t) \leq \rho_0$ for all $t \geq 0$, from (44), we have

$$s_{i,k}(t) \leq c\Bigg((|\mathcal{N}_i| + p_i)(\bar{\mu}_i + \delta_{i,0}) + p_i\rho_0 + \sum_{j\in\mathcal{N}} (\bar{\mu}_j + \delta_{j,0}) \Bigg), \tag{45}$$

where $\bar{\mu}_j$ denotes the maximum value attained by the function $\mu_j(t)$. Denoting the right-hand side of (45) as $\bar{s}_{i,k}$, and substituting (42) and (45) into (43), we have

$$\sigma_{i,k}(t) \leq \varsigma_i(t_{i,k})/2 + \bar{s}_{i,k}(t - t_{i,k}) \tag{46}$$

From (46), a necessary condition for having $\sigma_{i,k}(t) \geq \varsigma_i(t)$ is

$$\varsigma_i(t_{i,k})/2 + \bar{s}_{i,k}(t - t_{i,k}) \geq \varsigma_i(t). \tag{47}$$

Observing that $\varsigma_i(t) = (\varsigma_i(t_{i,k}) - \varsigma_{i,\infty})\,\mathrm{e}^{-\lambda_\varsigma(t-t_{i,k})} + \varsigma_{i,\infty}$, we can rewrite (47) as $\varsigma_i(t_{i,k})/2 + \bar{s}_{i,k}(t - t_{i,k}) \geq (\varsigma_i(t_{i,k}) - \varsigma_{i,\infty})\,\mathrm{e}^{-\lambda_\varsigma(t-t_{i,k})} + \varsigma_{i,\infty}$. For any $\varsigma_i(t_{i,k}) > \varsigma_{i,\infty} \geq 0$ and any $\lambda_\varsigma > 0$, the positive solutions in the unknown τ of the equation $\varsigma_i(t_{i,k})/2 + \bar{s}_{i,k}\tau \geq (\varsigma_i(t_{i,k}) - \varsigma_{i,\infty})\,\mathrm{e}^{-\lambda_\varsigma\tau} + \varsigma_{i,\infty}$ is lower-bounded. Therefore, condition (47) cannot be satisfied for $t - t_{i,k}$ arbitrarily small. Consequently, the triggering condition $\varsigma_{i,k}(t) \geq \varsigma_i(t)$ cannot generate Zeno behavior either. We can conclude that the closed-loop system defined by (2), (5)–(12) and (20) does not exhibit Zeno behavior. □

Remark IV.2. *Agent i can compute $\mu_j(t)$ for $j \in \mathcal{N}$ by* (20) *and* (28)*, and therefore by only using some neighborhood information on the network topology (β_j for $j \in \mathcal{N}$) and the initial conditions ($\|y_\mathcal{T}(0)\|$).*

Theorems IV.1 and IV.2 amount to our main result, which is formalized as follows.

Theorem IV.3. *Consider the multi-agent system* (2)*, let Assumptions III.1 and IV.1 hold, and let the system be controlled by the algorithm defined by* (5)–(12) *and* (20)*. Then the closed-loop system does not exhibit Zeno behavior and achieves practical consensus as by Definition III.1, with tolerance $\epsilon = \max_{i\in\mathcal{N}}\{\sqrt{m_i}\}\eta_\infty$, where m_i is the number of edges in the shortest path from vertex 0 to vertex i in the graph $(\mathcal{V}, \mathcal{T})$, and $\eta_\infty = \lim_{t\to\infty}\eta(t) = \alpha\|B_\mathcal{T}\|\frac{\|\delta_\infty+\varsigma_\infty\|}{c\lambda}$, where $\delta_\infty = [\delta_{0,\infty}, \delta_{1,\infty}, \dots, \delta_{N,\infty}]^\top$ and $\varsigma_\infty = [0, \varsigma_{1,\infty}, \dots, \varsigma_{N,\infty}]^\top$.*

Proof. From Theorems IV.1 and IV.2, we have $\|y_\mathcal{T}(t)\| \leq \eta(t)$ for all $t \geq 0$, where $\eta(t)$ is defined by (20). Letting $t \to \infty$, we have therefore $\limsup_{t\to\infty}\|y_\mathcal{T}(t)\| \leq \eta_\infty$. Finally, observing that $\|r(t)-x_i(t)\| \leq \sqrt{m_i}\|y_\mathcal{T}(t)\|$ yields the desired result. □

V. Asymptotic Convergence

If the disturbances vanish quickly enough, and the reference trajectory converges quickly enough to a fixed point, then the proposed algorithm, with only small adjustments, is capable to drive all the agents to the reference point asymptotically. In this case, the following assumption is needed.

Assumption V.1. *Assumption III.1 holds with $\rho_\infty, \delta_{1,\infty}, \dots, \delta_{N,\infty} = 0$ and $\lambda_\delta < c\min\{\mathrm{eig}(L_r)\}$.*

In this scenario, the threshold functions are chosen as

$$\varsigma_i(t) = \varsigma_{i,0}\,\mathrm{e}^{-\lambda_\varsigma t}, \tag{48}$$

with

$$0 < \lambda_\varsigma < \lambda_\delta < c\min\{\mathrm{eig}(L_r)\}. \tag{49}$$

Note that Theorem IV.1 and Corollary IV.1 still hold. Moreover, solving the integral in (20), and using Assumption V.1 and (49), we have

$$\eta(t) \leq \bar{\eta}\,\mathrm{e}^{-\lambda_\varsigma t}, \tag{50}$$

where

$$\bar{\eta} = \alpha\left(\|y_\mathcal{T}(0)\| + \frac{\|\delta(0)\|}{c\lambda - \lambda_\delta} + \frac{\|\varsigma(0)\|}{c\lambda - \lambda_\varsigma} \right), \tag{51}$$

and $\delta(t)$, $\varsigma(t)$ are defined in the statement of Theorem IV.1. In the following theorem we show that this version of the proposed algorithm is still Zeno-free.

Theorem V.1. *Consider the multi-agent system* (2)*, let Assumptions IV.1 and V.1 hold, and let the system be controlled by the algorithm defined by* (5)–(11)*,* (20) *and* (48)*, with λ_ς satisfying* (49)*. Then, the closed-loop system does not exhibit Zeno behavior and $\|\tilde{u}_i(t)\| \leq \varsigma_i(t)$ holds for all $t \geq 0$ and all $i \in \mathcal{N}$.*

Proof. Reasoning as in Theorem IV.2, we can show that the scheduling rule (8)–(11), (20) and (48) guarantees $\|\tilde{u}_i(t)\| \leq \varsigma_i(t)$ for any $t \geq 0$. To show that the closed-loop system is Zeno free, consider again the condition (8) that triggers the cloud accesses. From (9), we can see that the triggering condition $\Delta_{i,k}(t) \geq \zeta_{i,k}(t)$ requires $\frac{\delta_{q,0}}{\lambda_\delta}\,\mathrm{e}^{-\lambda_\delta t_{i,k}}(1 - \mathrm{e}^{-\lambda_\delta(t-t_{i,k})}) \geq$

$\frac{\varsigma_{q,0}}{2c|\mathcal{N}_q|} \mathrm{e}^{-\lambda_\varsigma t_{i,k}} \mathrm{e}^{-\lambda_\varsigma (t-t_{i,k})}$ for some $q \in \mathcal{N}$. By (49), the previous inequality implies $(\frac{\varsigma_{q,0}}{2c|\mathcal{N}_q|} + \frac{\delta_{q,0}}{\lambda_\delta}) e^{-\lambda_\varsigma (t-t_{i,k})} \leq \frac{\delta_{q,0}}{\lambda_\delta}$. Therefore, the condition $\Delta_{i,k}(t) \geq \zeta_{i,k}(t)$ cannot generate Zeno behavior. Next, consider the condition $\sigma_{i,k}(t) \geq \varsigma_i(t)$. With similar reasoning as in Theorem IV.2, we can show that (42) and (43) still hold. Using Corollary IV.1 and (48) and (50), and recalling that $t \in [t_{i,k}, t_{i,k+1})$, we have $\|u_{i,k}\| = \|u_i(t)\| \leq \mu_i(t) \leq (\beta_i \bar{\eta} + \varsigma_{i,0}) \mathrm{e}^{-\lambda_\varsigma t}$ and $\|u_{j,h_j}\| = \|u_j(t)\| \leq \mu_j(t) \leq (\beta_j \bar{\eta} + \varsigma_{j,0}) \mathrm{e}^{-\lambda_\varsigma t}$ for all $j \in \mathcal{N}_i$ such that $t < t_{j,h_j+1}$. Replacing these two inequalities into (44), and using (48)–(50), we have

$$\begin{aligned} s_{i,k}(t) \leq & c\Bigg((|\mathcal{N}_i| + p_i)(\beta_i \bar{\eta} + \varsigma_{i,0} + \delta_{i,0}) + p_i \rho_0 \\ & + \sum_{j \in \mathcal{N}} (\beta_j \bar{\eta} + \varsigma_{j,0}) \Bigg) \mathrm{e}^{-\lambda_\varsigma t} \quad \forall t \in [t_{i,k}, t_{i,k+1}), \end{aligned} \tag{52}$$

where $\bar{\eta}$ is defined in (51). Substituting (42) and (52) in (43), we have

$$\sigma_{i,k}(t) \leq \frac{\varsigma_i(t_{i,k})}{2} + \frac{\xi_{i,k}}{\lambda_\varsigma} \mathrm{e}^{-\lambda_\varsigma t_{i,k}} (1 - \mathrm{e}^{-\lambda_\varsigma (t-t_{i,k})}), \tag{53}$$

where $\xi_{i,k}$ denotes the coefficient that multiplies $\mathrm{e}^{-\lambda_\varsigma t}$ in (52). From (53), a necessary condition for having $\sigma_{i,k}(t) \geq \varsigma_i(t)$ is

$$\frac{\varsigma_i(t_{i,k})}{2} + \frac{\xi_{i,k}}{\lambda_\varsigma} \mathrm{e}^{-\lambda_\varsigma t_{i,k}} (1 - \mathrm{e}^{-\lambda_\varsigma (t-t_{i,k})}) \geq \varsigma_i(t). \tag{54}$$

Observing that $\varsigma_i(t) = \varsigma_i(t_{i,k}) \mathrm{e}^{-\lambda_\varsigma (t-t_{i,k})}$ and $\varsigma_i(t_{i,k}) = \varsigma_{i,0} \mathrm{e}^{-\lambda_\varsigma t_{i,k}}$, (54) can be rewritten as

$$\frac{\varsigma_{i,0}}{2} + \frac{\xi_{i,k}}{\lambda_\varsigma} \geq \left(\varsigma_{i,0} + \frac{\xi_{i,k}}{\lambda_\varsigma} \right) \mathrm{e}^{-\lambda_\varsigma (t-t_{i,k})},$$

which has lower-bounded solutions in the unknown $t - t_{i,k}$. Hence, the triggering condition $\sigma_{i,k}(t) \geq \varsigma_i(t)$ cannot generate Zeno behavior either. We can conclude that the closed-loop system does not exhibit Zeno behavior. □

Theorems IV.1 and V.1 amount to our asymptotic convergence result, which is formalized as follows.

Theorem V.2. *Consider the multi-agent system* (2)*, let Assumptions IV.1 and V.1 hold, and let the system be controlled by the algorithm defined by* (5)–(11)*,* (20) *and* (48)*, with* λ_ς *satisfying* (49)*. Then the closed-loop system does not exhibit Zeno behavior and* $\lim_{t\to\infty} \|r(t) - x_i(t)\| = 0$ *for all* $i \in \mathcal{N}$.

Proof. Reasoning as in Theorem IV.3, we have $\|r(t) - x_i(t)\| \leq \sqrt{m_i}\eta(t)$, where m_i is defined in Theorem IV.3. From (50), letting $t \to \infty$ in the previous inequality yields the desired convergence result. □

VI. Numerical Simulations

In this section, we present two numerical simulations of the proposed algorithm, which demonstrate respectively practical consensus and asymptotic convergence. We consider a system made up of $N = 4$ agents plus the reference trajectory, with graph topology as in Figure 1. Note that for

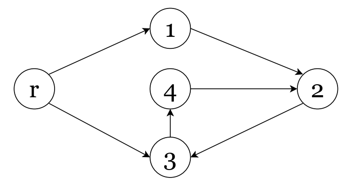

Fig. 1. Topology of the multi-agent system used in the simulations. The node r represents the reference trajectory.

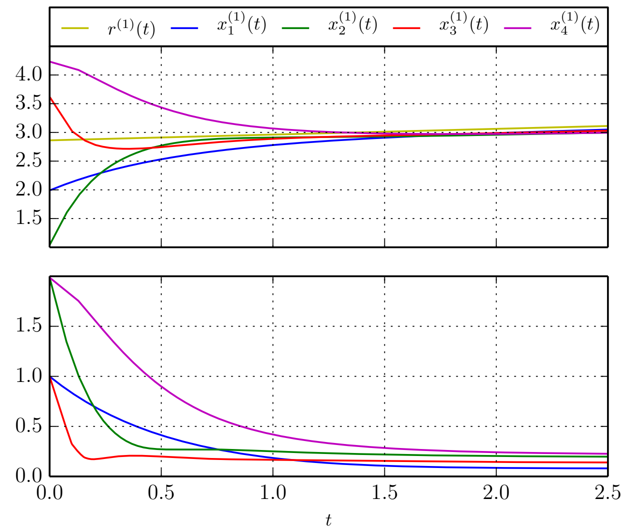

Fig. 2. Results of the first simulation scenario. Top: first position variable $x_i^{(1)}(t)$ for each agent and $r^{(1)}(t)$ for the reference trajectory. Bottom: error norm $||x_i(t) - r(t)||$ for each agent.

this topology Assumption IV.1 is satisfied, and we have $\lambda = \min\{\mathrm{eig}(L_r)\} = 0.53$, $\|B_\mathcal{T}\| = 1.90$ and $\alpha = 3.61$. For the first simulation, we let $c = 1.8$, $\varsigma(t) = [0.0, 9.5 \cdot 10^{-3}, 0.48, 0.57, 0.67]^\top 5.0\,\mathrm{e}^{-4.0t} + 5.0$, $\delta(t) = [99.9, 1.0, 2.0, 3.0, 4.0] \cdot 10^{-3}$ for all $t \geq 0$. We let the derivative of the reference trajectory and the disturbances be sinusoidal, namely $\dot{r}(t) = \delta_0(t)[\cos(2\pi f_0 t), \sin(2\pi f_0 t)]$, $d_i(t) = \delta_i(t)[\cos(2\pi f_i t), \sin(2\pi f_i t)]$. Note that with this choice Assumption III.1 is satisfied. The frequencies are chosen as $f_i = i/T$ for $i \in \{0, 1, \ldots, N\}$, where T is the simulation time. The simulation runs for $t \in [0.0, 2.5]$, with a fixed step of 10^{-4} (the physical time scale can be chosen according to the particular application). Figures 2 and 4 show the results of this simulation. We have a total of 305 updates corresponding to an average inter-update time of 0.033, two order of magnitudes larger than the simulation step. Hence, the simulation corroborates the absence of Zeno behavior. For the second simulation, we let $c = 1.0$, $\varsigma(t) = [0.0, 9.5 \cdot 10^{-3}, 0.48, 0.57, 0.67]^\top 10.0\,\mathrm{e}^{-1.0t}$, $\delta(t) = [99.9, 1.0, 2.0, 3.0, 4.0] \cdot 10^{-3}\,\mathrm{e}^{-1.1t}$. Note that, with these choices, Assumption V.1 and (48) are satisfied, and the derivative of the reference trajectory and the disturbances vanish asymptotically. The simulation runs for $t \in [0.0, 10.0]$, with the same step as before. Figures 3 and 4 show the results of this simulation. In this case, we have a total of 1418 updates, corresponding to an average inter-event time of 0.0282, still two order of magnitudes larger than the simulation step. Hence, the simulation corroborates

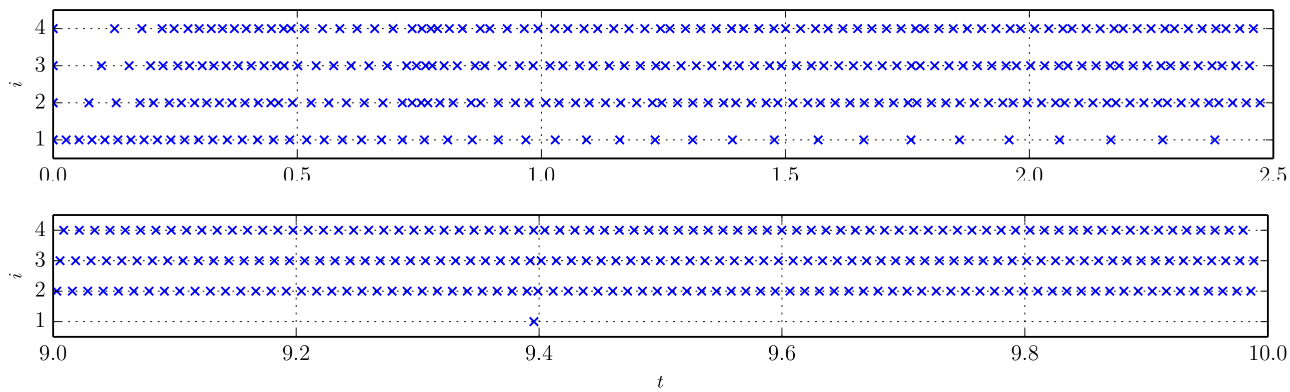

Fig. 4. Top: updates in the first simulation scenario for each agent. Bottom: updates in the second simulation scenario for $t \in [9.0, 10.0]$ for each agent.

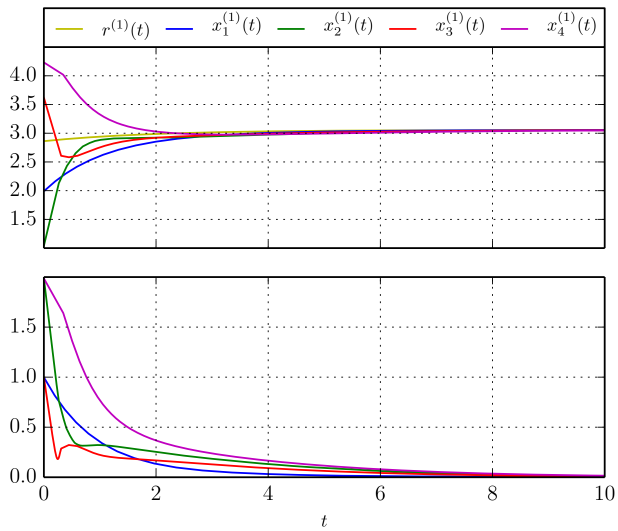

Fig. 3. Results of the second simulation scenario. Top: first position variable $x_i^{(1)}(t)$ for each agent and $r^{(1)}(t)$ for the reference trajectory. Bottom: error norm $||x_i(t) - r(t)||$ for each agent.

the absence of Zeno behavior.

VII. Conclusions and Future Developments

A cloud-supported control algorithm for leader-follower trajectory tracking in a network of mobile agents under disturbances has been proposed. The considered setup allows multi-agent coordination in case of interdicted communication among the agents. Specifically, the scenario of controlling a formation of AUVs has been considered as a motivating example. The control algorithm overcomes the limitation of having a pre-assigned trajectory for the whole fleet as well as synchronizing the surfacing of the agents [7]. Sufficient conditions for both bounded and asymptotic convergence have been identified, in terms of the topology of the information exchanges with the cloud, and of the scheduling of the control updates. The proposed control algorithm is effective in guaranteeing the overall stability despite each agent receiving outdated information and not knowing other agents' future control inputs. Future work will further develop the approach of the paper considering more complex agent dynamics and control objectives. Also, non-idealities in the cloud access, such as delays and packet losses, will be taken into account.

References

[1] W. Ren, "Distributed Consensus in Multi-vehicle Cooperative Control," *IEEE Control Systems Magazine*, April 2007.

[2] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, 2006.

[3] J. A. Fax and R. M. Murray, "Graph laplacians and stabilization of vehicle formations," *IEEE Transactions on Automatic Control*, vol. 15, no. 1, 2002.

[4] L. Stutters, H. Liu, C. Tiltman, and D. J. Brown, "Navigation technologies for autonomous underwater vehicles," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 38, no. 4, 2008.

[5] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, 2014.

[6] J. W. Nicholson and A. J. Healey, "The Present State of Autonomous Underwater Vehicle (AUV) Applications and Technologies," *Marine Technology Society Journal*, vol. 42, no. 1, 2008.

[7] E. Fiorelli, N. E. Leonard, P. Bhatta, D. A. Paley, R. Bachmayer, and D. M. Fratantoni, "Multi-AUV control and adaptive sampling in Monterey Bay," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 4, pp. 935–948, 2006.

[8] W. P. M. H. Heemels, K. H. Johansson, and P. Tabuada, "An Introduction to Event-triggered and Self-triggered Control," in *IEEE Conference on Decision and Control*, 2012.

[9] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2012.

[10] M. T. Hale and M. Egerstedt, "Differentially private cloud-based multi-agent optimization with constraints," in *American Control Conference*, 2015.

[11] P. Pandey, D. Pompili, and J. Yi, "Dynamic Collaboration between Networked Robots and Clouds in Resource-Constrained Environments," *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 2, pp. 471–480, 2015.

[12] A. Adaldo, D. Liuzza, D. V. Dimarogonas, and K. H. Johansson, "Control of Multi-Agent Systems with Event-Triggered Cloud Access," in *European Control Conference*, 2015.

[13] D. Zelazo, A. Rahmani, and M. Mesbahi, "Agreement via the edge Laplacian," in *IEEE Conference on Decision and Control*, 2007.

[14] Z. Zeng, X. Wang, and Z. Zheng, "Nonlinear Consensus under Directed Graph via the Edge Laplacian," in *Chinese Control and Decision Conference*, 2014.

[15] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton Univerisity Press, 2010.

[16] R. A. Horn and C. R. Johnson, *Matrix Analysis*, 2nd ed. Cambridge University Press, 2012.