Scheduling Local Information Exchange in Linear Multiagent Systems Through an Event-Triggering Approach[⋆]

Stefan Ristevski[*]
Combine Control Systems AB

Tansel Yucelen[†]
University of South Florida

Jonathan A. Muse[‡]
Air Force Research Laboratory

This paper presents a distributed event-triggered control algorithm for linear time-invariant multiagent systems to schedule local information exchange. The proposed distributed event-triggered control involves a dynamic threshold, which is a function of the error between a dynamical system and its reference model and, in addition, contains an exponentially decaying term to minimize local information exchange during the transient response. This dynamic threshold significantly decreases network utilization (i.e., number of events). Moreover, in contrast to the sampled data exchange approach, which is widely used in the event-triggered control literature, we use a solution-predictor curve exchange method. This method predicts the time trajectories of agents and has the ability to significantly decrease network utilization compared to sampled data exchange. Using graph theory and Lyapunov stability tools, we provide rigorous system-theoretic analysis and show the efficacy of the proposed approach through a numerical example.

I Introduction

A Literature Review

Small and low cost devices with mobility, computing, communication, and sensing capabilities have gained the attention of researchers in the several past decades. This trend has led to establishing strong theoretical foundations on how these devices working together are able to accomplish a common task by only sharing local information between each other through a network (hereinafter referred to as multiagent systems). Application range of networked multiagent systems encompasses many scientific, civilian, and military operations including, for example, traffic management, transportation, power distribution, internet delivery,

[*]Stefan Ristevski is a Controls Engineer at Combine Controls Systems AB (http://www.combine.se) Lund 41755, Sweden (email: stefan.ristevski@combine.se).

[†]T. Yucelen is an Associate Professor of the Department of Mechanical Engineering and the Director of the Laboratory for Autonomy, Control, Information, and Systems (LACIS, http://lacis.eng.usf.edu/) at the University of South Florida, Tampa, FL 33620, USA (email: yucelen@usf.edu). T. Yucelen is also a Senior Member of the American Institute of Aeronautics and Astronautics and a Member of the National Academy of Inventors.

[‡]J. A. Muse is a Research Aerospace Engineer of the Autonomous Control Branch at the Air Force Research Laboratory Aerospace Systems Directorate, WPAFB, Ohio 45433, United States of America (email: jonathan.muse.2@us.af.mil).

[⋆]This research was supported by the Air Force Research Laboratory Aerospace Systems Directorate under the Universal Technology Corporation Grant 162642-20-25-C1.

and emergency response (see, e.g., [1–4]). These systems have several important features that are beneficial to be studied. One such key feature is how they can accomplish a given application with guaranteed closed-loop system stability by only utilizing inter-agent (i.e., local) information exchange. Furthermore, it is essential to prevent potential network overload and decrease wireless communication costs in the design and implementation of these systems. Therefore, it is a practical need to also study how to schedule inter-agent information exchange.

Whether for discrete-time algorithms or for continuous-time algorithms, information exchange is performed over fixed periods (i.e., naturally for algorithms in discrete-time form and through approximation for algorithms in continuous-time form). To prevent the fixed period demand, event-triggered control theory has become an important method in the literature, where it allows information exchange predicated on sampled data in an aperiodic and asynchronous way such that exchange information is performed when an error of interest violates a-priori defined fixed or dynamic threshold (see, e.g., [5–11]). In the networked multiagent systems literature, local information exchange is also predicated on sampled data (see, e.g., [12–21]).

In contrast to sampled data information exchange, the authors of [10] propose information exchange based on curve-fitted command functions and exact command functions. In particular, they show that the number of events reduces dramatically with these functions as compared with its sampled data counterpart. Yet, their study is outside the scope of networked multiagent systems. Motivated from this standpoint, they generalize their results to these systems in [22], where local information exchange is predicated on solution-predictor curves. Similar in spirit to curve-fitted command functions and exact command functions, these curves predict the solution of each agent and this prediction significantly reduces the number of agent-wise events as compared with the sampled data local information exchange. However, the results in [22] hold only for agents having single integrator dynamics. Note that the authors of [22] have extended their approach from single integrator models to linear time-invariant models in [23], yet here there is a drawback that when an event occurs, in addition to the system state the state of the reference model is exchanged as well.

B Contribution and Organization

The contribution of this paper is a distributed event-triggered control algorithm for linear time-invariant multiagent systems to schedule local information exchange. The proposed distributed event-triggered control involves a dynamic threshold, which is a function of the error between a dynamical system and its reference model and, in addition, contains an exponentially decaying term to minimize local information exchange during the transient response. This dynamic threshold significantly decreases network utilization (i.e., number of events). Moreover, in contrast to the sampled data exchange approach, which is widely used in the event-triggered control literature, we use a solution-predictor curve exchange method in [22]. This method predicts the time trajectories of agents and has the ability to significantly decrease network utilization compared to sampled data exchange. System-theoretic analysis of the proposed method is also included using tools from graph theory and Lyapunov stability theory.

Organization of this paper is as follows. In Section II, we provide the notation used throughout this paper and basic definitions from graph-theory. Problem formulation is given in Section III and system-theoretical analysis is given in Section IV. In Section V, we provide an illustrate numerical example to show the efficacy of the proposed approach and finally concluding remarks are given in Section VI.

II Notation and Definitions

The notation used in this paper is given in Table 1. In addition, we consider a connected and undirected graph $\mathcal{G}$ throughout this paper, where $\mathcal{D}(\mathcal{G}) \triangleq \mathrm{diag}([d_1, \ldots, d_N]) \in \mathbb{R}^{N\times N}$ denotes the degree matrix with d_i being the degree of agent i, $\mathcal{A}(\mathcal{G}) \in \mathbb{R}^{N\times N}$ denotes the adjacency matrix that is defined as $[\mathcal{A}(\mathcal{G})]_{ij} = 1$ when agents i and j are neighbors and otherwise $[\mathcal{A}(\mathcal{G})]_{ij} = 0$, $\mathcal{L}(\mathcal{G}) \triangleq \mathcal{D}(\mathcal{G}) - \mathcal{A}(\mathcal{G})$ denotes the Laplacian matrix, and $i \sim j$ denotes the neighboring relationship (for details we refer to [3, 24]). Note that the eigenvalues of the Laplacian matrix of a connected and undirected graph are ordered as $0 = \lambda_1(\mathcal{L}(\mathcal{G})) \leq \lambda_2(\mathcal{L}(\mathcal{G})) \leq \cdots \leq \lambda_N(\mathcal{L}(\mathcal{G}))$, where the eigenvector 1_N corresponds to the zero eigenvalue [3]. For a connected and undirected graph $\mathcal{G}$, note also that $\mathcal{F}(\mathcal{G}) \triangleq \mathcal{L}(\mathcal{G}) + \mathcal{K} \in \mathbb{R}_+^{N\times N}$ is positive-definite matrix (here, $\mathcal{K} = \mathrm{diag}([k_1, \ldots, k_N])$, $k_i \in \overline{\mathbb{Z}}_+$, $i = 1, \ldots, N$, such that at least one diagonal entry of $\mathcal{K}$ is nonzero) [4].

$\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Z}_+$	the sets of natural numbers, integers, positive integers
$\mathbb{R}$, $\mathbb{R}^n$, $\mathbb{R}^{n\times m}$	the sets of real numbers, $n \times 1$ real vectors, $n \times m$ real matrices
$\mathbb{R}_+$, $\overline{\mathbb{R}}_+$	the sets of positive, nonnegative real numbers
$\mathbb{R}_+^{n\times n}$, $\overline{\mathbb{R}}_+^{n\times n}$	the sets of positive-definite, nonnegative-definite real matrices
I_n, 0_n, $0_{n\times m}$	$n\times n$ identity matrix, $n\times 1$ zero vector, $n\times m$ zero matrix
$\triangleq$, $(\cdot)^{\mathrm{T}}$, $(\cdot)^{-1}$	equality by definition, transpose, inverse
$\overline{\lambda}(A)$, $\underline{\lambda}(A)$	maximum, minimum eigenvalue of $A \in \mathbb{R}^{n\times n}$
$\|\cdot\|_2$, $\|A\|_2 \triangleq \left(\overline{\lambda}(A^{\mathrm{T}}A)\right)^{\frac{1}{2}}$	Euclidean norm, induced 2-norm of $A \in \mathbb{R}^{n\times m}$

Table 1. Notation

III Problem Formulation

Using the results in [22] as building blocks in the following subsections, they are generalized to networked multiagent systems with agents having linear time-invariant dynamics.

A Distributed Control Architecture

Let the dynamics of agent i in a connected and undirected graph $\mathcal{G}$ be

$$\dot{x}_i(t) = Ax_i(t) + Bu_i(t), \tag{1a}$$

$$z_i(t) = Ex_i(t), \tag{1b}$$

where $A \in \mathbb{R}^{n\times n}$ denotes the system matrix, $B \in \mathbb{R}^{n\times m}$ denotes the input matrix, $E \in \mathbb{R}^{p\times n}$ denotes the output matrix, $x_i(t) \in \mathbb{R}^n$ denotes the state vector of agent i, and $z_i(t) \in \mathbb{R}^p$ denotes the output of agent i. Each agent i has continuous access to its state, $x_i(t)$, and its output, $z_i(t)$. Let the local control architecture of agent i be given by

$$u_i(t) = -K_1 x_i(t) - K_2 x_{ei}(t). \tag{2a}$$

$$\dot{x}_{ei} = z_i(t) - \hat{\mu}_i(t). \tag{2b}$$

Here, $K_1 \in \mathbb{R}^{m\times n}$ and $K_2 \in \mathbb{R}^{m\times p}$ are feedback control gain matrices and $x_{ei}(t) \in \mathbb{R}^p$ is the integral state of agent i. Moreover, $\hat{\mu}_i(t) \in \mathbb{R}^p$ is the last broadcast of $\mu_i(t) \in \mathbb{R}^p$ of agent i to its neighbors through an event-triggering approach (details below) and $\mu_i(t)$ is given by [22]

$$\dot{\mu}_i(t) = -\gamma\left[\sum_{i\sim j}\left(\hat{\mu}_i(t)-\hat{\mu}_j(t)\right)+k_i\left(\hat{\mu}_i(t)-c(t)\right)\right], \quad \mu_i(0)=\mu_{i0}. \tag{3}$$

In (3), $\gamma \in \mathbb{R}$ is a design parameter, $k_i \in \{0,\ 1\}$ denotes whether agent i is a leader agent or a follower agent (i.e., $k_i = 0$ for follower agents and $k_i = 1$ for leader agents), and $\hat{\mu}_j \in \mathbb{R}^p$ for $i \sim j$ denotes the latest broadcasted value of $\mu_j(t) \in \mathbb{R}^p$ from agent j to its neighbors through an event-triggering approach (once again, details below).

Next, for agent i, using the control signal given by (2a) in (1a), one can write

$$\dot{x}_i(t) = A_{\mathrm{m}}x_i(t) + B_{\mathrm{m}}x_{ei}(t), \tag{4}$$

where $A_{\mathrm{m}} \triangleq A - BK_1 \in \mathbb{R}^{n\times n}$ is Hurwitz and $B_{\mathrm{m}} \triangleq -BK_2 \in \mathbb{R}^{n\times m}$. By combining agent i's state vector with its integral state (i.e., by letting $\overline{x}_i \triangleq [x_i,\ x_{ei}]^{\mathrm{T}} \in \mathbb{R}^{\overline{n}}$ with $\overline{n} \triangleq n+p$), we have

$$\dot{\overline{x}}_i(t) = \underbrace{\begin{bmatrix} A_{\mathrm{m}} & B_{\mathrm{m}} \\ E & 0_{p\times m}\end{bmatrix}}_{\overline{A}_{\mathrm{m}}}\overline{x}_i(t) + \underbrace{\begin{bmatrix} 0_{n\times m} \\ -I_p\end{bmatrix}}_{\overline{B}_{\mathrm{m}}}\hat{\mu}_i(t), \quad \overline{x}_i(0) = \overline{x}_{i0}. \tag{5}$$

Here, gain feedback matrices, K_1 and K_2, are designed such that the matrix $\overline{A}_{\mathrm{r}}$ is Hurwitz. In addition, note that in the dynamics of agent i given by (5), the last broadcasted value of $\mu_i(t)$ is used (i.e., $\hat{\mu}_i(t)$). Yet, agent i has continuous access to the value of $\mu_i(t)$ through (3). We now formalize the ideal continuous information exchange dynamics by defining

$$\dot{\overline{x}}_{\mathrm{m}i}(t) = \overline{A}_{\mathrm{m}}\overline{x}_{\mathrm{m}i}(t) + \overline{B}_{\mathrm{m}}\mu_i(t) + \overline{L}\left(\overline{x}_i(t) - \overline{x}_{\mathrm{m}i}(t)\right), \quad \overline{x}_{\mathrm{m}}(0) = \overline{x}_{\mathrm{m}i0}, \tag{6}$$

where $\overline{x}_{\mathrm{m}i} \in \mathbb{R}^{\overline{n}}$ and $\overline{L} \triangleq \alpha I_{\overline{n}} \in \mathbb{R}^{\overline{n}\times\overline{n}}$ with $\alpha \in \mathbb{R}_+$ being a design parameter. Hereinafter we will refer to (6) as the ideal closed-loop reference model of agent i [25–30]. We now define the error signal of agent i as $\overline{e}_i(t) \triangleq \overline{x}_i(t) - \overline{x}_{\mathrm{m}i}(t)$ to capture deviation of agent i's states from its closed-loop reference model. Then, using (5) and (6) one can write the error dynamics as

$$\dot{\overline{e}}_i(t) = \left(\overline{A}_{\mathrm{m}} - \overline{L}\right)\overline{e}_i(t) + \overline{B}_{\mathrm{m}}\tilde{\mu}_i(t), \tag{7}$$

where $\tilde{\mu}_i(t) \triangleq \hat{\mu}_i(t) - \mu_i(t) \in \mathbb{R}$ is the difference between the latest broadcasted value $\hat{\mu}_i(t)$ and the current value $\mu_i(t)$ for agent i.

Our objective is to design a distributed control architecture such that all agents in the multiagent system that have linear time-invariant dynamics follow a desired trajectory available only to the leader agents predicated on an event-triggering approach to minimize network utilization. We next outline the event-triggering architecture with dynamic threshold and solution-predictor method.

B Event-Triggering Architecture

In this paper, we consider the event-triggering rule given by

$$\|\tilde{\mu}_i(t)\|_2 \leq \varepsilon\|\overline{e}_i(t)\|_2 + \phi(t). \tag{8}$$

In (8), $\varepsilon \in \mathbb{R}_+$ is a design parameter and $\phi(t) \in \overline{\mathbb{R}}_+$ is an exponentially decaying term whose dynamics satisfies

$$\dot{\phi}(t) = -\kappa(\phi(t) - \phi_{\rm f}), \quad \phi(t) = \phi_0, \tag{9}$$

where $\kappa \in \mathbb{R}_+$ represents the time constant of $\phi(t)$. We refer to [Remark 3.1, 22] for details on the selection of $\phi_{\rm f}$.

For the event-triggering scenario given by (8) we consider two cases [22]:

a) Agent i will broadcast a smapled data of its μ-dynamics value, $\mu_i(t)$, to its neighbors (i.e., $\hat{\mu}_i(t) = \mu_i(t_{di})$ for $t \in [t_{di}, t_{(d+1)i})$) through a zero-order-hold operator, when it detect that the event-triggering rule given by (8) is violated. Moreover, all time instants when the event-triggering rule (8) is violated are stored in the sequences $\{t_{di}\}_{d\in\mathrm{N}}$ for $i = 1, \ldots, N$.

b) Agent i will broadcast a solution-predictor curve of its μ-dynamics, $\mu_i(t)$, to its neighbors (i.e., $\hat{\mu}_i(t) = \mathcal{S}_{di}(t)$ for $t \in [t_{di}, t_{(d+1)i})$), when it detect that the event-triggering rule given by (8) is violated. Once again, all time instants when the event-triggering rule (8) is violated are stored in the sequences $\{t_{di}\}_{d\in\mathrm{N}}$ for $i = 1, \ldots, N$. Note that, the solution-predictor curve method (details in Section III.C) as compared with the state-of-the-art results on event-triggered networked multiagent systems is new.

Finally, to better illustrate our problem formulation we refer to Figure 1. In this figure, we show a multiagent system consisting of 4 agents and, without loss of any generality, agent 1 is chosen as leader agent. Moreover, inter-agent communication is predicated ont he event-triggering approach above where the goal is that all agents converge to the desired trajectory only available to the leader agent (i.e., agent 1 in this figure).

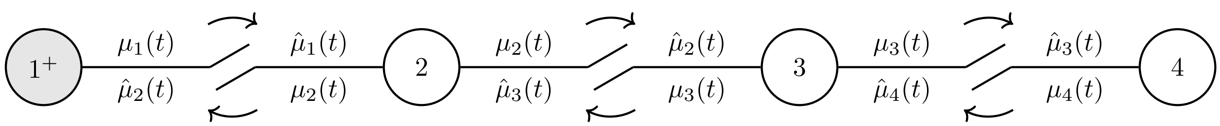

Figure 1. An example of linear time-invariant multiagent system consisting of 4 agents such that inter-agent communication is based on the event-triggering approach. Here, agent 1 is leader while the remaining agents are followers.

C Solution Predictor Curve [22]

We here provide the definition of the solution-predictor curve method for information exchange when the event-triggering rule (8) is violated for case b) as outlined in Section III.B. In particular, consider the μ-dynamics of agent i as given by (3). When an event occurs at agent i (i.e., when (8) is violated) at time $t = t_{di}$, the solution of (3) can be approximated between two events with the initial condition $\mu_i(t_{di})$, the inputs $\hat{\mu}_j(t_{dj})$ for $i \sim j$ and $c(t_{di})$ such that these values are considered as constant, as

$$\mathcal{S}_{di}(t) = \left(\hat{\mu}_{i0} - \frac{\sum\limits_{i\sim j} \hat{\mu}_j(t_{dj}) + k_i c(t_{di})}{d_i + k_i} \right) e^{-\gamma_2 d_i (t-t_0)} + \frac{\sum\limits_{i\sim j} \hat{\mu}_j(t_{dj}) + k_i c(t_{di})}{d_i + k_i}. \tag{10}$$

Note that with (10) the evolution of $\mu_i(t)$ for agent i over time can be approximated. Every agent in the networked multiagent system stores a function in the form given by (10) for each of its neighbors. When an event occurs, agent i will broadcast to its neighbors $\hat{\mu}_{i0} = \mu_i(t_{di})$, $t_0 = t_{di}$, and the value of $\left(\sum_{i\sim j} \hat{\mu}_j(t_{dj}) + k_i c(t_{di})\right)/(k_i + d_i)$.

IV System Theoretical Analysis

In this section, we provide a rigorous stability analysis for the proposed distributed event-triggered control architecture. To this end, let $\overline{x}(t) \triangleq [\overline{x}_1(t), \ldots, \overline{x}_N(t)]^{\mathrm{T}} \in \mathbb{R}^{N\overline{n}}$ denote the aggregated state vector, $\overline{x}_{\mathrm{m}}(t) \triangleq [\overline{x}_{\mathrm{m}1}(t), \ldots, \overline{x}_{\mathrm{m}N}(t)]^{\mathrm{T}} \in \mathbb{R}^{N\overline{n}}$ denote the aggregated closed-loop reference model state vector, $\overline{e}(t) \triangleq [\overline{e}_1(t), \ldots, \overline{e}_N(t)]^{\mathrm{T}} \in \mathbb{R}^{N\overline{n}}$ denote the aggregated error vector, $\mu(t) \triangleq [\mu_1(t), \ldots, \mu_N(t)]^{\mathrm{T}} \in \mathbb{R}^{Np}$ denote the aggregated μ vector, and $\hat{\mu}(t) \triangleq [\hat{\mu}_1(t), \ldots, \hat{\mu}_N(t)]^{\mathrm{T}} \in \mathbb{R}^{Np}$ denote the aggregated $\hat{\mu}$ vector. Then, one can the overall system error dynamics as

$$\dot{\overline{e}}(t) = \mathcal{A}\overline{e}(t) + \mathcal{B}\tilde{\mu}(t), \tag{11}$$

where $\mathcal{A} \triangleq (\overline{A}_{\mathrm{m}} - \overline{L}) \otimes I_N \in \mathbb{R}^{N\overline{n}\times N\overline{n}}$, $\mathcal{B} \triangleq \overline{B}_{\mathrm{m}} \otimes I_N \in \mathbb{R}^{N\overline{n}\times Nm}$, and $\tilde{\mu}(t) \triangleq \hat{\mu}(t) - \mu(t) \in \mathbb{R}^N$. Note that $\mathcal{A}$ is Hurwitz and thus there exists a unique positive definite matrix $\mathcal{P} \in \mathbb{R}^{N\overline{n}\times N\overline{n}}$ solving the Lyapunov equation $\mathcal{A}^{\mathrm{T}}\mathcal{P} + \mathcal{P}\mathcal{A} + \mathcal{R} = 0$ for $\mathcal{R} \in \mathbb{R}_+^{N\overline{n}\times N\overline{n}}$.

Next, the aggregated μ-dynamics becomes

$$\begin{aligned} \dot{\mu}(t) =& -\gamma\left[(\mathcal{L} \otimes I_p)\,\hat{\mu}(t) + (\mathcal{K} \otimes I_p)\,(\hat{\mu}(t) - c(t))\right] \\ =& -\gamma\left[((\mathcal{L} + \mathcal{K}) \otimes I_p)\hat{\mu}(t) - (\mathcal{K} \otimes I_p)c(t)\right] \\ =& -\gamma\left[(\mathcal{F} \otimes I_p)\hat{\mu}(t) - (\mathcal{K} \otimes I_p)c(t)\right], \end{aligned} \tag{12}$$

where $\mathcal{L} \in \mathbb{R}^{N\times N}$ is the Laplacian matrix of the connected and undirected graph. Moreover, $\mathcal{F} \in \mathbb{R}_+^{N\times N}$ and $\mathcal{K} \in \mathbb{R}^{N\times N}$ are defined in Section II. Then, by letting $\overline{\mathcal{F}} \triangleq \mathcal{F} \otimes I_p \in \mathbb{R}^{Np\times Np}$ and $\overline{\mathcal{K}} \triangleq \mathcal{K} \otimes I_p \in \mathbb{R}^{Np\times Np}$, (12) can be identically rewritten as

$$\dot{\mu}(t) = -\gamma\overline{\mathcal{F}}\hat{\mu}(t) + \gamma\overline{\mathcal{K}}c(t). \tag{13}$$

By considering continuous inter-agent information exchange, one can define an auxiliary ideal μ-dynamics as

$$\dot{\mu}_{\mathrm{m}}(t) = -\gamma\overline{\mathcal{F}}\mu_{\mathrm{m}}(t) + \gamma\overline{\mathcal{K}}c(t), \tag{14}$$

where $\mu_{\mathrm{m}} \in \mathbb{R}^{Np\times Np}$. Next, we define the error between the dynamics given by (13) and (14) as $\overline{\mu}(t) \triangleq \mu(t) - \mu_{\mathrm{m}}(t) \in \mathbb{R}^{Np}$. Then, one can write

$$\dot{\overline{\mu}}(t) = -\gamma\overline{\mathcal{F}}\overline{\mu}(t) - \gamma\overline{\mathcal{F}}\tilde{\mu}(t). \tag{15}$$

Next, we present a lemma essential to stability analysis of the proposed distributed event-triggered control architecture from [22].

Lemma IV.1 [*Lemma 4.1, 22*] *The agent-wise event-triggering rule given by (8) can be equivalently rewritten as*

$$\|\tilde{\mu}(t)\|_2 \le \varepsilon\,\|\overline{e}(t)\|_2 + \sqrt{N}\phi(t) \tag{16}$$

for the overall networked multiagent system.

We now define

$$\mathcal{O} \triangleq \begin{bmatrix} \underline{\lambda}(\mathcal{R}) - 2\varepsilon\|\mathcal{P}\mathcal{B}\|_2 - \frac{1}{\overline{\sigma}_1} & 0 & 0 \\ 0 & \rho_1\gamma\left(\underline{\lambda}\left(\overline{\mathcal{F}}\right) - \overline{\rho}_1\right) & 0 \\ 0 & 0 & \overline{\rho}_2 \end{bmatrix}, \tag{17}$$

where $\overline{\sigma}_1 \in \mathbb{R}_+$, $\overline{\rho}_1 \in \mathbb{R}_+$, and $\overline{\rho}_2 \in \mathbb{R}_+$ are free parameters to ensure the positive definiteness of $\mathcal{O}$. In particular, $\overline{\rho}_2$ is unrestricted, $\overline{\rho}_1$ should be chosen such that $\underline{\lambda}(\overline{\mathcal{F}}) - \overline{\rho}_1 > 0$ holds, and the event-triggering parameter ε

should be chosen such that $\varepsilon < (\underline{\lambda}(\mathcal{R}) - \overline{\sigma}_1^{-1})/2\|\mathcal{PB}\|_2$. Note that this bound on ε can be made randomly large by carefully choosing α for $\overline{L}$ and judiciously large $\overline{\sigma}_1$.

Next, we present the system-theoretical stability analysis of the proposed distributed event-triggered control architecture.

Theorem IV.1 *Consider a networked multiagent system consisting of N agents. Consider also the dynamics of agent i for $i = 1, \ldots, N$ given by (1), the control signal given by (2), the μ-dynamics given by (3), and the closed-loop reference model given by (6). In addition, consider the event-triggering rule for scheduling inter-agent information exchange given by (8) and (9) subject to case $a)$ sampled data exchange and case $b)$ solution-predictor curve exchange. When $\varepsilon < (\underline{\lambda}(\mathcal{R}) - \overline{\sigma}_1^{-1})/2\|\mathcal{PB}\|_2$ the closed-loop solution $(\overline{e}(t), \overline{\mu}(t), \phi(t))$ approaches to zero for $\phi_{\mathrm{f}} = 0$ and it remains bounded for $\phi_{\mathrm{f}} \in \mathbb{R}_+$.*

Proof. The proof of this theorem follows similar steps to the proof of the main results presented in [22], but is not identical since [22] only considers single integrator agents and we here consider linear time-invariant models. Thus, we present the proof in details below for completeness.

To show the stability of the proposed distributed event-triggered control architecture, consider the following Lyapunov-like function candidate

$$\mathcal{V}(\overline{e}, \overline{\mu}, \phi) = \overline{e}^{\mathrm{T}}\mathcal{P}\overline{e} + \frac{1}{2}\rho_1\overline{\mu}^{\mathrm{T}}\overline{\mu} + \frac{1}{2}\rho_2\phi^2, \tag{18}$$

where $\rho_1 \in \mathbb{R}_+$ and $\rho_2 \in \mathbb{R}_+$ are free parameters. Observe that $\mathcal{V}(\overline{e}, \overline{\mu}, \phi) = 0$ for $(\overline{e}, \overline{\mu}, \phi) = (0, 0, 0)$ and $\mathcal{V}(\overline{e}, \overline{\mu}, \phi) > 0$ for $(\overline{e}, \overline{\mu}, \phi) \neq (0, 0, 0)$. Taking the time derivative of (18) along closed-loop system trajectories yields

$$\dot{\mathcal{V}}\left(\overline{e}(t), \overline{\mu}(t), \phi(t)\right) = 2\overline{e}^{\mathrm{T}}(t)\mathcal{P}\dot{\overline{e}}(t) + \rho_1\overline{\mu}(t)\dot{\overline{\mu}}(t) + \rho_2\phi(t)\dot{\phi}(t). \tag{19}$$

Using (9), (11), and (12) in (19) yields

$$\begin{aligned}\dot{\mathcal{V}}\left(\overline{e}(t), \overline{\mu}(t), \phi(t)\right) =& 2\overline{e}^{\mathrm{T}}\mathcal{P}\left(\mathcal{A}\overline{e}(t) + \mathcal{B}\tilde{\mu}(t)\right) + \rho_1\overline{\mu}^{\mathrm{T}}(t)\left(-\gamma\overline{\mathcal{F}}\overline{\mu}(t) - \gamma\overline{\mathcal{F}}\tilde{\mu}(t)\right) + \rho_2\phi(t)\left(-\kappa(\phi(t) - \phi_{\mathrm{f}})\right)\\
=& -\overline{e}^{\mathrm{T}}(t)\mathcal{R}\overline{e}(t) + 2\overline{e}^{\mathrm{T}}(t)\mathcal{PB}\tilde{\mu}(t) - \rho_1\gamma\overline{\mu}^{\mathrm{T}}(t)\overline{\mathcal{F}}\overline{\mu}(t) - \rho_1\gamma\overline{\mu}^{\mathrm{T}}(t)\overline{\mathcal{F}}\tilde{\mu}(t)\\
&- \rho_2\kappa\phi^2(t) + \rho_2\kappa\phi(t)\phi_{\mathrm{f}}\\
\leq& -\underline{\lambda}\left(\mathcal{R}\right)\|\overline{e}(t)\|_2^2 - \rho_1\gamma\underline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2^2 - \rho_2\kappa\phi^2(t) + 2\|\mathcal{PB}\|_2\|\overline{e}(t)\|_2\|\tilde{\mu}(t)\|_2\\
&+ \rho_1\gamma\overline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2\|\tilde{\mu}(t)\|_2 + \rho_2\kappa\phi(t)\phi_{\mathrm{f}}.\end{aligned} \tag{20}$$

Next, using the event-triggering rule given by (16), which follows from the agent-wise event-triggering rule given by (8) that holds for both case $a)$ sampled data exchange and case $b)$ solution-predictor curves exchange in (20), further yields

$$\begin{aligned}\dot{\mathcal{V}}\left(\overline{e}(t), \overline{\mu}(t), \phi(t)\right) \leq& -\underline{\lambda}\left(\mathcal{R}\right)\|\overline{e}(t)\|_2^2 - \rho_1\gamma\underline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2^2 - \rho_2\kappa\phi^2(t) + \rho_2\kappa\phi(t)\phi_{\mathrm{f}}\\
&+ 2\|\mathcal{PB}\|_2\|\overline{e}(t)\|_2\left(\varepsilon\|\overline{e}(t)\|_2 + \sqrt{N}\phi(t)\right) + \rho_1\gamma\overline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2\left(\varepsilon\|\overline{e}(t)\|_2 + \sqrt{N}\phi(t)\right)\\
=& -\underline{\lambda}\left(\mathcal{R}\right)\|\overline{e}(t)\|_2^2 - \rho_1\gamma\underline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2^2 - \rho_2\kappa\phi^2(t) + \rho_2\kappa\phi(t)\phi_{\mathrm{f}}\\
&+ 2\varepsilon\|\mathcal{PB}\|_2\|\overline{e}(t)\|_2^2 + 2\sqrt{N}\|\mathcal{PB}\|_2\|\overline{e}(t)\|_2\phi(t) + \rho_1\gamma\varepsilon\overline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2\|\overline{e}(t)\|_2\\
&+ \rho_1\gamma\sqrt{N}\overline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2\phi(t).\end{aligned} \tag{21}$$

Applying Young's inequality $ab \leq (1/2)(\sigma^{-1}a^2 + \sigma b^2)$ for $a \in \mathbb{R}$, $b \in \mathbb{R}$, and $\sigma \in \mathbb{R}_+$ on the sixth, seventh, and eight term of (21) it further yields

$$\begin{aligned}\dot{\mathcal{V}}\left(\overline{e}(t),\overline{\mu}(t),\phi(t)\right) \leq & -\left(\underline{\lambda}\left(\mathcal{R}\right)-2\varepsilon\|\mathcal{P}\mathcal{B}\|_2\right)\|\overline{e}(t)\|_2^2-\rho_1\gamma\underline{\lambda}\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2^2-\rho_2\kappa\phi^2(t)+\rho_2\kappa\phi(t)\phi_{\mathrm{f}} \\ & +\frac{1}{\sigma_1}\|\overline{e}(t)\|_2^2+\sigma_1 N\|\mathcal{P}\mathcal{B}\|_2^2\phi^2(t)+\frac{1}{2\sigma_2}\|\overline{e}(t)\|_2^2+\frac{\sigma_2}{2}\rho_1^2\gamma^2\varepsilon^2\overline{\lambda}^2\left(\overline{\mathcal{F}}\right)\|\overline{\mu}(t)\|_2^2 \\ & +\frac{\sigma_3}{2}\|\overline{\mu}(t)\|_2^2+\frac{1}{2\sigma_3}\rho_1^2\gamma^2 N\overline{\lambda}^2\left(\overline{\mathcal{F}}\right)\phi^2(t) \\ = & -\left(\underline{\lambda}\left(\mathcal{R}\right)-2\varepsilon\|\mathcal{P}\mathcal{B}\|_2-\frac{1}{\sigma_1}-\frac{1}{2\sigma_2}\right)\|\overline{e}(t)\|_2^2 \\ & -\left(\rho_1\gamma\underline{\lambda}\left(\overline{\mathcal{F}}\right)-\frac{\sigma_2}{2}\rho_1^2\gamma^2\varepsilon^2\overline{\lambda}^2\left(\mathcal{F}\right)-\frac{\sigma_3}{2}\right)\|\overline{\mu}(t)\|_2^2 \\ & -\left(\rho_2\kappa-\sigma_1 N\|\mathcal{P}\mathcal{B}\|_2^2-\frac{1}{2\sigma_3}\rho_1^2\gamma^2 N\overline{\lambda}^2\left(\overline{\mathcal{F}}\right)\right)\phi^2(t)+\rho_2\kappa\phi(t)\phi_{\mathrm{f}}.\end{aligned} \tag{22}$$

Letting $\sigma_1 = 2\overline{\sigma}_1$, $\sigma_2 = \overline{\sigma}_1$, $\sigma_3 = \sigma_2\rho_1^2\gamma^2\varepsilon^2\overline{\lambda}^2(\overline{F})$, $\rho_1 = \overline{\rho}_1(\sigma_2\gamma\varepsilon^2\overline{\lambda}^2(\overline{\mathcal{F}}))^{-1}$, and $\rho_2 = \kappa^{-1}(\overline{\rho}_2 + \sigma_2 N\|\mathcal{P}\mathcal{B}\|_2^2 + (2\sigma_3)^{-1}\rho_1^2\gamma^2 N\overline{\lambda}^2(\overline{\mathcal{F}}))$ one can rewrite (22) as

$$\dot{\mathcal{V}}\left(\overline{e}(t),\overline{\mu}(t),\phi(t)\right) \leq -\left(\underline{\lambda}\left(R\right)-2\varepsilon\|\mathcal{P}\mathcal{B}\|_2-\frac{1}{\overline{\sigma}_1}\right)\|\overline{e}(t)\|_2^2-\rho_1\gamma\left(\underline{\lambda}\left(\overline{\mathcal{F}}\right)-\overline{\rho}_1\right)\|\overline{\mu}(t)\|_2^2-\overline{\rho}_2\phi^2(t)+\rho_2\kappa\phi(t)\phi_{\mathrm{f}}. \tag{23}$$

Furthermore, (23) can be rewritten in compact form as

$$\dot{\mathcal{V}}\left(\overline{e}(t),\overline{\mu}(t),\phi(t)\right) \leq -\xi(t)^{\mathrm{T}}\mathcal{O}\xi(t)+\Gamma\xi(t), \tag{24}$$

where $\xi(t) \triangleq [\|\overline{e}(t)\|_2, \|\overline{\mu}(t)\|_2, \phi(t)]^{\mathrm{T}} \in \mathbb{R}^3$, $\Gamma \triangleq [0, 0, \rho_2\kappa\phi_{\mathrm{f}}] \in \mathbb{R}^{1\times 3}$, and $\mathcal{O}$ given by (17). An immediate consequence of (24) is that the closed-loop solution $(\overline{e}(t), \overline{\mu}(t), \phi(t))$ is Lyapunov stable and $\lim_{t\to\infty}(\overline{e}(t), \overline{\mu}(t), \phi(t)) \to (0, 0, 0)$ when $\phi_{\mathrm{f}} = 0$ since $\mathcal{O}$ is positive definite and Γ is zero in this case.

We next show the boundedness of the closed-loop solution $(\overline{e}(t), \overline{\mu}(t), \phi(t))$ when $\phi_{\mathrm{f}} \in (0, \phi_0)$. To this end, (24) can be upper bounded by

$$\begin{aligned}\dot{\mathcal{V}}\left(\overline{e}(t),\overline{\mu}(t),\phi(t)\right) \leq & -\underline{\lambda}\left(\mathcal{O}\right)\|\xi(t)\|_2^2+\|\Gamma\|_2\|\xi(t)\|_2 \\ = & -\|\xi(t)\|_2\left(\underline{\lambda}\left(\mathcal{O}\right)\|\xi(t)\|_2-\|\Gamma\|_2\right).\end{aligned} \tag{25}$$

It follows from (25) that there exists a compact set $C \triangleq \{\xi(t) : \|\xi(t)\|_2 \leq \|\Gamma\|_2/\underline{\lambda}(\mathcal{O})\}$ such that $\dot{\mathcal{V}}(\overline{e}(t), \overline{\mu}(t), \phi(t)) \leq 0$ outside of this set. Boundedness of the closed loop solution $(\overline{e}(t), \overline{\mu}(t), \phi(t))$ is now immediate. ■

V Numerical Example

To show the efficacy of the proposed distributed event-triggered control architecture for linear time-invariant multiagent systems predicated on the solution-predictor curve method, we provide an illustrative numerical example in this section. Consider a group of 4 agents whose inter-agent communication are encoded by the graph topology shown in Figure 1. We set the leader to agent 1 and let the dynamics of agent i for $i = 1, \ldots, 4$ be given by

$$\dot{x}_i(t) = \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}}_{A} x_i(t) + \underbrace{\begin{bmatrix} 0 \\ 2 \end{bmatrix}}_{B} u(t), \quad x_i(0) = x_{i0}, \tag{26a}$$

$$z_i(t) = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{E} x_i(t). \tag{26b}$$

Moreover, the initial conditions of agent i for $i = 1, \ldots, 4$ are chosen as

$$x_{10} = \begin{bmatrix} 0.8 & 0.0 \end{bmatrix}^{\mathrm{T}}, \; x_{20} = \begin{bmatrix} 0.5 & 0.0 \end{bmatrix}^{\mathrm{T}}, \; x_{30} = \begin{bmatrix} -0.5 & 0.0 \end{bmatrix}^{\mathrm{T}}, \; x_{40} = \begin{bmatrix} -0.8 & 0.0 \end{bmatrix}^{\mathrm{T}}, \tag{27}$$

We set the control feedback gains K_1 and K_2, respectively as $K_1 = [18.8, \; 17.8]^{\mathrm{T}}$ and $K_2 = 10.0$. In addition, we set the event-triggering parameters as $\varepsilon = 0.9$, $\phi_0 = 3$, $\phi_{\mathrm{f}} = 0$, $\kappa = 4/150$, and $\gamma = 1$.

We use Python 3.7.7 to preform the illustrative numerical study such that we run independent script for each agent (for this example we run 4 independent scripts) and required data as explained in Section III.B is shared between scripts only when an event occurs. We run the simulations for a total of 150 seconds at a sampling rate of $10\ 000$ Hz. Observe that if this numerical study was performed in the absence of the proposed event-triggered control, there would have been a total of $4 \cdot (150/10^{-4}) = 6 \cdot 10^6$ sampled data points exchanged.

First, we perform the numerical study by applying the proposed event-triggering architecture to schedule networked transmission data in the multiagent system such that neighboring agents share sampled data points through zero-order-hold operator, Figure 2. An immediate observation from Figure 2 is the reduction of events as compared to continuous communication (i.e., number of events decreased to 601 from $6 \cdot 10^6$).

Second, we perform the numerical study by applying the proposed event-triggering architecture to schedule networked transmission data in the multiagent system such that neighboring agents share solution-predictor curves, Figure 3. This method, in contrast to the widely adopted sampled data exchange in the event-triggering literature, has the potential to further reduce network utilization (number of events) as can be seen from Figure 3 (i.e., number of events decreased to 279 from 601).

In summary, with this illustrative numerical study, we show the efficacy of the proposed distributed event-triggered control algorithm for linear time-invariant multiagent systems to schedule local information exchange predicated on a solution-predictor curve. By applying this method, network utilization (number of events) has been dramatically reduced by more than 50% (i.e., from 601 to 279) as compared to the standard sampled data exchange approach; compare Figure 2 to Figure 3. Finally, from these figures, one can also observe that the solution-predictor method provides a smoother control signal.

VI Conclusion

To prevent potential network overload and decrease wireless communication costs, a distributed event-triggered control algorithm with a dynamic threshold was proposed for linear time-invariant multiagent systems by generalizing the results in [22]. Within the context of this architecture, for dramatically reducing network utilization, we utilized the solution-predictor curve exchange method. This method was shown the capability to decrease network utilization as compared to its standard sampled-data counterpart. We also presented rigorous stability analysis for our architecture and gave an illustrative numerical example to validate our theoretical contribution.

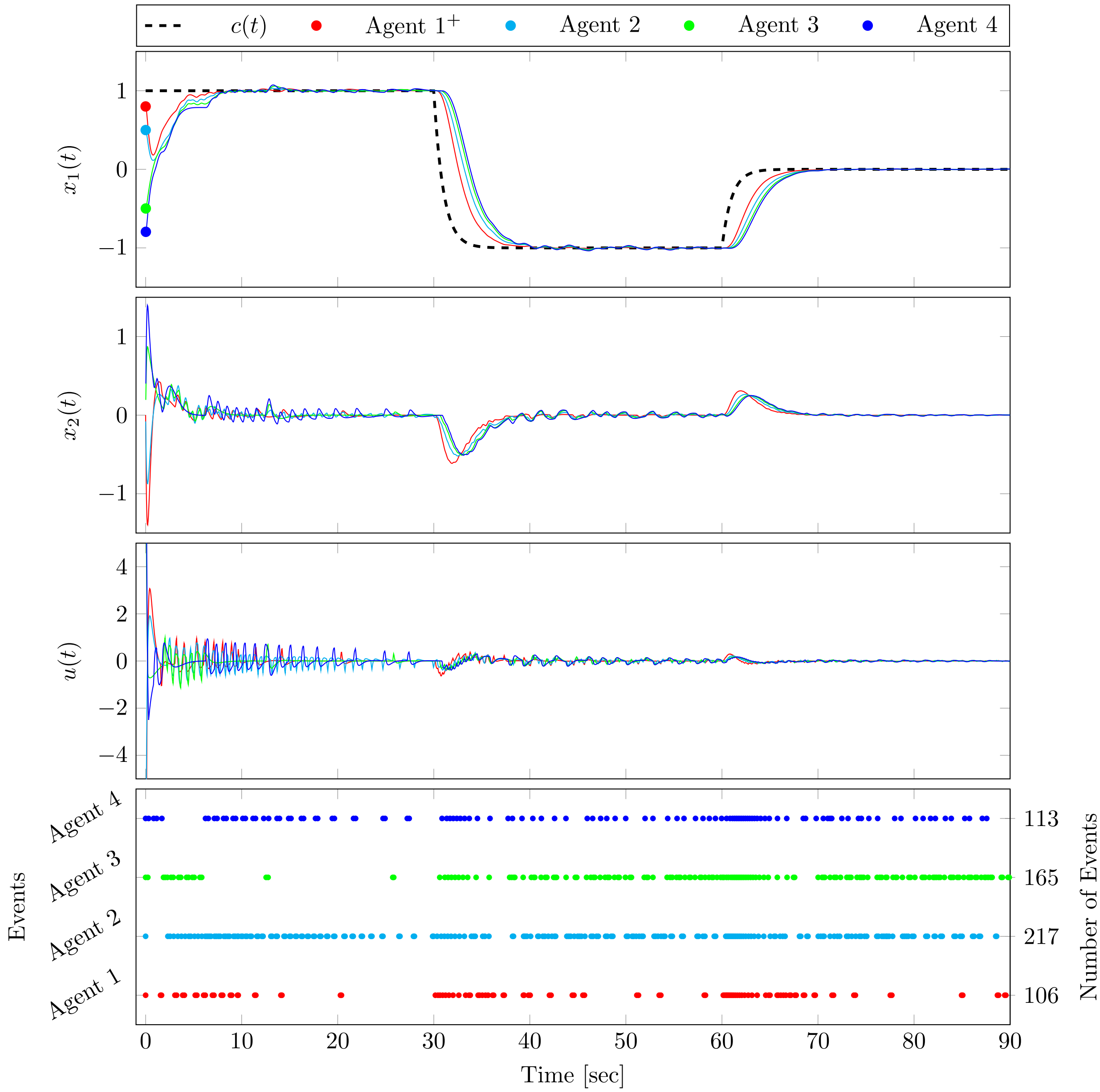

Figure 2. Sampled data exchange for event-triggering case a).

References

[1] J. Shamma, *Cooperative control of distributed multi-agent systems*. John Wiley & Sons, 2008.

[2] F. Bullo, J. Cortes, and S. Martinez, *Distributed control of robotic networks: A mathematical approach to motion coordination algorithms*. Princeton University Press, 2009, vol. 27.

[3] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*. Princeton University Press, 2010, vol. 33.

[4] F. L. Lewis, H. Zhang, K. Hengster-Movric, and A. Das, *Cooperative control of multi-agent systems: Optimal and adaptive design approaches*. Springer Science & Business Media, 2013.

[5] M. Mazo and P. Tabuada, "Decentralized event-triggered control over wireless sensor/actuator networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 10, pp. 2456–2461, 2011.

[6] ——, "On event-triggered and self-triggered control over sensor/actuator networks," in *Conference on Decision and Control*. IEEE, 2008, pp. 435–440.

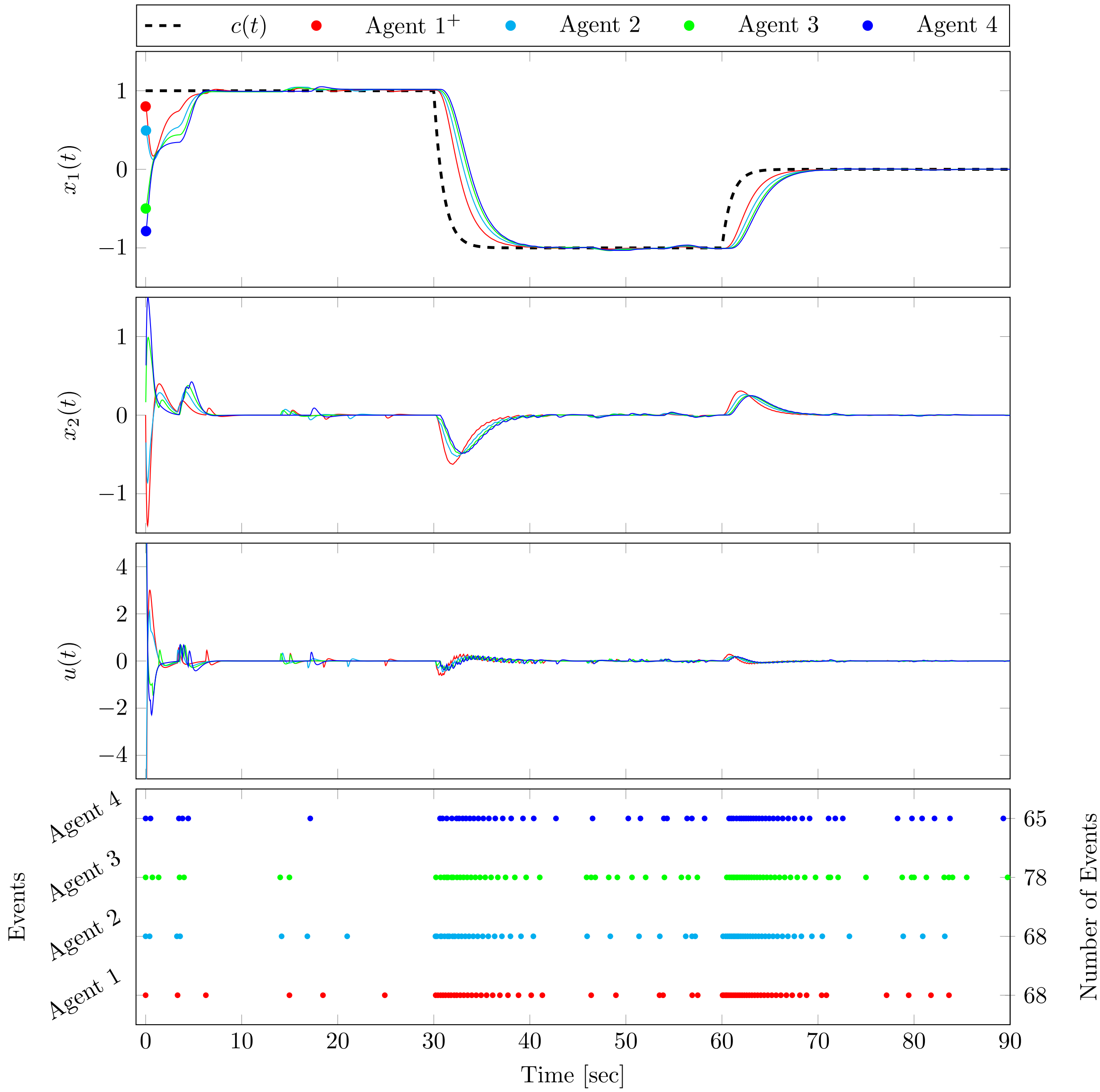

Figure 3. Local solution-predictor curves exchange.

[7]A. Anta and P. Tabuada, "Self-triggered stabilization of homogeneous control systems," in *American Control Conference*. IEEE, 2008, pp. 4129–4134.

[8]Y. Shoukry and P. Tabuada, "Event-triggered state observers for sparse sensor noise/attacks," *IEEE Transactions on Automatic Control*, vol. 61, no. 8, pp. 2079–2091, 2015.

[9]Q. Liu, Z. Wang, X. He, and D. Zhou, "A survey of event-based strategies on control and estimation," *Systems Science & Control Engineering: An Open Access Journal*, vol. 2, no. 1, pp. 90–97, 2014.

[10]S. Ristevski, K. M. Dogan, T. Yucelen, and J. A. Muse, "A new event-triggering approach for scheduling guidance data transmissions in networked control systems," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 4411–4416.

[11]W. Heemels, K. H. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 3270–3285.

[12]X. Yi, K. Liu, D. V. Dimarogonas, and K. H. Johansson, "Dynamic event-triggered and self-triggered control for multi-agent systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 8, pp. 3300–3307, 2018.

[13]D. Yang, W. Ren, X. Liu, and W. Chen, "Decentralized event-triggered consensus for linear multi-agent systems under general

directed graphs," *Automatica*, vol. 69, pp. 242–249, 2016.

[14]G. Guo, L. Ding, and Q.-L. Han, "A distributed event-triggered transmission strategy for sampled-data consensus of multi-agent systems," *Automatica*, vol. 50, no. 5, pp. 1489–1496, 2014.

[15]X. Zhou, P. Shi, C.-C. Lim, C. Yang, and W. Gui, "Event based guaranteed cost consensus for distributed multi-agent systems," *Journal of the Franklin Institute*, vol. 352, no. 9, pp. 3546–3563, 2015.

[16]A. Wang, B. Mu, and Y. Shi, "Consensus control for a multi-agent system with integral-type event-triggering condition and asynchronous periodic detection," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 7, pp. 5629–5639, 2017.

[17]Y. Zou, X. Su, S. Li, Y. Niu, and D. Li, "Event-triggered distributed predictive control for asynchronous coordination of multi-agent systems," *Automatica*, vol. 99, pp. 92–98, 2019.

[18]B. Zhou, X. Liao, T. Huang, and G. Chen, "Leader-following exponential consensus of general linear multi-agent systems via event-triggered control with combinational measurements," *Applied Mathematics Letters*, vol. 40, pp. 35–39, 2015.

[19]E. Garcia, Y. Cao, H. Yu, P. Antsaklis, and D. Casbeer, "Decentralised event-triggered cooperative control with limited communication," *International Journal of Control*, vol. 86, no. 9, pp. 1479–1488, 2013.

[20]X. Chen, F. Hao, and A. Rahmani, "Asynchronous decentralised event-triggered control of multi-agent systems," *International Journal of Control*, vol. 87, no. 10, pp. 2130–2139, 2014.

[21]W. H. Heemels, M. Donkers, and A. R. Teel, "Periodic event-triggered control for linear systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 847–861, 2012.

[22]S. Ristevski, T. Yucelen, and J. A. Muse, "An event-triggered distributed control architecture for scheduling information exchange in networked multiagent systems," *IEEE Transactions on Control Systems Technology*, (Under review).

[23]——, "Distributed event-triggered control with local solution-predictor curves for linear multiagent systems," in *IEEE Conference on Decision and Control (CDC)*, (Submitted).

[24]C. Godsil and G. F. Royle, *Algebraic graph theory*. Springer Science & Business Media, 2013, vol. 207.

[25]T. Yucelen, G. De La Torre, and E. N. Johnson, "Improving transient performance of adaptive control architectures using frequency-limited system error dynamics," *International Journal of Control*, vol. 87, no. 11, pp. 2383–2397, 2014.

[26]V. Stepanyan and K. Krishnakumar, "Mrac revisited: guaranteed performance with reference model modification," in *Proceedings of the 2010 American Control Conference*. IEEE, 2010, pp. 93–98.

[27]——, "Adaptive control with reference model modification," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 4, pp. 1370–1374, 2012.

[28]——, "M-mrac for nonlinear systems with bounded disturbances," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 5419–5424.

[29]E. Lavretsky, "Reference dynamics modification in adaptive controllers for improved transient performance," in *AIAA guidance, navigation, and control conference*, 2011, p. 6200.

[30]T. E. Gibson, A. M. Annaswamy, and E. Lavretsky, "Adaptive systems with closed-loop reference models: Stability, robustness and transient performance," *arXiv preprint arXiv:1201.4897*, 2012.