

Event-Triggering in Distributed Networked Control Systems

Xiaofeng Wang, *Member, IEEE*, and Michael D. Lemmon, *Member, IEEE*

Abstract—This paper examines event-triggered data transmission in distributed networked control systems with packet loss and transmission delays. We propose a distributed event-triggering scheme, where a subsystem broadcasts its state information to its neighbors only when the subsystem’s local state error exceeds a specified threshold. In this scheme, a subsystem is able to make broadcast decisions using its locally sampled data. It can also locally predict the maximal allowable number of successive data dropouts (MANSD) and the state-based deadlines for transmission delays. Moreover, the designer’s selection of the local event for a subsystem only requires information on that individual subsystem. Our analysis applies to both linear and nonlinear subsystems. Designing local events for a nonlinear subsystem requires us to find a controller that ensures that subsystem to be input-to-state stable. For linear subsystems, the design problem becomes a linear matrix inequality feasibility problem. With the assumption that the number of each subsystem’s successive data dropouts is less than its MANSD, we show that if the transmission delays are zero, the resulting system is finite-gain \mathcal{L}_p stable. If the delays are bounded by given deadlines, the system is asymptotically stable. We also show that those state-based deadlines for transmission delays are always greater than a positive constant.

Index Terms—Event-Triggering; Networked Control Systems; Distributed Systems

I. INTRODUCTION

A distributed networked control system (NCS) consists of numerous coupled subsystems (also called “agents”), which are geographically distributed. In such a system, individual subsystems exchange information over a communication network. These networked systems are found throughout our national infrastructure with specific examples being the electrical power grid and transportation networks. Networking not only refers to the communication infrastructure supporting feedback control, but also refers to the fact that individual subsystems are physically interconnected in a way that can be modelled as a network. The networking of control effort can be advantageous in terms of lower system costs due to streamlined installation and maintenance costs.

The introduction of a communication network, however, raises important issues regarding the impact that such communication has on the control system’s performance. In practice, communication, especially wireless communication, takes place over digital networks where the data is transmitted in

discrete packets. These packets may be lost during communication. Moreover, the communication media is a resource that is usually accessed in a mutually exclusive manner by neighborhood agents. This means that the throughput capacity of such networks is limited. So one important issue in the implementation of such systems is to identify methods that more effectively use the limited network bandwidth available for transmitting state information.

For this reason, some researchers began investigating the timing issue in NCS. In other words, how frequently should subsystems communicate to ensure that the NCS has a desired level of performance? In traditional approaches, one first designs the controllers under the assumption of perfect communication and then determines the *maximum allowable transfer interval* (MATI) between two subsequent message transmissions that ensure closed-loop stability under a network protocol, such as Try-Once-Discard (TOD) or Round-Robin (RR) protocol.

The computation of the MATI, however, is often done in a highly centralized manner. This is impractical for large-scale systems. Moreover, because the MATI is computed before the system is deployed, it must ensure performance levels over all possible system states. As a result, the MATI may be conservative in the sense of being shorter than necessary to assure a specified performance level. Consequently, the bandwidth of the network has to be higher than necessary to ensure the MATI is not violated.

This paper addresses the timing issue through the use of a distributed event-triggered feedback scheme in NCS, where packet loss and transmission delays are allowed. Event-triggering has an agent broadcast its state information only when “needed”. In this case, “needed” means that some measure of the agent’s state error is above a specified threshold. Our scheme is decentralized in the sense that an agent is able to make broadcast decisions using its locally sampled data. An agent can also locally predict the *maximal allowable number of successive data dropouts* (MANSD) as well as the state-based bounds for transmission delays (also called “deadlines”). Such information may be used to help schedule an agent’s access to the communication network. Moreover, the selection of the event-triggering threshold only requires local information about that agent, so that the design is decentralized.

Our analysis applies to both linear and nonlinear subsystems. Designing “local” events for a nonlinear subsystem requires us to find a controller that ensures the subsystem is input-to-state stable. By “local”, it means the event only depends on that subsystem’s local state and error. For linear subsystems, the design problem becomes a linear matrix

Xiaofeng Wang is with the Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801; e-mail: wangx@illinois.edu; Michael Lemmon is with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556; e-mail: lemmon@nd.edu. The authors gratefully acknowledge the partial financial support of the National Science Foundation (grants NSF-ECCS-0925229 and NSF-CNS-0720457)

inequality (LMI) feasibility problem. With the assumption that the transmission delays are zero and the number of each agent's successive data dropouts is less than its MANSF, we show that the resulting NCS is finite-gain \mathcal{L}_p stable. When the transmission delays are not zero, we provide state-based deadlines for those delays, which are always greater than a positive constant. As long as the delay in each transmission is less than the associated deadline, we show that the resulting NCS is asymptotically stable. Simulation results show that the average broadcast period generated by our scheme scales well with respect to the number of agents. In addition to this, simulation results suggest that the computational time required to select event thresholds also scales well with respect to system size. These results can serve as the basis for the design of firm real-time systems that guarantee system performance at levels traditionally seen in hard real-time systems.

The paper is organized as follows. Prior work is discussed in section II. Section III formulates the problem. Distributed approaches to design the local triggering event are introduced in section IV for both nonlinear and linear systems. Data dropouts and transmission delays are considered in section V. Simulation results are presented in section VI. Section VII draws the conclusions.

II. PRIOR WORK

To the best of our knowledge, there is little prior work on the distributed implementation of event-triggering in NCS. Preliminary results [1], [2] proposed decentralized event-triggered feedback schemes for linear and nonlinear systems, respectively. This work studied the asymptotic stability of NCS without considering packet loss and transmission delays. An event-triggering scheme was introduced for sensor-actuator networks [3]. This work, however, adopted a *centralized* approach to event-design. Scheduling of event-triggered controllers over networks was studied in [4], where different MAC protocols were compared in simulations. A recent study [5] applied distributed event-triggering in Network Utility Maximization (NUM) problem. Other than these papers, we are aware of no other work formally analyzing distributed implementations of event-triggering in NCS. There is, however, a great deal of related work dealing with event-triggered feedback in embedded control systems, sample period selection, and packet loss in NCS. We will review these areas and discuss their relationship to our distributed event-triggering scheme.

Early work [6], [7] analyzed the scheduling of real-time network traffic. The impact of communication constraints on system performance, however, was not addressed in this work. It was noticed [8], [9] that communication delay had a harmful effect on system stability. These papers considered the one packet transmission problem, in which all of the system outputs were packaged into a single packet. Agents in the network, therefore, do not have to compete for channel access. One packet transmission strategies, however, require a supervisor who gathers the data from all subsystems into a single packet. The cost and complexity of implementing such centralized supervisors will not scale well with system size. As a result, such schemes may be impractical for large-scale systems with limited network bandwidth.

Asynchronous transmissions were considered in [10]. In this work, several sensors and actuators attempt to access the communication channel at the same time, but only one of them actually gains access. Which agent gains access depends on the media access control (MAC) protocol being used. Commonly used MAC protocols include Try-Once-Discard (TOD) and Round-Robin (RR) [10]. For these protocols, an upper bound on the MATI was derived [10] that guarantees asymptotic stability of the system. It led to scheduling methods [11] that were able to assure the MATI was not violated. Further work [12], [13] derived tighter bounds on the MATI. All of this prior work confined its attention to control area network (CAN) buses where centralized computers coordinate the information flow across the network. The length of the MATI heavily relied on the choice of network protocols.

The aforementioned work computed bounds on the MATI in a centralized manner. This earlier work also assumed that MAC protocols were also realized using a central supervisor. A centralized approach in analysis and implementation is impractical for large-scale systems. Moreover, because the MATI is computed before the system is deployed, the selected MATI must ensure performance levels over a wide range of possible system states. As a result, the MATI may be conservative in the sense of being shorter than necessary to assure a specific performance level. Consequently, the network bandwidth has to be higher than necessary to ensure the MATI is not violated.

One approach for reducing the bandwidth requirements within NCS is to reduce the frequency with which agents communicate. Recent work considering event-triggered feedback sampled-data systems [14], [15] shows that the sampling rates under event-triggering are well below those in periodic task models. This is because the system can adaptively adjust the rates in a manner that is sensitive to what is currently happening within the system. It should therefore be possible to reduce the transmission frequency in NCS using event-triggering. Event-triggering [16] has appeared under a number of names that include interrupt-based feedback [17], Lebesgue sampling [18], state-triggered feedback [19], and self-triggered feedback [20], [15], [21]. All of this prior work, however, focused on using event-triggered feedback in single processor real-time systems.

Another related research direction is to study packet loss in NCS. In [22], a 2-state Markov model was used to describe the packet loss. The system can either use past control inputs or compute new control inputs based on an estimate of the lost data. In [23], packet loss is modelled as an identically independently distributed (i.i.d.) process in a single-input single-output NCS. These results were extended in [24] by modelling data dropouts as a Markov chain instead of an i.i.d. process. Optimal dropout compensation for NCS was presented in [25]. A packet-based multi-control strategy was examined in [26] to improve the performance of NCS, where packet loss is assumed to follow a stochastic 2-state Markov model. All of this work focused on modelling data dropouts as stochastic processes in a centralized manner. This paper, on the other hand, presents a method by which agents can locally estimate their MANSF. Again, this information may be used

in scheduling agent access to the communication medium.

This paper considers a more general framework of asynchronous information transmission in NCS using distributed event-triggering. This work addresses the impact that both data dropouts and transmission delays have on overall system performance. To the best of our knowledge, this is the first result examining the requirements for distributed implementation of NCS. It is also the first result on packet loss in event-triggering. Another important contribution over prior work [10], [12], [13] is that this paper derives state-based bounds on stabilizing transmission delays. Furthermore, we show that the existence of strictly positive bounds on stabilizing delays. These bounds can be used to select realistic deadlines that can be achieved by communication network middleware. Our results can therefore serve as the basis for the design of firm real-time systems that guarantee system performance at levels traditionally seen in hard real-time systems.

III. PROBLEM FORMULATION

Consider a distributed NCS containing N agents. These N agents are coupled together and each agent receives information from neighboring agents. Let $\mathcal{N} = \{1, 2, \dots, N\}$. The coupling and information flow in the NCS can be described by a coupling graph and a communication graph, which are defined as follows.

Definition 3.1: A graph $\mathcal{G}_{\text{cp}} = (\mathcal{N}, \mathcal{E}_{\text{cp}})$ is called the coupling graph of a NCS, where each node $i \in \mathcal{N}$ represents an agent in the NCS. The ordered pair (edge) (i, j) is in \mathcal{E}_{cp} if agent j is directly driven by agent i .

Definition 3.2: A graph $\mathcal{G}_{\text{cm}} = (\mathcal{N}, \mathcal{E}_{\text{cm}})$ is called the communication graph of a NCS, where each node $i \in \mathcal{N}$ represents an agent in the NCS. The ordered pair (edge) (i, j) is in \mathcal{E}_{cm} if agent j can receive broadcasts from agent i .

In this paper, the coupling and communication graphs need not be the same. Both graphs are directed. This provides us a general framework for the network topology. For notational convenience, we let

- $Z_i \triangleq \{j \in \mathcal{N} \mid (j, i) \in \mathcal{E}_{\text{cm}}\}$ denote the set of agents that agent i can get information from,
- $U_i \triangleq \{j \in \mathcal{N} \mid (i, j) \in \mathcal{E}_{\text{cm}}\}$ denote the set of agents that can receive agent i 's information,
- $D_i \triangleq \{j \in \mathcal{N} \mid (j, i) \in \mathcal{E}_{\text{cp}}\}$ denote the set of agents that directly drive agent i , and
- $S_i \triangleq \{j \in \mathcal{N} \mid (i, j) \in \mathcal{E}_{\text{cp}}\}$ denote the set of agents who are directly driven by agent i .

Notice that $i \notin Z_i \cup D_i \cup U_i \cup S_i$. We let $\bar{\Sigma}_i = \Sigma_i \cup \{i\}$ for any set $\Sigma_i \in \{Z_i, U_i, D_i, S_i\}$. For any set $\Sigma \subseteq \mathcal{N}$, $|\Sigma|$ denotes the number of the elements in Σ . $\|\cdot\|_2$ denotes the Euclidean 2-norm of a vector, $\|\cdot\|$ denotes the matrix norm induced by the Euclidean vector norm, and $\lambda_{\min}(A)$, $\lambda_{\max}(A)$ denote the minimal and maximal eigenvalues of matrix A , respectively.

The state equation of agent i is

$$\begin{aligned} \dot{x}_i(t) &= f_i(x_{\bar{D}_i}(t), u_i(t), w_i(t)) \\ u_i(t) &= g_i(x_{\bar{Z}_i}(t)), \quad x_i(t_0) = x_{i0} \end{aligned}$$

where $x_i : \mathbb{R} \rightarrow \mathbb{R}^n$ is the state trajectory of agent i , $u_i : \mathbb{R} \rightarrow \mathbb{R}^m$ is a control input, $w_i : \mathbb{R} \rightarrow \mathbb{R}^l$ is an

exogenous disturbance in \mathcal{L}_p space, $g_i : \mathbb{R}^{n|\bar{Z}_i|} \rightarrow \mathbb{R}^m$ is the feedback strategy of agent i satisfying $g_i(0) = 0$, $f_i : \mathbb{R}^{n|\bar{D}_i|} \times \mathbb{R}^m \times \mathbb{R}^l \rightarrow \mathbb{R}^n$ is a locally Lipschitz function satisfying $f_i(0, 0, 0) = 0$, and $x_{\bar{D}_i} = \{x_j\}_{j \in \bar{D}_i}$, $x_{\bar{Z}_i} = \{x_j\}_{j \in \bar{Z}_i}$. For convenience, this paper assumes that the states, inputs, and disturbances of agents have the same dimension. The results in this paper can be easily extended to cases where the dimensions of agents' states, inputs, and disturbances are different from each other.

This paper assumes agent i can only detect its own state, x_i , and receive the broadcast states of its neighbors in Z_i . If a local "error" signal exceeds a given threshold, which can be detected by hardware detectors, agent i will sample and broadcast its state information to all agents in the set U_i over a real-time network. Meanwhile, agent i 's control, u_i , at time t is computed based on the latest states that were successfully broadcast by those agents in \bar{Z}_i . These broadcast states are denoted as $\hat{x}_{\bar{Z}_i}(t)$. The control signal used by agent i is computed based on $\hat{x}_{\bar{Z}_i}(t)$. This means that the state equation of agent i can be written as

$$\begin{aligned} \dot{x}_i(t) &= f_i(x_{\bar{D}_i}(t), u_i(t), w_i(t)) \\ u_i(t) &= g_i(\hat{x}_{\bar{Z}_i}(t)), \quad x_i(t_0) = x_{i0}, \quad \forall t \geq t_0. \end{aligned} \quad (1)$$

One thing worth mentioning is that the control input $u_i(t)$ depends on $\hat{x}_i(t)$ instead of $x_i(t)$. In fact, if agent i knows its exact state $x_i(t)$, the control input $u_i(t)$ can be computed based on $x_i(t)$ and $\hat{x}_{Z_i}(t)$. The analysis in this paper is still applicable. But in this case, agent i has to continuously compute u_i . From the perspective of saving computational resource, we relate $u_i(t)$ to $\hat{x}_i(t)$. u_i is, therefore, constant between two consecutive successful broadcasts by the agents in \bar{Z}_i , because $\hat{x}_{\bar{Z}_i}(t)$ remains the same during that time period. Of course, agent i can also update its control input between broadcasts, subject to its local computational resource available. But this requires a more sophisticated strategy to manage the computational resource that will be studied in the future.

Agent i 's broadcast is characterized by three monotone increasing sequences of time instants: the broadcast release time, $\{r_j^i\}_{j=1}^\infty$, the successful broadcast release, $\{b_k^i\}_{k=1}^\infty$, and the broadcast finishing time $\{f_k^i\}_{k=1}^\infty$. The time r_k^i denotes the time instant when the k th broadcast of agent i is released. The data, however, may not be transmitted successfully. The time b_k^i denotes the time instant when the i th agent "successfully" broadcasts its state for the k th time. "Successfully" means the data in this broadcast is received by ALL agents in U_i successfully. The sequence, $\{b_k^i\}_{k=1}^\infty$, is clearly a subsequence of sequence $\{r_j^i\}_{j=1}^\infty$. The j th inter-release time of agent i is defined as $T_j^i = r_{j+1}^i - r_j^i$. It is also known as the j th broadcast period of agent i . We use d_k^i to denote the number of times agent i broadcasts its state between time instants b_k^i and b_{k+1}^i . This is also equal to the number of data dropouts between agent i and U_i over the interval $[b_k^i, b_{k+1}^i]$. We assume there is no delay between sampling and broadcasting data. The delay considered in this paper is the time between sampling and "successful" receipt of the data. The time f_k^i denotes the time instant when all of agent i 's neighbors (agents in U_i) receive

the k th successfully broadcasted data from agent i . Notice that $\hat{x}_i(t) = x_i(b_k^i)$ for all $t \in [f_k^i, f_{k+1}^i)$. In the following we let $e_i : [t_0, \infty) \rightarrow \mathbb{R}^n$ be defined as $e_i(t) \triangleq x_i(t) - \hat{x}_i(t)$ for $\forall t \geq t_0$. This function therefore represents the error between agent i 's current state and the state at its last successful broadcast. We also define the function $\varepsilon_i^j : [t_0, \infty) \rightarrow \mathbb{R}^n$ as $\varepsilon_i^j(t) \triangleq x_i(t) - x_i(r_j^i)$. This signal therefore represents the error between agent i 's current state and its state at r_j^i , whether or not the broadcast at r_j^i was successful.

Definition 3.3: The system in equation (1) is said to be finite-gain \mathcal{L}_p stable from w to x with an induced gain less than γ if there exist non-negative constants γ and ξ such that for $\forall w \in \mathcal{L}_p$

$$\left(\int_{t_0}^{\infty} \|x(t)\|_2^p dt \right)^{\frac{1}{p}} \leq \gamma \left(\int_{t_0}^{\infty} \|w(t)\|_2^p dt \right)^{\frac{1}{p}} + \xi. \quad (2)$$

The objective of this paper is to develop distributed event-triggering schemes to identify $\{r_j^i\}_{j=1}^{\infty}$, $\{b_k^i\}_{k=1}^{\infty}$, and $\{f_k^i\}_{k=1}^{\infty}$ such that the NCS defined in equation (1) achieves a desired level of performance with regard to the stability concepts introduced above.

IV. DISTRIBUTED BROADCAST-TRIGGERING EVENT DESIGN

This section describes a distributed approach to the design of "local" events for both nonlinear and linear subsystems. By "local", we mean that a subsystem's event can be triggered using information that is "locally" available to the agent. This information includes the agent's local state as well as local models of the agent's dynamics.

A. Local Event Design in Nonlinear Systems

The following assumption will be used throughout this paper.

Assumption 4.1: Consider subsystem $i \in \mathcal{N}$ in equation (1), i.e.

$$\begin{aligned} \dot{x}_i(t) &= f_i(x_{\bar{D}_i}(t), u_i(t), w_i(t)) \\ u_i(t) &= g_i(\hat{x}_{\bar{Z}_i}(t)), \quad x_i(t_0) = x_{i0}, \quad \forall t \geq t_0. \end{aligned} \quad (3)$$

Assume that for any $i \in \mathcal{N}$, there exist a continuously differentiable, positive-definite function $V_i : \mathbb{R}^n \rightarrow \mathbb{R}$, class \mathcal{K} functions $\zeta_1^i, \zeta_2^i : \mathbb{R} \rightarrow \mathbb{R}$, positive constants $\alpha_i, \beta_i, \delta_i, \gamma_i \in \mathbb{R}$, and control law $g_i : \mathbb{R}^{n|\bar{Z}_i|} \rightarrow \mathbb{R}^{m_i}$ satisfying

$$\zeta_1^i(\|x_i\|_2) \leq V_i(x_i) \leq \zeta_2^i(\|x_i\|_2) \quad (4)$$

$$\frac{\partial V_i}{\partial x_i}(x_i) f_i(x_{\bar{D}_i}, g_i(x_{\bar{Z}_i} - e_{\bar{Z}_i}), w_i) \leq \quad (5)$$

$$\sum_{j \in D_i \cup Z_i} \beta_j \|x_j\|_2^p + \gamma_i^p \|w_i\|_2^p + \sum_{j \in \bar{Z}_i} \delta_j \|e_j\|_2^p - \alpha_i \|x_i\|_2^p$$

$$\alpha_i - |S_i \cup U_i| \beta_i > 0 \quad (6)$$

with some $p \geq 1$.

Remark 4.2: Equation (5) suggests that subsystem i is \mathcal{L}_p stable from $\{x_j\}_{j \in D_i \cup Z_i}$, $\{e_j\}_{j \in \bar{Z}_i}$, and w_i to x_i . Equation (6) requires α_i to be large. Taking this into equation (5), it puts a requirement on the \mathcal{L}_p gain of this subsystem. This

assumption can be satisfied when the interconnections between subsystems are weak. Here is a distributed way to design the control strategy g_i such that Assumption 4.1 holds. We first select $\{\delta_i\}_{i=1}^N$ and $\{\beta_i\}_{i=1}^N$ that are shared by all agents. Then agent i only needs to find V_i , α_i , and g_i to fulfill equations (4) – (6). Notice that once agent i knows $\{\delta_i\}_{i=1}^N$ and $\{\beta_i\}_{i=1}^N$, equations (4) – (6) are only associated with agent i 's local dynamics. It turns to be a local input-to-state stabilization problem for agent i . Solving such a stabilization problem is not easy in general. For linear systems, however, it can be solved by solving an LMI. This will be further discussed in Section IV-B.

Theorem 4.3: Consider the NCS defined in equation (1). Suppose Assumption 4.1 holds. For all $i \in \mathcal{N}$ and any $\rho_i \in (0, 1)$, let

$$\sigma_i = \left(\frac{|\bar{U}_i| \delta_i}{\alpha_i - |S_i \cup U_i| \beta_i} \right)^{\frac{1}{p}}. \quad (7)$$

If

$$-\rho_i \|x_i(t)\|_2 + \sigma_i \|e_i(t)\|_2 \leq 0 \quad (8)$$

holds for all $i \in \mathcal{N}$ and all $t \geq t_0$, then the NCS is \mathcal{L}_p stable from w to x .

Proof: Let $V(x) = \sum_{i \in \mathcal{N}} V_i(x_i)$. It is easy to see that

$$\begin{aligned} \dot{V} &= \sum_{i \in \mathcal{N}} \frac{\partial V_i}{\partial x_i}(x_i) f_i(x_{\bar{D}_i}, g_i(x_{\bar{Z}_i} - e_{\bar{Z}_i}), w_i) \\ &\leq \sum_{i \in \mathcal{N}} \left[-\alpha_i \|x_i\|_2^p + \sum_{j \in D_i \cup Z_i} \beta_j \|x_j\|_2^p \right] \\ &\quad + \sum_{i \in \mathcal{N}} \left[\sum_{j \in \bar{Z}_i} \delta_j \|e_j\|_2^p + \gamma_i^p \|w_i\|_2^p \right] \\ &= \sum_{i \in \mathcal{N}} \left[-(\alpha_i - |S_i \cup U_i| \beta_i) \|x_i\|_2^p + \delta_i |\bar{U}_i| \|e_i\|_2^p \right] \\ &\quad + \sum_{i \in \mathcal{N}} \gamma_i^p \|w_i\|_2^p \\ &\leq \sum_{i \in \mathcal{N}} \left[-(1 - \rho_i^p) (\alpha_i - |S_i \cup U_i| \beta_i) \|x_i\|_2^p + \gamma_i^p \|w_i\|_2^p \right], \end{aligned}$$

where the second equality is obtained by resorting all of the items according to index i . Equation (6) and the above inequality are sufficient to show the NCS is \mathcal{L}_p stable from w to x . ■

Remark 4.4: The inequality in (8) is the real-time constraint to ensure the system stability. This constraint is only associated with the agent's local state, x_i , and the local error, e_i , between the agent's current state and its last successfully broadcast state. Note that these signals are locally available to agent i . If there are no packet loss and transmission delays, we can directly use the violation of the inequality in (8) to trigger agent i 's broadcasts. Even if transmission delays and packet loss exist, as long as the the resulting measure error is bounded by the given threshold, the system is still \mathcal{L}_p stable. How to quantify the allowable delays and packet loss will be discussed in section V.

Remark 4.5: The functions $\alpha_i \|x_i\|_2^p$, $\beta_j \|x_j\|_2^p$, and $\delta_j \|e_j\|_2^p$ may be replaced by more general class \mathcal{K} functions that are Lipschitz continuous.

We will find it convenient to use a slightly stronger sufficient condition for \mathcal{L}_p stability. This condition is stated in the following corollary. The corollary recasts the real-time constraint in equation (8) as a function of the local error, e_i , and the successfully broadcast local state, \hat{x}_i .

Corollary 4.6: Consider the NCS in (1). Suppose Assumption 4.1 holds. For any $i \in \mathcal{N}$ and $\rho_i \in (0, 1)$, let $c_i = 1 + \sigma_i$ where σ_i is given by equation 7. If

$$c_i \|e_i(t)\|_2 \leq \rho_i \|\hat{x}_i(t)\|_2 \quad (9)$$

holds for all $i \in \mathcal{N}$ and all $t \geq t_0$, then the NCS is \mathcal{L}_p stable from w to x .

Proof: By the definition of c_i , equation (9) is equivalent to $\sigma_i \|e_i(t)\|_2 + \|e_i(t)\|_2 \leq \rho_i \|\hat{x}_i(t)\|_2$ for all $t \geq t_0$. So we have

$$\begin{aligned} \sigma_i \|e_i(t)\|_2 &\leq \rho_i \|\hat{x}_i(t)\|_2 - \rho_i \|e_i(t)\|_2 \\ &\leq \rho_i \|\hat{x}_i(t) + e_i(t)\|_2 = \rho_i \|x_i(t)\|_2 \end{aligned}$$

for all $t \geq t_0$. Since the hypotheses of Theorem 4.3 are satisfied, we can conclude that the NCS is \mathcal{L}_p stable from w to x . ■

Remark 4.7: In the later discussion, we design event trigger to ensure the inequality in (9). In equation (9), the threshold on the local error is fixed between two successive transmissions. Such a threshold makes it more convenient to predict deadlines. These predictions are discussed in section V.

B. Linear Systems

This section shows how to implement the distributed scheme proposed in section IV-A for linear systems. We confine our attention to \mathcal{L}_2 stability. For linear systems, events are designed by solving LMI feasibility problems. With the linear structure, the state equation of agent i is

$$\begin{aligned} \dot{x}_i(t) &= A_{ii}x_i(t) + B_i u_i(t) + \sum_{j \in D_i} A_{ij}x_j(t) + C_i w_i(t) \\ u_i(t) &= K_{ii}\hat{x}_i(t) + \sum_{j \in Z_i} K_{ij}\hat{x}_j(t) \end{aligned} \quad (10)$$

and the state equation of the overall NCS is

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + Cw(t) \\ u(t) &= K\hat{x}(t), \end{aligned} \quad (11)$$

where $x = (x_1^T, \dots, x_N^T)^T$, $u = (u_1^T, \dots, u_N^T)^T$, $w = (w_1^T, \dots, w_N^T)^T$, and $\hat{x} = (\hat{x}_1^T, \dots, \hat{x}_N^T)^T$. We will use equation (10) and (11) interchangeably to denote the dynamics of the NCS.

We first introduce a lemma, in which LMIs are used to identify the parameters in local events.

Lemma 4.8: Consider the NCS in equation (11). Given $\gamma \in \mathbb{R}^+$, assume that there exist positive-definite matrices $P, Q \in$

$\mathbb{R}^{nN \times nN}$ and $W_i, M_i \in \mathbb{R}^{n \times n}$, $i = 1, 2, \dots, N$ such that:

$$\begin{bmatrix} -P(A+BK) - (A+BK)^T P - Q & PC \\ C^T P & \gamma^2 I_{LN \times LN} \end{bmatrix} \geq 0, \quad (12)$$

$$\begin{bmatrix} Q - W & PBK \\ K^T B^T P & M \end{bmatrix} \geq 0, \quad (13)$$

$$P, Q, M_i, W_i > 0, \quad (14)$$

where $M = \text{diag}\{M_j\}_{j \in \mathcal{N}}$ and $W = \text{diag}\{W_j\}_{j \in \mathcal{N}}$. For all $i \in \mathcal{N}$ and any $\rho_i \in (0, 1)$, let $a_i = 1 + \sqrt{\frac{\lambda_{\max}(M_i)}{\lambda_{\min}(W_i)}}$ If

$$a_i \|e_i(t)\|_2 \leq \rho_i \|\hat{x}_i(t)\|_2 \quad (15)$$

for all $i \in \mathcal{N}$ and all $t \geq t_0$, then the NCS is \mathcal{L}_2 stable from w to x .

Remark 4.9: When equation (48) holds, there always exist positive definite matrices W_i, M_i satisfying (49). For some $\epsilon \in (0, \lambda_{\min}(Q))$, the matrices $W_i = \epsilon I_{n \times n}$ and $M_i = \frac{\|PBK\|^2}{\lambda_{\min}(Q) - \epsilon} I_{n \times n}$, for example, clearly satisfy equations (48) and (49).

Lemma 4.8 provides a way to design local events. Agent i can use the violation of the inequality in (15) to trigger its broadcasts. Directly solving the LMIs in equation (12) – (14), however, may not be suitable for large-scale systems. We now propose a way to solve this LMI feasibility problem in a decentralized manner.

Let us look at agent i . Assume $|\bar{Z}_i| = q_i$, $|\bar{Z}_i \cup \bar{D}_i| = s_i$,

$$\bar{Z}_i = \{i_1, i_2, \dots, i_{q_i}\} \subseteq \mathcal{N}, \text{ and}$$

$$\bar{Z}_i \cup \bar{D}_i = \{i_1, \dots, i_{q_i}, i_{q_i+1}, \dots, i_{s_i}\} \subseteq \mathcal{N}.$$

Without loss of the generality, we assume $i_1 = i$. We also define four matrices $A_i \in \mathbb{R}^{n \times n s_i}$, $K_i \in \mathbb{R}^{m \times n s_i}$, and $\tilde{K}_i \in \mathbb{R}^{m \times n q_i}$, $H_i \in \mathbb{R}^{n s_i \times l}$ by

$$\begin{aligned} A_i &= (A_{i,i_1}, A_{i,i_2}, \dots, A_{i,i_{s_i}}), \\ K_i &= (K_{i,i_1}, K_{i,i_2}, \dots, K_{i,i_{s_i}}), \\ \tilde{K}_i &= (K_{i,i_1}, K_{i,i_2}, \dots, K_{i,i_{q_i}}), \end{aligned} \quad H_i = \begin{bmatrix} P_i C_i \\ \mathbf{0} \end{bmatrix},$$

where $K_{ij} = 0$ if $j \in D_i \setminus Z_i$, and two functions $F_i : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n s_i \times n s_i}$, $G_i : \mathbb{R}^{n \times n} \times \mathbb{R} \rightarrow \mathbb{R}^{n s_i \times n s_i}$ by

$$F_i(P_i) = \begin{bmatrix} P_i(A_i + B_i K_i) \\ \mathbf{0} \end{bmatrix}, \quad G_i(Q_i; \beta) = \begin{bmatrix} Q_i & \mathbf{0} \\ \mathbf{0} & -\beta I \end{bmatrix}.$$

With these matrices and functions, we define the local LMI problem associated with agent i :

Problem 4.10 (Local LMI): Given constants $\delta, \beta > 0$, find $P_i, Q_i, W_i \in \mathbb{R}^{n \times n}$ and $\hat{\gamma}_i \in \mathbb{R}$ such that

$$\begin{bmatrix} -F_i(P_i) - F_i^T(P_i) - G_i(Q_i; \beta) & H_i \\ H_i^T & \hat{\gamma}_i I_{l \times l} \end{bmatrix} \geq 0 \quad (16)$$

$$\begin{bmatrix} Q_i - |S_i \cup U_i| \beta I_{n \times n} - W_i & P_i B_i \tilde{K}_i \\ \tilde{K}_i^T B_i^T P_i & \delta I_{n q_i \times n q_i} \end{bmatrix} \geq 0 \quad (17)$$

$$P_i, W_i > 0, \hat{\gamma}_i > 0. \quad (18)$$

The following theorem shows that solving Problem 4.10 ensures \mathcal{L}_2 stability of the resulting event-triggered NCS.

Theorem 4.11: Consider the NCS in equation (10). Given $\delta, \beta \in \mathbb{R}^+$, assume that for any $i \in \mathcal{N}$, the local LMI in Problem 4.10 is feasible and $P_i, Q_i, W_i \in \mathbb{R}^{n \times n}$, and $\gamma_i \in \mathbb{R}$

are the solutions. For all $i \in \mathcal{N}$ and any $\rho_i \in (0, 1)$, let $b_i = 1 + \sqrt{\frac{\delta|\bar{U}_i|}{\lambda_{\min}(W_i)}}$. If

$$b_i \|e_i(t)\|_2 \leq \rho_i \|\hat{x}_i(t)\|_2 \quad (19)$$

holds for all $t \in \mathcal{N}$ and all $t \geq t_0$, then the NCS is \mathcal{L}_2 stable from w to x .

Proof: Notice that the inequality still holds when we expand the matrices in equation (16) into $nN \times nN$ dimension by appropriately adding zero. Summing both sides of the expanded matrix inequalities shows that equation (12) is satisfied and therefore equation (48) with

$$P = \text{diag}\{P_i\}_{i=1}^N, \quad \gamma = \max_i \{\sqrt{\hat{\gamma}_i}\},$$

$$Q = \text{diag}\{Q_i - |S_i \cup U_i| \beta I_{n \times n}\}_{i=1}^N,$$

where $Q_i - |S_i \cup U_i| \beta I_{n \times n} > 0$ holds by equation (17).

Similarly, we can show that equation (49) is satisfied with

$$W = \text{diag}\{W_i\}_{i=1}^N, \quad M = \text{diag}\{\delta(|\bar{U}_i|)I_{n \times n}\}_{i=1}^N.$$

Since the hypotheses in Theorem 4.8 are satisfied, we conclude that the NCS is \mathcal{L}_2 stable with an induced gain less than $\frac{\gamma}{\sqrt{\min_i \{(1-\rho_i^2)\lambda_{\min}(W_i)\}}}$. ■

Remark 4.12: Since δ and β are pre-selected, the local problem associated with agent i only requires the information on agent i 's system dynamics. To design the local events, agents do not have to know other agents' system information and, therefore, the design scheme is distributed.

Remark 4.13: The dimensions of the matrices on the left-hand side of the LMIs in (16) and (17) are $(ns_i + l) \times (ns_i + l)$ and $(nq_i + n) \times (nq_i + n)$, respectively. These dimensions are much smaller than the dimensions of the matrices in the equations (12) and (13).

In Problem 4.10, two parameters, δ and β , are pre-selected and all agents share the same δ and β . The following corollaries 4.14 and 4.15 discuss the selection of these parameters so that the local LMIs are feasible. A more general setup is to pre-select a group of parameters $\{\delta_i\}_{i=1}^N$ and $\{\beta_i\}_{i=1}^N$. The preceding results can be easily modified to handle this more general setup.

Corollary 4.14: Consider the NCS in equation (1). For any $i \in \mathcal{N}$, if there exists a positive-definite matrix $P_i \in \mathbb{R}^{n \times n}$ such that

$$F_i(P_i) + F_i^T(P_i) + G_i(|S_i \cup U_i| \beta I_{n \times n}; \beta) < 0, \quad (20)$$

then there always exists a positive constant $\delta^* \in \mathbb{R}^+$, such that for any $\delta \geq \delta^*$, the LMI in Problem 4.10 is feasible.

Proof: Equation (20) implies there exists a positive definite matrix $Q_i \in \mathbb{R}^{n \times n}$ such that

$$F_i(P_i) + F_i^T(P_i) + G_i(Q_i; \beta) < 0 \quad (21)$$

$$Q_i - |S_i \cup U_i| \beta I_{n \times n} > 0 \quad (22)$$

Since (21) holds, we know that there always exists a positive constant $\hat{\gamma}_i^* \in \mathbb{R}^+$ such that for all $\hat{\gamma}_i \geq \hat{\gamma}_i^*$, equation (16) holds. Equation (22) implies that there exists a positive definite matrix $W_i \in \mathbb{R}^{n \times n}$ such that $Q_i - |S_i \cup U_i| \beta I_{n \times n} - W_i > 0$, which suggests that there always exists a positive constant $\delta^* \in \mathbb{R}^+$ such that for all $\delta \geq \delta^*$, equation (17) holds. ■

Corollary 4.14 suggests that δ must be large enough to guarantee the feasibility of the local LMI, provided equation (20) holds. We still need to know how to select β . The following corollary shows that the feasibility of equation (20) is independent of the selection of β .

Corollary 4.15: If there exist a positive-definite matrix $P_i \in \mathbb{R}^{n \times n}$ and a positive constant $\beta \in \mathbb{R}$ such that equation (20) holds, then for any $\hat{\beta} > 0$, $\frac{\hat{\beta}}{\beta} P_i$ and $\hat{\beta}$ also satisfy equation (20).

Proof: This can be easily proven by the definitions of F_i and G_i . ■

V. EVENT-TRIGGERING WITH DATA DROPOUTS AND TRANSMISSION DELAYS

The previous section provides real-time constraints that guarantee \mathcal{L}_p stability. If there are no dropouts and delays during data transmissions, agents can directly use the violation of the inequality in (9) to trigger the broadcast. When dropouts and delays are involved, agent i uses the violation of

$$\|x_i(t) - x_i(r_j^i)\|_2 \leq \frac{\rho_i}{c_i} \|x_i(r_j^i)\|_2 \quad (23)$$

to trigger the $j + 1$ st broadcast release, where r_j^i denotes the time instant when agent i samples and releases the j th broadcast. Notice that the difference between equation (23) and (9) is that $x_i(r_j^i)$ in (23) might be lost during the transmission; while $x_i(b_k^i)$ in (9) is always successfully transmitted. Under this triggering mechanism, we provide a maximal allowable number of successive dropouts (MANSD) and a bound on delays (also called ‘‘deadline’’) for each agent to ensure asymptotic stability of the overall system. These parameters can be identified by agents using local information. For notational convenience, let $\varepsilon_i^j(t) = x_i(t) - x_i(r_j^i)$ and d_{MANSD}^i denote agent i 's MANSD.

To analyze data dropouts and transmission delays in networks, we first need to introduce the transmission procedure. Let us take agent i as an example. Agent i uses the violation of inequality in (23) to trigger the next broadcast. When the local event occurs, agent i samples and then sends a DATA message to neighboring agents in U_i . The DATA packet contains a time tag and the sampled state $x_i(r_{j+1}^i)$. At the same time, the triggering event is updated to be the violation of

$$\|\varepsilon_i^{j+1}(t)\|_2 \leq \frac{\rho_i}{c_i} \|x_i(r_{j+1}^i)\|_2.$$

Those agents who receive the packet from agent i need to send acknowledgement messages (ACK) back to agent i . Notice that at this point, agents in \bar{U}_i are not allowed to use this DATA packet to update their control inputs.

If agent i receives confirmations from **ALL** of its neighbors in U_i within τ_k^i seconds, i.e. during the interval $[r_{j+1}^i, r_{j+1}^i + \tau_k^i)$, it sends out a permission message (PERM) to its neighboring agents. The PERM message gives neighboring agents permission to use the previously transmitted data. Otherwise, if agent i does not receive confirmations from **ALL** of its neighbors in U_i within τ_k^i seconds, the DATA packet is treated as a lost packet and discarded. Notice that sending PERM indicates a successful broadcast. r_{j+1}^i is, therefore, the time

instant of a successful broadcast. We use the symbol b_k^i to denote the release time of the k th successful broadcast ($\{b_k^i\}_{j=1}^\infty$ is a subsequence of $\{r_j^i\}_{j=1}^\infty$). Only after the agents in \bar{U}_i receive the permission from agent i , they are allowed to use this packet to update their control inputs. Notice that the sizes of the ACK and PERM packets are very small. Therefore, they will not take a lot communication resource. We can assume that transmission of PERM packet is instantaneous and will not be lost.

Following this transmission protocol, it is easy to see that a packet sent by an agent is either lost or transmitted to all of its neighbors. The broadcast state $\hat{x}_i(t)$ of agent i , therefore remains consistent across all of its neighbors. It is important to have this consistency in the system. Otherwise, different versions of measurement errors will be generated that are difficult for agent i to track. For example, assume that agent 1 receives and uses the DATA package from agent i while agent 2, another neighbor of agent i , does not receive that packet. Agent 2 therefore still uses the previously received state of agent i , say x_{i2} . Then agent 2's measurement error in the state of agent i is $x_i(t) - x_{i2}$. This error, however, is difficult for agent i to track because there is no way for agent i to know exactly what x_{i2} is. How to handle different versions of measurement errors in distributed systems might be an interesting topic in the future..

We currently know how to design local events. But to ensure asymptotic stability of the system, we still need to determine d_{MANSD}^i and derive upper bounds on τ_k^i so that $c_i \|e_i(t)\|_2 \leq \bar{\rho}_i \|\hat{x}_i(t)\|_2$ is always valid with some $\bar{\rho}_i \in (\rho_i, 1)$ for all $t \geq t_0$ and all $i \in \mathcal{N}$. Notice that $\hat{x}_i(t) = x_i(b_k^i)$ for all $t \in [f_k^i, f_{k+1}^i)$ and therefore $e_i(t) = x_i(t) - x_i(b_k^i)$ for all $t \in [f_k^i, f_{k+1}^i)$. Recall that f_k^i is the time instant when the k th successful broadcast is completed. This suggests that we only need to ensure $c_i \|x_i(t) - x_i(b_k^i)\|_2 \leq \bar{\rho}_i \|x_i(b_k^i)\|_2$ over the time interval $[f_k^i, f_{k+1}^i)$.

We may actually split $[f_k^i, f_{k+1}^i)$ into two subintervals: $[f_k^i, b_{k+1}^i)$ and $[b_{k+1}^i, f_{k+1}^i)$. To determine d_{MANSD}^i , we focus on the time interval $[f_k^i, b_{k+1}^i)$ since data dropouts happen during this time interval. d_{MANSD}^i is selected in a way that even if packets are lost, the real-time constraint, $c_i \|e_i(t)\|_2 \leq \bar{\rho}_i \|\hat{x}_i(t)\|_2$, is still valid over that interval. To determine bounds on τ_k^i , we focus on the interval $[b_{k+1}^i, f_{k+1}^i)$ because τ_k^i is basically the delay in the k th transmission. We need to find an upper bound on $f_k^i - b_k^i$, denoted as η_k^i , that ensures asymptotic stability.

Before we present the main results, we need two lemmas. The first lemma (Lemma 5.1) describes the behavior of $e_i(t)$ over $[f_k^i, b_{k+1}^i)$ when data dropouts happen. The second lemma (Lemma 5.3) shows that if there is a bound on the delays and the overall system dynamics is bounded, the resulting NCS is uniformly ultimately bounded.

Lemma 5.1: Consider the NCS in equation (1). Suppose that Assumption 4.1 holds. Given two collections of positive constants $\rho_i \in (0, 1)$ and $\varrho_i \in [\rho_i, 1)$ for $i = 1, 2, \dots, N$, if for any $i \in \mathcal{N}$, the inequality

$$c_i \|e_i^j(t)\|_2 \leq \rho_i \|x_i(r_j^i)\|_2, \quad (24)$$

holds for all $t \in [r_j^i, r_{j+1}^i)$, where $c_i = 1 + \sigma_i$, and σ_i is defined

by equation (7), and the number of successive dropouts, $d_k^i \in \mathbb{Z}$, satisfies

$$d_k^i \leq d_{\text{MANSD}}^i \triangleq \left\lfloor \log_{\left(1 + \frac{\rho_i}{c_i}\right)} \left(1 + \frac{\varrho_i}{c_i}\right) - 1 \right\rfloor, \quad (25)$$

then the inequality

$$\|x_i(t) - x_i(b_k^i)\|_2 \leq \frac{\xi_i(d_k^i)}{c_i} \|x_i(b_k^i)\|_2 \leq \frac{\varrho_i}{c_i} \|x_i(b_k^i)\|_2 \quad (26)$$

holds for all $t \in [b_k^i, b_{k+1}^i)$ and all $k \in \mathbb{N}$, where $\xi_i : \mathbb{Z} \rightarrow (0, \varrho_i)$ is defined by

$$\xi_i(d_k^i) \triangleq c_i \left(1 + \frac{\rho_i}{c_i}\right)^{d_k^i+1} - c_i \in (0, \varrho_i). \quad (27)$$

Remark 5.2: If all the hypotheses in Lemma 5.1 hold and $b_k^i = f_k^i$ holds for all $i \in \mathcal{N}$ and all $k \in \mathbb{N}$, then the NCS is finite-gain \mathcal{L}_p stable from w to x . This is because when $b_k^i = f_k^i$ holds, $x(b_k^i) = \hat{x}_i(t)$ and $\|e_i(t)\|_2 = \|x_i(t) - x_i(b_k^i)\|_2$ for $t \in [f_k^i, f_{k+1}^i)$. Equation (26), therefore, implies $\|e_i(t)\|_2 \leq \frac{\varrho_i}{c_i} \|\hat{x}_i(t)\|_2$ for all $t \geq 0$ with $\varrho_i \in (0, 1)$. This is sufficient to show that the NCS is finite-gain \mathcal{L}_p stable from w to x according to corollary 4.6.

Lemma 5.3: Consider the NCS in equation (1) with $w_i = 0$ for all $i \in \mathcal{N}$. Suppose that Assumption 4.1 holds and ζ_1^i, ζ_2^i in equation (4) satisfy

$$\zeta_1^i(\|x_i\|_2) \geq \underline{L}_i \|x_i\|_2^q \quad \text{and} \quad \zeta_2^i(\|x_i\|_2) \leq \bar{L}_i \|x_i\|_2^q, \quad (28)$$

respectively, with some positive constants $\underline{L}_i, \bar{L}_i > 0, q \geq 1$. Also assume that there exist a collection of positive constants $\theta_i \in \mathbb{R}^+$ for $i = 1, 2, \dots, N$ such that

$$\|f_i(x_{\bar{D}_i}(t), g_i(\hat{x}_{\bar{Z}_i}(t)), 0)\|_2 \leq \theta_i, \quad (29)$$

holds for all $t \geq t_0$ and all $i \in \mathcal{N}$. Given a constant $\Delta \in \mathbb{R}_0^+$ and two collections of positive constants $\rho_i \in (0, 1)$, $\varrho_i \in [\rho_i, 1)$ for $i = 1, 2, \dots, N$, if for any $i \in \mathcal{N}$, equation (24) holds for all $t \in [r_j^i, r_{j+1}^i)$, the number of successive dropouts, d_k^i , satisfies equation (25), and the delay in the $k+1$ st successful transmission satisfies

$$f_{k+1}^i - b_{k+1}^i \leq \frac{1 - \xi_i(d_k^i)}{c_i \theta_i} \max \left\{ \frac{\|x_i(b_k^i)\|_2}{2}, \Delta \right\} \quad (30)$$

where ξ_i is defined in equation (27), then for any $\bar{\pi} \in \mathbb{R}^+$ such that $\bar{\pi} > \pi$, there exists $T \geq t_0$ such that

$$\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \leq \max_{i, j \in \mathcal{N}} \left\{ \frac{\bar{L}_i}{\underline{L}_j} \right\} \mu \bar{\pi}^q \Delta^q$$

holds for all $t \geq T$, where

$$\mu = \begin{cases} 1 & p \leq q \\ N^{1-\frac{q}{p}} & p > q \end{cases} \quad (31)$$

$$\pi = \left(\frac{\sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) (1 - \rho_i)}{\min_{i \in \mathcal{N}} \{(\alpha_i - |S_i \cup U_i| \beta_i) (1 - \bar{\varsigma}_i)\}} \right)^{\frac{1}{q}} \quad (32)$$

$$\bar{\varsigma}_i = \max \left\{ \left(\frac{1 + \varrho_i}{2} \right)^p, \varrho_i \right\}. \quad (33)$$

Remark 5.4: If the delays are less than the first term (associated with $\|x_i(b_k^i)\|_2$) in the bound in equation (30), then the real-time constraint in equation (9) can be guaranteed. However, we cannot simply use $\frac{1 - \xi_i(d_k^i)}{c_i \theta_i} \frac{\|x_i(b_k^i)\|_2}{2}$ as the bound

on delays because this bound may shrink to zero as the state approaches zero, which is not practical. Alternatively, we introduce $\frac{1-\xi_i(d_k^i)}{c_i\theta_i}\Delta$, which can be viewed as the worst-case bound. When Δ dominates the max operator, the real-time constraint in equation (9) will be violated. This violation, however, only happens when the state is close to zero.

Lemma 5.3 suggests that, with the assumption that the system dynamics is bounded, the overall system is globally uniformly ultimately bounded for any Δ . It is, however, still not clear how to select Δ so that this assumption of bounded system dynamics holds. The following lemma solves this issue and therefore helps us relax the assumption of equation (29) in Lemma 5.3. It shows that if Δ is small enough, the system dynamics will be in a pre-selected compact set. We now define this compact set. Suppose Assumption 4.1 holds and f_i, g_i are locally Lipschitz for all $i \in \mathcal{N}$. Then we can define a compact set, $\Lambda \subset \mathbb{R}^{nN}$, as

$$\Lambda \triangleq \{x \in \mathbb{R}^{nN} \mid V(x) \leq V(x_0)\} \quad (34)$$

and find positive constants, $\theta_i, L_i, \underline{L}_i, \bar{L}_i \in R$ for $i = 1, 2, \dots, N$ such that for $\forall x, \hat{x} \in \Lambda$,

$$\|f_i(x_{\bar{D}_i}, g_i(\hat{x}_{\bar{Z}_i}), 0)\|_2 \leq L_i \sum_{i \in \mathcal{N}} (\|x_i\|_2 + \|\hat{x}_i\|_2), \quad (35)$$

$$\zeta_1^i(\|x_i\|_2) \geq \underline{L}_i \|x_i\|_2^q, \quad \zeta_2^i(\|x_i\|_2) \leq \bar{L}_i \|x_i\|_2^q, \quad (36)$$

$$\theta_i \triangleq (1 + |\bar{Z}_i|) L_i N^{\frac{q-1}{q}} \left(\frac{V(x_0)}{\min_{i \in \mathcal{N}} \underline{L}_i} \right)^{\frac{1}{q}} \quad (37)$$

with some $q \geq 1$. For the notational convenience, we use $V(t)$ to denote $V(x(t))$ for all $t \geq t_0$.

Lemma 5.5: Consider the NCS in equation (1) with $w_i = 0$ for all $i \in \mathcal{N}$. Suppose that Assumption 4.1 and equation (35), (36) hold with some $q \geq 1$. Given positive constants $\rho_i \in (0, 1)$, $\varrho_i \in [\rho_i, 1)$ for all $i \in \mathcal{N}$ and $\bar{\pi} \in (\pi, \infty)$, where π is defined in equation (32), if for any $i \in \mathcal{N}$, equation (24) holds for all $t \in [r_j^i, r_{j+1}^i)$, the number of successive data dropouts, $d_k^i \in \mathbb{Z}$, satisfies equation (25), and the delay in the $k+1$ st successful transmission satisfies

$$f_{k+1}^i - b_{k+1}^i \leq \eta_k^i \triangleq \max \left\{ \frac{(1-\xi_i(d_k^i))\|x_i(b_k^i)\|_2}{2c_i\theta_i}, \frac{(1-\xi_i(d_k^i)) \min_i \underline{L}_i^{\frac{1}{q}}}{(1+|\bar{Z}_i|)c_i L_i \bar{\pi} N^{1-\frac{1}{\max\{p,q\}}} \max_i \bar{L}_i^{\frac{1}{q}}} \right\} \quad (38)$$

where $L_i, \bar{L}_i, \underline{L}_i, \xi_i$ are defined in equation (35), (36), (27), respectively, then $x(t) \in \Lambda$ for all $t \geq t_0$, where Λ is defined in equation (34), .

With the upper bound on delays derived in Lemma 5.5, we know that the system dynamics is bounded. Therefore, by Lemma 5.3 the NCS is at least uniformly ultimately bounded. The following theorem shows that with such an upper bound on delays, the ultimate bound will gradually shrink to zero, which implies asymptotic stability of the system.

Theorem 5.6: Consider the NCS in equation (1) with $w_i = 0$ for all $i \in \mathcal{N}$. Suppose that Assumption 4.1 and equation (35), (36) hold with some $q \geq 1$. Given positive constants $\nu \in \mathbb{R}^+$, $\rho_i \in (0, 1)$, $\varrho_i \in [\rho_i, 1)$ for all $i \in \mathcal{N}$ and $\bar{\pi} \in (\pi, \infty)$, where π is defined in equation (32), if for any $i \in \mathcal{N}$, the

$j+1$ st broadcast is released by the violation of

$$E_1 \wedge E_2, \quad (39)$$

where E_1 is the inequality in (24) and

$$E_2 : t \leq r_j^i + \nu, \quad (40)$$

the number of successive data dropouts, $d_k^i \in \mathbb{Z}$, satisfies equation (25), and the delay in the $k+1$ st successful transmission satisfies equation (38), then the NCS is asymptotically stable.

Remark 5.7: The introduction of ν in E_2 can be viewed as the security requirement of the system. It requires each agent broadcasts at least every ν units of time. ν is arbitrarily chosen. E_2 ensures that agents still communicate even when they stay in the ultimate set.

Remark 5.8: η_k^i in equation (38) serves as the deadline for the k th successful broadcast of agent i . With the fact that $\xi_i(d_k^i) \leq \varrho_i$ holds, we have η_k^i always greater than a positive constant, τ_{SPD}^i . In other words,

$$\eta_k^i \geq \tau_{\text{SPD}}^i = \frac{(1-\varrho_i) \min_i \underline{L}_i^{\frac{1}{q}}}{(1+|\bar{Z}_i|)c_i L_i \bar{\pi} N^{1-\frac{1}{\max\{p,q\}}} \max_i \bar{L}_i^{\frac{1}{q}}} > 0 \quad (41)$$

holds for all $k \in \mathbb{N}$. τ_{SPD}^i is the *smallest predicted deadline* (SPD) of agent i . To show the SPD is greater than zero is important in establishing that our scheme does not require the network to transmit data infinitely fast.

Remark 5.9: The number of successive dropouts, d_k^i determines the deadline η_k^i . As d_k^i increases, the value of $\xi_i(d_k^i)$ increases. It, therefore, results in a short deadline according to equation (38). There is a trade-off between the number of successive dropouts and the deadline.

Remark 5.10: Two parameters ρ_i, ϱ_i are used in the scheme. The parameter ρ_i determines $d_{\text{MANS}}^i, \tau_{\text{SPD}}^i$, and the transmission periods, T_j^i . The large ρ_i is, the longer T_j^i is and the smaller d_{MANS}^i is, according to equation (24) and (25). Large ρ_i may also result in a small π according to equation (32) and therefore leads to a larger τ_{SPD}^i . The parameter ϱ_i determines d_{MANS}^i and τ_{SPD}^i . The large ϱ_i is, the larger d_{MANS}^i is and the smaller τ_{SPD}^i is. As a ‘‘rule of thumb’’, a reasonable strategy is to choose ρ_i and ϱ_i so that the periods and the SPDs are as large as possible; as this makes the task easier to schedule under an earliest-deadline first (EDF) scheduling discipline.

Remark 5.11: To design a system with special requirements on d_{MANS}^i and τ_{SPD}^i , we solve equation (25) and (38) for ρ_i and ϱ_i . There is no constraint on the selection of d_{MANS}^i . It can be arbitrarily large. τ_{SPD}^i , however, must be less than some positive constant so that equation (25) and (38) have solutions.

Remark 5.12: It is not necessary that the transmission periods reduce to zero as the state approaches to the origin. The transmission period is determined by the growth rate of the ratio $\frac{\|\varepsilon_i^j(t)\|_2}{\|x_i(r_j^i)\|_2}$. For certain systems, when the threshold $(\|x_i(r_j^i)\|_2)$ goes to zero, the growth rate of the error is also decreasing. As long as the growth rate of $\frac{\|\varepsilon_i^j(t)\|_2}{\|x_i(r_j^i)\|_2}$ is bounded from above, there must be a constant lower bound on periods.

Such a time-space relation is further discussed in [21], where a scaling law is provided.

Remark 5.13: When disturbances hit agent i , $\|\varepsilon_i^j(t)\|_2$ might grow faster and therefore the growth rate of $\frac{\|\varepsilon_i^j(t)\|_2}{\|x_i(r_j^i)\|_2}$ will be large. As a result, to ensure inequality (24), the required transmission frequency will be very high. This is especially true when the state is close to the equilibrium. When the required transmission frequency exceeds the frequency that the network can provide, the inequality (24) will be violated. It might result in the violation of the real-time constraints in (9). In this case, the state will not converge to the equilibrium, but stay in a neighborhood of the equilibrium. The size of the neighborhood may be determined by the size of the disturbances and the capacity of the communication network. It is also possible that by successfully communicating enough times among agents, equation (9) is still valid. This reflects a tradeoff between the transmission periods and the number of allowable dropouts. By reducing the number of dropouts, the ‘‘allowable’’ transmission periods can be enlarged. By ‘‘allowable’’, it mean the transmission periods to ensure equation (9). In summary, whether the real-time constraint in (9) will be violated or not depends on the size of the disturbance, how close the state is to the origin, and the number of dropouts during that time interval.

We also provide a lower bound on the transmission periods in the following corollary.

Corollary 5.14: If the hypotheses in Theorem 5.6 hold, then

$$T_j^i = r_{j+1}^i - r_j^i \geq \min \left\{ \nu, \frac{\rho_i}{c_i \theta_i} \|x_i(r_j^i)\|_2 \right\} \quad (42)$$

holds for all $j \in \mathbb{N}$ and all $i \in \mathcal{N}$.

Proof: By Lemma 5.5, we have

$$\|f_i(x_{\bar{D}_i}(t), g_i(\hat{x}_{\bar{Z}_i}(t)), 0)\|_2 \leq \theta_i$$

for all $i \in \mathcal{N}$ and all $t \geq t_0$. Consider the derivative of $\|\varepsilon_i^j(t)\|_2$ over the time interval $[r_j^i, r_{j+1}^i)$.

$$\begin{aligned} \frac{d}{dt} \|\varepsilon_i^j(t)\|_2 &\leq \|\dot{\varepsilon}_i^j(t)\|_2 = \|\dot{x}_i(t)\|_2 \\ &= \|f_i(x_{\bar{D}_i}(t), g_i(\hat{x}_{\bar{Z}_i}(t)), 0)\|_2 \leq \theta_i \end{aligned}$$

holds for $\forall t \in [r_j^i, r_{j+1}^i)$. Solving this inequality with the initial condition $\|\varepsilon_i^j(r_j^i)\|_2 = 0$ implies

$$\|\varepsilon_i^j(t)\|_2 \leq \theta_i(t - r_j^i) \quad (43)$$

holds for all $t \in [r_j^i, r_{j+1}^i)$. If the $j + 1$ st broadcast is triggered by the violation of E_1 in equation (24), then $\|\varepsilon_i^j(r_{j+1}^i)\|_2 = \frac{\rho_i}{c_i} \|x_i(r_j^i)\|_2$ holds and therefore equation (43) implies $\|\varepsilon_i^j(r_{j+1}^i)\|_2 = \frac{\rho_i}{c_i} \|x_i(r_j^i)\|_2 \leq \theta_i(r_{j+1}^i - r_j^i)$, which means $r_{j+1}^i - r_j^i \geq \frac{\rho_i}{c_i \theta_i} \|x_i(r_j^i)\|_2$. If the broadcast is triggered by the violation of E_2 in equation (40), then $T_j^i = \nu$ holds. Combining these two cases yields the satisfaction of equation (42). ■

Remark 5.15: It is easy to see that in corollary 5.14, the lower bound on the transmission period goes to zero as the state goes to the equilibrium. It, however, does not mean the

actual periods go to zero. In fact, in the simulations in section VI, the periods are always greater than a positive constant. It suggests that the bound in corollary 5.14 may be conservative for certain systems.

Remark 5.16: With the bound on periods given in corollary 5.14, the MANS D given in equation (25), and the deadline given in equation (38), we can not only do hard real-time scheduling of the transmission tasks, but also study firm real-time scheduling schemes, in which the case that a task misses its deadline is allowed.

VI. SIMULATIONS

This section presents simulation results demonstrating the distributed event-triggering scheme. The system under study is a collection of carts coupled by springs (Figure 1). Both soft spring models (nonlinear) and normal spring models (linear) are considered. The state of the i th subsystem is the vector $x_i = [y_i \ \dot{y}_i]^T$ where y_i is the i th cart’s position. We assume that at equilibrium, all springs are unstretched. We also assume that cart i can only receive the broadcast state information from cart $i + 1$.

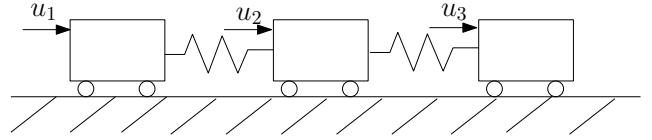


Fig. 1. Three carts coupled by springs

This section consists of four subsections. Subsection VI-A shows how to implement our scheme in nonlinear systems. Subsection VI-B investigates the robustness of the event-triggering scheme to transmission delays, data dropouts, and exogenous disturbances. Subsection VI-C explores how the parameters in the scheme affects transmission periods, the MANS D, and the SPD. Finally, subsection VI-D examines the communication cost and the complexity of our scheme by comparing it against the approach in [12] that derives the bound on the MATI. These simulations results show that our scheme is robust to transmission delays, data dropouts, and exogenous disturbances. The average broadcast period and the time spent in event design in our scheme scales well with respect to the number of agents.

A. Implementation in Nonlinear Systems

This subsection considers how to implement our scheme in nonlinear systems. In this simulation, the carts are coupled together by soften springs [27]. The state equation for the i th cart is

$$\dot{x}_i = \begin{bmatrix} \dot{y}_i \\ u_i + \kappa_i^1 \tanh(y_{i+1} - y_i) + \kappa_i^2 \tanh(y_{i-1} - y_i) \end{bmatrix} \quad (44)$$

In the preceding equation, $\kappa_i^2 = \kappa_N^1 = 0$. Otherwise, $\kappa_i^1 = \kappa_i^2 = 1$.

The control input of subsystem i is

$$u_i = K_i \hat{x}_i - \kappa_i^1 \tanh(\hat{y}_{i+1} - \hat{y}_i),$$

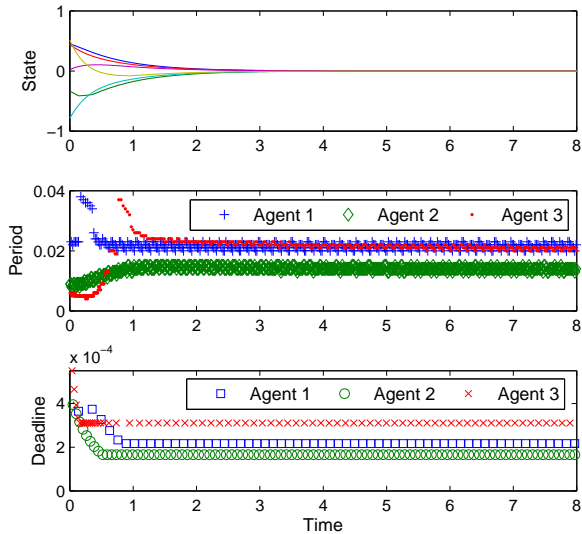


Fig. 2. State trajectory, broadcast periods, and predicted deadlines in an event-triggered NCS

where $K_i = \begin{bmatrix} -5 & -5 \end{bmatrix}$ for $i = 1, \dots, N$.

We set $\nu = 100$, $\delta_i = 10$, $\beta_i = 1$, $\rho_i = 0.1$, $\varrho_i = 0.65$, and $\gamma_i = 40$ for all $i \in \mathcal{N}$. The triggering events are

$$\begin{aligned} 0.1 \|x_1(r_j^i)\|_2 &= 3.6847 \|\varepsilon_1^j(t)\|_2 \\ 0.1 \|x_i(r_j^i)\|_2 &= 5.7202 \|\varepsilon_i^j(t)\|_2, \text{ for } i = 2, \dots, N-1 \\ 0.1 \|x_N(r_j^N)\|_2 &= 3.8700 \|\varepsilon_N^j(t)\|_2 \end{aligned}$$

according to equation (24) and the MANSDs for agents are all 5 according to equation (25).

We set $N = 3$ and ran the event-triggered NCS for 6 seconds. We assumed that for each agent, the number of data dropouts between successive transmissions is equal to its MANSD. We also assumed that transmission delays are equal to the predicted deadlines defined in equation (38). The initial state x_0 was randomly generated satisfying $\|x_0\|_\infty \leq 1$. From the top plot of figure 2, we can see that the system is asymptotically stable. The broadcast periods of agent 1 (cross), agent 2 (diamond), and agent 3 (dot) are shown in the middle plot of figure 2 that vary in a wide range before the system approaches its equilibrium. It demonstrates the ability of event-triggering in adjusting broadcast periods in response to variations in the system's states. Also notice that in the simulation the periods are always greater than a positive constant, even when the states are close to the equilibrium. This is because as $x_i(t)$ approaches zero, although the threshold ($\|x_i(r_j^i)\|_2$) goes to zero, the growth rate of the error is also decreasing. For this specific system, the growth rate of the ratio ($\frac{\|\varepsilon_i^j(t)\|_2}{\|x_i(r_j^i)\|_2}$) happens to be bounded from above, which results in a constant lower bound on periods in this simulation. On the other hand, it means the lower bound on periods obtained in corollary 5.14 might be conservative. The bottom plot in figure 2 shows the history of predicted deadlines, which are reduced to fixed constants as the states approach to the equilibrium. This is because as the states get small,

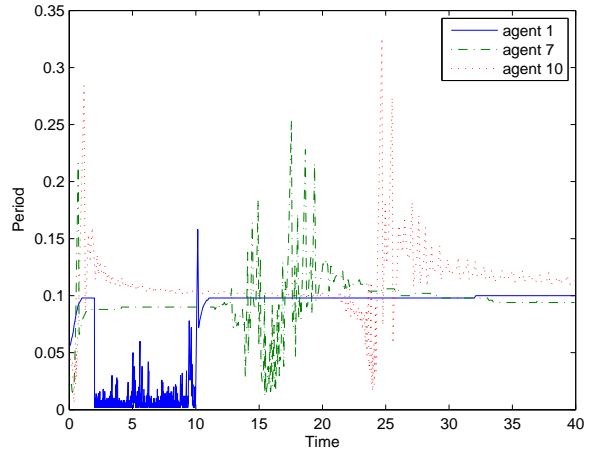


Fig. 3. Successful broadcast periods versus time in an event-triggered NCS with disturbances in agent 1

$\frac{(1-\xi_i(d_k^i)) \min_i L_i^{\frac{1}{q}}}{(1+|\bar{Z}_i|)c_i L_i \bar{\pi} N^{1-\frac{1}{\max\{p,q\}}} \max_i L_i^{\frac{1}{q}}}$ dominates the deadlines. The SPDs of agent 1, 2, 3 are $280 \mu\text{s}$, $140 \mu\text{s}$, and $300 \mu\text{s}$, respectively.

B. Robustness

Robustness of our scheme is studied with respect to delays, dropouts, and exogenous disturbances in this subsection. We first considered increasing the transmission delays in the simulation. We kept all settings the same as the simulation in subsection VI-A except that the transmission delays can exceed the predicted deadlines. The system becomes unstable when the delay in each transmission is larger than 0.002 s. The time, 0.002 s, is, therefore, the maximal delay that the system can actually tolerate. Notice that our predicted SPDs ($280 \mu\text{s}$, $140 \mu\text{s}$, and $300 \mu\text{s}$ in agent 1,2,3, respectively) are around 10% of the actual allowable delay.

The next simulation used the model in section VI-A except that data dropouts are modeled as a stochastic way, instead of setting it equal to the MANSD. The probability of a data dropout is set to be a constant $p \in [0, 1]$. Simulation results show that the system can be stable even when p is as large as 0.9. The maximal number of successive dropouts that occurred in the simulation is 41. These results show that the event-triggered system is very robust to data dropouts.

We finally considered the effect of exogenous disturbances on the event-triggered system. We assumed that there are neither transmission delays nor data dropouts in the system. We ran the system for 40 seconds with $N = 10$. $\rho_i = \varrho_i = 0.9$ for all $i \in \mathcal{N}$. An external disturbance was added into agent 1, where $|w_1(t)| \leq 5$ for $t \in [2, 10]$ and $w_i(t) = 0$ otherwise. The broadcast periods of agent 1, 7, 10 are plotted in Figure 3. We see from the figure that agent 1's broadcast periods became short when the disturbance came in during $t \in [2, 10]$. It is because event-triggering can adjust the agent's broadcast periods in response to variations in the system's external inputs. Although no disturbance directly came into agent 7, 10, their periods were also shortened. Also notice that the decrease

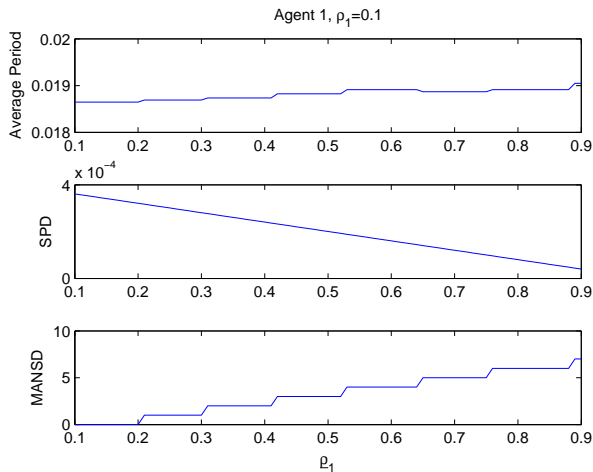


Fig. 4. The average period, the SPD, and the MANSND in agent 1 versus ϱ_1

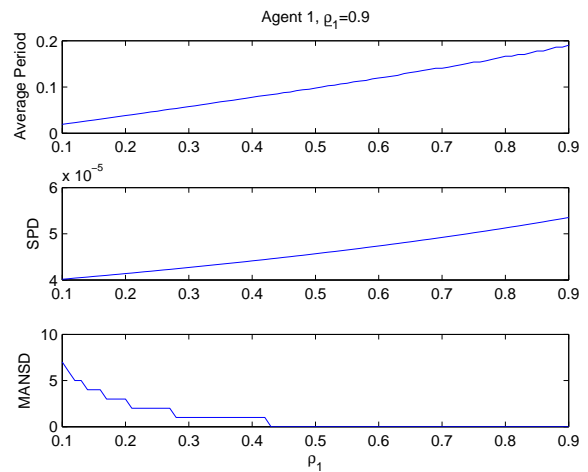


Fig. 5. The average period, the SPD, and the MANSND in agent 1 versus ρ_1

of agents' periods happens over different time intervals. This is because the effect of the disturbance in agent 1 was passed to each agent, from 1 to 10. The spatial distance causes a time delay in passing the effect of the disturbance. Another thing worth mentioning is that, although the periods in agent 7 and 10 decrease for a while, the intensity of such decrease is much less than that in agent 1. This is because the effect of the disturbance decreases when it is passed from agent 1 to other agents.

C. Selection of Parameters

The simulations in this subsection examined the effect of parameters ρ_i and ϱ_i on the broadcast periods, the MANSNDs, and the SPDs. In particular, we studied agent 1. We assume that the delays in each agent are equal to its SPD and the number of each agent's successive dropouts is equal to its MANSND. The parameters $\rho_2 = \rho_3 = 0.1$ and $\varrho_2 = \varrho_3 = 0.9$.

We first fixed $\rho_1 = 0.1$ and varied ϱ_1 from 0.1 to 0.9. The simulation results are shown in figure 4. The top plot in figure 4 is the average broadcast periods versus ϱ_1 . It shows that the average period almost remains the same as ϱ_1 changes. It suggests that ϱ_i does not affect the broadcast periods. The middle plot in figure 4 is the SPD versus ϱ_1 . We can see that when ϱ_1 increases, the SPD decreases. Therefore, to have a longer SPD, we need a small ϱ_i . The bottom plot in figure 4 is the MANSND versus ϱ_1 . As ϱ_1 increases, the MANSND increases, which means ϱ_i has to be large to ensure large MANSND.

We then fixed $\varrho_1 = 0.9$ and varied ρ_1 from 0.1 to 0.9. The simulation results are shown in figure 5. The top plot in figure 4 is the average broadcast periods versus ρ_1 . As ρ_1 increases, the average period increases. It implies that to obtain long periods, ρ_1 needs to be large. The middle plot in figure 4 is the SPD versus ρ_1 . We can see that when ρ_1 increases, the SPD increases. The bottom plot in figure 4 is the MANSND versus ρ_1 . As ρ_1 increases, the MANSND decreases. These simulations verify the comments in Remark 5.10. It suggests a tradeoff between the broadcast periods, the MANSND, and the SPD.

D. Scalability

This subsection studied the scalability of the distributed event-triggered system. We compared the average broadcast period and the design complexity of our scheme against the approach in [12] that derives the bound on the MATI. The simulations were done using normal spring models (linear). The state equation of the i th cart is

$$\begin{aligned} \dot{x}_i &= \begin{bmatrix} \dot{y}_i \\ u_i + \kappa_i^1(y_{i+1} - y_i) + \kappa_i^2(y_{i-1} - y_i) + w_i \end{bmatrix} \\ u_i &= K_i \hat{x}_i - \kappa_i^1(\hat{y}_{i+1} - \hat{y}_i), \end{aligned} \quad (45)$$

where $K_i = \begin{bmatrix} -5 & -5 \end{bmatrix}$ for $i = 1, \dots, N$ and $\kappa_1^2 = \kappa_N^1 = 0$, $\kappa_i^1 = \kappa_i^2 = 1$ otherwise.

We set $\delta = 10$ and $\beta = 1$ and solved local LMI problems 4.10 using MATLAB toolbox. With $\rho_i = 0.9$ for $\forall i \in \mathcal{N}$, the events are

$$\begin{aligned} 0.9 \|x_1(r_j^1)\|_2 &= 2.5908 \|\varepsilon_1^j(t)\|_2 \\ 0.9 \|x_i(r_j^i)\|_2 &= 4.1626 \|\varepsilon_i^j(t)\|_2, \text{ for } i = 2, \dots, N-1 \\ 0.9 \|x_N(r_j^N)\|_2 &= 3.7833 \|\varepsilon_N^j(t)\|_2. \end{aligned}$$

We first compared the average broadcast period generated by our scheme with the MATI in [12]. Recall that for a NCS containing N subsystems, the MATI in [12], denoted as T_{MATI}^N , is,

$$T_{\text{MATI}}^N = \frac{1}{L} \ln \frac{L + \gamma}{\rho L + \gamma}, \quad (46)$$

where, with TOD protocol, $\rho = \sqrt{\frac{N-1}{N}}$, $L = \max(0.5\lambda_{\max}(-BK - K^T B^T), 0)$, γ is the \mathcal{L}_2 gain for the system ($\dot{x} = (A + BK)x + BKe + Cw$) from (e, w) to $-(A + BK)x$, and A, B, C, K are in equation (11).

The average broadcast period generated by our scheme in a NCS containing N agents, denoted as \bar{T}_N , is defined as

$$\bar{T}_N = \frac{\text{System Runtime}}{\text{Total Number of Broadcasts}}. \quad (47)$$

The simulation was done under the assumption that there are no data dropouts, transmission delays or disturbances

in NCS. We varied N from 3 to 100. The system ran for 10 seconds. We compared \bar{T}_N and T_{MATI}^N as N varied. In the simulation, the MATI, T_{MATI}^N , is almost 6 times smaller than the average broadcast period generated by our scheme, \bar{T}_N , for all $N \in [3, 100]$. This means our scheme uses less communication resource, compared with the MATI. Also notice that the average broadcast frequency, $\frac{1}{\bar{T}_N}$, is almost proportional to the number of agents increases. Moreover, the increase of $\frac{1}{\bar{T}_N}$ is much slower than that of $\frac{1}{T_{\text{MATI}}^N}$ as N increases. This suggests that the average broadcast period in our scheme scales well with respect to the number of agents.

We also compared the time spent in event design in our scheme with the time spent in computing the MATI in [12]. The simulation was run with 3.0GHZ P4 CPU. It took around 0.04 seconds in MATLAB to compute the parameter, c_i , in one local event. So the total time used to design N events is $0.04 \times N$. The MATI was computed according to equation (46), where γ was obtained using MATLAB robust control toolbox. It took 21 seconds to compute the MATI when $N = 150$, 225 seconds when $N = 200$, and 463 seconds when $N = 250$. By our scheme, it only took 6 seconds to design all events when $N = 150$, 8 seconds when $N = 200$, 10 seconds when $N = 250$. Our scheme took much less time in event design than the computation of the MATI when the number of agents is large. It is obvious that the time spent in event design in our scheme scales well with respect to the number of agents.

VII. CONCLUSIONS

This paper examines event-triggered broadcasting of state information in distributed networked control systems with data dropouts and transmission delay. We propose an event-triggering scheme, where a subsystem broadcasts its state information to its neighbors only when the subsystem's local state error exceeds a specified threshold. This scheme is decentralized in a sense that a subsystem's broadcast decisions are made using its local sampled data; a subsystem is able to locally predict its MANSD and deadlines for transmission delay; and the designer's selection of the local event for a subsystem only requires information about that individual agent.

Our analysis applies to both linear and nonlinear subsystems. For nonlinear subsystems, the local event design is transformed into local ISS design problems; for linear subsystems, the design is simplified to be local linear matrix inequality (LMI) feasibility problems. With the assumption that the transmission delay is zero and the number of each agent's successive data dropouts is less than its MANSD, we show that the resulting NCS is finite-gain \mathcal{L}_p stable using our distributed scheme. When the transmission delay is not zero, we provide state-based deadlines that are always greater than a positive constant. As long as the delay in each transmission is less than the associated deadline, we show that the resulting NCS is asymptotically stable, provided the external disturbance vanishes. Simulation results show that our scheme has a good scalability with respect to the system maintenance and the average broadcast period.

These results are significant because they show how one might stabilize distributed control systems over ad hoc net-

works without necessarily requiring a high degree of synchronization within the communication network. They can serve as the basis for the design of firm real-time systems that guarantee network control system performance at levels traditionally seen in hard real-time systems. How to use these results to schedule the broadcasts over the network will be an interesting research direction in the future.

APPENDIX

Proof of Lemma 4.8: Notice that equation (12) and (13) are equivalent to

$$P(A + BK) + (A + BK)^T P + \frac{1}{\gamma^2} P C C^T P \leq -Q \quad (48)$$

$$Q - P B K M^{-1} K^T B^T P \geq W, \quad (49)$$

respectively. Consider \dot{V} with $V(x) = x^T P x$ at time t .

$$\begin{aligned} \dot{V} &= x^T (P A + A^T P) x + 2x^T P B K \hat{x} + 2x^T P C w \\ &\leq -x^T Q x - 2x^T P B K e + \gamma^2 \|w\|_2^2 \\ &\leq -x^T (Q - P B K M^{-1} K^T B^T P) x + e^T M e + \gamma^2 \|w\|_2^2 \\ &\leq -x^T W x + e^T M e + \gamma^2 \|w\|_2^2 \\ &\leq -\sum_{i \in \mathcal{N}} (1 - \rho_i^2) \lambda_{\min}(W_i) \|x_i\|_2^2 + \gamma^2 \|w\|_2^2 \\ &\leq -\min_{i \in \mathcal{N}} \{(1 - \rho_i^2) \lambda_{\min}(W_i)\} \|x\|_2^2 + \gamma^2 \|w\|_2^2 \end{aligned}$$

for any $t \geq t_0$, which is sufficient to show that the NCS in equation (1) is \mathcal{L}_2 stable with an induced gain less than

$$\frac{\gamma}{\sqrt{\min_{i \in \mathcal{N}} \{(1 - \rho_i^2) \lambda_{\min}(W_i)\}}}. \quad \blacksquare$$

Proof of Lemma 5.1: Consider agent i over the time interval $[b_k^i, b_{k+1}^i)$. For notational convenience, we assume $b_k^i = r_0^i < r_1^i < \dots < r_{d_k^i}^i < r_{d_k^i+1}^i = b_{k+1}^i$.

Consider $\|x_i(t) - x_i(b_k^i)\|_2$ for any $t \in [r_j^i, r_{j+1}^i)$. We have

$$\begin{aligned} \|x_i(t) - x_i(b_k^i)\|_2 &= \|x_i(t) - \hat{x}_i(t)\|_2 = \|x_i(t) - x_i(r_0^i)\|_2 \\ &\leq \sum_{l=0}^{j-1} \|x_i(r_{l+1}^i) - x_i(r_l^i)\|_2 + \|x_i(t) - x_i(r_j^i)\|_2 \end{aligned}$$

for $\forall t \in (r_j^i, r_{j+1}^i)$. Applying equation (24) into the preceding equation yields

$$\|x_i(t) - x_i(b_k^i)\|_2 \leq \sum_{l=0}^j \frac{\rho_i}{c_i} \|x_i(r_l^i)\|_2 \quad (50)$$

for all $t \in (r_j^i, r_{j+1}^i)$. Therefore,

$$\|x_i(t) - x_i(b_k^i)\|_2 \leq \sum_{l=0}^{d_k^i} \frac{\rho_i}{c_i} \|x_i(r_l^i)\|_2 \quad (51)$$

holds for all $t \in [b_k^i, b_{k+1}^i)$. Because $\|\varepsilon_i^j(r_{j+1}^i)\|_2 = \|x_i(r_{j+1}^i) - x_i(r_j^i)\|_2 \leq \frac{\rho_i}{c_i} \|x_i(r_j^i)\|_2$, we have

$$\|x_i(r_{j+1}^i)\|_2 \leq (1 + \frac{\rho_i}{c_i}) \|x_i(r_j^i)\|_2$$

and therefore

$$\begin{aligned} \|x_i(r_{j+1}^i)\|_2 &\leq (1 + \frac{\rho_i}{c_i})^{j+1} \|x_i(r_0^i)\|_2 \\ &= (1 + \frac{\rho_i}{c_i})^{j+1} \|x_i(b_k^i)\|_2 \end{aligned} \quad (52)$$

for $j = 0, 1, 2, \dots, d_k^i$. Applying equation (52) into (51) yields

$$\begin{aligned} \|x_i(t) - x_i(b_k^i)\|_2 &\leq \sum_{l=0}^{d_k^i} \frac{\rho_i}{c_i} (1 + \frac{\rho_i}{c_i})^l \|x_i(b_k^i)\|_2 \\ &= \left((1 + \frac{\rho_i}{c_i})^{d_k^i+1} - 1 \right) \|x_i(b_k^i)\|_2 = \frac{\xi_k^i}{c_i} \|x_i(b_k^i)\|_2 \end{aligned}$$

for all $t \in [b_k^i, b_{k+1}^i)$. Also notice that $d_k^i \leq d_{\text{MANS}}^i$ holds according to equation (25). It is easy to show that $\xi_k^i = c_i(1 + \frac{\rho_i}{c_i})^{d_k^i+1} - c_i \leq c_i(1 + \frac{\rho_i}{c_i})^{d_{\text{MANS}}^i+1} - c_i \leq \varrho_i$. ■

Proof of Lemma 5.3: We first consider the behavior of agent i after a successful transmission occurs, say the k th successful transmission of agent i . For notational convenience, we assume $b_{k-1}^i = r_0^i \leq r_1^i \leq \dots \leq r_{d_k^i}^i \leq r_{d_k^i+1}^i = b_k^i$. Consider the derivative of $\left\| \varepsilon_i^{d_k^i+1}(t) \right\|_2 \triangleq \|x_i(t) - x_i(r_{d_k^i+1}^i)\|_2$ over the time interval $[b_k^i, f_k^i)$.

$$\begin{aligned} \frac{d}{dt} \left\| \varepsilon_i^{d_k^i+1}(t) \right\|_2 &\leq \left\| \dot{\varepsilon}_i^{d_k^i+1}(t) \right\|_2 = \|\dot{x}_i(t)\|_2 \\ &= \|f_i(x_{\mathcal{D}_i}, g_i(\hat{x}_{\mathcal{Z}_i}), 0)\|_2 \leq \theta_i \end{aligned}$$

holds for all $t \in [b_k^i, f_k^i)$. Solving the preceding inequality with the initial condition $\left\| \varepsilon_i^{d_k^i+1}(b_k^i) \right\|_2 = 0$ implies

$$\begin{aligned} \left\| \varepsilon_i^{d_k^i+1}(t) \right\|_2 &= \|x_i(t) - x_i(b_k^i)\|_2 \leq \theta_i(t - b_k^i) \\ &\leq \frac{1 - \xi_k^i}{c_i} \max \left\{ \frac{\|x_i(b_{k-1}^i)\|_2}{2}, \Delta \right\} \end{aligned} \quad (53)$$

holds for all $t \in [b_k^i, f_k^i)$, where the inequality on the right side is obtained by applying (30).

Since the hypotheses in Lemma 5.1 are satisfied, we know

$$\begin{aligned} \|x_i(t) - x_i(b_{k-1}^i)\|_2 &\leq \frac{\xi_k^i}{c_i} \|x_i(b_{k-1}^i)\|_2 \\ &\leq \frac{\varrho_i}{c_i} \|x_i(b_{k-1}^i)\|_2 \end{aligned} \quad (54)$$

holds for all $t \in [b_{k-1}^i, b_k^i)$ and therefore,

$$\|x_i(b_k^i) - x_i(b_{k-1}^i)\|_2 \leq \frac{\xi_k^i}{c_i} \|x_i(b_{k-1}^i)\|_2. \quad (55)$$

Combining equation (53) and (55) implies that for $t \in [b_k^i, f_k^i)$,

$$\begin{aligned} &\|x_i(t) - x_i(b_{k-1}^i)\|_2 \quad (56) \\ &\leq \|x_i(t) - x_i(b_k^i)\|_2 + \|x_i(b_k^i) - x_i(b_{k-1}^i)\|_2 \\ &\leq \frac{1 - \xi_k^i}{c_i} \max \left\{ \frac{\|x_i(b_{k-1}^i)\|_2}{2}, \Delta \right\} + \frac{\xi_k^i}{c_i} \|x_i(b_{k-1}^i)\|_2. \end{aligned}$$

Let $\varsigma_k^i = \frac{1 + \xi_k^i}{2}$. Therefore, equation (56), with equation (54), implies

$$\begin{aligned} &\|x_i(t) - x_i(b_{k-1}^i)\|_2 \quad (57) \\ &\leq \max \left\{ \frac{\varsigma_k^i}{c_i} \|x_i(b_{k-1}^i)\|_2, \frac{\Delta(1 - \xi_k^i)}{c_i} + \frac{\xi_k^i}{c_i} \|x_i(b_{k-1}^i)\|_2 \right\} \end{aligned}$$

for all $t \in [b_{k-1}^i, f_k^i)$. Because $0 < \rho_i \leq \xi_k^i \leq \varrho_i < 1$, we know

$$\rho_i \leq \xi_k^i < \varsigma_k^i \leq \frac{1 + \varrho_i}{2} < 1. \quad (58)$$

Equation (57) then suggests that

$$\begin{aligned} \sigma_i \|e_i(t)\|_2 &= \sigma_i \|x_i(t) - x_i(b_{k-1}^i)\|_2 \\ &= (c_i - 1) \|x_i(t) - x_i(b_{k-1}^i)\|_2 \\ &\leq \max \left\{ \varsigma_k^i \|x_i(b_{k-1}^i)\|_2, (1 - \xi_k^i)\Delta + \xi_k^i \|x_i(b_{k-1}^i)\|_2 \right\} \\ &\quad - \|x_i(t) - x_i(b_{k-1}^i)\|_2 \\ &\leq \max \left\{ \varsigma_k^i \|x_i(t)\|_2, \right. \\ &\quad \left. (1 - \xi_k^i)\Delta + \xi_k^i \|x_i(b_{k-1}^i)\|_2 - \xi_k^i \|x_i(t) - x_i(b_{k-1}^i)\|_2 \right\} \\ &\leq \max \left\{ \varsigma_k^i \|x_i(t)\|_2, (1 - \xi_k^i)\Delta + \xi_k^i \|x_i(t)\|_2 \right\} \end{aligned}$$

holds for $t \in [f_{k-1}^i, f_k^i)$, where σ_i is defined in equation (7) and $c_i = 1 + \sigma_i$. Therefore,

$$\sigma_i^p \|e_i(t)\|_2^p \leq \max \left\{ \varsigma_k^{i,p} \|x_i(t)\|_2^p, ((1 - \xi_k^i)\Delta + \xi_k^i \|x_i(t)\|_2)^p \right\}. \quad (59)$$

With the fact that $((1 - \xi_k^i)\Delta + \xi_k^i \|x_i(t)\|_2)^p \leq (1 - \xi_k^i)\Delta^p + \xi_k^i \|x_i(t)\|_2^p$ holds for $p \geq 1$, equation (59) implies

$$\begin{aligned} &\sigma_i^p \|e_i(t)\|_2^p \quad (60) \\ &\leq \max \left\{ \varsigma_k^{i,p} \|x_i(t)\|_2^p, (1 - \xi_k^i)\Delta^p + \xi_k^i \|x_i(t)\|_2^p \right\}. \end{aligned}$$

We now consider \dot{V} for any $t \geq 0$. Equation (5) implies

$$\begin{aligned} \dot{V} &\leq \sum_{i \in \mathcal{N}} [- (\alpha_i - |S_i \cup U_i| \beta_i) \|x_i(t)\|_2^p + \delta_i |\bar{U}_i| \|e_i(t)\|_2^p] \\ &= \sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) [- \|x_i(t)\|_2^p + \sigma_i^p \|e_i(t)\|_2^p]. \end{aligned}$$

Because $\alpha_i - |S_i \cup U_i| \beta_i > 0$ holds, applying equation (60) into the preceding equation yields

$$\begin{aligned} \dot{V} &\leq \sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) [- \|x_i(t)\|_2^p + \\ &\quad \max \left\{ \varsigma_k^{i,p} \|x_i(t)\|_2^p, (1 - \xi_k^i)\Delta^p + \xi_k^i \|x_i(t)\|_2^p \right\}]. \end{aligned}$$

Let

$$\Omega_t = \left\{ i \in \mathcal{N} \mid \varsigma_k^{i,p} \|x_i(t)\|_2^p > (1 - \xi_k^i)\Delta^p + \xi_k^i \|x_i(t)\|_2^p \right\}.$$

Therefore, the preceding equation is equivalent to

$$\begin{aligned} \dot{V} &\leq \sum_{i \in \Omega_t} (\alpha_i - |S_i \cup U_i| \beta_i) (\varsigma_k^{i,p} - 1) \|x_i(t)\|_2^p \\ &\quad + \sum_{i \in \mathcal{N} \setminus \Omega_t} (\alpha_i - |S_i \cup U_i| \beta_i) (1 - \xi_k^i)\Delta^p \\ &\quad + \sum_{i \in \mathcal{N} \setminus \Omega_t} (\alpha_i - |S_i \cup U_i| \beta_i) (\xi_k^i - 1) \|x_i(t)\|_2^p. \end{aligned}$$

Applying equation (58), (33), to the preceding equation

implies

$$\begin{aligned}
\dot{V} &\leq \sum_{i \in \mathcal{N} \setminus \Omega_t} (\alpha_i - |S_i \cup U_i| \beta_i) (1 - \rho_i) \Delta^p \\
&\quad + \sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) (\bar{\varsigma}_i - 1) \|x_i(t)\|_2^p \\
&\leq \sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) (1 - \rho_i) \Delta^p \\
&\quad + \sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) (\bar{\varsigma}_i - 1) \|x_i(t)\|_2^p \\
&\leq \Delta^p \sum_{i \in \mathcal{N}} (\alpha_i - |S_i \cup U_i| \beta_i) (1 - \rho_i) \\
&\quad - \min_i \{(\alpha_i - |S_i \cup U_i| \beta_i) (1 - \bar{\varsigma}_i)\} \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^p \\
&= \left(\Delta^p \pi^p - \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^p \right) \\
&\quad \min_i \{(\alpha_i - |S_i \cup U_i| \beta_i) (1 - \bar{\varsigma}_i)\}, \quad (61)
\end{aligned}$$

where π is defined in equation (32). This inequality means

$$\dot{V} \leq \min_i \{(\alpha_i - |S_i \cup U_i| \beta_i) (1 - \bar{\varsigma}_i)\} \Delta^p (\pi^p - \bar{\pi}^p) < 0, \quad (62)$$

if $\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^p \geq \Delta^p \bar{\pi}^p$.

We know if $1 \leq p \leq q < \infty$, the inequality

$$\left(\sum_{i=1}^N \|x_i\|_2^q \right)^{\frac{1}{q}} \leq \left(\sum_{i=1}^N \|x_i\|_2^p \right)^{\frac{1}{p}} \leq N^{\frac{1}{p} - \frac{1}{q}} \left(\sum_{i=1}^N \|x_i\|_2^q \right)^{\frac{1}{q}} \quad (63)$$

holds. So equation (61) implies that if $p \leq q$, then equation (62) holds when $\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \geq \Delta^q \bar{\pi}^q$. Similarly, if $p \geq q$, then we have equation (62) when $\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \geq N^{1 - \frac{p}{q}} \Delta^q \bar{\pi}^q$. Combining these two cases, equation (62) holds when

$$\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \geq \mu \Delta^q \bar{\pi}^q, \quad (64)$$

where μ is defined in equation (31). By equation (28),

$$\min_{i \in \mathcal{N}} \underline{L}_i \sum_{i \in \mathcal{N}} \|x_i\|_2^q \leq V(x) = \sum_{i \in \mathcal{N}} V_i(x_i) \leq \max_{i \in \mathcal{N}} \bar{L}_i \sum_{i \in \mathcal{N}} \|x_i\|_2^q$$

holds, which, with equation (64), is sufficient to show that there exists $T \geq t_0$, such that $\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \leq \max_{i,j \in \mathcal{N}} \left\{ \frac{\bar{L}_i}{\underline{L}_j} \right\} \mu \bar{\pi}^q \Delta^q$ holds for any $t \geq T$, as shown in [28]. ■

Proof of Lemma 5.5: Consider the set

$$\Gamma = \left\{ x \in \Lambda \mid \sum_{i \in \mathcal{N}} \|x_i\|_2^q \leq \frac{V(x_0)}{\max_{i \in \mathcal{N}} \bar{L}_i} \right\}. \quad (65)$$

According to equation (36), we have

$$\min_{i \in \mathcal{N}} \underline{L}_i \sum_{i \in \mathcal{N}} \|x_i\|_2^q \leq V(x) = \sum_{i \in \mathcal{N}} V_i(x_i) \leq \max_{i \in \mathcal{N}} \bar{L}_i \sum_{i \in \mathcal{N}} \|x_i\|_2^q, \quad (66)$$

which implies $\Gamma \subseteq \Lambda$ and $\max_{i,j \in \mathcal{N}} \frac{\bar{L}_i}{\underline{L}_j} \geq 1$.

We now show that $V(t) \leq V(t_0)$ holds for all $t > t_0$. We prove it by contradiction. Suppose that there is time instant $\hat{t} > t_0$ such that $V(\hat{t}) > V(t_0)$.

Notice that before the first time the inequality in (24) is violated, the inequality

$$\dot{V} \leq \sum_{i \in \mathcal{N}} [-(1 - \rho_i^p) (\alpha_i - |S_i \cup U_i| \beta_i) \|x_i\|_2^p]$$

holds. Therefore, there must exist time instant $\bar{t} > t_0$ such that $V(t) < V(t_0)$ for all $t \in (t_0, \bar{t}]$. Since $V(t)$ is continuous and $V(\hat{t}) > V(t_0)$, we know there must exist at least one time interval $(s - \epsilon_1, s + \epsilon_1) \subset (\bar{t}, \hat{t})$ such that

$$V(s) = V(t_0) \quad (67)$$

$$\dot{V}(t) \geq 0, \quad \forall t \in (s - \epsilon, s). \quad (68)$$

Assume that s is the first time in (t_0, \hat{t}) satisfying equation (67), (68) with a parameter ϵ_1 . Then we have

$$t_0 < \bar{t} < s < \hat{t} \quad (69)$$

$$V(t) \leq V(t_0), \quad \forall t \in [t_0, s). \quad (70)$$

Equation (70) implies

$$x(t) \in \Lambda \quad \text{and} \quad \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \leq \frac{V(t_0)}{\min_{i \in \mathcal{N}} \underline{L}_i} \quad (71)$$

for all $t \in [t_0, s)$ according to equation (66). We know the fact that

$$\frac{(\sum_{i \in \mathcal{N}} \|x_i\|_2)^q}{N^{q-1}} \leq \sum_{i \in \mathcal{N}} \|x_i\|_2^q, \quad \forall q \geq 1, \forall x_i \in \mathbb{R}^n. \quad (72)$$

Equation (70), (71), (72), therefore, suggest that

$$\frac{1}{N^{q-1}} \left(\sum_{i \in \mathcal{N}} \|x_i(t)\|_2 \right)^q \leq \frac{V(t_0)}{\min_{i \in \mathcal{N}} \underline{L}_i}$$

holds for all $t \in [t_0, s)$ and any $q \geq 1$. Combining this inequality with equation (35), we have

$$\begin{aligned}
&f_i(x_{\bar{D}_i}(t), g_i(x_{\bar{Z}_i}(t)), 0) \\
&\leq (1 + |\bar{Z}_i|) L_i N^{\frac{q-1}{q}} \left(\frac{V(t_0)}{\min_{i \in \mathcal{N}} \underline{L}_i} \right)^{\frac{1}{q}} = \theta_i
\end{aligned}$$

for all $t \in [t_0, s)$. Also equation (38) implies

$$f_k^i - b_k^i \leq \max \left\{ \frac{(1 - \xi_k^i)}{2c_i \theta_i} \|x_i(b_{k-1}^i)\|_2, \frac{(1 - \xi_k^i)}{c_i \theta_i} \Delta \right\} \quad (73)$$

with

$$\Delta = \frac{\theta_i}{(1 + |\bar{Z}_i|) L_i \bar{\pi} N^{1 - \frac{1}{\max\{p, q\}}} \max_{i,j \in \mathcal{N}} \left\{ \left(\frac{\bar{L}_i}{\underline{L}_j} \right)^{\frac{1}{q}} \right\}}. \quad (74)$$

Then, following the same reasoning in Lemma 5.3, we have

$$\begin{aligned}
\dot{V}(t) &\leq \left[\Delta^p \pi^p - \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^p \right] \\
&\quad \min_{i \in \mathcal{N}} \{(\alpha_i - |S_i \cup U_i| \beta_i) (1 - \bar{\varsigma}_i)\} \quad (75)
\end{aligned}$$

for all $t \in [t_0, s)$, where $\bar{\varsigma}_i$ is defined in equation (33). Since $\dot{V}(t) \geq 0$ for all $t \in (s - \epsilon_1, s)$, from equation (75), we know

$$\Delta^p \pi^p \geq \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^p, \quad \forall t \in (s - \epsilon_1, s), \quad (76)$$

which implies

$$\mu \Delta^q \pi^q \geq \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q, \quad \forall t \in (s - \epsilon_1, s), \quad (77)$$

where μ is defined in equation (31).

Therefore, implementing equation (74) into the preceding equation implies

$$\mu \Delta^q \pi^q = \frac{V(t_0) \pi^q}{\bar{\pi}^q \max_{i \in \mathcal{N}} \bar{L}_i} \geq \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q, \quad \forall t \in (s - \epsilon_1, s). \quad (78)$$

Since $x(t)$ is continuous, equation (78) implies

$$\frac{V(t_0) \pi^q}{\bar{\pi}^q \max_{i \in \mathcal{N}} \{\bar{L}_i\}} \geq \lim_{t \rightarrow s} \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q = \sum_{i \in \mathcal{N}} \|x_i(s)\|_2^q.$$

Because $\bar{\pi} > \pi$,

$$\frac{V(t_0)}{\max_{i \in \mathcal{N}} \{\bar{L}_i\}} > \frac{V(t_0) \pi^q}{\bar{\pi}^q \max_{i \in \mathcal{N}} \{\bar{L}_i\}} \geq \sum_{i \in \mathcal{N}} \|x_i(s)\|_2^q$$

holds, which implies that

$$V(t_0) > \max_{i \in \mathcal{N}} \{\bar{L}_i\} \sum_{i \in \mathcal{N}} \|x_i(s)\|_2^q \geq V(s).$$

This makes a contradiction with equation (67). Therefore, we conclude that $V(t) \leq V(t_0)$ holds for all $t \geq t_0$. ■

Proof of Theorem 5.6: By Lemma 5.5, we know the state trajectory $x(t) \in \Lambda$ for all $t \in t_0$. Therefore, by equation (36),

$$\begin{aligned} & \min_{i \in \mathcal{N}} \bar{L}_i \frac{(\sum_{i \in \mathcal{N}} \|x_i(t)\|_2)^q}{N^{q-1}} \\ & \leq \min_{i \in \mathcal{N}} \bar{L}_i \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \leq V(t) \leq V(t_0), \quad \forall t \geq t_0 \end{aligned}$$

holds, where the inequality on the left most is obtained using Holder's inequality. Therefore,

$$\sum_{i \in \mathcal{N}} \|x_i(t)\|_2 \leq N^{\frac{q-1}{q}} \left(\frac{V(t_0)}{\min_{i \in \mathcal{N}} \bar{L}_i} \right)^{\frac{1}{q}}, \quad \forall t \geq t_0. \quad (79)$$

According to equation (35), we have

$$\begin{aligned} & f_i(x_{\bar{D}_i}(t), g_i(\hat{x}_{\bar{Z}_i}(t)), 0) \\ & \leq (1 + |\bar{Z}_i|) \bar{L}_i N^{\frac{q-1}{q}} \left(\frac{V(t_0)}{\min_{i \in \mathcal{N}} \bar{L}_i} \right)^{\frac{1}{q}} = \theta_i \end{aligned} \quad (80)$$

for all $t \geq t_0$.

Let $\hat{\pi} = \frac{\pi + \bar{\pi}}{2}$. Since the hypotheses of Lemma 5.3 are satisfied with

$$\Delta := \Delta_1 = \frac{\theta_i}{(1 + |\bar{Z}_i|) \bar{L}_i \bar{\pi} N^{1 - \frac{1}{\max\{p, q\}}} \max_{i, j \in \mathcal{N}} \left\{ \left(\frac{\bar{L}_i}{\bar{L}_j} \right)^{\frac{1}{q}} \right\}},$$

we know that there exists a positive number $t_1 > t_0$, such that

$$\begin{aligned} & \frac{(\sum_{i \in \mathcal{N}} \|x_i(t)\|_2)^q}{N^{q-1}} \leq \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \\ & \leq \max_{i, j \in \mathcal{N}} \left\{ \frac{\bar{L}_i}{\bar{L}_j} \right\} \mu \hat{\pi}^q \Delta_1^q = \left(\frac{\hat{\pi}}{\bar{\pi}} \right)^q \frac{V(t_0)}{\min_{i \in \mathcal{N}} \bar{L}_i}, \quad \forall t \geq t_1, \end{aligned} \quad (81)$$

where μ is defined in equation (31).

By E_2 , we know each agent will broadcast after t_1 . Let s_1 be the time when each agent in \mathcal{N} broadcasts successfully at least once after t_1 . Then we have

$$\frac{(\sum_{i \in \mathcal{N}} \|\hat{x}_i(t)\|_2)^q}{N^{q-1}} \leq \left(\frac{\hat{\pi}}{\bar{\pi}} \right)^q \frac{V(t_0)}{\min_{i \in \mathcal{N}} \bar{L}_i}, \quad \forall t \geq s_1 > t_1. \quad (82)$$

Applying the preceding two equations into equation (35) yields

$$f_i(x_{\bar{D}_i}(t), g_i(\hat{x}_{\bar{Z}_i}(t)), 0) \leq \frac{\hat{\pi}}{\bar{\pi}} \theta_i, \quad \forall t \geq s_1. \quad (83)$$

We now set $\Delta := \Delta_2 = \frac{\hat{\pi}}{\bar{\pi}} \Delta_1$ and use the preceding equation to bound the behavior of f_i over $[s_1, \infty)$. Then Lemma 5.3 suggests that there exists $t_2 \geq s_1$ such that

$$\frac{(\sum_{i \in \mathcal{N}} \|x_i(t)\|_2)^q}{N^{q-1}} \leq \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q$$

$$\leq \max_{i, j \in \mathcal{N}} \left\{ \frac{\bar{L}_i}{\bar{L}_j} \right\} \mu \hat{\pi}^q \Delta_2^q = \left(\frac{\hat{\pi}}{\bar{\pi}} \right)^{2q} \frac{V(t_0)}{\min_{i \in \mathcal{N}} \bar{L}_i}, \quad \forall t \geq t_2.$$

Let s_2 be the time when each agent in \mathcal{N} broadcasts successfully at least once after t_2 . Then we have

$$\frac{1}{N^{q-1}} \left(\sum_{i \in \mathcal{N}} \|\hat{x}_i(t)\|_2 \right)^q \leq \left(\frac{\hat{\pi}}{\bar{\pi}} \right)^{2q} \frac{V(t_0)}{\min_{i \in \mathcal{N}} \bar{L}_i}, \quad \forall t \geq s_2.$$

With the preceding two equations, we can re-compute the bound on f_i over $[s_2, \infty)$ and re-apply Lemma 5.3 to get new bounds on $\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q$ and $\sum_{i \in \mathcal{N}} \|\hat{x}_i(t)\|_2^q$, so on and so forth. Then there exists $s_k > t_0$ such that

$$\begin{aligned} & \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \leq \left(\frac{\hat{\pi}}{\bar{\pi}} \right)^{kq} \frac{V(t_0)}{\min_{i \in \mathcal{N}} \{\bar{L}_i\}} \quad \text{and} \\ & f_i(x_{\bar{D}_i}(t), g_i(\hat{x}_{\bar{Z}_i}(t)), 0) \leq \left(\frac{\hat{\pi}}{\bar{\pi}} \right)^k \theta_i \end{aligned}$$

hold for all $t \geq s_k$. Since $\frac{\hat{\pi}}{\bar{\pi}} \in (0, 1)$, as $k \rightarrow \infty$, the preceding equation implies $x(t) \rightarrow 0$, which means the NCS is asymptotically stable. ■

REFERENCES

- [1] X. Wang and M. Lemmon, "Event-triggered broadcasting across distributed networked control systems," in *American Control Conference*, 2008, pp. 3139–3144.
- [2] —, "Decentralized event-triggering broadcast over networked systems," in *Hybrid Systems: Computation and Control*, 2008, pp. 674–677.
- [3] M. Mazo and P. Tabuada, "On event-triggered and self-triggered control over sensor/actuator networks," in *IEEE Conference on Decision and Control*, 2008, pp. 435–440.
- [4] A. Cervin and T. Henningsson, "Scheduling of event-triggered controllers on a shared network," in *IEEE Conference on Decision and Control*, 2008, pp. 3601–3606.
- [5] P. Wan and M. Lemmon, "An event-triggered distributed primal-dual algorithm for network utility maximization," in *IEEE Conference on Decision and Control*, 2009, pp. 5863–5868.
- [6] K. Shin, "Real-time communications in a computer-controlled workcell," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 1, pp. 105–113, 1991.
- [7] K. Zuberi and K. Shin, "Scheduling messages on controller area network for real-time CIM applications," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 2, pp. 310–316, 1997.
- [8] R. Krtolica, U. Ozguner, H. Chan, H. Goktas, J. Winkelman, and M. Liubakka, "Stability of linear feedback systems with random communication delays," *International Journal of Control*, vol. 59, no. 4, pp. 925–953, 1994.

- [9] W. S. Wong and R. W. Brockett, "Systems with finite communication bandwidth constraints – Part II: Stabilization with limited information feedback," *IEEE Transactions on Automatic Control*, vol. 44, no. 5, pp. 1049–1053, May 1999.
- [10] G. Walsh, H. Ye, and L. Bushnell, "Stability analysis of networked control systems," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 3, pp. 438–446, 2002.
- [11] W. Zhang, M. Branicky, and S. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–99, 2001.
- [12] D. Nedic and A. Teel, "Input-output stability properties of networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, no. 10, pp. 1650–1667, Oct. 2004.
- [13] D. Carnevale, A. R. Teel, and D. Nedic, "Further results on stability of networked control systems: a Lyapunov approach," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 892–897, May 2007.
- [14] P. Tabuada, "Event-Triggered Real-Time Scheduling of Stabilizing Control Tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, Sep. 2007.
- [15] X. Wang and M. Lemmon, "Self-triggered Feedback Control Systems with Finite-Gain \mathcal{L}_2 Stability," *IEEE Transactions on Automatic Control*, vol. 54, no. 3, pp. 452–467, Mar. 2009.
- [16] K. Arzen, "A simple event-based PID controller," in *Proceedings of the 14th IFAC World Congress*, vol. 18, 1999, pp. 423–428.
- [17] D. Hristu-Varsakelis and P. Kumar, "Interrupt-based feedback control over a shared communication medium," in *IEEE Conference on Decision and Control*, 2002, pp. 3223–3228.
- [18] K. Astrom and B. Bernhardsson, "Comparison of Riemann and Lebesgue sampling for first order stochastic systems," in *IEEE Conference on Decision and Control*, 2002, pp. 2011–2016.
- [19] P. Tabuada and X. Wang, "Preliminary results on state-triggered scheduling of stabilizing control tasks," in *IEEE Conference on Decision and Control*, 2006, pp. 282–287.
- [20] M. Lemmon, T. Chantem, X. Hu, and M. Zyskowski, "On self-triggered full information H-infinity controllers," in *Hybrid Systems: computation and control*, 2007, pp. 371–384.
- [21] A. Anta and P. Tabuada, "Self-triggered stabilization of homogeneous control systems," in *American Control Conference*, 2008, pp. 4129–4134.
- [22] J. Nilsson, "Real-time control systems with delays," *Lund, Sweden: Lund Institute of Technology*, 1998.
- [23] Q. Ling and M. Lemmon, "Robust performance of soft real-time networked control systems with data dropouts," in *IEEE Conference on Decision and Control*, 2002, pp. 1225 – 1230.
- [24] —, "Soft real-time scheduling of networked control systems with dropouts governed by a Markov chain," in *American Control Conference*, 2003, pp. 4845–4850.
- [25] —, "Optimal dropout compensation in networked control systems," in *IEEE Conference on Decision and Control*, 2003, pp. 670–675.
- [26] P. Kawka and A. Alleyne, "Stability and performance of packet-based feedback control over a Markov channel," in *American Control Conference*, 2006, pp. 2807–2812.
- [27] X. Chen, K. Tamma, and D. Sha, "Virtual-pulse time integral methodology: A new approach for computational dynamics. Part 2. Theory for nonlinear structural dynamics," *Finite Elements in Analysis & Design*, vol. 20, no. 3, pp. 195–204, 1995.
- [28] H. Khalil, *Nonlinear systems*. Prentice Hall Upper Saddle River, NJ, 2002.



Michael D. Lemmon is a professor of electrical engineering at the University of Notre Dame. He received a B.S. in electrical engineering from Stanford University (1979) and a Ph.D. in electrical and computer engineering from Carnegie-Mellon University (1990). Dr. Lemmon has served as an associate editor for the *IEEE Transactions on Neural Networks* and the *IEEE Transactions on Control Systems Technology*. He chaired the first IEEE working group on hybrid dynamical systems and was the program chair for a hybrid systems workshop in 1997. Most recently he helped forge a consortium of academic, private and public sector partners to build one of the first metropolitan scale sensor-actuator networks (CSOnet) used in monitoring and controlling combined-sewer overflow events. Dr. Lemmon's research deals with real-time networked control systems with an emphasis on understanding the impact that reduced feedback information has on overall system performance. This work has been funded by a variety of state and federal agencies that include the National Science Foundation, Army Research Office, Defense Advanced Research Project Agency, and Indiana's 21st Century Technology Fund.



Xiaofeng Wang was born in Nantong, Jiangsu, China, in 1978. He received B.S. and M.S. in mathematics from East China Normal University in 2000 and 2003, respectively. After one year working at IBM, China, he went to University of Notre Dame and received Ph.D. in electrical engineering in 2009. He is currently a postdoctoral research associate in the Department of Mechanical Science and Engineering at the University of Illinois at Urbana-Champaign.

Dr. Wang's research interests include networked control systems, adaptive control systems, real-time systems, distributed systems, and optimization.