This Dissertation

entitled

Event-Triggering in Cyber-Physical Systems

typeset with NDdiss2$_\varepsilon$ v3.0 (2005/07/27) on June 3, 2009 for

Xiaofeng Wang

This LaTeX 2$_\varepsilon$ classfile conforms to the University of Notre Dame style guidelines established in Spring 2004. However it is still possible to generate a non-conformant document if the instructions in the class file documentation are not followed!

> Be sure to refer to the published Graduate School guidelines at `http://graduateschool.nd.edu` as well. Those guidelines override everything mentioned about formatting in the documentation for this NDdiss2$_\varepsilon$ class file.

It is YOUR responsibility to ensure that the Chapter titles and Table caption titles are put in CAPS LETTERS. This classfile does *NOT* do that!

This page can be disabled by specifying the "`noinfo`*" option to the class invocation.* (i.e.,`\documentclass[...,noinfo]{nddiss2e}`)

This page is *NOT* part of the dissertation/thesis, but MUST be turned in to the proofreader(s) or the reviwer(s)!

NDdiss2$_\varepsilon$ documentation can be found at these locations:

`http://www.gsu.nd.edu`
`http://graduateschool.nd.edu`

Event-Triggering in Cyber-Physical Systems

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy in Electrical Engineering

by

Xiaofeng Wang, Ph.D. Candidate in Electrical Engineering

Michael D. Lemmon, Director

Graduate Program in Department of Electrical Engineering
Notre Dame, Indiana
June 2009

Event-Triggering in Cyber-Physical Systems

Abstract

by

Xiaofeng Wang

Cyber-Physical Systems (CPS) are the systems integrating physical processes with computation and communication. Embedded computers and networks are used to monitor and control these physical processes. A great challenge in implementing such systems is the timing issue. Our work addresses this issue using event-triggering schemes. We first study event/self-triggering in embedded control systems. Both event-triggering and self-triggering schemes are proposed to enforce specified stability concepts of the resulting sampled-data systems. These results provide a solid analytical basis for the development of aperiodic sampled-data control systems. Based on these results, distributed event-triggering is considered in networked control systems with packet loss and transmission delays. A distributed event-triggering scheme is proposed, 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 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. 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.

CONTENTS

FIGURES

CHAPTER 1

Introduction

1.1 Motivation

As digital technologies become popular, more and more digital-based applications are present in real life. In these applications, communication and computation are integrated with physical processes. Specific examples include electrical power grids, transportation networks, advanced automotive systems, process control, environmental monitoring, medical systems, telecommunication, distributed robotics, etc. In recent years, it has been popular to refer to such systems as *Cyber-Physical Systems* (CPS) [28]. In such systems, pervasive networks are used to monitor a large number of local physical processes (also called "subsystem" or "agent") that might be geographically distributed. Based on the information gathered through the networks, the control decisions are made by embedded computers located in physical components.

One important attribute of CPS is "networking". Networking is not only referred to the interconnection between physical processes, but also referred to the interaction in cyber side. One example is distributed networked control systems where subsystems are not only physically coupled together but also share the same network to transmit information. To be considered as CPS, the system should also be deeply embedded in a sense that each subsystem has the capability of computing. It must integrate logic rules/events with timed-feedback control and have high degree of automation.

To better understand CPS, we may compare CPS with real-time control, sensor network, and traditional distributed/decentralized control. In traditional real-time control, embedded systems can be viewed as “closed” units that do not explore the computing capability to the outside, whereas in CPS subsystems are networked and work cooperatively to ensure the performance of the overall system. As to sensor networks, the distributed sensors only have the function of monitoring the environment, whereas in CPS each subsystem has the capability of making control decisions as well as monitoring. A cyber-physical system is also different from traditional distributed/decentralized control in a sense that traditional distributed/decentralized control ignores the timing effect on the system performance. In other words, it is assumed each agent can sample the states and communicate with its neighbors at will. This might be impractical in the real world. In practice, the delays in computation and information transmission always exist due to the limitation of embedded computers and communication media. In CPS this timing issue must be taken into account.

Several interesting questions are raised in the research on CPS:

1. *What are the real-time constraints in CPS such that some desired level of performance can be achieved? How conservative are these constraints?*

   The introduction of real-time networking and the use of embedded processors raise important issues regarding the impact of limited communication and computation resource on the control system’s performance. In practice, it might be impossible to continuously sample the environment. The control decisions are also made in a discrete-time manner because of the use of embedded computers. Moreover, the communication behavior has to be well scheduled due to the limited communication resource. All of these suggest that there must be predictable real-time constraints in CPS.

2. *How can we design CPS that scale well with respect to system cost, performance, and maintenance?*

   As we mentioned before, the number of physical processes in CPS may be huge. Therefore, scalability becomes extremely important in such systems.

3. *How can we design reliable networks to provide real-time guarantee and information flow required by embedded subsystems?*

   Even if the real-time constraints mentioned in the first question are available, we still need to construct a reliable communication network such that those real-time constraints can be enforced. This question focuses more on the communication framework that is out of our scope.

The first question is associated with the problem of task period and deadline selection in networked control systems. We are interested in how frequently subsystems should sample and communicate such that the overall system can achieve a desired level of performance. One approach is to first design the controllers under the assumption of perfect communication and then determine the *maximum allowable transfer interval* (MATI) between two subsequent message transmissions that ensures the stability of the overall system under a network protocol, such as Try-Once-Discard (TOD) or Round-Robin (RR).

The computation of the MATI, however, is often done in a highly centralized manner. It requires extremely detailed models of the overall system. This is impractical for large-scale systems. Moreover, because the MATI is computed before the system is deployed, it must ensure the performance level over all possible system states. As a result, the MATI may be conservative in the sense of being shorter than necessary. Consequently, the network bandwidth has to be higher than necessary to ensure the MATI is not violated. These limitations suggest a great need for alternative well-scaled approaches to address the timing issue in CPS in a way that enables the system to use network bandwidth in a extremely frugal manner as suggested in [9, 10, 21].

Our research is to find such approaches to address the first and second questions simultaneously. In particular, to characterize the real-time constraints in a less conservative way, we use sporadic task models. To ensure good scalability, distributed approaches are considered. Our research, therefore, are separated into two steps: (1) because sporadic task models are still a relatively open area even in real-time control, we first would like to study how to realize sporadic task models in embedded control systems, (2) we try to figure out how to implement such models into CPS in a distributed manner.

Sporadic task models can be realized in two ways. A hardware realization is called event-triggering. Under event-triggering the system states are sampled when some error signal exceeds a given threshold. Event-triggering requires a hardware event detector that may be implemented using application-specific integrated circuit (ASIC) or field-programmable gate array (FPGA) processors. A software realization of sporadic task models is called self-triggering. Under self-triggering the next task release time is predicted by the processing computer based on the sampled data. In reality, one might consider periodic task models as self-triggered tasks since many implementations release tasks upon expiration of a one-shot timer that was started by the previous invocation of the task. Under a periodic task model, the period of this one-shot timer is always a constant value. Self-triggering has a more adaptive form in which the value loaded into the one-shot timer is actually a function of the system state sampled by the current job. Under this "state-based" self-triggering, each task releases its next job based on the system state.

Self-triggering approach may be appropriate when the hardware implementation is unacceptable. But if the cost of using ASIC/FPGA hardware is acceptable, event-triggering uses lower computation and/or communication resource than self-triggering and usually generates longer task periods since self-triggering periods are usually conservative estimates of the periods generated by event-triggering. Both event-triggering and self-triggering can be considered as a closed-loop form of releasing tasks for execution, whereas periodic task models release their jobs in an open-loop fashion. This closed-loop form enables event/self-triggering to generate longer task periods than periodic task models.

Our research starts from event-triggering in embedded control systems. Chapter 2 presents a novel event-triggering scheme for embedded control systems. The approach pertains to nonlinear state-feedback systems. The resulting event-triggered feedback systems are guaranteed to be asymptotically stable. We also show that the task periods and deadlines generated by our scheme are bounded strictly away from zero if the continuous closed-loop systems are input-to-state stable with respect to measurement errors.

Chapter 3 presents a self-triggering scheme for embedded control systems. Other than considering asymptotic stability of the systems, our approach ensures $\mathcal{L}_2$ stability of the resulting self-triggered feedback systems. The results pertain to linear time-invariant systems with state feedback. To our best knowledge, this is the first rigorous examination of what might be required to implement self-triggered feedback control systems. The results in Chapter 2 and Chapter 3 serve as the foundation of our research on distributed event-triggering in CPS.

A distributed event-triggering scheme is presented in Chapter 4. This scheme is used to address the issues raised in NCS, such as broadcast period selection, packet loss and transmission delays. 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 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.

The 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 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 MANSD, 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 will 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.

Chapter 5 raises some problems to be solved in the future. One is the robustness of event-triggered NCSs since Chapter 4 only discusses asymptotic stability. Another problem is to address the impact of quantization on event-triggered NCSs. Promising approaches are provided to solve these two problems.

1.2 Prior Work on Embedded Control Systems

There is a great deal of related work dealing with event/self-triggered feedback systems, sample period selection, and real-time control system co-design. We will review each of these areas in more detail below.

Traditional methods for sample period selection [5] are usually based on Nyquist sampling. Nyquist sampling ensures that the sampled signal can be perfectly reconstructed from its samples. In practice, however, feedback within the control system means the system's performance will be somewhat insensitive to errors in the feedback signal, so that perfect reconstruction is much more than we require in a feedback control system. An alternative approach to the sample period selection problem makes use of Lyapunov techniques. This was done in Zheng et al. [64] for a class of nonlinear sampled-data system. Nesic et al. [38] used input-to-state stability (ISS) techniques to bound the inter-sample behavior of nonlinear systems. $\mathcal{L}_p$ stability of sampled-data systems was considered in [62]. As to networked control systems, upper bounds on the task periods were provided, known as the maximal allowable transfer interval (MATI) [13, 37, 48].

The sampling periods determined by the aforecited methods can be conservative because they are essentially "open-loop". Sample periods are selected before the system is deployed, so this selection must ensure adequate behavior over a wide range of possible input disturbances. As a result, these selected periods may be shorter than necessary. This fact was demonstrated by Tabuada et al. [45] where sampling instants were determined on-line using the current system state. In this case, the average sampling periods of Tabuada's event-triggered scheme appeared to be significantly longer than what one would have chosen using traditional estimates of the MATI.

Another related research direction viewed sample period selection as a "co-design" problem that involves both the control system and the real-time system. In this case, sample periods are selected to minimize some penalty on control system performance subject to a schedulability condition. Early statements of this problem may be found in Seto et al. [41] with more recent studies in [14] and [35]. The penalty function is often a performance index for an infinite horizon optimal control problem. It has, however, been demonstrated [6] that under slow sampling such indices may not be monotone functions of the sampling period. As a result, it only appears to be feasible to do off-line determination of these "optimal" sampling periods. Instead of considering quadratic cost functions, [40] presented an approach to maximize the stability radius subject to a schedulability condition. Although this approach can enlarge the family of stabilizing controllers, it did not provide a direct relationship between the sampling periods and the control performance of the systems.

The prior work on co-design really focuses on optimizing performance subject to scheduling constraints. The scheduling constraints are Liu-Layland [34] schedulability conditions for earliest deadline first (EDF) scheduling. It is not always clear, however, that these are the best set of constraints to be using. Our work actually derives a set of constraints on both the periods and deadlines that we can then use as a quality-of-service (QoS) constraint that the real-time scheduler needs to meet. The schedulability of these QoS constraints is addressed in [17], [46].

In recent years, a number of researchers have proposed aperiodic and sporadic task models in which tasks are event-triggered [3, 44, 53]. By event-triggering, we usually mean that the system state is sampled when some function of the system state exceeds a threshold. The idea of event-triggered feedback has appeared under a variety of names, such as interrupt-based feedback [22], Lebesgue sampling [4], or state-triggered feedback [45]. Event-triggering usually requires some form of hardware event detector to generate a hardware interrupt to release the control task. This can be done using either application-specific integrated circuit (ASIC) or field-programmable gate array (FPGA) processors.

Event triggering provides a useful way of adaptively adjusting task periods at run time, provided the cost associated with using ASIC/FPGA hardware is acceptable. In some applications, however, it may be unreasonable or impractical to retrofit an existing system with such "event detectors". In these cases, a software approach such as a self-triggering scheme may be more appropriate. A self-triggered task model was introduced by Velasco et al. [47] in which a heuristic rule was used to adjust task periods. A self-triggered task model was also introduced by Lemmon et al. [29] which chose task periods based on a Lyapunov-based technique. But the authors did not provide analytic bounds for task periods and the task delays were considered only in the simulation results. The first rigorous examination of what might be required to implement self-triggered feedback control systems was presented in [49, 55]. A space-time scaling law was provided for homogenous systems in [1, 2]. Based on this law, self-triggering schemes were presented.

The prior work in [44] is probably most closely related to our work in Chapter 2. The difference is that in [44] the same Lyapunov function V is required to be always decreasing in both the original closed-loop system (continuously sampling) and the event-triggered feedback system, whereas in Chapter 2 the requirement of always decreasing is relaxed. Temporary increases in V are allowed. It means that we can lengthen the period between events by adopting this less restrictive condition on V in the sampled-data system.

The techniques used in Chapter 3 are similar to the input-to-state stability (ISS) methods used in [13, 37] for bounding the MATI. Compared with the prior work in [29], our result derives explicit bounds on the sampling periods and deadlines, which appear to be tight and computationally efficient. Based on these bounds, we present a practical self-triggering scheme, where the sampling periods and deadlines are uniformly bounded from below by a positive constant. These bounds provide less conservative sampling times than those obtained using the MATI estimates in [37]. Moreover, in many cases the average sampling periods generated by our approach are less conservative than those generated by [44]. Finally, a major contribution of our work in Chapter 3 is an explicit state-dependent bound on the acceptable delay, which is not found in either [13, 37].

1.3 Prior Work on Networked Control Systems

To the best of our knowledge, there is little prior work on the distributed implementation of event-triggering in NCS. Preliminary results [52, 54] proposed decentralized event-triggered feedback schemes for both linear and nonlinear systems, respectively. This work studied asymptotic stability of NCS without considering packet loss and transmission delays. A more complete framework was considered in [50, 51, 56], which includes delays and dropouts. A recent study [36] introduced an event-triggering scheme for sensor-actuator networks. This work, however, adopted a centralized approach to event-design. 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 distributed control, transmission period selection and packet loss in NCS. We will review these areas and discuss their relationship to our distributed event-triggering scheme.

In distributed control, each subsystem uses its state and the states of its immediate neighbors to determine its control action. In [7] it was shown that optimal controllers with a quadratic objective possess an inherent degree of spatial localization. This suggests that it should be possible to effectively regulate the behavior of distributed systems using local interactions between spatially adjacent subsystems. One approach to distributed control builds upon model predictive control [12, 20]. Significant progress was made toward this goal in an approach that modelled system coupling using linear fractional transformations [19] [27]. More recent work has used integrator backstepping to extend this approach to networks of nonlinear systems [23].

One thing worth mentioning is that in all of this prior work, it is assumed that subsystem controllers can communicate with their neighbors at will. In practice, however, communication, especially wireless communication, takes place over a digital network which means that information is transmitted in discrete time rather than continuous-time. Moreover, all real networks have bandwidth limitation that can cause delays in message delivery that may have a major impact on overall system stability [30]. Therefore, the preceding distributed approaches may have the entire systems unstable in real networks.

For this reason, some researchers began investigating the timing issue in networked control. Early work [42, 65] 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 [26, 59, 60] 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 [48]. In this work, several sensors and actuators attempt to access to 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) [48]. For these protocols, an upper bound on the MATI was derived [48] that guarantees asymptotic stability of the system. It led to scheduling methods [63] that were able to assure the MATI was not violated. Further work [13, 37] 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 [44, 49] 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.

Another related research direction is to study packet loss in NCS. In [39], 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 [31], 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 [33] by modelling data dropouts as a Markov chain instead of an i.i.d. process. Optimal dropout compensation for NCS was presented in [32]. A packet-based multi-control strategy was examined in [24] 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. Chapter 4, on the other hand, presents a method by which agents can locally estimate their MANSD. Again, this information may be used in scheduling agent access to the communication medium.

Chapter 4 provides a complete analysis of asynchronous information transmission in NCS using distributed event-triggering. These results are summarized in [50, 51]. 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 [13, 37, 48] is that our work 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.

1.4 Notations

$\mathbb{R}^n$	the n-dimensional Euclidean space
$\mathcal{L}_p$	$\mathcal{L}_p$ space
$\lambda_{\min}(P)$	the minimum singular value of a matrix P
$\lambda_{\max}(P)$	the maximum singular value of a matrix P
$\|x\|$	the norm of a vector x
$\|x\|_p$	the p-norm of a vector x
$\|A\|$	the norm of a matrix A
$\forall$	for all
$\exists$	exists
$\in$	belongs to
$\subset$	subset of
$\rightarrow$	tends to
x^T	the transpose of a vector x
A^T	the transpose of a matrix A
$\lvert s\rvert$	the absolute value of a scaler $s \in \mathbb{R}$
$\lvert S\rvert$	the number of elements in a set S
$L_g h$	the Lie derivative of h with respect to the vector field g
sgn	the signum function
$\mathbb{N}$	the set of natural numbers
$\mathbb{Z}$	the set of integers

CHAPTER 2

Asymptotic Stability in Event-Triggered Feedback Systems

2.1 Problem Formulation

This chapter discusses how to design event-triggering schemes in embedded control systems [53]. The primary goal is to ensure that the resulting event-triggered feedback back scheme is asymptotically stable. The main idea is that one first designs the controllers without the consideration of real-time issues and then derives real-time constraints on the task release and finishing time, with which system stability can still be maintained. This method is so-called **emulation-based** method [37].

Let us first look at how an event-triggered feedback system works. In such a system, as shown in Figure 2.1, a hardware event-detector (HED), either ASIC or FPGA, can monitor the continuous signal $x(t)$. Once an event happens, i.e., a pre-specified logic rule is violated, the detector has the sensor sample the state information. The control signal, u, is computed based on this sampled state. Notice that this computation happens in a discrete-time manner. To hold the control signal continuous, a zero-order hold (ZOH) is placed.

The system considered in this chapter is nonlinear. Let $x : [0, \infty) \rightarrow \mathbb{R}^n$ denote the state trajectory and $u : [0, \infty) \rightarrow \mathbb{R}^m$ denote the control input. The state equation of the sampled-data system is:

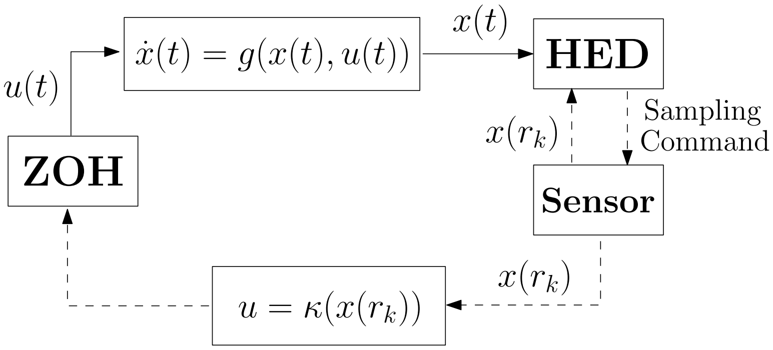

Figure 2.1. An event-triggered feedback system

$$\begin{aligned} \dot{x}(t) &= g(x(t), u(t)) \\ u(t) &= \kappa(x(r_k)) \\ x(0) &= x_0 \end{aligned} \tag{2.1}$$

for all $t \in [f_k, f_{k+1})$ and all $k = 0, \ldots, \infty$, where $x_0 \in \mathbb{R}^n$ is the non-zero initial state, r_k is the time when the kth invocation of a control task (also called "job") is released for execution on the computer and f_k is the time when the kth job has finished executing. In the above equation, $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $\kappa : \mathbb{R}^n \to \mathbb{R}^m$ are locally Lipschitz functions. In particular, we assume that as soon as the system samples, a control task is released. In other words, there is no delay between sampling and task release. To simplify the notation, we define $e_k : \mathbb{R}^+ \to \mathbb{R}$ as $e_k(t) = x(t) - x(r_k)$ for all $t \geq 0$ and all $k \in \mathbb{N}$, which are the measurement errors.

Let $T_k = r_{k+1} - r_k$ denote the kth inter-release time ($k = 0, \ldots, \infty$). T_k can therefore be interpreted as a time-varying "sampling" period by control engineers and a time-varying "task" period by real-time system engineers. We let $\tau_k = f_k - r_k$

denote the time interval between the kth job's release and finishing time. Control engineers would view τ_k as the "delay" of the kth job whereas real-time system engineers would view τ_k as the "jitter" of the kth job. If the control task satisfies a hard real-time constraint, then the delay τ_k is required to lie below a specified "deadline". Figure 2.2 illustrates the relationship between the task period, T_k, the delay τ_k, the task finishing time f_k, and the release time r_k. The x-axis in figure 2.2 is time with the period (T_k), delay (τ_k), finishing time (f_k), and release (r_k) marked on the axis. The black rectangles above the time axis mark intervals over which the task is executing.

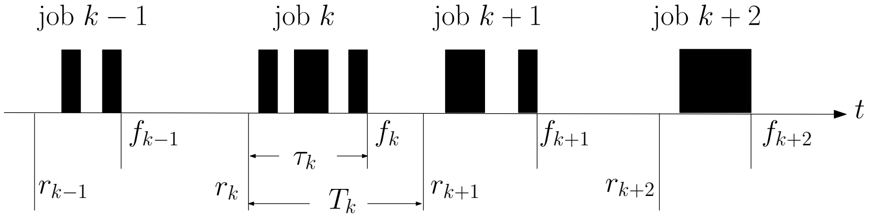

Figure 2.2. Relationship between task period (T_k), delay (τ_k), release time (r_k), and finishing time (f_k)

The problem is to design the triggering event such that the sampling periods can be largely saved; meanwhile, asymptotic stability of the resulting sampled-data system has to be guaranteed. To solve this problem, an emulation-based approach is proposed in [45]. In [45], One is to find Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$

and two class $\mathcal{K}$ functions $\alpha_1, \alpha_2 : \mathbb{R} \to \mathbb{R}$ such that the equation

$$\frac{\partial V(x)}{\partial x} g(x, \kappa(x-e)) \le \alpha_1(\|x\|) + \alpha_2(\|e\|) \tag{2.2}$$

holds for all $x, e \in \mathbb{R}^n$. This inequality implies that the system in equation (2.1) is input-to-state stable (ISS) with respect to the measurement error e. As a result, if the system uses the violation of

$$\delta\alpha_2(\|e_k(t)\|) < \alpha_1(\|x(t)\|), \;\; \delta \in (0,1)$$

to trigger the next sampling release time, the time derivation of V satisfies

$$\dot{V} \le (\delta - 1)\alpha_1(\|x(t)\|),$$

which implies asymptotic stability of the sampled-data system. The main idea [45] is to ensure the Lyapunov function $V(x)$ for continuous closed-loop system is also a Lyapunov function for the sampled-data system. It requires that V is monotonically decreasing in both systems. This is, however, not necessary. As noted in earlier work on switched system stability, we can still guarantee asymptotic stability as long as an appropriate subsequence of V is monotone decreasing [61]. V does not have to be decreasing all the time. It suggests that we might be able to lengthen the period between events by adopting this less restrictive condition on V in the sampled-data system. The event design scheme in this chapter is based on this idea.

The rest of this chapter is organized as follows. Section 2.2 presents an event-triggering scheme without the consideration of delays. The non-zero delay case is considered in Section 2.3. Simulation results are presented in Section 2.4. Finally, conclusions are stated in Section 2.5.

2.2 Event Design in Systems without Task Delay

This section studies the sampled-data systems without task delay ($r_k = f_k$). We introduce a design approach that ensures asymptotic stability of the overall system. To show asymptotic stability of such a system, we only need to show the existence of a piecewise continuous function $h : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^+$ such that

$$h(t|x_0) \geq V(x(t|x_0)), \text{ for all } t \in \mathbb{R}^+ \tag{2.3}$$

$$\lim_{t\to\infty} h(t|x_0) = 0, \tag{2.4}$$

hold, where x is the state trajectories of the resulting event-triggered feedback system and $V \in \mathbb{R}^n \to \mathbb{R}$ is Lyapunov function of the continuous closed-loop system

$$\begin{aligned} \dot{x}(t) &= g(x(t), u(t)) \\ u(t) &= \kappa(x(t)) \\ x(0) &= x_0. \end{aligned} \tag{2.5}$$

Since x is also a function of t, we use $V(t)$ to denote $V(x(t|x_0))$.

Notice that $h(t|x_0)$ does not have to be a class $\mathcal{KL}$ function because it does not have to be continuous. Also, temporary increases in V are allowed as long as V is bounded by $h(t|x_0)$. Accordingly in our event-triggered feedback systems, the derivative of $V(t)$ does not have to be negative all the time.

The results in this section focus on sampled-data systems where $r_k = f_k$ holds for all $k \in \mathbb{N}$. We show that with our event-triggering scheme, the resulting system is asymptotically stable and the sampling period T_k is bounded from below by a positive constant. To show this, we first introduce a lemma, which will be used in the later proofs.

Lemma 2.2.1 *For two $\mathbb{C}^1$ functions $p, q : \mathbb{R}^+ \to \mathbb{R}$, assume $\xi \in \mathbb{R}^+$ is the smallest positive solution to $p(t) = q(t)$. The following statements hold:*

1. *If $p(0) = q(0)$, $\dot{q}(0) < \dot{p}(0)$, and $t^* > 0$ satisfies $p(t^*) \leq q(t^*)$, then $t^* \geq \xi$;*
2. *If $p(0) > q(0)$ and $t^* > 0$ satisfies $p(t^*) \leq q(t^*)$, then $t^* \geq \xi$;*
3. *If $p(0) = q(0)$ and $\dot{q}(0) < \dot{p}(0)$, then $p(t) \geq q(t)$ for all $t \in [0, \xi)$;*
4. *If $p(0) > q(0)$, then $p(t) \geq q(t)$ for all $t \in [0, \xi)$.*

Proof: It can be easily shown taking advantage of the continuity of $p, q, \dot{p}, \dot{q}$. □

To design the event with the guarantee of asymptotic stability of the system, we need an assumption on the system in equation (2.1).

Assumption 2.2.1 *For the sampled-data system in equation* (2.1)*, assume that there exist positive constants $L, \underline{a}, \bar{a}, \underline{b}, \bar{b}, L_1 \in \mathbb{R}^+$, a positive definite, $\mathbb{C}^1$ function $V : \mathbb{R}^n \to \mathbb{R}^+$, and two class $\mathcal{K}$ functions α_1, $\alpha_2 : \mathbb{R}^+ \to \mathbb{R}^+$ such that*

$$\|g(x, \kappa(x+e))\|_2 \leq L\|x\|_2 + L\|e\|_2 \tag{2.6}$$

$$\alpha_1(\|x\|_2) \leq V(x) \leq \alpha_2(\|x\|_2) \tag{2.7}$$

$$-\underline{a}V(x) - \underline{b}\|e\|_2 \leq \frac{\partial V(x)}{\partial x} g(x, \kappa(x-e)) \leq -\bar{a}V(x) + \bar{b}\|e\|_2 \tag{2.8}$$

$$\alpha_1^{-1}(\|x\|_2) \leq L_1\|x\|_2 \tag{2.9}$$

hold for all x, e in a compact set.

Remark 2.2.1 *Equation* (2.8) *implies the continuous system* $\dot{x} = g(x, \kappa(x+e))$ *is ISS with respect to e. It also suggests that the continuous closed-loop system is exponentially stable. More discussion on this assumption can be seen in [45].*

Remark 2.2.2 *For a linear time invariant system*

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ u(t) &= Kx(r_k) \\ x(0) &= x_0, \end{aligned} \tag{2.10}$$

assumption 2.2.1 is satisfied as long as the system defined in equation (2.10) *is asymptotically stable. Assume* $W(x) = x^T Px$ *is Lyapunov function of this LTI system. Then* $V(x) = \sqrt{W(x)} = \sqrt{x^T Px}$ *is also a Lyapunov function. Let* $Q = -P(A+BK) - (A+BK)^T P > 0$. *So we have*

$$L = \max\{\|A+BK\|, \|BK\|\}, \tag{2.11}$$

$$\underline{a} = \frac{1}{2}\lambda_{\max}(P^{-1}Q), \quad \bar{a} = \frac{1}{2}\lambda_{\min}(P^{-1}Q), \tag{2.12}$$

$$\underline{b} = \bar{b} = \|\sqrt{P}BK\|, \quad L_1 = \frac{1}{\sqrt{\lambda_{\min}(P)}}. \tag{2.13}$$

where $\lambda_{\min}(P)$ *denote the minimum singular value of* P.

Theorem 2.2.2 *For the sampled-data system in equation* (2.1)*, let assumption 2.2.1 holds and* $\tau_k = 0$ *for all* $k \in \mathbb{N}$. *If* $r_0 = 0$ *and the* $k+1$*th task release is triggered by the violation of*

$$V(t) \leq -\delta\bar{a}V(r_k)(t - r_k) + V(r_k), \tag{2.14}$$

where $\delta \in (0,1)$*, then the sampled-data system is asymptotically stable and there* *exists a positive constant $\xi > 0$, such that the sample period, T_k, satisfies $T_k \geq \xi$.*

Proof: The proof is in Appendix A.1. □

Remark 2.2.3 *For any $k \in \mathbb{N}$, r_{k+1} is triggered when $V(t)$ intersects the straight line $V(t) = -\delta\bar{a}V(r_k)(t - r_k) + V(r_k)$. This line serves as the threshold. However, using linear functions of t as the thresholds is not the only choice for the threshold. Nonlinear functions can also be used as long as the sequence $\{x(r_k)\}_{k=1}^{\infty}$ converges to zero and $r_{k+1} - r_k > 0$ holds for all $k \in \mathbb{N}$.*

2.3 Event Design in Systems with Task Delay

This section considers event-triggered feedback systems with non-zero delays and show the existence of a non-zero deadline for the delays. The main idea is to use an upper bound of $V(f_k)$ as the starting point of the threshold line. r_{k+1} is still triggered when $V(t)$ intersects the threshold line. To ensure the system stability, we need to properly choose the deadline such that the sequence $\{V(r_k)\}_{k=1}^{\infty}$ converges to zero. In that way, a piecewise continuous $h(t|x_0)$ can be constructed satisfying equation (2.3) and (2.4). This is shown in Figure 2.3, where the horizontal axis is time, the vertical axis is the energy V, the solid curve is the trajectory of $V(t)$, and the dashed lines are the threshold lines.

As we mentioned above, the first step is to find the upper bound for $V(t)$ for $t \in [r_k, f_k)$, which is shown in the following lemma.

Lemma 2.3.1 *For the sampled-data system in equation (2.1), let assumption 2.2.1 hold. For any $k \in \mathbb{N}$, if $V(r_{k-1}) \leq \sigma V(r_k)$ and the delay τ_k satisfies $\tau_k < \min\{\Delta_1, \Delta_2\}$, where $\sigma \in (1, \infty)$ and Δ_1, Δ_2 are the smallest positive so-*

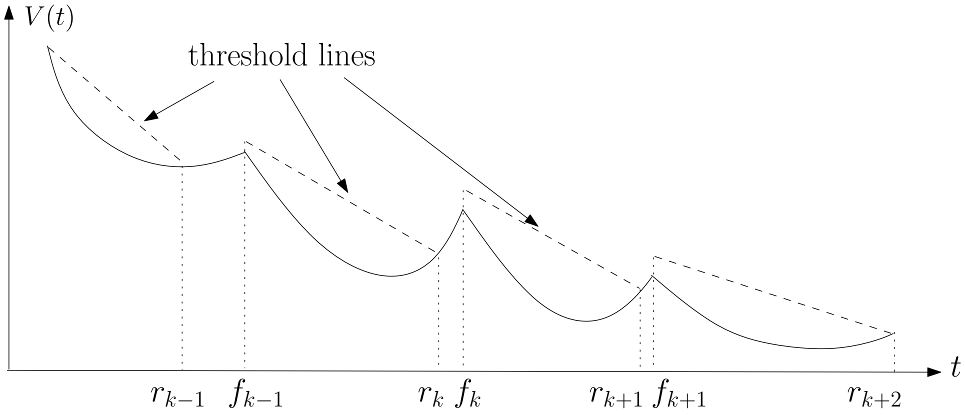

Figure 2.3. The trajectory of V and the threshold lines in event-triggered systems with non-zero delays

lutions to the equation

$$\frac{\bar{b}L_1}{\bar{a}}\left[(1+\frac{3\sigma}{2})e^{2L\Delta_1}-\frac{\sigma}{2}\right]\left(1-e^{-\bar{a}\Delta_1}\right)=\epsilon\Delta_1, \quad (2.15)$$

$$e^{-\underline{a}\Delta_2}+\frac{\underline{b}L_1(1+\sigma)}{\underline{a}}e^{2L\Delta_2}\left(e^{-\underline{a}\Delta_2}-1\right)+\frac{\sigma\underline{b}L_1}{2\underline{a}}\left(e^{2L\Delta_2}-1\right)\left(e^{-\underline{a}\Delta_2}-1\right)=\frac{1}{\sigma}, \quad (2.16)$$

respectively, then $V(t) \leq V(r_k)(1+\epsilon\tau_k)$ *and* $\sigma V(t) \geq V(r_k)$ *hold for all* $t \in [r_k, f_k)$*, where* $\epsilon \in \mathbb{R}^+$ *is a positive constant satisfying* $\epsilon > \bar{b}L_1(1+\sigma)$*.*

Proof: The proof is in Appendix A.2. □

Lemma 2.3.1 shows that $V(t)$ is bounded by a linear function of delays for $t \in [r_k, f_k)$ as long as $\tau_k < \Delta_1$. Based on this lemma, we can use the point $(f_k, V(r_k)(1+\epsilon\Delta))$ as the start of the threshold line with the slope $-\delta\bar{a}V(r_k)$, where $\Delta \leq \Delta_1$ is the deadline for the delays. Asymptotic stability of the event-triggered feedback system is guaranteed by theorem 2.3.3. The proof of Theorem 2.3.3 requires the following lemma, which shows that the time length between the finishing time, f_k, and the time instant when $V(t)$ intersects the threshold line is always bounded from below by a positive function of the deadline.

Lemma 2.3.2 *For the sampled-data system in equation* (2.1)*, let assumption 2.2.1 hold. If* $V(r_{k-1}) \leq \sigma V(r_k)$ *and the delay* τ_k *satisfies* $\tau_k < \Delta = \min\{\Delta_1, \Delta_3\}$*, where* $\sigma > 1$*,* $\Delta \in \mathbb{R}^+$*,* Δ_1 *is given by equation* (2.15)*, and* $\Delta_3 \in \mathbb{R}^+$ *is the smallest positive solution to*

$$\bar{b}L_1 \frac{2+\sigma}{2}\left(e^{2L\Delta_3} - 1\right) - \bar{a}(1 + \epsilon\Delta_3 - \delta) = 0, \tag{2.17}$$

then $t_k^* - f_k \geq \xi(\Delta) > 0$ *holds, where* $t_k^* \geq f_k$ *is the first time when*

$$V(t_k^*) = (1 + \epsilon\Delta)V(r_k) - \delta\bar{a}V(r_k)(t_k^* - f_k). \tag{2.18}$$

holds after f_k*,* $\epsilon > \bar{b}L_1(1+\sigma)$*,* $\xi(\Delta)$ *is the smallest positive solution to*

$$1 + \epsilon\Delta - \delta\bar{a}\pi = (1+\epsilon\Delta)e^{-\bar{a}\pi} - \frac{\eta(\pi,\Delta)\left(e^{-\bar{a}\pi} - 1\right)}{\bar{a}}, \tag{2.19}$$

with respect to π*, and* $g : \mathbb{R}^+ \times \mathbb{R}^+ \to \mathbb{R}^+$ *is defined by*

$$\eta(\pi,\Delta) = \bar{b}L_1\left[\frac{2+\sigma}{2}\left(e^{2L\Delta} - 1\right)e^{2L\pi} + \frac{1}{2}\left(e^{2L\pi} - 1\right)\right]. \tag{2.20}$$

Proof: The proof is in Appendix A.2. □

Theorem 2.3.3 *For the sampled-data system in equation* (2.1)*, let assumption 2.2.1 hold. If* $r_0 = f_0 = 0$ *and for any* $k \in \mathbb{N}$*,*

1. $\tau_k < \Delta = \min\{\Delta_1, \Delta_2, \Delta_3, \Delta_4\}$ *holds, where* $\Delta_1, \Delta_2, \Delta_3 \in \mathbb{R}^+$ *are defined in equation* (2.15)*,* (2.16) *and* (2.17)*, respectively,* $\Delta_4 \in \mathbb{R}^+$ *is the smallest*

positive solution to

$$\epsilon\Delta_4 - \xi(\Delta_4)\delta\bar{a} = 0 \tag{2.21}$$

or ∞ *if the positive solution to equation* (2.21) *does not exist,* ξ *is defined in equation* (2.19), $\epsilon > \bar{b}L_1(1+\sigma)$, $\delta \in (0,1)$, *and* $\sigma \in (1,\infty)$.

2. r_{k+1} *is triggered by the violation of*

$$\left(E_1 \bigwedge E_2\right) \bigvee E_3, \tag{2.22}$$

where

$$E_1 : V(t) \leq (1 + \epsilon\Delta - \delta\bar{a}(t - f_k))V(r_k) \tag{2.23}$$

$$E_2 : \sigma V(t) > V(r_k) \tag{2.24}$$

$$E_3 : r_k \leq t \leq f_k \tag{2.25}$$

then the sampled-data system (2.1) *is asymptotically stable.*

Proof: The proof is in Appendix A.4. □

Remark 2.3.1 *Event* E_2 *in equation* (2.24) *is used to control the distance between* $V(r_{k+1})$ *and* $V(r_k)$. *The reason to do this is that if* $V(r_{k+1})$ *is arbitrarily small, the deadline will go to zero, although the sample period might be enlarged. There is a tradeoff between periods and the predicted deadlines.*

Remark 2.3.2 *Event* E_3 *in equation* (2.25) *means the system does not have to consider the behavior of* $V(t)$ *in* $[r_k, f_k)$. *It is because when* $t \in [r_k, f_k)$, $V(t)$ *will be bounded by* $V(r_k)(1+\epsilon\Delta)$ *anyway.*

Remark 2.3.3 *Theorem 2.3.3 is only used to show the existence of non-zero deadline for the delays. The predicted deadline may be conservative because it is selected before the system is deployed and adequate behavior should be ensured over a wide range of possible input disturbances. An alternative way is to use dynamic deadlines that are computed based on the previous sampled information. This is demonstrated in [49]. How to extend this work with the dynamic deadlines would be addressed in the future.*

2.4 Simulations

In this section, we used the inverted pendulum problem to demonstrate the proposed event-triggered scheme. The plant's linearized state equations were

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -mg/M & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & g/\ell & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1/M \\ 0 \\ -1/(M\ell) \end{bmatrix} u = Ax + Bu$$

where M was the cart mass, m was the mass of the pendulum bob, ℓ was the length of the pendulum arm, and g was gravitational acceleration. For these simulations, we let $M = 10$, $m = 1$, $\ell = 3$, and $g = 10$. The system state was the vector $x = \begin{bmatrix} y & \dot{y} & \theta & \dot{\theta} \end{bmatrix}^T$ where y was the cart's position and θ was the pendulum bob's angle with respect to the vertical. The system's initial state was the vector $x_0 = \begin{bmatrix} 0.98 & 0 & 0.2 & 0 \end{bmatrix}^T$. The controller is $u = Kx$, where $K = \begin{bmatrix} 2 & 12 & 378 & 210 \end{bmatrix}$. The Lyapunov function we used for the continuous closed-loop system is $V(x) = \sqrt{x^T P x}$, where

$$P = \begin{pmatrix} 7 & 21 & 222 & 127 \\ 21 & 106 & 1180 & 675 \\ 222 & 1180 & 26578 & 14873 \\ 127 & 675 & 14873 & 8327 \end{pmatrix} \tag{2.26}$$

We used the violation of equation (2.14) to trigger the sampling. In the first simulation, $\tau_k = 0$ was assumed and $\delta = 0.2$. The top plot of Figure 2.4 shows the state trajectories versus time in the resulting event-triggered feedback system. Obviously, the state converges to zero. The bottom plot of Figure 2.4 is the sample periods (cross) generated in the system versus time. We can see a wide range of variations in periods that have an average of 0.4816. This is mainly because event-triggering can dynamically adjust task periods in response to variations in the system state. Figure 2.5 plots the trajectory of V. As we mentioned before, there are temporary increases in $V(t)$. These increases directly result in longer sample periods, as shown in the next simulation.

In the second simulation, we compared our event-triggering scheme with the event-triggering scheme in [45] and the bound on the MATI in [13] with $\tau_k = 0$. Recall that the event-triggering scheme in [45] samples the state when

$$e_k^T(t) P e_k(t) = b^2 x^T(t) P x(t).$$

where b is the real constant

$$b = \frac{\lambda_{\min}(P)}{2\lambda_{\max}(P)} \frac{\lambda_{\min}(-P(A+BK) - (A+BK)^T P)}{\|PBK\|}.$$

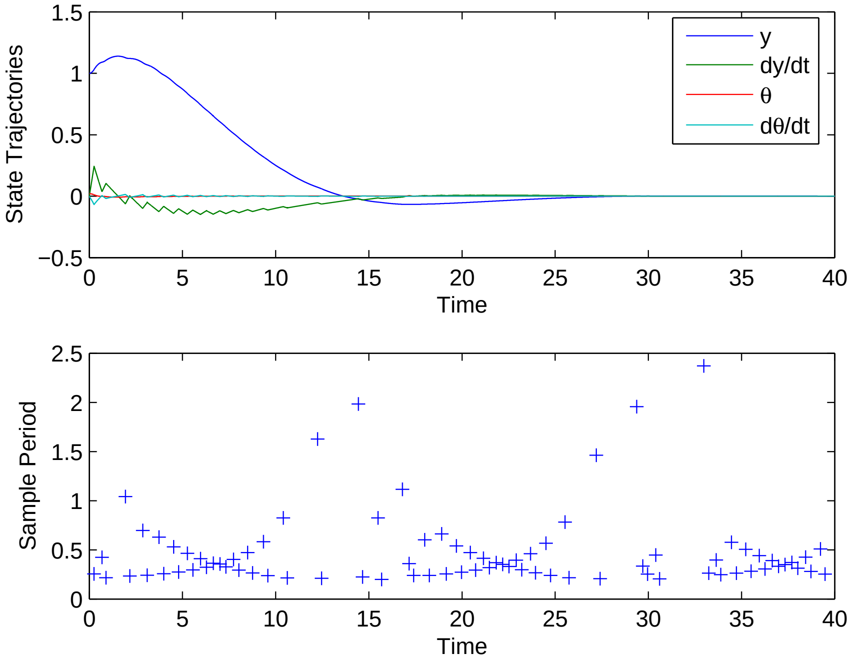

Figure 2.4. An event-triggered feedback system

The bound on the MATI in [13] is defined by

$$T_{\text{MATI}} = \begin{cases} \frac{1}{lr}\arctan\frac{r(1-c)}{2\frac{c}{1+c}\left(\frac{\gamma}{l}-1\right)+1+c} & \gamma > l \\ \frac{1-c}{l(1+c)} & \gamma = l \\ \frac{1}{lr}\text{arctanh}\frac{r(1-c)}{2\frac{c}{1+c}\left(\frac{\gamma}{l}-1\right)+1+c} & \gamma < l \end{cases}, \tag{2.27}$$

where, in this inverted pendulum case, $c = 0$, $l = \max(0.5\lambda_{\max}(-BK - K^T B^T), 0)$, γ is the $\mathcal{L}_2$ gain for the closed-loop system, $\dot{x} = (A + BK)x + BKe$, from e to $-(A + BK)x$, and $r = \sqrt{\left|\frac{\gamma^2}{l^2} - 1\right|}$.

The average periods generated by different schemes are listed in Table 2.1. It is obvious that our event-triggered scheme has a much longer average sample

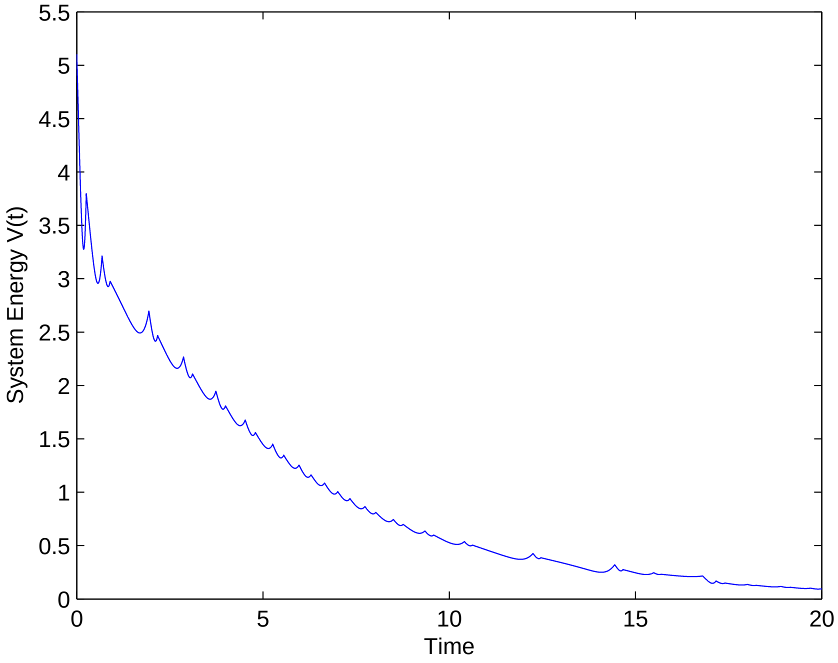

Figure 2.5. The trajectory of system energy versus time for an event-triggered feedback system

period. One thing worth mentioning is that the event-triggering scheme in [45] generates sampling periods less than 10^{-5}. This is even smaller than the bound on the MATI, which is derived based on periodic task models. The reason for this is that the condition number of the particular P matrix is extremely large due to the great difference in the time constants associated with the dynamics of the cart and pendulum bob. Such a matrix leads to a very small b, which limits the size of the sampling periods generated by the approach in [45]. However, for different systems which allow P with a small condition number, the approach in [45] may also generate large sampling periods.

We also examined the robustness of our event-triggered feedback system to the

TABLE 2.1

Comparison of Different Schemes

Schemes	Average Periods
Our event-triggering scheme ($\delta = 0.2$)	0.4816
Event-triggering scheme in [45]	$< 10^{-5}$
The bound on the MATI in [13]	0.0169

external disturbance. The delays were still assumed to be zero and the disturbance was $w(t) = [1, 1, 0.1, 0.1]^T \nu(t)$, where $\nu : \mathbb{R}^+ \rightarrow \mathbb{R}$ is a white noise satisfying $|\nu(t)| \leq 0.5$. The state equations were $\dot{x} = Ax + Bu + w$. The results are plotted in Figure 2.6. The top plot of Figure 2.6 shows the state trajectories of the event-triggered feedback system. The system still converges to a small neighborhood of the equilibrium point. The bottom plot of Figure 2.6 provides the sample periods generated by this system. Although the periods still vary a lot, they are in general much smaller than those in non-disturbance case. The average period is 0.0285. This verifies the ability of event-triggered feedback systems in adjusting sample periods in response to changes in the control system's external inputs. Based on the results of this simulation, our event-triggering scheme appears to be robust to the external disturbance.

Finally, we took a look at non-zero delay cases. The parameters were computed based on equation (2.13) in remark 2.2.2, where $\bar{a} = 0.015$, $\underline{b} = \bar{b} = 1042.8$, $L = 45.58$, $L_1 = 1.91$, $\epsilon = 5993$, $\sigma = 2$, $\delta = 0.8$. The deadline based on Theorem 2.3.3 is around 10^{-12}. This deadline is extremely small, although it is at the same level of the predicted deadline in [45] which is around 10^{-13}. It is because the

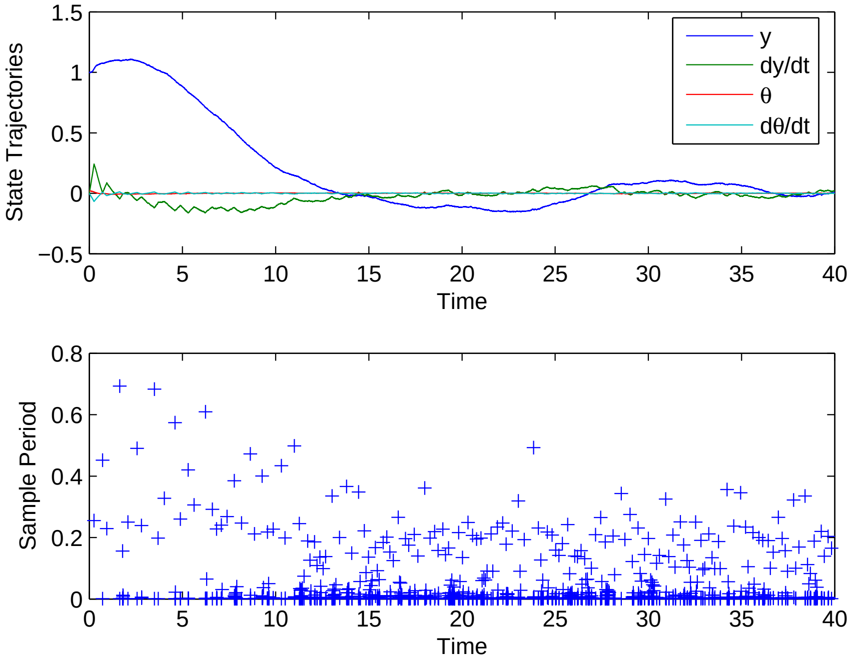

Figure 2.6. An event-triggered feedback system with external disturbance $w(t)$

large condition number of P leads to a small $\bar{a}$ and a big $\bar{b}$, which directly affect the solutions to equation (2.17) and (2.21). Notice that the method we proposed is only for showing the existence of non-zero deadlines. In practice, for systems with a large condition number of P, it is better to use dynamic deadlines because of its "on-line" nature as suggested in Remark 2.3.3.

We then added random delays satisfying $\tau_k \leq 0.1$ into the proposed event-triggered feedback system to see how robust this system can be to delays. We used the violation of equation (2.22) to trigger the next release with $\Delta = 0$. The results are presented in Figure 2.7. The state trajectories are shown in the top plot of Figure 2.7. From this plot, we can see that, the event-triggered feedback system still converges to the equilibrium even when τ_k can be as large as 0.1. The bottom plot of Figure 2.7 provides the sample periods in this system. The average period is 0.1882, which is definitely larger than the periods offered by the prior work. These simulation results suggest that the event-triggered feedback system is robust to delays. How to obtain a tighter estimate of the deadline would be an interesting topic in the future.

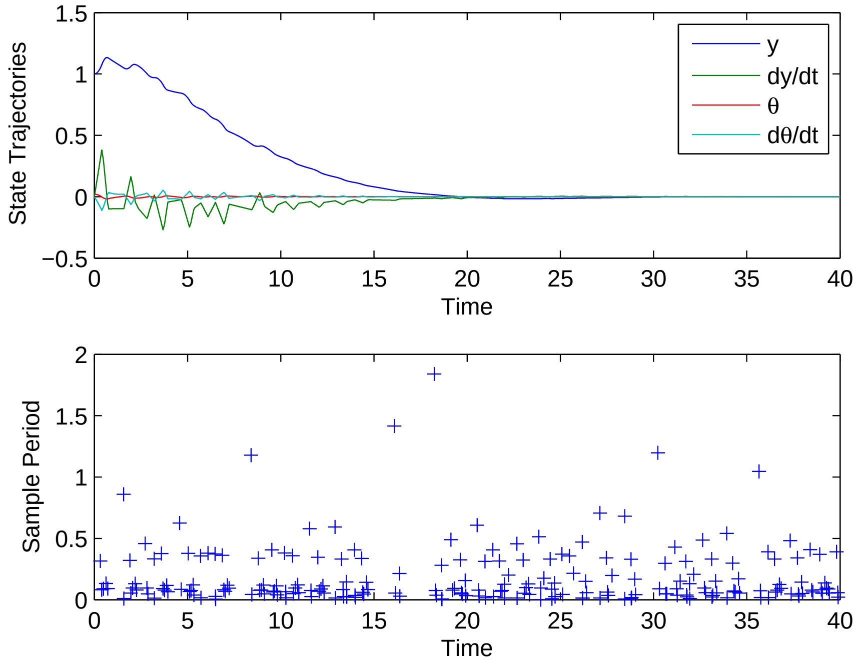

Figure 2.7. An event-triggered feedback system with $\tau_k \leq 0.1$

2.5 Summary

In this chapter, event design in sampled-data systems is discuessed. The resulting event-triggered feedback systems are guaranteed to be asymptotically stable, provided that the continuous systems are stabilizable. We show that the task periods and deadlines generated by our scheme are bounded strictly away from zero if the continuous systems are input-to-state stable with respect to measurement errors. Simulation results indicate that our event-triggered scheme has a much larger average period than the previous event/self-triggered schemes. Moreover, our scheme also appears to be robust to task delays.

CHAPTER 3

Finite-Gain $\mathcal{L}_2$ Stability in Self-Triggered Feedback Systems

3.1 Problem Formulation

This chapter considers stability of self-triggered feedback systems [49, 55]. Under self-triggering the next task release time is predicted by the processing computer based on the previously sampled data. Different from event-triggering that requires hardware detectors, self-triggering is a software implementation of sporadic task models.

To better understand how self-triggering works, let us take a look at a linear time-invariant system,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1u(t) + B_2w(t) \\ u(t) &= Kx(t) \\ x(0) &= x_0 \end{aligned} \tag{3.1}$$

where $x : [0, \infty)$ is the state trajectory, $x_0 \in \mathbb{R}^n$ is the non-zero initial state, $u : [0, \infty) \to \mathbb{R}^m$ is a control input and $w : [0, \infty) \to \mathbb{R}^l$ is an exogenous disturbance function in $\mathcal{L}_2$.

A sampled-data implementation of the closed-loop system in equation (3.1) satisfies the following set of state equations,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1u(t) + B_2w(t) \\ u(t) &= Kx(r_k) \\ x(0) &= x_0 \end{aligned} \tag{3.2}$$

for $t \in [f_k, f_{k+1})$ and all $k = 0, \dots, \infty$. The time r_k denotes the time when the kth invocation of a control task (also called a job) is released for execution on the computer's central processing unit (CPU). The time f_k denotes the time when then kth job has finished executing. At this time, we assume that the system state is sampled so that r_k also represents the kth sampling time instant. In such systems, the plant's control, u, is computed by a computer task. Each job of the control task computes the control u based on the last sampled data. Upon finishing, the control job outputs this control to the plant. The control signal used by the plant is held constant by a zero-order hold (ZOH) until the next finishing time f_{k+1}. We let $T_k = r_{k+1} - r_k$ denote the kth inter-release time (period) and $\tau_k = f_k - r_k$ denote the time interval between the kth job's release and finishing time (delay).

A self-triggering scheme characterizes $\{r_k\}_{i=0}^{\infty}$ and $\{f_k\}_{i=0}^{\infty}$ in the following way. The $k+1$st task release time r_{k+1} is predicted at time f_k such that

$$r_{k+1} = f_k + \eta_2(\{x(r_j)\}_{j=0}^{k}) \tag{3.3}$$

and the $k+1$st task finishing time satisfies

$$f_{k+1} - r_{k+1} \le \Delta(\{x(r_j)\}_{j=0}^{k}). \tag{3.4}$$

The scheme is called *admissible* if $r_k \leq f_k \leq r_{k+1}$ for all $k = 0, 1, \cdots, \infty$.

The key problem in self-triggering design is to find a control gain matrix K and two functions $\eta_2, \Delta : \mathbb{R}^{n(k+1)} \to \mathbb{R}^+$ such that the sampled-data system in equation (3.2) achieves a desired level of performance. One might consider periodic task models as a special class of self-triggered tasks where $\eta_2(\{x(r_j)\}_{j=0}^k)$ and $\Delta(\{x(r_j)\}_{j=0}^k)$ are two fixed constants. Self-triggering has a more adaptive form in which the prediction of task release and finishing time is based on the past information. Under this "state-based" self-triggering, each task releases its next job based on the system state. We can therefore consider "state-based" self-triggering as a closed-loop form of releasing tasks for execution, whereas periodic task models release their jobs in an open-loop fashion.

Since we are interested in finite-gain $\mathcal{L}_2$ stability of the self-triggered feedback system, we assume there exists a full-information $\mathcal{H}_\infty$ controller for the system in equation (3.1). In particular, we assume there exists a symmetric positive semi-definite matrix P satisfies the $\mathcal{H}_\infty$ algebraic Riccati equation (ARE) [8, p. 138],

$$0 = PA + A^T P - Q + R \tag{3.5}$$

where

$$\begin{aligned} Q &= PB_1B_1^TP && (3.6) \\ R &= I + \frac{1}{\gamma^2}PB_2B_2^TP && (3.7) \end{aligned}$$

for some real constant $\gamma > 0$. The state feedback gain matrix in equation (3.1) is

$$K = -B_1^T P. \tag{3.8}$$

The sampled-data system, therefore, satisfies

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_1 u(t) + B_2 w(t) \\ u(t) &= -B_1^T P x(r_k) \\ x(0) &= x_0. \end{aligned} \tag{3.9}$$

For notational convenience, let $A_{cl} = A - B_1 B_1^T P$.

Definition 3.1.1 *The system in equation* (3.9) *is said to be finite-gain $\mathcal{L}_2$ stable from w to x with an induced gain less than γ if there exist non-negative constants γ and ξ such that*

$$\left(\int_{t_0}^{\infty} \|x(t)\|_2^2 dt \right)^{\frac{1}{2}} \leq \gamma \left(\int_{t_0}^{\infty} \|w(t)\|_2^2 dt \right)^{\frac{1}{2}} + \xi \tag{3.10}$$

for any $w \in \mathcal{L}_2$.

In the following discussion, we will present an admissible self-triggering scheme which ensures finite-gain $\mathcal{L}_2$ stability of the sampled-data system in equation (3.9) from w to x.

The remainder of this chapter is organized as follows. Section 3.2 derives a sufficient threshold condition that can serve as an event trigger for state sampling. Section 3.3 presents a self-triggering scheme and proves that it is $\mathcal{L}_2$ stable. Simulations are shown in section 3.4. Finally, conclusions are presented in section 3.5.

Consider the sampled-data system in equation (3.9) with a set of admissible release and finishing time sequences. For all k, define the kth job's error function $e_k : \mathbb{R}^+ \to \mathbb{R}^n$ by $e_k(t) = x(t) - x(r_k)$. This error represents the difference between the current system state and the system state at the last release time, r_k. This section presents two inequality constraints on $e_k(t)$ (see Theorem 3.2.1 and Corollary 3.2.2 below) whose satisfaction is sufficient to ensure that the sampled-data system's $\mathcal{L}_2$ gain is less than γ/β for some parameter $\beta \in (0,1]$.

The following theorem states that if a function of the state error $e_k(t)$ and state $x(t)$ satisfies a certain inequality constraint, then the closed-loop system in equation (3.9) is finite-gain $\mathcal{L}_2$ stable.

Theorem 3.2.1 *Consider the sampled-data system in equation* (3.9) *with admissible release and finishing time sequences. Let* $r_0 = 0$ *and* β *be any real constant in the interval* $(0,1]$ *with the matrix* Q *as given in equation* (3.6)*. If*

$$e_k^T(t)Qe_k(t) \le (1-\beta^2)\|x(t)\|_2^2 + x^T(r_k)Qx(r_k) \tag{3.11}$$

holds for all $t \in [f_k, f_{k+1})$ *and any* $k = 0, \ldots, \infty$*, then the sampled-data system is finite-gain* $\mathcal{L}_2$ *stable from* w *to* x *with a gain less than* γ/β*.*

Proof: The proof is in Appendix A.5. □

In our following work, we will find it convenient to use a slightly weaker sufficient condition for $\mathcal{L}_2$ stability which is only a function of the state error $e_k(t)$. The following corollary states this result.

Corollary 3.2.2 *Consider the sampled-data system in equation* (3.9) *with admissible sequences of release and finishing times. Let* $x(r_0) = x_0$ *and* Q *be a real* *matrix that satisfies equation* (3.6)*. For any* $\beta \in (0,1]$*, let*

$$M = (1-\beta^2)I + Q \tag{3.12}$$

$$N = \frac{1}{2}(1-\beta^2)I + Q \tag{3.13}$$

If the state error trajectory satisfies

$$e_k(t)^T M e_k(t) \leq x^T(r_k) N x(r_k) \tag{3.14}$$

for $t \in [f_k, f_{k+1})$ *and all* $k = 0, \ldots, \infty$*, then the sampled data system is finite-gain* $\mathcal{L}_2$ *stable from* w *to* x *with a gain less than* γ/β*.*

Proof: The proof is in Appendix A.6. □

Remark 3.2.1 *The inequalities in equations* (3.11) *or* (3.14) *can both be used as the basis for an event-triggered feedback control system. Note that both inequalities are trivially satisfied at* $t = r_k$*. If we let the delay,* τ_k*, be zero for each job, then by triggering the release times* $\{r_k\}_{k=0}^{\infty}$ *anytime before the inequalities in equations* (3.11) *or* (3.14) *are violated, we will ensure the sampled-data system's induced* $\mathcal{L}_2$ *gain remains below* γ/β*. The resulting event-triggered feedback system is very similar to the state-triggering scheme proposed by Tabuada et al. [45] for asymptotic stability. The main difference between that result and this one is that our proposed event-triggering condition provides a stronger assurance on the sampled-data system's performance as measured by its induced* $\mathcal{L}_2$ *gain.*

3.3 Self-Triggered Feedback Schemes

This section establishes sufficient conditions for the existence of admissible sequences of release and finishing times that ensure the sampled data system in equation (3.9) is finite-gain $\mathcal{L}_2$ stable with a specified gain. These conditions take the form of admissible bounds on the task sampling periods, T_k, and task delays, τ_k. Based on these bounds, we present a self-triggered scheme, where the sampling periods and deadlines are uniformly bounded from below by a positive constant. The following assumption is placed on the disturbance $w(t)$ to ensure these bounds are nonzero.

Assumption 3.3.1 *Consider the sampled-data system in equation* (3.9). *Assume that there exists a positive real constant $a > 0$ so that $\|w(t)\|_2 \leq a\|x(t)\|_2$ for all $t \geq 0$.*

Remark 3.3.1 *Assumption 3.3.1 consists of a restricted class of signals whose norm is bounded by a linear function of the state's norm. The more precise way to state this assumption is $\|w(t, x(t))\|_2 \leq a\|x(t)\|_2$ for all $t \geq 0$, which means that w depends on x as well as t. But, to make the notation consistent, we still use $w(t)$ to denote the disturbance instead of $w(t, x)$. Such disturbances may arise in uncertain systems when there are unmodeled dynamics caused by fluctuations in plant parameters. To cover a wider class of disturbances, this assumption is relaxed in our recent work [58].*

For notational convenience let $z_k : [r_k, f_{k+1}) \to \mathbb{R}^n$ be given as

$$z_k(t) \quad = \quad \sqrt{(1-\beta^2)I + Q}e_k(t) = \sqrt{M}e_k(t) \tag{3.15}$$

where M is defined in equation (3.12) and $\sqrt{M}$ is the matrix square root of M. We refer to z_k as the kth job's "trigger signal". Note that M is dependent on the parameter β. In the following discussion, we assume M has full rank by properly choosing β. It also implies that $\sqrt{M}$ has full rank. Notice that $M \geq N$ always holds and, if M has full rank, M, N will be both positive definite, where N is defined in equation (3.13).

We define the function $\psi : \mathbb{R}^n \to \mathbb{R}$ given by

$$\psi(x) = \sqrt{x^T N x} \tag{3.16}$$

where $x \in \mathbb{R}^n$. So if we can guarantee for any $\delta \in (0, 1]$ that

$$\|z_k(t)\|_2 \;\; \leq \;\; \delta\psi(x(r_k)) \tag{3.17}$$

for all $t \in [f_k, f_{k+1})$ for any $k = 0, \ldots, \infty$, then the hypotheses in Corollary 3.2.2 are satisfied and we can conclude that the sampled-data system is finite-gain $\mathcal{L}_2$ stable from w to x with a gain less than γ/β.

The first major result examines what happens if we use equation (3.17) as the basis for an event-triggered feedback control system. In particular, let us assume that the kth job's release, r_k, is precisely that time when $\|z_k(t)\|_2 = \delta\psi(x(r_k))$ under the assumption that the kth job's delay, τ_k, is zero. The following theorem states a lower bound on the sampling period for which a sampled-data system with zero delay (i.e. $\tau_k = 0$) has an induced $\mathcal{L}_2$ gain less than γ/β.

Theorem 3.3.1 *Consider the sampled-data system in equation* (3.9) *satisfying assumption 3.3.1. Assume that M has full rank and for some $\delta \in (0, 1]$ that the* *sequence of release times* $\{r_k\}_{k=0}^{\infty}$ *satisfy*

$$\|z(r_{k+1})\|_2 = \delta\psi(x(r_k)) \tag{3.18}$$

where $f_k = r_k$ *for all* $k = 0, \ldots, \infty$.

The sequence of release and finishing times is admissible and the sampled-data system is finite-gain $\mathcal{L}_2$ *stable from* w *to* x *with a gain less than* γ/β. *Furthermore, the task sampling periods satisfy*

$$T_k \geq \frac{1}{\sigma}\ln\left(1 + \delta\sigma\frac{\psi(x(r_k))}{\mu_0(x(r_k))}\right) \tag{3.19}$$

where σ *is a real constant*

$$\sigma = \left\|\sqrt{M}A\sqrt{M}^{-1}\right\| + a\left\|\sqrt{M}B_2\right\|\left\|\sqrt{M}^{-1}\right\| \tag{3.20}$$

and $\mu_0 : \mathbb{R}^n \to \mathbb{R}$ *is a real-valued function given by*

$$\mu_0(x(r_k)) = \left\|\sqrt{M}A_{\mathrm{cl}}x(r_k)\right\|_2 + a\left\|\sqrt{M}B_2\right\|\|x(r_k)\|_2 . \tag{3.21}$$

Proof: The proof is in Appendix A.7. □

Remark 3.3.2 *Note that the righthand side of equation* (3.19) *will always be strictly greater than zero. We can therefore conclude that if we trigger release times when* $\delta\psi(x(r_k)) = \|z_k(r_{k+1})\|$, *then the sampling period* T_k *can never be zero.*

Remark 3.3.3 *The admissibility of sequences* $\{r_k\}_{k=0}^{\infty}$ *and* $\{f_k\}_{k=0}^{\infty}$ *can be restated in terms of the sequences* $\{\tau_k\}_{k=0}^{\infty}$ *and* $\{T_k\}_{k=0}^{\infty}$. *By definition, the release and* *finishing time sequences are admissible if and only if $r_k \leq f_k \leq r_{k+1}$ for all k. Clearly this holds if and only if $0 \leq \tau_k \leq T_k$ for all k.*

The previous theorem presumes there is no task delay (i.e. $\tau_k = 0$). Under this assumption, Theorem 3.3.1 states that triggering release times when equation (3.18) holds assures the closed-loop system's induced $\mathcal{L}_2$ gain. This theorem, however, also provides a lower bound on the task sampling period, which suggests that we can also use theorem 3.3.1 as the basis for state-based self-triggered feedback. In this scenario, if the kth job would set the next job's release time as

$$r_{k+1} = r_k + \frac{1}{\sigma} \ln \left(1 + \delta\sigma \frac{\psi(x(r_k))}{\mu_0(x(r_k))} \right) \tag{3.22}$$

then we are again assured that the system's induced $\mathcal{L}_2$ gain is less than γ/β.

The problem faced in using equation (3.22) for self-triggering is the assumption of no task delay. In many applications, the task delay may not be small enough to neglect. If we consider non-zero delay, then the triggering signals appear as shown in Figure 3.1. This figure shows the time history for the triggering signals, z_{k-1}, z_k, and z_{k+1}. With non-zero delay, we can partition the time interval $[r_k, f_{k+1})$ into two subintervals $[r_k, f_k)$ and $[f_k, f_{k+1})$. The differential equations associated with subintervals $[r_k, f_k)$ and $[f_k, f_{k+1})$ are

$$\begin{aligned} \dot{x}(t) &= Ax(t) - B_1 B_1^T P x(r_{k-1}) + B_2 w(t) \quad \text{and} \\ \dot{x}(t) &= Ax(t) - B_1 B_1^T P x(r_k) + B_2 w(t), \end{aligned}$$

respectively. In a manner similar to the proof of theorem 3.3.1, we can use differential inequalities to bound $z_k(t)$ for all $t \in [r_k, f_{k+1})$ and thereby determine sufficient conditions assuring the admissibility of the release/finishing times while preserving the closed-loop system's $\mathcal{L}_2$-stability. The next two lemmas (Lemma 3.3.2 and 3.3.3) characterize the behavior of $z_k(t)$ over these two subintervals. We then use Lemma 3.3.3 to establish sufficient conditions assuring the $\mathcal{L}_2$-stability of the sampled-data system with non-zero delay. The proofs of these lemmas have been moved to the appendix.

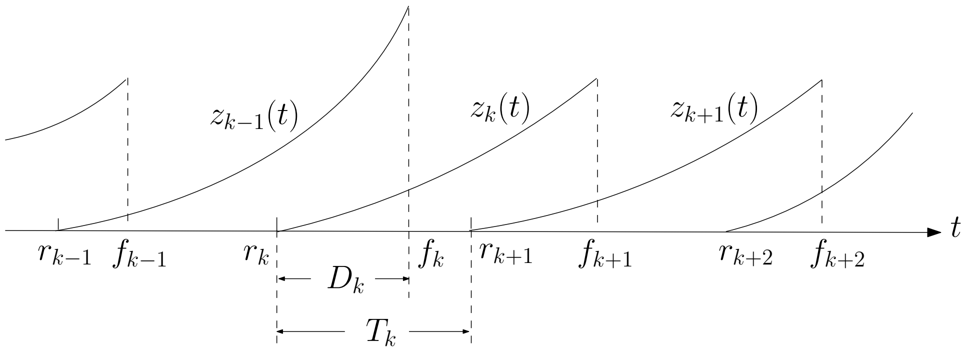

Figure 3.1. Time history of $z_k(t)$ with non-zero task delay.

Lemma 3.3.2 *Consider the sampled-data system in equation* (3.9) *satisfying assumption 3.3.1. Assume that M has full rank and for some k, $r_{k-1} \leq f_{k-1} \leq r_k$. Given some $\epsilon \in (0,1)$, let $\eta_1 : \mathbb{R}^n \times \mathbb{R}^n \times (0,1) \to \mathbb{R}$, $\phi : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, and*

$\mu_1 : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ *be real-valued functions given by*

$$\begin{aligned}
\eta_1(x(r_k), x(r_{k-1}); \epsilon) &= \frac{1}{\sigma} \ln \left(1 + \frac{\epsilon \sigma \psi(x(r_k))}{\mu_1(x(r_k), x(r_{k-1}))} \right), && (3.23) \\
\phi(x(r_k), x(r_{k-1}); t - r_k) &= \frac{\mu_1(x(r_k), x(r_{k-1}))}{\sigma} \left(e^{\sigma(t-r_k)} - 1 \right), \ \textit{and} && (3.24) \\
\mu_1(x(r_k), x(r_{k-1})) &= a \left\| \sqrt{M} B_2 \right\| \|x(r_k)\|_2 \\
&\quad + \left\| \sqrt{M} \left(Ax(r_k) - B_1 B_1^T P x(r_{k-1}) \right) \right\|_2, && (3.25)
\end{aligned}$$

respectively, where σ *is a positive real constant given by equation* (3.20) *and* $\psi : \mathbb{R}^n \to \mathbb{R}$ *is given by equation* (3.16)*. If the kth finishing time* f_k *satisfies*

$$0 \leq \tau_k = f_k - r_k \leq \eta_1(x(r_k), x(r_{k-1}); \epsilon) \tag{3.26}$$

for all $t \in [r_k, f_k)$*, then the kth trigger signal,* z_k*, satisfies*

$$\|z_k(t)\|_2 \leq \phi(x(r_k), x(r_{k-1}); t - r_k) \leq \epsilon \psi(x(r_k)) \tag{3.27}$$

for all $t \in [r_k, f_k)$*.*

Proof: The proof is in Appendix A.8. □

Remark 3.3.4 *In Lemma 3.3.2,* $\eta_1(x(r_k), x(r_{k-1}); \epsilon)$ *serves as a bound on the maximal allowable delay. Since we are considering non-zero delays, we would like* $\eta_1(x(r_k), x(r_{k-1}); \epsilon)$ *to be strictly away from zero. To ensure that, we need to guarantee* $\frac{\|x(r_{k-1})\|_2}{\|x(r_k)\|_2}$ *is bounded by a positive constant from above. The upper bound of* $\frac{\|x(r_{k-1})\|_2}{\|x(r_k)\|_2}$ *can be obtained provided that* $\|z_{k-1}(r_k)\|_2 \leq \delta \psi(x(r_{k-1}))$ *and*

$x(r_{k-1}) \neq 0$ hold. This is because

$$\begin{aligned} 0 &\geq \|z_{k-1}(r_k)\|_2^2 - \delta^2\psi(x(r_{k-1}))^2 \\ &= x(r_k)^T M x(r_k) - 2x(r_k)^T M x(r_{k-1}) + x(r_{k-1})^T (M - \delta^2 N) x(r_{k-1}) \\ &\geq -2\|x(r_k)\|_2 \, \|M\| \, \|x(r_{k-1})\|_2 + \lambda_{\min}(M - \delta^2 N)\|x(r_{k-1})\|_2^2 \end{aligned}$$

Because $\|x(r_{k-1})\|_2 > 0$, the inequality above implies

$$0 \geq -2\|x(r_k)\|_2 \, \|M\| + \lambda_{\min}(M - \delta^2 N)\|x(r_{k-1})\|_2,$$

which means

$$\frac{\|x(r_{k-1})\|_2}{\|x(r_k)\|_2} \leq \frac{2\|M\|}{\lambda_{\min}(M - \delta^2 N)}$$

since $M \geq N > \delta^2 N > 0$. With the fact that $\frac{\|x(r_{k-1})\|_2}{\|x(r_k)\|_2}$ is bounded by a positive constant from above, it is easy to show that $\eta_1(x(r_k), x(r_{k-1}); \epsilon)$ is greater than a positive constant, if $\|z_{k-1}(r_k)\|_2 \leq \delta\psi(x(r_{k-1}))$ holds.

Lemma 3.3.3 *Consider the sampled-data system in equation (3.9) satisfying assumption 3.3.1. Assume M has full rank. For a given integer k and some $\epsilon \in (0,1)$, assume that $r_{k-1} \leq f_{k-1} \leq r_k$. For any $\pi \in (\epsilon, 1]$, let*

$$d_\pi = f_k + \eta_2(x(r_k), x(r_{k-1}); \tau_k, \pi), \tag{3.28}$$

where $\eta_2 : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \times (0,1] \to \mathbb{R}$ is given by

$$\eta_2(x(r_k), x(r_{k-1}); \tau_k, \pi) = \frac{1}{\sigma} \ln \left(1 + \sigma \frac{\pi\psi(x(r_k)) - \phi(x(r_k), x(r_{k-1}); \tau_k)}{\mu_0(x(r_k)) + \sigma\phi(x(r_k), x(r_{k-1}); \tau_k)} \right). \tag{3.29}$$

if

$$0 \leq \tau_k \leq \eta_1(x(r_k), x(r_{k-1}); \epsilon) \tag{3.30}$$

then

$$d_\pi > f_k \text{ and} \tag{3.31}$$

$$\|z_k(t)\|_2 \leq \pi\psi(x(r_k)) \text{ for all } t \in [f_k, d_\pi]. \tag{3.32}$$

Proof: The proof is in Appendix A.9. □

According to Lemma 3.3.3, for a positive constant $\delta \in (\epsilon, 1)$, if $r_{k+1} = f_k + \eta_2(x(r_k), x(r_{k-1}); \tau_k, \delta)$ and $f_{k+1} \leq f_k + \eta_2(x(r_k), x(r_{k-1}); \tau_k, 1)$ hold, we will always have $\|z_k(r_{k+1})\|_2 \leq \delta\psi(x(r_k))$ and $\|z_k(f_{k+1})\|_2 \leq \psi(x(r_k))$. We will use this fact below to characterize a self-triggering scheme that preserves the sampled-data system induced $\mathcal{L}_2$ gain. Theorem 3.3.5 formally states this self-triggering scheme. The proof of theorem 3.3.5 requires the following lemma which shows that the upper bound for delays given in Lemma 3.3.2 is bounded below by a positive function of $x(r_{k-1})$. In that case, the deadline for τ_k can be predicted at time r_{k-1}. The proof of this lemma will be found in the appendix.

Lemma 3.3.4 *Consider the sampled-data system in equation* (3.9) *satisfying assumption 3.3.1. Assume that M has full rank and for a constant $\delta \in (0, 1)$, the release time r_{k-1} and r_k satisfy*

$$\|z_{k-1}(r_k)\|_2 \leq \delta\psi(x(r_{k-1})) \tag{3.33}$$

for any given k. Then η_1 given by equation (3.23) *satisfies*

$$\eta_1(x(r_k), x(r_{k-1}); \epsilon) \geq \Delta(x(r_{k-1}); \epsilon, \delta) > 0, \tag{3.34}$$

where $\epsilon \in (0, \delta)$ and $\Delta : \mathbb{R}^n \times (0,1) \times (0,1) \to \mathbb{R}$ is a real-valued function given by

$$\Delta(x(r_{k-1}); \epsilon, \delta) = \frac{1}{\sigma} \ln \left(1 + \frac{\epsilon(1-\delta)\psi(x(r_{k-1}))}{\delta\psi(x(r_{k-1})) + \mu_0(x(r_{k-1}))/\sigma} \right). \tag{3.35}$$

Proof: The proof is in Appendix A.10. □

With the preceding technical lemma we can now state a self-triggered feedback scheme which can guarantee the sampled-data system's induced $\mathcal{L}_2$ gain. The basis for this self-triggering scheme will be found in the following theorem.

Theorem 3.3.5 *Consider the sampled-data system in equation* (3.9) *satisfying assumption 3.3.1. Assume M has full rank. For given $\epsilon \in (0,1)$ and $\delta \in (\epsilon, 1)$, we assume that*

- *The initial release and finishing times satisfy*

$$r_{-1} = r_0 = f_0 = 0$$

- *For any non-negative integer k, the release times are generated by the following recursion,*

$$r_{k+1} \quad = \quad f_k + \eta_2(x(r_k), x(r_{k-1}); \tau_k, \delta) \tag{3.36}$$

and the finishing times satisfy

$$r_{k+1} \leq f_{k+1} \leq r_{k+1} + \Delta(x(r_k); \epsilon, \delta), \tag{3.37}$$

where η_2 *is given in equation* (3.29) *and* Δ *is given in equation* (3.35). *Then the sequence of release times,* $\{r_k\}_{k=0}^{\infty}$, *and finishing time,* $\{f_k\}_{k=0}^{\infty}$, *will be admissible and the sampled-data system is finite-gain* $\mathcal{L}_2$ *stable from* w *to* x *with an induced gain less than* γ/β.

Proof: The proof is in Appendix A.11. □

Remark 3.3.5 $\Delta(x(r_k); \epsilon, \delta)$ *serves as the deadline for the delay* τ_{k+1} *in Theorem 3.3.5.*

Remark 3.3.6 *As is evident from the way it was constructed,* δ *controls the next job's release time. We might therefore expect to see a larger* δ *result in larger sampling periods. This is indeed confirmed by the analysis. Since*

$$T_k \geq r_{k+1} - f_k = \eta_2(x(r_k), x(r_{k-1}); \tau_k, \delta)$$

and since η_2 *is an increasing function of* δ *we can see that larger* δ *result in larger sampling periods.*

Remark 3.3.7 *By our construction of the parameter* ϵ, *we see that it controls the current job's finishing time. Since this*

$$\tau_k = f_k - r_k \leq \Delta(x(r_k); \epsilon, \delta)$$

and since Δ *is an increasing function of* ϵ, *we can expect to see the allowable delay increase as we increase* ϵ. *Note also that* Δ *is a decreasing function of* δ *so that* *adopting a longer sampling period by increasing δ will have the effect of reducing the maximum allowable task delay.*

Remark 3.3.8 *From the previous two remarks we see that the parameters δ and ϵ can be used to control the task's deadline and period. One way to choose ϵ and δ is to enforce real-time schedulability constraints such as those discussed in [16]. As a "rule of thumb" a reasonable strategy is to choose δ and ϵ so that η_2 and Δ are as large as possible; as this makes the task easier to schedule under an earliest-deadline first (EDF) scheduling discipline. This suggests that ϵ and δ may be seen as parameters in a scheduling-controller co-design method similar in philosophy to the approach introduced in [41]. We are currently working to see if this idea indeed provides a useful formalism for the systematic co-design of real-time self-triggered control systems.*

Remark 3.3.9 *The prior techniques can also be applied to self-triggered systems in which $\|w(t)\|_2 \leq a$, thereby relaxing assumption 3.3.1. In this case, however, it is easy to see that the bounds on sampling periods and deadlines will asymptotically go to zero, thereby leading to "chattering" behavior. A topic for future research is how best to address this issue when we can only guarantee the disturbance is uniformly bounded by a constant.*

The following corollary to the above theorem shows that the task periods and deadlines generated by our self-triggered scheme are all bounded away from zero. This is important in establishing that our scheme does not generate infinite sampling frequencies.

Corollary 3.3.6 *Let the assumptions in theorem 3.3.5 hold. Then there exist two positive constants $\zeta_1, \zeta_2 > 0$ such that $T_k \geq \zeta_1$ and $\Delta(x(r_k); \epsilon, \delta) \geq \zeta_2$.*

Proof: The proof is in Appendix A.12. □

3.4 Simulation

This section presents the results of simulation studies that empirically compare the performance of self-triggered controllers against periodically triggered and event-triggered controllers. This section's main finding is that self-triggered systems appear to generate longer sampling periods than the bound on the MATI presented in [37]. The simulation results suggest that periodically-triggered systems with a sampling period T have worse disturbance rejection abilities (as measured by energy in the tracking error) than self-triggered systems whose average sampling period is equal to T. Finally we provide examples illustrating that the self-triggering's computational cost (as measured by the ratio of the task's execution time over period) is comparable and sometimes better than the computational cost of periodically triggered systems using the the bound on the MATI in [37].

The remainder of this section is organized as follows. Subsection 3.4.1 describes the system under study. Simulations of the system's self-triggered controller will be found in subsection 3.4.2. The performance of the self-triggered system is then compared against comparable event-triggered schemes (subsection 3.4.3) and periodically-triggered schemes (subsection 3.4.4). A discussion of self-triggering's computational cost is found in 3.4.5.

3.4.1 System Model

The following simulation results were generated for event-triggered and self-triggered feedback systems. The plant was an inverted pendulum on top of a moving cart. The plant's linearized state equations were

$$\dot{x}(t) \;=\; \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -m_1 g/m_2 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & g/\ell & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1/m_2 \\ 0 \\ -1/(m_2\ell) \end{bmatrix} u(t) + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} w(t)$$

where m_2 was the cart mass, m_1 was the mass of the pendulum bob, ℓ was the length of the pendulum arm, and g was gravitational acceleration. For these simulations, we let $m_1 = 1$, $m_2 = 10$, $\ell = 3$, and $g = 10$. The system state was the vector $x = \begin{bmatrix} y & \dot{y} & \theta & \dot{\theta} \end{bmatrix}^T$ where y was the cart's position and θ was the pendulum bob's angle with respect to the vertical. The control input $u(t)$ was generated by either an event-triggered or self-triggered controller. The function w was an external disturbance to the system. The system's initial state was the vector $x_0 = \begin{bmatrix} 0.98 & 0 & 0.2 & 0 \end{bmatrix}^T$.

We designed a continuous-time state feedback control system (equation (3.1)) in which the performance level, γ, was set to 200. Solving the Riccati equation in equation (3.5) yielded a positive definite matrix P such that the state-feedback gains were

$$B_1^T P = \begin{bmatrix} -2 & -12 & -378 & -210 \end{bmatrix}. \tag{3.38}$$

The state trajectory of the resulting closed-loop system is denoted below as x_c. Figure 3.2 plots the system states as a function of time under the assumption that $w(t) = 0$ for all t. Figure 3.2 is therefore the impulse response of the inverted pendulum system.

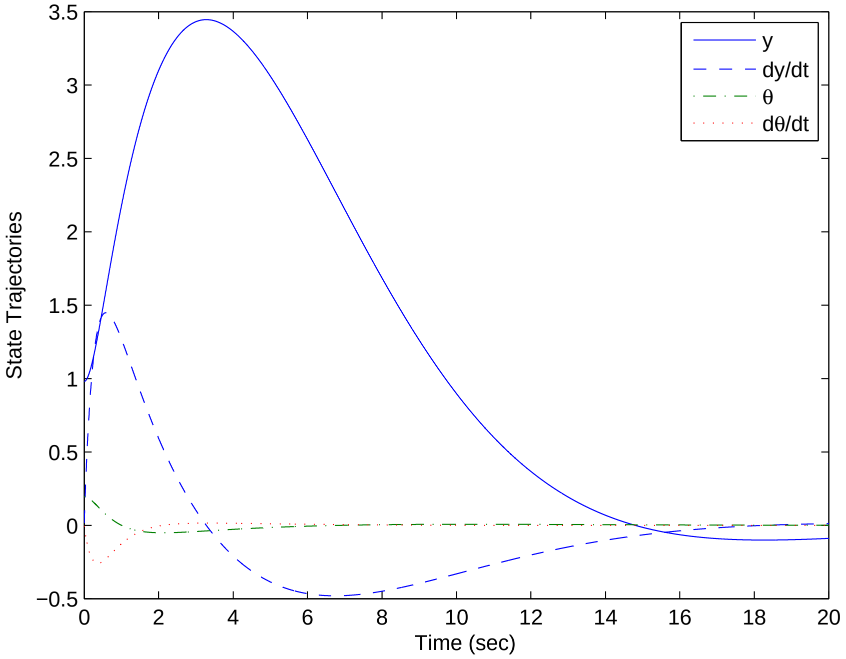

Figure 3.2. State trajectories of continuous-time closed-loop systems in equation (3.1)

3.4.2 Self-triggered Feedback

The simulations in this subsection are for the self-triggered feedback scheme associated with equation (3.36) and (3.37) in Theorem 3.3.5 with $\beta = 0.5$. In this case, the task release times were generated at time f_k using the equation

$$r_{k+1} = f_k + \eta_2(x(r_k), x(r_{k-1}), \tau_k, \delta)$$

and the finishing times were assumed to satisfy

$$f_{k+1} = r_{k+1} + \Delta(x(r_k); \epsilon, \delta),$$

which means the delays are equal to the deadlines. The plant is the inverted pendulum plant of the preceding subsection in which the external disturbance $w(t)$ was again zero. The ϵ and δ parameters were chosen to be 0.65 and 0.7, respectively.

In comparing the performance of the self-triggered versus the continuous-time system, we examine the "normalized" error, Let x_s denote the self-triggered system's response and let x_c denote the continuous-time system's response. The normalized self-triggered system's error, $E(t; x_s)$ is defined by

$$E(t; x_s) = \frac{\left| \sqrt{V(x_s(t))} - \sqrt{V(x_c(t))} \right|}{\sqrt{V(x_c(t))}} \tag{3.39}$$

where $V(x) = x^T P x$ and P is the positive definite matrix satisfying the algebraic Riccati equation (3.5). This normalization of the state error allows us to fairly measure those states (i.e. the pendulum bob angle) that are most directly affected when input disturbances exist.

Figure 3.3 plots the normalized error, $E(t; x_s)$ of the self-triggered system assuming $w(t) = 0$ and $\|w(t)\|_2 \leq 0.01\|x_s(t)\|_2$. For both cases, the normalized error is small over time, thereby suggesting that the continuous-time and self-triggered systems have nearly identical impulse responses.

Figure 3.4 plots the task periods, T_k, (crosses) and deadlines, Δ, (dots) generated by the self-triggered scheme assuming $w(t) = 0$. The sampling periods range between 0.021 and 0.185 seconds. Note that these sampling periods show significant variability. The shortest and most aggressive sampling periods occurred in response to the system's non-zero initial condition. Longer and relatively constant sampling periods were generated once the system state has returned to the neighborhood of the system's equilibrium point. This seems to confirm the conjecture

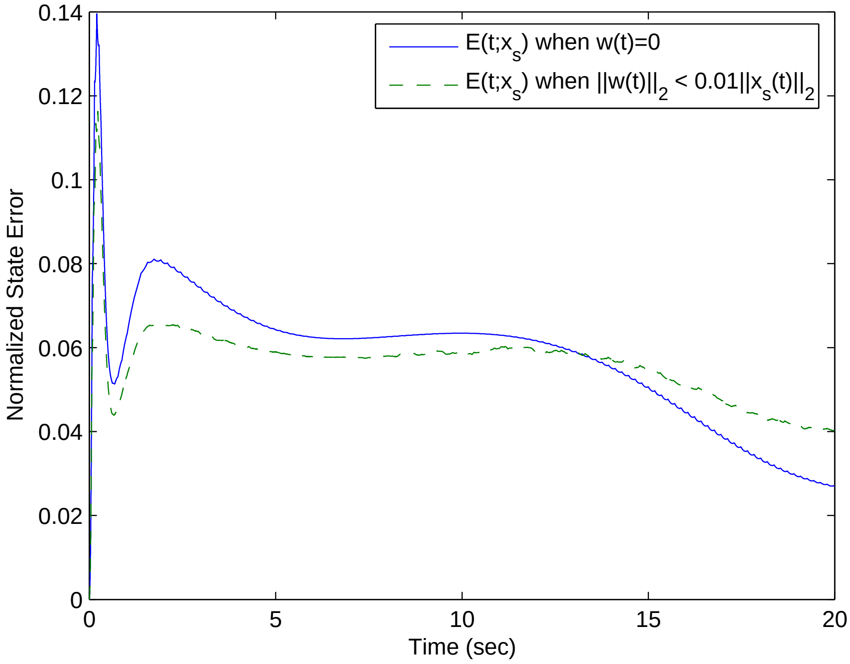

Figure 3.3. Normalized state error versus time for a self-triggered systems with $w(t) = 0$ and a self-triggered system with $\|w(t)\|_2 \leq 0.01\|x_s(t)\|_2$ ($\delta = 0.7$, $\epsilon = 0.65$).

that self-triggering can effectively adjust sampling periods in response to changes in the control system's external inputs.

Figure 3.5 plots the sample periods, T_k (crosses), and predicted deadlines (dots), generated by the self-triggered system when it is driven by the disturbance w where $\|w(t)\| \leq 0.01\|x_s(t)\|$. After the initial transient in the system's response, the sampling periods converge to a periodic signal in which the sample periods range between 0.037 and 0.092. It is interesting to note that T_k shows significant periodic variation. Other simulations have shown similar results. These observations suggest that the choice of "optimal" sampling period has its own dynamic that leads to a periodic variation in sampling periods. One interesting

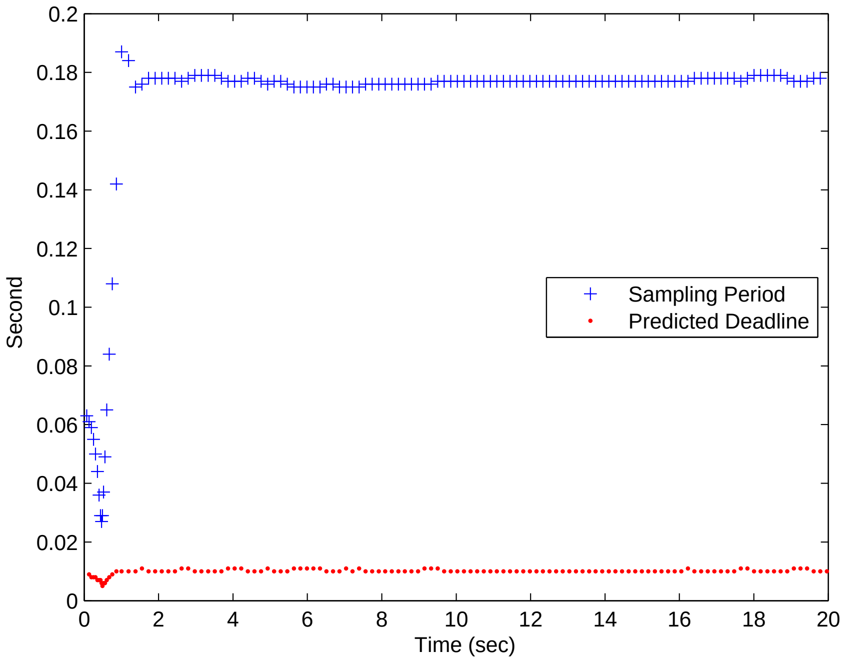

Figure 3.4. Sampling period and predicted deadline for a self-triggered system in which $\delta = 0.7$ and $\epsilon = 0.65$.

issue for future research is whether or not we can take advantage of this variability in scheduling multiple real-time control tasks.

Figures 3.6 and 3.7 show what happens to task periods and deadlines when we varied δ and ϵ. In Figure 3.6, $\delta = 0.7$ and ϵ was varied between 0.1, 0.4 and 0.65. The top two plots show histograms of the sampling period (left) and deadline (right) for $\epsilon = 0.65$. The middle two plots are histograms of the sampling periods and deadlines for $\epsilon = 0.4$. The bottom two plots display results when $\epsilon = 0.1$. Examining the three histograms on the left side of Figure 3.6 shows little change in sampling period as a function of ϵ. The three histograms on the right side of Figure 3.6 show significant variation in deadline as a function of ϵ. These results

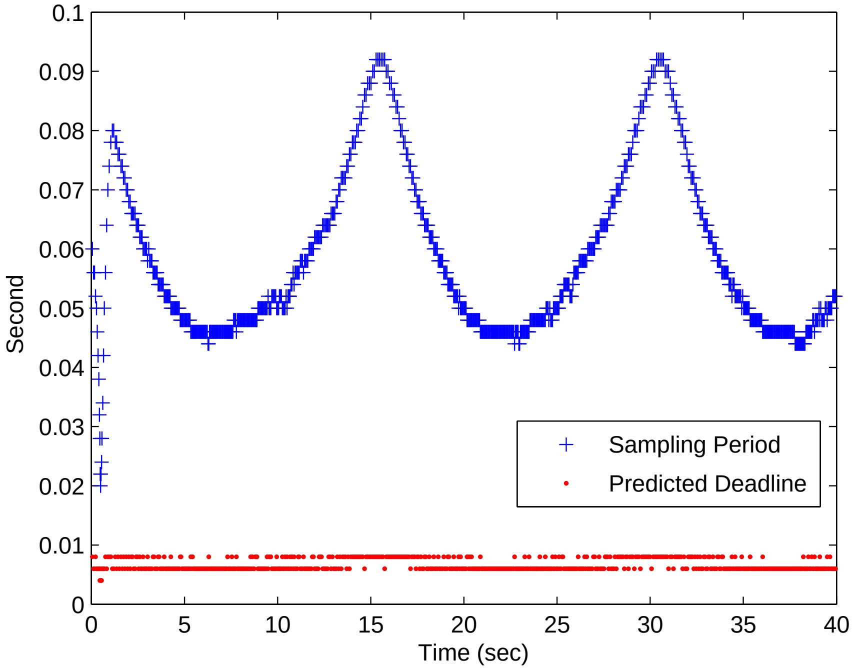

Figure 3.5. Sample periods and predicted deadlines versus time for a self-triggered system ($\delta = 0.7$, $\epsilon = 0.65$, and $\|w(t)\|_2 \leq 0.01\|x_s(t)\|_2$).

are consistent with our earlier discussion in Remark 3.3.7. Recall that ϵ controls the time when the kth task finishes. So by changing ϵ we expect to see a large impact on the predicted deadline (Δ) and little impact on the task period.

Figure 3.7 is similar to Figure 3.6 except that we keep ϵ fixed at 0.1 and vary δ from 0.15 (bottom) to 0.4 (middle) to 0.9 (top). These histograms show that as we increase δ we also enlarge the task periods. Recall that δ controls the time interval $f_{k+1} - f_k$ so that what we observe in the simulation is again consistent with our comments in Remark 3.3.6. As we increase the sampling period, however, we can expect smaller predicted deadlines because the average sampling frequency is lower. This too is seen in the histograms on the righthand side of Figure 3.7.

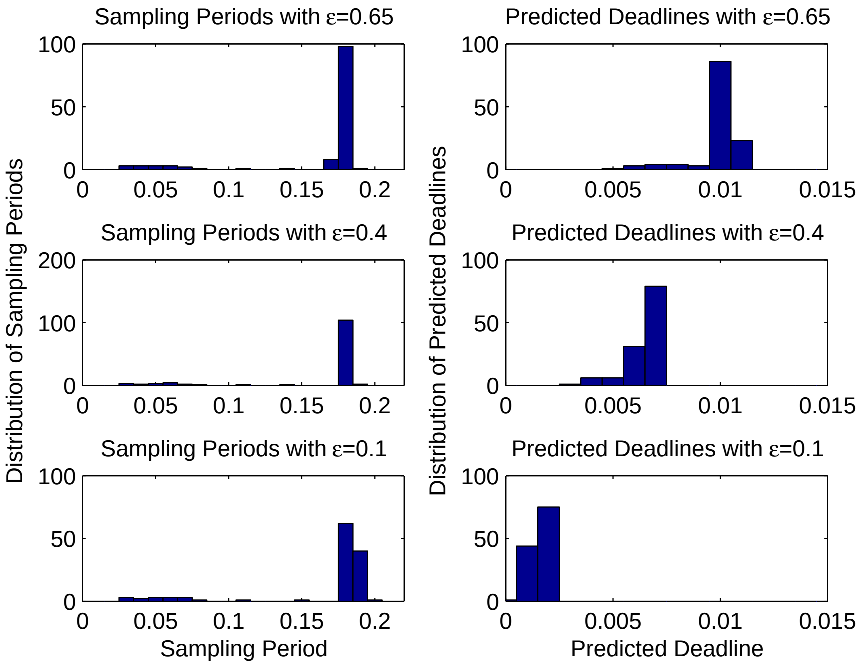

Figure 3.6. Histogram of sample period and predicted deadline for a self-triggered system in which $\delta = 0.7$ and $\epsilon \in \{0.1, 0.4, 0.65\}$.

The results in this subsection clearly show that we can effectively bound the task periods and deadlines in a way that preserves the closed-loop system's $\mathcal{L}_2$ stability. An interesting future research topic concerns how we might use these bounds on period and deadline in a systematic manner to schedule multiple real-time control tasks.

3.4.3 Comparison against Event-triggered Feedback

This subsection compares the self-triggered scheme against two event-triggered schemes; our own event-triggered scheme in theorem 3.3.1 and the event-triggered scheme in [45]. To make a fair comparison, we set $\epsilon = 0$ and $\delta = 1$ so self-triggering

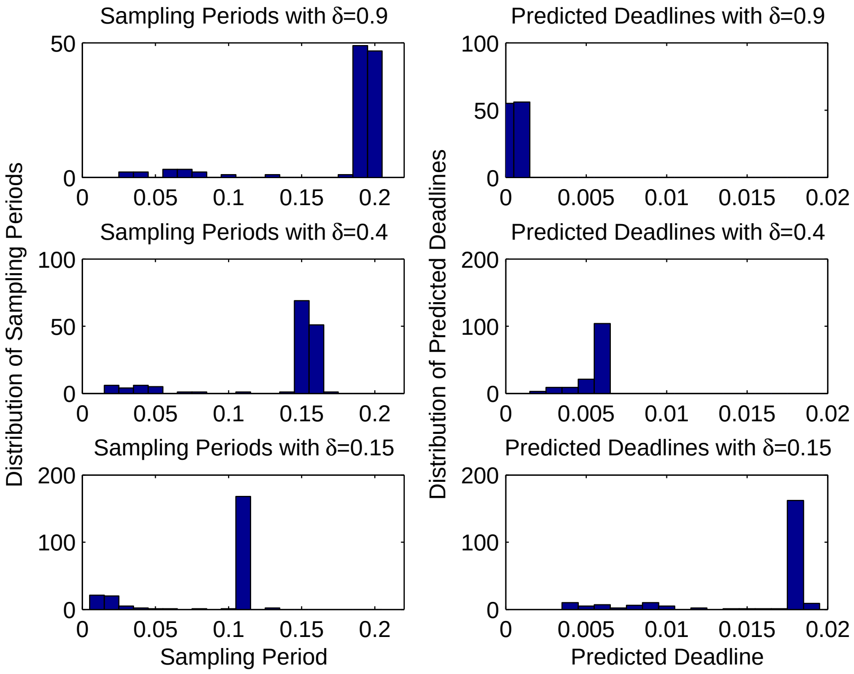

Figure 3.7. Histogram of sample period and predicted deadline for a self-triggered system in which $\epsilon = 0.1$ and $\delta \in \{0.15, 0.40.9\}$.

occurs with zero delay.

Let $x_e(t)$ denote the state trajectory of the event-triggered system based on Theorem 3.3.1's threshold test. Let $E(t; x_e)$ denote the normalized error (see equation (3.39)) of the event-triggered trajectory. Figure 3.8 plots the normalized error for the self-triggered system, $E(t; x_s)$ (solid line), and the event-triggered system, $E(t, x_e)$ (dashed line) as functions of time for $w(t) = 0$. In both cases the normalized errors are small, though the event-triggered system has a slightly larger error.

Figure 3.9 plots the sampling periods generated by the self-triggered scheme (top plot) and the event-triggered scheme (bottom plot). The self-triggered sam-

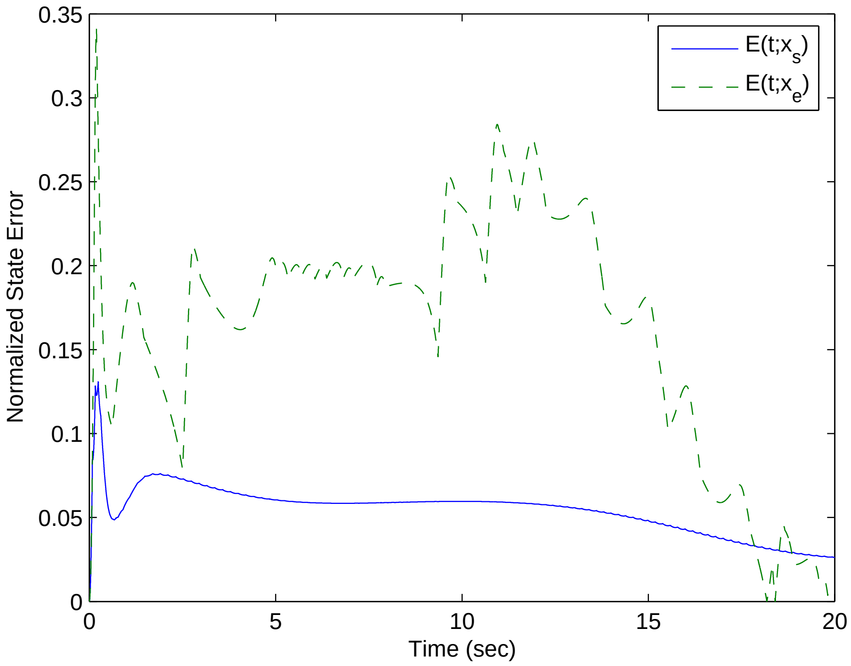

Figure 3.8. Normalized state errors versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$, and $w(t) = 0$)

pling periods range between 0.0300 and 0.2060 with an average period of 0.1782. The event-triggered sampling periods range between 0.0340 and 1.3890 with an average period of 0.3375. Note that the self-triggered sampling periods are an order of magnitude smaller than the periods of the event-triggered scheme. These results suggest that even-triggered feedback is better able to reduce sampling period frequency than the self-triggered feedback.

We then added a square wave input to the system to see how the self-triggered and event-triggered systems react to external disturbances. The results from this comparison are shown in Figure 3.10. This figure plots the time history of the normalized error signals, $E(t; x_s)$ (solid line) and $E(t; x_e)$ (dashed line), for the

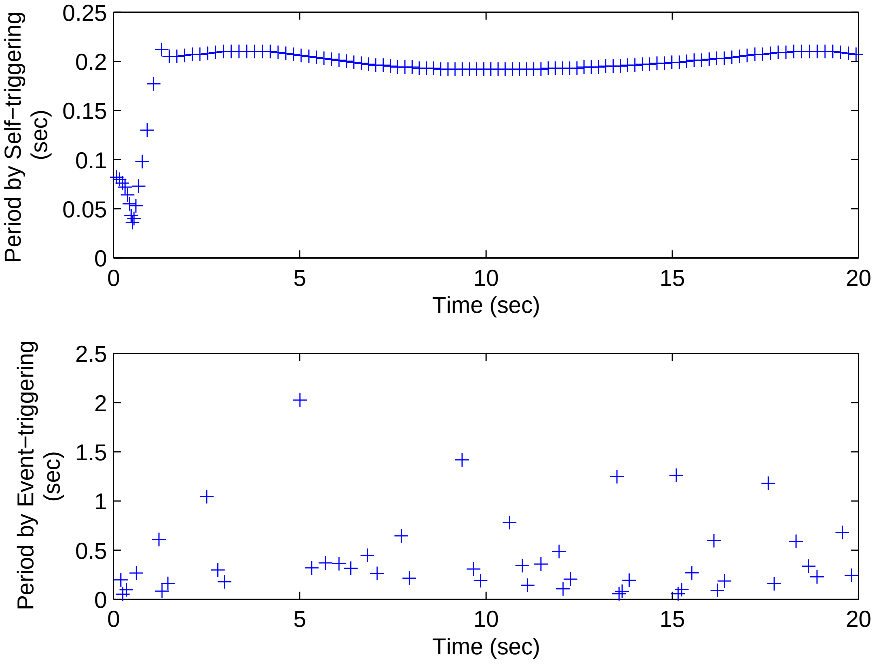

Figure 3.9. Sampling period versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$, and $w(t) = 0$)

inverted pendulum using the input signal, $w(t) = \mu(t)$ where $\mu : \mathbb{R} \rightarrow \mathbb{R}$ takes the values

$$\mu(t) = \begin{cases} \text{sgn}(\text{sin}t) & \text{if } 0 \leq t < 10 \\ 0 & \text{otherwise} \end{cases} . \tag{3.40}$$

Again, the error in the self-triggered system is smaller than that in the event-triggered system.

Figure 3.11 plots the sampling periods generated by the self-triggered (top plot) and event-triggered (bottom plot) systems when $w(t) = \mu(t)$. The top plot shows that the sampling periods in the self-triggered system readjust and get

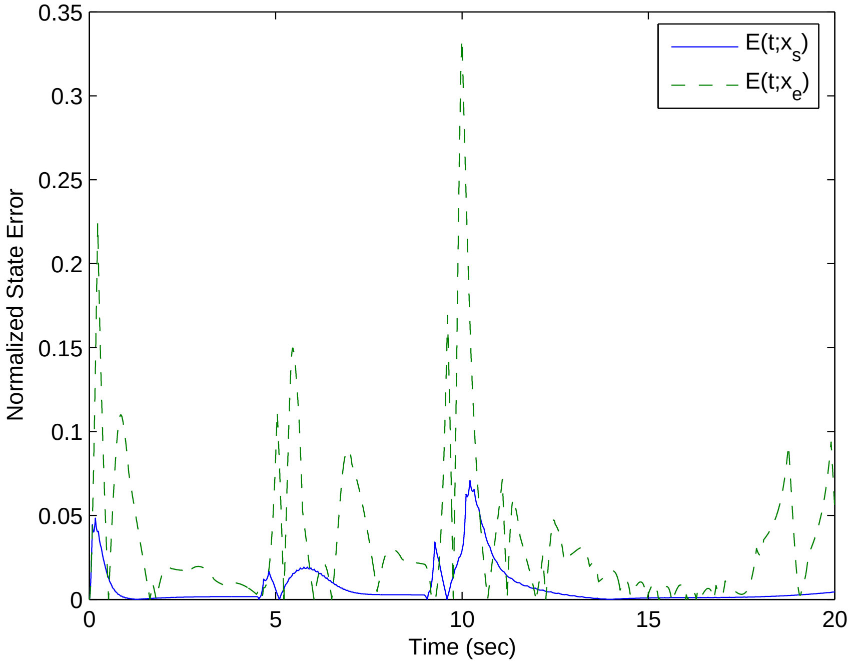

Figure 3.10. Normalized error versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$, and $w(t) = \mu(t)$)

smaller when the square wave input hits the system over the time interval $[0, 10]$. In the event-triggered system, as shown in the bottom plot, the average period, 0.2830, also get smaller compared with the periods in the bottom plot of Figure 3.9, although the decrease is not very obvious. These results again demonstrate the ability of self-triggering and event-triggering to successfully adapt to changes in the system's input disturbances.

It is instructive to compare the sampling periods generated by self-triggering (see the top plot in Figure 3.9) against the periods that would have been generated by the event-triggering scheme in [45]. Recall that the event-triggering scheme in

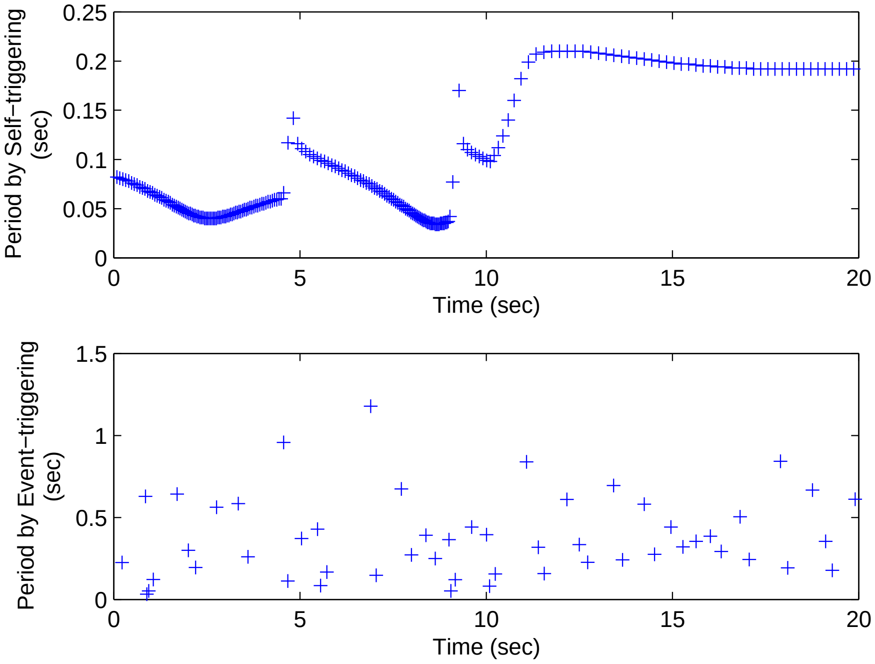

Figure 3.11. Sampling period versus time for a self-triggered system and an event-triggered system ($\delta = 1$, $\epsilon = 0$ and $w(t) = \mu(t)$).

[45] samples the state when

$$e_k^T(t)Ge_k(t) = b^2 x^T(t)Gx(t).$$

G is a positive definite matrix associated with a control Lyapunov function $V(x) = x^T Gx$ for the closed-loop system with state feedback gains K. Since V is a control Lyapunov function, we can find a matrix H such that the directional derivative of the unforced closed-loop system satisfies the inequality $\dot{V} \leq -x^T Hx$. In the above equation, p is the real constant

$$b = \frac{\lambda_{\min}(G)}{2\lambda_{\max}(G)} \frac{\lambda_{\min}(H)}{\|GBK\|}.$$

For this particular simulation, we set G equal to the P associated with our controller to obtain $b = 4.17 \times 10^{-11}$. This event-triggering threshold generates sampling periods less than 10^{-5}. This is much smaller than the sampling periods generated by the self-triggering scheme.

The reason for this difference is that the condition number of the particular G matrix is extremely large due to the great difference in the time constants associated with the dynamics of the cart and pendulum bob. Such a matrix leads to a very small b, which limits the size of the sampling periods generated by the approach in [45]. In fact, for the inverted pendulum model with the control gain K given by equation (3.38), the smallest condition number of G matrix is 409.05. The resulting p is equal to 1.35×10^{-11}. If we directly consider the value of p, the largest p we can get is 1.20×10^{-7}. The resulting sampling periods are still less than 10^{-5}. Therefore, for the inverted pendulum model, our event-triggering threshold generates much longer sampling periods.

However, for different systems which allow G with a small condition number, the approach in [45] may also generate large sampling periods. For example, for a scalar system, $\dot{x} = -x + u + w$, with $\gamma = 1/\sqrt{2}$, we get $P = 1$ and $K = -1$. The average sampling period generated by self-triggering is 0.3670 associated with $\beta = 0.5$, $\delta = 1$, $\epsilon = 0$ and $w(t) = 0$. The threshold condition in [45] is $e_k^2(t) < 4x(t)^2$ for the same P and K. The minimal, average, maximal sampling periods by the approach in [45] are the same, 0.4060, which are longer than the average period generated by our approach.

3.4.4 Comparison against Periodically-triggered Feedback

The simulations in this subsection compare the performance of self-triggered and "comparable" periodically triggered feedback control systems using the inverted pendulum system described above. Again to make a fair comparison, we enforce zero delays by setting $\delta = 1$ and $\epsilon = 0$ in the self-triggered controller.

We first compare the sampling period in the self-triggered system with the bound on the MATI given by [37]. The bound on the MATI ensuring an $\mathcal{L}_2$ gain of γ is,

$$T_{\text{MATI}} = \frac{1}{L} \ln \frac{L + \gamma_1}{\bar{\rho} L + \gamma_1}, \tag{3.41}$$

where, in the inverted pendulum model, $\bar{\rho} = 0$, $L = \max(0.5\lambda_{\max}(-B_1 K - K^T B_1^T), 0)$, $\gamma_1 \geq 0$ satisfies

$$\gamma = \frac{1 + \max\{\gamma_1, \gamma_2\}}{1 - \gamma_1 \gamma_2},$$

and γ_2 is the $\mathcal{L}_2$ gain for the closed-loop system ($\dot{x} = A_{cl} x + B_1 K e + B_2 w$) from (e, w) to $-A_{cl} x$.

From equation (3.41), we compute the bound on the MATI consistent with an $\mathcal{L}_2$ gain $\gamma = 400$. This results in $T_{\text{MATI}} = 0.0092$. The corresponding average sampling period for a self-triggered system with gain $\gamma = 400$ is equal to 0.1782 (see Figure 3.9). Clearly the average period generated by the self-triggered scheme is longer than the estimate of the MATI for systems with the same induced $\mathcal{L}_2$ gain. Note that the bound on the MATI obtained assuming an infinite-gain for γ is still only 0.0112 which is still much smaller than the average sampling period generated by the self-triggered controller.

Note that the above self-triggered system generated sampling periods under the assumption that the noise magnitude $a = 0$. For non-zero a the average sampling period will decrease. For instance if $a = 0.01$, then the average self-triggered period shrinks to 0.0629. Though this is still larger than the bound on the MATI, it is apparent that as a increases, the average period will continue shrinking until it is less than the MATI. This appears to be one weakness of the current result in theorem 3.3.5. We believe this can be relaxed, but that will need to be addressed in future work.

One thing worth mentioning is that the self-triggering scheme is compared with a theoretically derived bound on the MATI [37]. This bound may be conservative due to the sampling scheme and the conservatism of the proof techniques. It does not mean that the actual maximal allowable transfer interval is conservative.

We should also notice that the bound on the MATI can be predicted before the system is deployed by methods such as the one in [37]. So when designers try to build physical devices, they know exactly what the requirements are on the sampling rate. On the contrary, it is difficult to predict ahead of time the minimal sampling period in self-triggered feedback systems. It may be possible that for a short interval, the controller insists on a sampling/control rate that the physical device cannot provide. Therefore, how to handle such unexpected delays would be an interesting direction to follow in the future.

We then compared the performance of the self-triggered system and a periodically triggered system with "comparable" task period, which is the average sampling period over the entire time zone, 0.1782, generated by the example in Figure 3.9. The results from this comparison are shown in Figure 3.12. This figure plots the time history of the normalized errors for the self-triggered system, $E(t; x_s)$

(solid line), and the periodically triggered system, $E(t; x_T)$ (dash-dot line) for an inverted pendulum with input signal $w(t) = \mu(t)$ where μ is defined in equation (3.40) and x_T denotes the periodically triggered system's response.

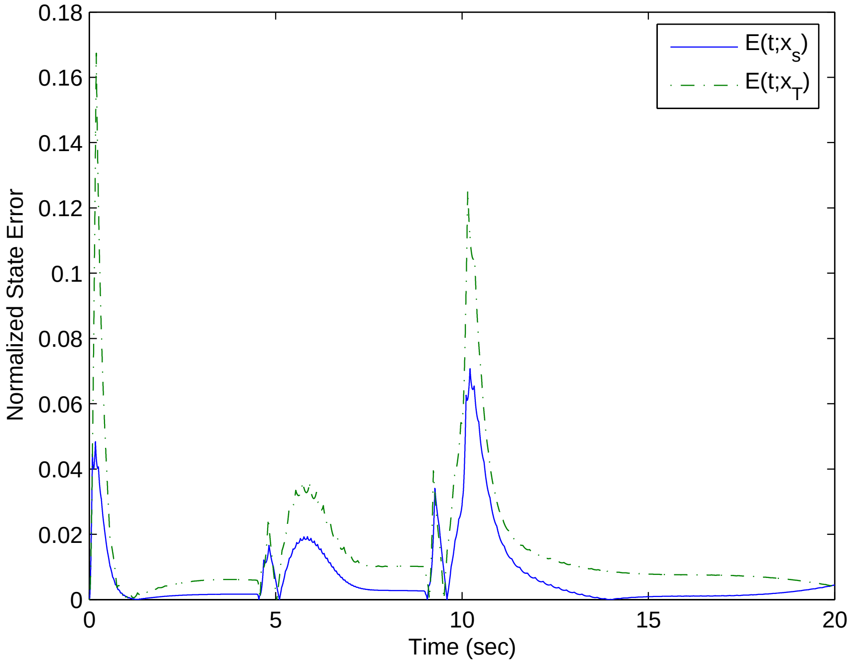

Figure 3.12. Normalized error versus time for a self-triggered system ($\delta = 1$ and $\epsilon = 0$) and a periodically triggered system whose period was chosen from the sample periods shown in the top plot of Figure 3.9.

Figure 3.12 clearly shows that the self-triggered error is significantly smaller than the error of the periodically triggered system. This error is a direct result of the self-triggered system's ability to adjust its sample period as shown in the top plot of Figure 3.11.

3.4.5 Self-triggered System's Computational Cost

This subsection compares the computational cost in the self-triggered system by comparing the average utilization of the self-triggered system against a periodically triggered system with period equal to the MATI, $T_{\text{MATI}} = 0.0092$. The average utilization is the quotient of the execution time over the average sampling period. In this section, we set $\delta = 1$ and $\epsilon = 0$ with $w(t) = 0$ for the self triggered system. In this case the average period is 0.1782.

The computational cost for one task will be measured by the number of multiplies required for a single update of the control. We focus on "multiplies" since they represent the most expensive floating point computation. Since other parameters (such as σ and $\sqrt{M}$) can be computed off-line, the computational cost for one task comes from the computation of the control, which uses n multiplies, and the prediction of the next release time, which requires about $2n^2 + 2n$ multiplies, where n is the state dimension. The total computational cost of the self-triggered scheme is therefore $2n^2 + 3n$, whereas the computational cost of the periodically triggered controller is only n multiplies.

While the computational cost of self-triggering is higher than that of periodically-triggered systems, we generally see that self-triggered systems have longer average periods. So a more appropriate comparison of each method's resource usage is provided by their "utilization" which we take here as the quotient of the computational cost (number of multiplies) over the sampling period. For the inverted pendulum system the periodically triggered system's utilization, U_T, may therefore be taken to be $\frac{n}{T_{\text{MATI}}}$ where we normalized out the cost of the control computation. For this example (with $n = 4$) we earlier computed the MATI to be $T_{\text{MATI}} = 0.0092$ so that the periodic system's utilization is $U_T = 434.7828$. The self-triggered system's utilization, U_s, is given by $\frac{2n^2+3n}{\overline{T}}$ where $\overline{T}$ is the average period generated by self-triggering. In our simulations $\overline{T} = 0.1782$, so that the self-triggered system's utilization becomes $U_s = 246.9136$. The main conclusion to be drawn here is that even though periodically-triggered feedback has lower computational cost per job, the average utilization of both methods appears to be comparable.

3.5 Summary

This chapter has presented a state-dependent threshold inequality whose satisfaction assures the induced $\mathcal{L}_2$ gain of a sampled-data linear state feedback control system. We derive state-dependent bounds on the task periods and deadlines enforcing this threshold inequality based on an event-triggered feedback scheme. These results were used to present a self-triggered feedback scheme with guaranteed $\mathcal{L}_2$ stability. Simulation results show that the proposed event- and self-triggered feedback schemes perform better than comparable periodically triggered feedback controllers. The results in this chapter, therefore, appear to provide a solid analytical basis for the development of aperiodic sampled-data control systems that adjust their periods and deadlines to variations in the system's external inputs.

There are a number of open directions for future study. The bounds derived in this chapter can be thought of as quality-of-control (QoC) constraints that a real-time scheduler must enforce to assure the application's (i.e. control system's) performance level. This may be beneficial in the development of soft real-time systems for controlling multiple plants. The bounds on task period and deadline suggest that real-time engineers can adjust both task period and task deadline to assure task set schedulability while meeting application performance requirements. It would be interesting to see whether such bounds can be used in generalizations of elastic scheduling algorithms [11] [15]. This might allow us to finally build soft real-time systems providing guarantees on application performance that have traditionally been found only in hard real-time control systems.

To our best knowledge, this is the first rigorous examination of what might be required to implement self-triggered feedback control systems. Self-triggering on single processor systems may not be very useful since event-triggers can often be implemented in an inexpensive manner using programmable gate arrays (FPGA) or custom analog integrated circuits (ASIC). If, however, we are controlling multiple plants over a wireless network, then the inability of such networks to provide deterministic guarantees on message delivery make the use of self-triggered feedback much more attractive. An interesting future research direction would explore the use of self-triggered feedback over wireless sensor-actuator networks, which is partially addressed in [13, 37, 43, 52, 56].

CHAPTER 4

Distributed Event-Triggering in Networked Control Systems

A networked control system (NCS) is a system wherein numerous physically coupled subsystems are geographically distributed throughout the system. Control and feedback signals are exchanged through a real-time network among the system's components. Specific examples of NCS include electrical power grids and transportation networks. In recent years, it has been popular to refer to such networked systems as *cyber-physical systems*. The networking of control effort can be advantageous in terms of lower system costs due to streamlined installation and maintenance costs. The introduction of real-time network infrastructure, however, raises new challenges regarding the impact that communication reliability has on the control system's performance. In real-time networks, information is transmitted in discrete-time rather than continuous-time. Moreover, all real networks have bandwidth limitation that can cause delays in message delivery that may have a major impact on overall system stability [30].

For this reason, some researchers began investigating the timing issue in NCS. One packet transmission problem was considered in [26, 60], where a supervisor summarizes all subsystem data into this single packet. As a result such schemes may be impractical for large-scale systems. Asynchronous transmission was considered in [13, 37, 48], which derived bounds on the maximum allowable transfer interval (MATI) between two subsequent message transmissions so that the system stability can be guaranteed. All of this prior work confined its attention to control area network (CAN) buses where centralized computers are used to coordinate the information transmission.

One thing worth mentioning is that these schemes mentioned above require extremely detailed models of subsystem interactions and the execution of communication protocols must be done in a highly centralized manner. Both of these requirements can greatly limit the scalability of centralized approaches to NCS. On the other hand, the MATI is computed before the system is deployed, which means it is independent of the system state. So it must ensure adequate behavior over a wide range of possible system states. As a result, it may be conservative.

To overcome these issues, decentralized event-triggering feedback schemes were proposed in [52, 54] for linear and nonlinear systems, respectively. Most recently, an implementation of event-triggering in sensor-network was introduced in [36]. By event-triggering, a subsystem broadcasts its state information to its neighbors only when “needed”. In this case, “needed” means that some measure of the subsystem’s local state error exceeds a specified threshold [44, 49]. In this way, event-triggering makes it possible to reduce the frequency with which subsystems communicate and therefore use network bandwidth in an extremely frugal manner. An important assumption in [52, 54] is that neither data dropouts nor delays occur in such systems. In real-time networks, however, especially wireless networks, data dropouts and delays always exist. It, therefore, suggests a more sophisticated consideration of such systems.

In this chapter, we study asymptotic stability in the distributed NCS with data dropouts and transmission delays. A distributed event-triggering scheme is introduced to ensure system’s stability [56]. Unlike the prior work that modelled data dropouts as stochastic processes using a centralized approach [24, 33], this event-triggering scheme identifies the maximal allowable number of agents' successive dropouts (MANSD). It also provides the maximal allowable transmission delay (also called *deadline*) for each transmission.

This scheme is decentralized in a sense that (1) a subsystem's broadcast decisions are made using its local sampled data, (2) the maximal allowable transmission delay of a subsystem's broadcast can be predicted based on the local information, (3) a subsystem locally identifies the maximal allowable number of its successive data dropouts, and (4) the designer's selection of the threshold only requires information about an individual subsystem and its immediate neighbors.

This chapter is organized as follows. Section 4.1 formulates the problem. Section 4.2 presents the decentralized event-design approaches for both nonlinear and linear systems. Data dropouts and broadcast delays are considered in section 4.3. Section 4.4 demonstrates the simulation results. Conclusions are drawn in section 4.5.

4.1 Problem Formulation

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

Definition 4.1.1 *A graph $\mathcal{G}_{cp} = (\mathcal{N}, \mathcal{E}_{cp})$ is called the coupling graph of a NCS, if each node $i \in \mathcal{N}$ represents an agent in the NCS and each edge $(i, j) \in \mathcal{E}_{cp}$ represents that the system dynamics of agent j is directly driven by agent i.*

Definition 4.1.2 *A graph $\mathcal{G}_{cm} = (\mathcal{N}, \mathcal{E}_{cm})$ is called the communication graph of a*

NCS, if each node $i \in \mathcal{N}$ represents an agent in the NCS and each edge $(i,j) \in \mathcal{E}_{cm}$ represents that agent j can receive the broadcasts from agent i.

In this chapter, the coupling graph and the communication graph do not have to be the same. Furthermore, both graphs are directed, instead of undirected. This provides us a general framework on the network topology. For notational convenience, we let

- $Z_i \triangleq \{j \in \mathcal{N} \mid (j,i) \in \mathcal{E}_{\text{cm}}\}$ denotes the set of agents that agent i can get information from;
- $U_i \triangleq \{j \in \mathcal{N} \mid (i,j) \in \mathcal{E}_{\text{cm}}\}$ denotes the set of agents that can receive agent i's information;
- $D_i \triangleq \{j \in \mathcal{N} \mid (j,i) \in \mathcal{E}_{\text{cp}}\}$ denotes the set of agents that directly drive agent i's dynamics;
- $S_i \triangleq \{j \in \mathcal{N} \mid (i,j) \in \mathcal{E}_{\text{cp}}\}$ denotes the set of agents who are directly driven by agent i;
- For any set $\Sigma \subseteq \mathcal{N}$, $|\Sigma|$ to denote the number of the elements in Σ and $\bar{\Sigma} = \Sigma \cup \{i\}$.

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.

The state equation of agent i is

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

where $x_i : \mathbb{R} \to \mathbb{R}^n$ is the state trajectory of agent i, $u_i : \mathbb{R} \to \mathbb{R}^m$ is a control input, $\kappa_i : \mathbb{R}^{n|\bar{Z}_i|} \to \mathbb{R}^m$ is the feedback strategy of agent i satisfying $\kappa_i(0) = 0$, $g_i : \mathbb{R}^{n|\bar{D}_i|} \times \mathbb{R}^m \times \mathbb{R}^l \to \mathbb{R}^n$ is continuous and locally Lipschitz satisfying $g_i(0,0) = 0$, and $x_{\bar{D}_i} = \{x_j\}_{j\in\bar{D}_i}$, $x_{\bar{Z}_i} = \{x_j\}_{j\in\bar{Z}_i}$. To outline the analysis, we assume that the states/inputs/disturbances of agents have the same dimension. The results can be easily extended to the case where the dimensions of agents' states/inputs/disturbances are different from each other.

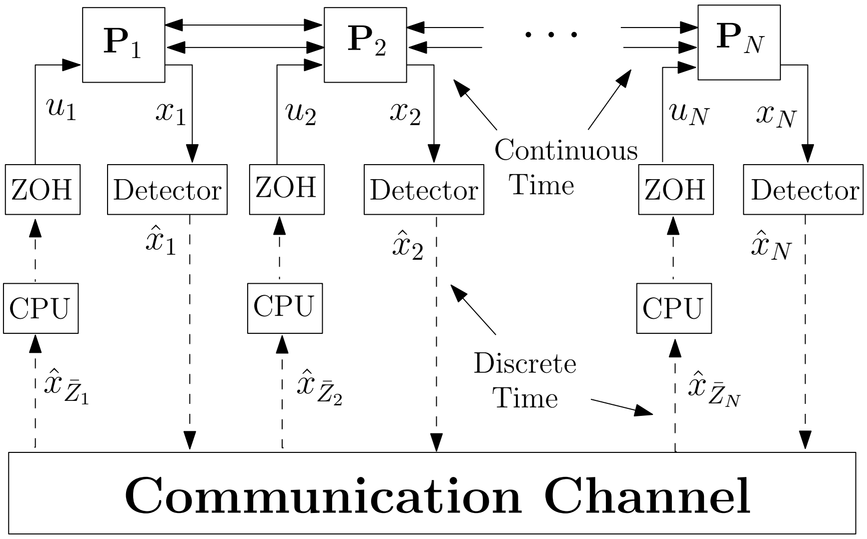

Figure 4.1. The architecture of the real-time NCS

The architecture of a real-time implementation of distributed NCS is plotted in Figure 4.1. In such a system, agent i can only detect its own state, x_i. If the local "error" signal exceeds some given threshold, which can be detected by hardware detectors, agent i will sample and broadcast its state information to those agents in U_i through a real-time network. Meanwhile, agent i's control, u_i, at time t is computed based on the latest states at time t that are successfully broadcasted by those agents in $\bar{Z}_i$, denoted as $\hat{x}_{\bar{Z}_i}(t)$. Therefore, the control signal used by agent i is held constant by a zero-order hold (ZOH) unless one of the agents in $\bar{Z}_i$ makes a successful broadcast. This means that the state equation of agent i can be written as

$$\begin{aligned} \dot{x}_i(t) &= g_i(x_{\bar{D}_i}(t), u_i(t)) \\ u_i(t) &= \kappa_i(\hat{x}_{\bar{Z}_i}(t)). \end{aligned} \tag{4.1}$$

Agent i's broadcast can be characterized by three monotone increasing sequences of time instants: the broadcast release time sequence $\{r_j^i\}_{j=1}^{\infty}$, 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 jth broadcast of agent i is released, but not necessarily transmitted successfully. The time b_k^i denotes the time instant when the kth "successful" broadcast of agent i is released. By "successful", it means the data in this broadcast is successfully transmitted to all agents in set $\bar{U}_i$ through the network. Obviously, $\{b_k^i\}_{k=1}^{\infty}$ is a subsequence of $\{r_j^i\}_{j=1}^{\infty}$. We use d_k^i to denote the number of agent i's broadcast releases between b_k^i and b_{k+1}^i, which is also the number of data dropouts of agent i between b_k^i and b_{k+1}^i. At this time, we assume there is no delay between sampling and broadcast release. The time f_k^i denotes the time instant when the kth broadcasted data of agent i is received by its neighbors. Notice that $\hat{x}_i(t) = x_i(b_k^i)$ for all $t \in [f_k^i, f_{k+1}^i)$.

Figure 4.2 illustrates the relationship between the broadcast release time r_j^i, successful release time b_k^i and broadcast finishing time f_k of agent i. The x-axis in Figure 4.2 is time with release (r^i_j), successful release (b^i_k), and finishing time (f^i_k) marked on the axis. The black rectangles above the time axis mark intervals over which the data is being transmitted.

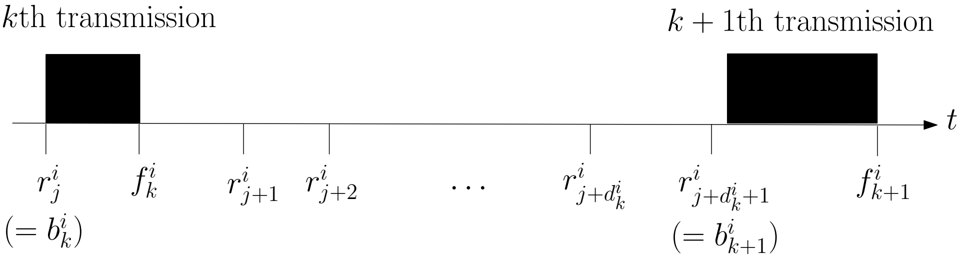

Figure 4.2. Relationship between release time (r^i_j), successful release (b^i_k), and finishing time (f^i_k)

The objective is to develop distributed event-triggering schemes to identify $\{r^i_j\}_{j=1}^{\infty}$, $\{b^i_k\}_{k=1}^{\infty}$, and $\{f^i_k\}_{k=1}^{\infty}$ such that the NCS defined in equation (4.1) is asymtotically stable. For notational convenience, let $e_i : \mathbb{R} \to \mathbb{R}^n$ be defined as $e_i(t) \triangleq x_i(t) - \hat{x}_i(t)$ for $\forall t \geq 0$ and $\varepsilon^j_i : \mathbb{R} \to \mathbb{R}^n$ defined as $\varepsilon^j_i(t) \triangleq x_i(t) - x_i(r^i_j)$.

4.2 Distributed Event Design for Asymptotic Stability

In this section, we proposed distributed schemes to design local events for both nonlinear and linear systems. For nonlinear subsystems, local event design is transformed into local ISS design problems; for linear subsystems, the design is simplified to be local LMI feasibility problems. The broadcast release can be triggered by these events.

4.2.1 Local Event Design in Nonlinear Systems

In this section, a distributed approach for nonlinear systems is proposed to construct local events such that the resulting event-triggered NCS is asymptotically stable. To obtain such a distributed method, we first introduce a theorem that provides a centralized approach for local event design. For notational convenience, we let $L_{g_i}V = \frac{\partial V}{\partial x_i} g_i(x_{\bar{D}_i}, u_i)$ with some function $V(x) \in \mathbb{R}$.

Theorem 4.2.1 *Consider the NCS in equation* (4.1)*. Assume that there exist a smooth, positive-definite function* $V : \mathbb{R}^{nN} \to \mathbb{R}$ *and class* $\mathcal{K}$ *functions* α_1*,* α_2*,* ϕ_i*,* $\psi_i : \mathbb{R} \to \mathbb{R}$ *for* $i = 1, \cdots, N$ *such that*

$$\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|) \tag{4.2}$$

$$\sum_{i \in \mathcal{N}} L_{g_i}V \leq \sum_{i \in \mathcal{N}} [-\phi_i(\|x_i\|) + \psi_i(\|e_i\|)] \tag{4.3}$$

$$-\delta_i \phi_i(\|x_i(t)\|) + \psi_i(\|e_i(t)\|) \leq 0 \tag{4.4}$$

hold for all $t \geq 0$ *and all* $i \in \mathcal{N}$ *with some* $\delta_i \in (0, 1)$*, then the NCS is asymptotically stable.*

Proof: Applying equation (4.4) into equation (4.3), we have

$$\sum_{i \in \mathcal{N}} L_{g_i}V \leq \sum_{i \in \mathcal{N}} [-(1 - \delta_i)\phi_i(\|x_i\|)] \tag{4.5}$$

which is, with equation (4.2), sufficient to show the NCS is asymptotically stable. □

Remark 4.2.1 *The event for agent* i *is only associated with* e_i *and* x_i*. Agent* i *just needs to use the violation of equation* (4.4) *to trigger the broadcast such that* *the inequality in equation* (4.4) *holds. If there are no transmission delay and data dropouts, the system stability of NCS can be maintained.*

Theorem 4.2.1 shows that the satisfaction of equation (4.3) and (4.4) guarantees $\mathcal{L}_p$ stability of the NCS. Based on this theorem, deriving local events is equivalent to constructing a collection of class $\mathcal{K}$ functions. In fact, we notice that in order to design its local event, agent i just needs to find two class $\mathcal{K}$ functions ϕ_i and ψ_i. This brings the possibility to decentralize the design procedure. The following theorem provides such a distributed approach for agent i to construct ϕ_i and ψ_i.

Assumption 4.2.1 *Assume that, for any $i \in \mathcal{N}$, there exist a constant $p \geq 1$, a continuous, positive-definite functions $V_i : \mathbb{R}^n \to \mathbb{R}$, class $\mathcal{K}$ functions α_1^i, $\alpha_2^i : \mathbb{R} \to \mathbb{R}$, positive constants $\zeta_i, \beta_i, \rho_i \in \mathbb{R}$, and control law $g_i : \mathbb{R}^{n|\bar{Z}_i|} \to \mathbb{R}^{m_i}$ satisfying*

$$\alpha_1^i(\|x_i\|) \leq V_i(x_i) \leq \alpha_2^i(\|x_i\|) \tag{4.6}$$

$$L_{g_i}V_i \leq -\zeta_i\|x_i\|^p + \sum_{j\in D_i\cup Z_i} \beta_j\|x_j\|^p + \sum_{j\in\bar{Z}_i} \rho_j\|e_j\|^p \tag{4.7}$$

$$\zeta_i - |S_i \cup U_i|\beta_i > 0. \tag{4.8}$$

Remark 4.2.2 *Equations* (4.7) *suggests that subsystem i is $\mathcal{L}_p$ stable from $\{x_j\}_{j\in D_i\cup Z_i}$ and $\{e_j\}_{j\in\bar{Z}_i}$ to x_i. Equation* (4.8) *specifies the performance level of the subsystem. This assumption is satisfied when the interconnections between subsystems are weak. If $\{\rho_i\}_{i=1}^N$ and $\{\beta_i\}_{i=1}^N$ are pre-selected, agent i needs to find V_i, ζ_i, and κ_i to fulfill equations* (4.6) – (4.8)*. This is an input-to-state stabilization problem for agent i. The local event design is based on the parameters ζ_i, β_i, and ρ_i. Notice that the stabilization problem is only associated with agent i's local dynamics.*

Theorem 4.2.2 *Consider the NCS defined in equation* (4.1). *Suppose that assumption 4.2.1 holds. If for any $i \in \mathcal{N}$, the inequality*

$$-\delta_i \|x_i(t)\| + \sigma_i \|e_i(t)\| \leq 0 \tag{4.9}$$

holds for all $t \geq 0$, where $\delta_i \in (0, 1)$ and

$$\sigma_i = \left(\frac{|\bar{U}_i| \rho_i}{\zeta_i - |S_i \cup U_i| \beta_i} \right)^{\frac{1}{p}}, \tag{4.10}$$

then the NCS is asymptotically stable.

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

$$\begin{aligned}
\dot{V} &= \sum_{i \in \mathcal{N}} L_{g_i} V_i \leq \sum_{i \in \mathcal{N}} \left[-\zeta_i \|x_i\|^p + \sum_{j \in D_i \cup Z_i} \beta_j \|x_j\|^p + \sum_{j \in \bar{Z}_i} \rho_j \|e_j\|^p \right] \\
&= \sum_{i \in \mathcal{N}} \left[-\left(\zeta_i - |S_i \cup U_i| \beta_i\right) \|x_i\|^p + \rho_i |\bar{U}_i| \|e_i\|^p \right],
\end{aligned}$$

where the equality is obtained by resorting the items according to index i.

Applying equation (4.9) into the preceding equation yields

$$\dot{V} \leq \sum_{i \in \mathcal{N}} [-(1 - \delta_i^p) \left(\zeta_i - |S_i \cup U_i| \beta_i\right) \|x_i\|^p,] \tag{4.11}$$

which is, with equation (4.8), sufficient to show the NCS is asymptotically stable. □

Remark 4.2.3 *We use the violation of the inequality in equation* (4.9) *to trigger agent i's broadcasts. This event is only a function of 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.*

Remark 4.2.4 *The functions $\zeta_i\|x_i\|_2^p$, $\beta_j\|x_j\|_2^p$, and $\rho_j\|e_j\|_2^p$ may be replaced by more general class $\mathcal{K}$ functions that are Lipschitz continuous.*

We will find it convenient to use a slightly weaker sufficient condition for asymptotic stability. This condition is stated in the following corollary. The corollary recasts the event trigger in equation (4.9) as a function of the local error, e_i, and the successfully broadcast local state, $\hat{x}_i$.

Corollary 4.2.3 *Consider the NCS in equation* (4.1)*. Suppose that assumption 4.2.1 holds. If for any $i \in \mathcal{N}$, the inequality*

$$c_i\|e_i(t)\| \leq \delta_i\|\hat{x}_i(t)\| \tag{4.12}$$

holds for all $t \geq 0$, where $\delta_i \in (0,1)$, $c_i = 1 + \sigma_i$, and σ_i is defined by equation (4.10)*, then the NCS is asymptotically stable.*

Proof: By the definition of c_i, equation (4.12) is equivalent to

$$\sigma_i\|e_i(t)\| + \|e_i(t)\| \leq \delta_i\|\hat{x}_i(t)\|. \tag{4.13}$$

for all $t \geq 0$. Therefore, we have

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

for all $t \geq 0$. Since the hypotheses of Theorem 4.2.2 are satisfied, we can conclude that the NCS is aymptotically stable. □

Remark 4.2.5 *In the later discussion, we use the violation of the inequality in* (4.12) *to trigger agents' broadcasts. In equation* (4.12)*, the threshold on the local* *error is fixed between two successive transmissions. It brings convenience to predict the deadlines for the delays, which will be discussed in section 4.3.*

4.2.2 Local Event Design in Linear Systems

This section shows how to implement the distributed scheme proposed in section 4.2.1 for linear systems. We still confine our attention to asymptotic stability. For linear systems, events are designed by solving LMI feasibility problems. With the linear structure, the state equation of agent i is

$$\begin{aligned} \dot{x}_i(t) &= A_{ii}x_i(t) + B_i u_i(t) + \sum_{j\in D_i} A_{ij}x_j(t) \\ u_i(t) &= K_{ii}\hat{x}_i(t) + \sum_{j\in Z_i} K_{ij}\hat{x}_j(t). \end{aligned} \tag{4.14}$$

The state equation of the overall NCS is

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ u(t) &= K\hat{x}(t). \end{aligned} \tag{4.15}$$

In equation (4.15), $x = (x_1^T, \cdots, x_N^T)^T$, $u = (u_1^T, \cdots, u_N^T)^T$, and

$$A = \begin{bmatrix} A_{11} & \cdots & A_{1n} \\ \cdots & \cdots & \cdots \\ A_{n1} & \cdots & A_{nn} \end{bmatrix}, \quad B = \begin{bmatrix} B_1 & \cdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \cdots & B_n \end{bmatrix}, \quad K = \begin{bmatrix} K_{11} & \cdots & K_{1n} \\ \cdots & \cdots & \cdots \\ K_{n1} & \cdots & K_{nn} \end{bmatrix} \tag{4.16}$$

where $A_{ij} = \mathbf{0} \in R^{n\times n}$ if $j \notin D_i$ and $K_{ij} = \mathbf{0} \in R^{m\times n}$ if $j \notin Z_i$.

We first propose a centralized approach to design local events for agents that are used to trigger the broadcast. Linear Matrix Inequalities (LMI) are used to identify the parameters in those events.

Theorem 4.2.4 *Consider the NCS in equation* (4.15). *If the matrices* $P, Q \in \mathbb{R}^{nN \times nN}$ *and* $W_i, M_i \in \mathbb{R}^{n \times n}$, $i = 1, 2, \cdots, N$ *satisfy:*

$$P(A+BK) + (A+BK)^T P \leq -Q \tag{4.17}$$

$$Q - PBKM^{-1}B^T K^T P \geq W \tag{4.18}$$

$$P, Q, M_i, W_i > 0 \tag{4.19}$$

where $M = diag\{M_j\}_{j \in \mathcal{N}}$ *and* $W = diag\{W_j\}_{j \in \mathcal{N}}$. *If for any* $i \in \mathcal{N}$, *the inequality*

$$e_i^T(t) M_i e_i(t) \leq \delta_i x_i^T(t) W_i x_i(t) \tag{4.20}$$

holds for all $t \geq 0$ *with some* $\delta_i \in (0, 1)$, *then the NCS is asymptotically stable.*

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

$$\begin{aligned} \dot{V} &= x^T(PA + A^T P)x + 2x^T PBK\hat{x} \\ &= x^T(P(A+BK) + (A+BK)^T P)x + 2x^T PBKe \end{aligned}$$

Since equation (4.17) holds, the inequality above can be further reduced as:

$$\begin{aligned} \dot{V} &\leq -x^T Q x + 2x^T PBKe \\ &\leq -x^T(Q - PBKM^{-1}K^T B^T P)x + e^T M e \end{aligned}$$

Combining equation (4.18) and the preceding inequality, we have

$$\dot{V} \leq -x^T W x + e^T M e = -\sum_{i \in \mathcal{N}} x_i^T W_i x_i + \sum_{i \in \mathcal{N}} e_i^T M_i e_i. \tag{4.21}$$

Applying equation (4.20) into equation (4.21) yields

$$\dot{V} \ \leq \ -\sum_{i\in\mathcal{N}}(1-\delta_i)x_i^T W_i x_i$$

for any $t \geq 0$, which is sufficient to show that the NCS in equation (4.15) is asymptotically stable. □

It can be shown that the matrices $\{W_j\}_{j\in\mathcal{N}}$ and $\{M_j\}_{j\in\mathcal{N}}$ required in Theorem 4.2.4 always exist, provided equation (4.17) holds (for example, let $W_i = \varepsilon I_{n_i\times n_i}$ and $M_i = \frac{\|PBK\|^2}{\sigma_{\min}(Q)-\varepsilon} I_{n_i\times n_i}$, where $\varepsilon \in (0, \sigma_{\min}(Q))$.

Remark 4.2.6 *Notice that equation* (4.18) *can be rewritten as the following*

$$\begin{bmatrix} Q-W & PBK \\ K^T B^T P & M \end{bmatrix} \geq 0 \tag{4.22}$$

Therefore, equation (4.17), (4.19), (4.22) *form a linear matrix inequalities (LMI), which characterizes the desired matrices.*

Theorem 4.2.4 provides a way to design local events. Agent i can use the violation of the inequality in equation (4.20) to trigger its broadcasts. Directly solving the LMIs in equation (4.17), (4.19), (4.22), 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 that

$$\bar{Z}_i = \{i_1, i_2, \cdots, i_{q_i}\} \subseteq \mathcal{N},$$
$$\bar{Z}_i \cup \bar{D}_i = \{i_1, \cdots, i_{q_i}, i_{q_i+1}, \cdots, i_{s_i}\} \subseteq \mathcal{N}.$$

Therefore, $q_i = |\bar{Z}_i|$ and $s_i = |\bar{Z}_i \cup \bar{D}_i|$. Without loss of the generality, we assume $i_1 = i$. For notational convenience, we define four matrices $A_i \in \mathbb{R}^{n \times ns_i}$, $K_i \in \mathbb{R}^{m \times ns_i}$, and $\tilde{K}_i \in \mathbb{R}^{m \times nq_i}$ by

$$\begin{aligned} A_i &= (A_{i,i_1}, A_{i,i_2}, \cdots, A_{i,i_{s_i}}) \in \mathbb{R}^{n \times ns_i}, \\ K_i &= (K_{i,i_1}, K_{i,i_1}, \cdots, K_{i,i_{s_i}}) \in \mathbb{R}^{m \times ns_i}, \\ \tilde{K}_i &= (K_{i,i_1}, K_{i,i_1}, \cdots, K_{i,i_{q_i}}) \in \mathbb{R}^{m \times nq_i} \end{aligned} \tag{4.23}$$

and two functions $F_i : \mathbb{R}^{n \times n} \to \mathbb{R}^{ns_i \times ns_i}$ and $G_i : \mathbb{R}^{n \times n} \times \mathbb{R} \to \mathbb{R}^{ns_i \times ns_i}$ by

$$F_i(P_i) = \begin{bmatrix} P_i(A_i + B_i K_i) \\ \mathbf{0} \end{bmatrix} \in \mathbb{R}^{ns_i \times ns_i} \tag{4.24}$$

$$G_i(Q_i; \beta) = \begin{bmatrix} Q_i & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & -\beta I & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \cdots & -\beta I \end{bmatrix} \in \mathbb{R}^{ns_i \times ns_i}. \tag{4.25}$$

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

Problem 4.2.1 (Local LMI) *For given constants $\rho, \beta > 0$, find $P_i, Q_i, W_i \in$*

$\mathbb{R}^{n\times n}$ *such that*

$$F_i(P_i) + F_i^T(P_i) + G_i(Q_i;\beta) \leq 0 \tag{4.26}$$

$$\begin{bmatrix} Q_i - |S_i \cup U_i|\beta I_{n\times n} - W_i & P_i B_i \tilde{K}_i \\ \tilde{K}_i^T B_i^T P_i & \rho I_{nq_i \times nq_i} \end{bmatrix} \geq 0 \tag{4.27}$$

$$P_i, W_i > 0. \tag{4.28}$$

The following theorem shows that if the LMI problem 4.2.1 is feasible, then the events can be constructed in a distributed manner.

Theorem 4.2.5 *Consider the NCS in equation* (4.14)*. Given ρ, β, assume that for any $i \in \mathcal{N}$, the local LMI in problem 4.2.1 is feasible and P_i, Q_i, $W_i \in \mathbb{R}^{n\times n}$ are the solutions. If for any $i \in \mathcal{N}$, the inequality*

$$\rho|\bar{U}_i|\|e_i(t)\|_2^2 \leq \delta_i x_i^T(t) W_i x_i(t) \tag{4.29}$$

holds for all $t \geq 0$ with some $\delta_i \in (0,1)$, then the NCS is asymptotically stable.

Proof: Notice that the inequality still holds when we expand the matrices in equation (4.26) into $nN \times nN$ dimension by appropriately adding zero. Summing both sides of the expanded matrix inequalities yields the satisfaction of equation (4.17) with

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

where $Q_i - |S_i \cup U_i|\beta I_{n\times n} > 0$ holds due to equation (4.27).

Similarly, we can show the satisfaction of equation (4.18) with

$$
\begin{aligned}
W &= \mathrm{diag}\{W_i\}_{i=1}^N \\
M &= \mathrm{diag}\{\rho|\bar{U}_i| I_{n\times n}\}_{i=1}^N.
\end{aligned}
$$

Since the hypotheses in Theorem 4.2.4 are satisfied, we conclude that the NCS is asymptotically stable. □

Remark 4.2.7 *Since ρ 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 4.2.8 *The dimensions of the matrices on the lefthand side of the LMIs in* (4.26) *and* (4.27) *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* (4.17) *and* (4.22).

In problem 4.2.1, two parameters, ρ and β, are pre-selected and all agents share the same ρ and β. The following corollaries 4.2.6 and 4.2.7 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 $\{\rho_i\}_{i=1}^N$ and $\{\beta_i\}_{i=1}^N$. The preceding results can be easily modified to handle this more general setup.

Corollary 4.2.6 *Consider the NCS in equation* (4.1)*. For any $i \in \mathcal{N}$, if there exist positive-definite matrices $P_i \in \mathbb{R}^{n\times n}$ such that*

$$
F_i(P_i) + F_i^T(P_i) + G_i(|S_i \cup U_i|\beta I_{n\times n}; \beta) < 0 \tag{4.30}
$$

then there always exists a positive constant $\rho^* \in \mathbb{R}^+$, such that for any $\rho \geq \rho^*$, the LMI in problem 4.2.1 is feasible.

Proof: Equation (4.30) implies that there exists a positive definite matrix $Q_i \in \mathbb{R}^{n\times n}$ such that

$$F_i(P_i) + F_i^T(P_i) + G_i(Q_i;\beta) \leq 0 \tag{4.31}$$

$$Q_i - |S_i \cup U_i|\beta I_{n\times n} > 0 \tag{4.32}$$

Equation (4.32) 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 $\rho^* \in \mathbb{R}^+$ such that for all $\rho \geq \rho^*$, equation (4.27) holds. □

Corollary 4.2.6 suggests that ρ must be large enough to guarantee the feasibility of the local LMI, provided equation (4.30) holds. We still need to know how to select β. In the following corollary, we show that the satisfaction of equation (4.30) is independent of the selection of β.

Corollary 4.2.7 *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* (4.30) *holds, then for any $\hat{\beta} > 0$, the pair $\frac{\hat{\beta}}{\beta}P_i$ and $\hat{\beta}$ also satisfies equation* (4.30).

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

Remark 4.2.9 *Corollary 4.2.7 suggests that the selection of β will not affect* *the existence of P_i satisfying equation* (4.30). *Therefore, if only considering the existence of P_i, we just need to arbitrarily pick a positive constant β.*

4.3 Event-Triggering with Data Dropouts and Transmission Delays

The previous section provides real-time constraints that guarantee asymptotic stability. If there are no dropouts and delays during data transmissions, agents can directly use the violation of the inequality in equation (4.12) to trigger the broadcast. When dropouts and delays are involved, agent i uses the violation of

$$\|x_i(t) - x_i(r_j^i)\|_2 \leq \frac{\delta_i}{c_i}\|x_i(r_j^i)\|_2 \tag{4.33}$$

to trigger broadcast, where r_j^i denotes the time instant when agent i samples and releases the jth broadcast. Notice that the difference between equation (4.33) and (4.12) is that $x_i(r_j^i)$ in (4.33) might be lost during the transmission; while $x_i(b_k^i)$ in (4.12) is always successfully transmitted. Under this triggering mechanism, we provide maximal allowable number of successive dropouts (MANSD) and bounds 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 (4.33) 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

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

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. Otherwise, the broadcast state $\hat{x}_i(t)$ may be inconsistent in subsystems whereas in our analysis we require $\hat{x}_i(t)$ to be consistent in all subsystems.

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 th neighboring agents permission to use the previously transmitted data. Otherwise, the DATA packet is treated as a lost packet. 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 kth successful broadcast ($\{b_k^i\}_{j=1}^{\infty}$ is a subsequence of $\{r_j^i\}_{j=1}^{\infty}$). 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. We assume that transmission of the permission takes $\hat{\tau}_k^i$ seconds. Following this transmission procedure, 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 $\hat{x}_i(t)$, therefore, remains consistent in all subsystems.

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 $\tau_k^i + \hat{\tau}_k^i$ in a way that $c_i\|e_i(t)\|_2 \leq \bar{\delta}_i\|x_i(t)\|_2$ is always valid with some $\bar{\delta}_i \in (\delta_i, 1)$ 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 kth successful broadcast is completed. This suggests that we only need to ensure $c_i \|x_i(t) - x_i(b_k^i)\|_2 \leq \bar{\delta}_i \|x_i(b_k^i)\|_2$ 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, $c_i\|e_i(t)\|_2 \leq \bar{\delta}_i \|\hat{x}_i(t)\|_2$, is still valid over that interval. To determine bounds on τ_k^i and $\hat{\tau}_k^i$, we focus on the interval $[b_{k+1}^i, f_{k+1}^i)$ because τ_k^i and $\hat{\tau}_k^i$ are basically transmission delays. The parameters τ_k^i and $\hat{\tau}_k^i$ are associated with two different types of delays (delay in transmitting DATA packets and delay in sending PERM message). To obtain the constraints on τ_k^i and $\hat{\tau}_k^i$, we just 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 4.3.1) describes the behavior of $e_i(t)$ over $[f_k^i, b_{k+1}^i)$ when data dropouts happen. The second lemma (Lemma 4.3.2) considers the effect of delays on the convergence of the overall system.

Lemma 4.3.1 *Consider the NCS in equation* (4.1). *Suppose that assumption 4.2.1 holds. Given two collections of positive constants* $\delta_i \in (0,1)$ *and* $\varrho_i \in [\delta_i, 1)$ *for* $i = 1, 2, \cdots, N$, *if for any* $i \in \mathcal{N}$, *the next broadcast release time,* r_{j+1}^i, *is triggered by the violation of*

$$c_i \|\varepsilon_i^j(t)\|_2 \leq \delta_i \|x_i(r_j^i)\|_2, \tag{4.34}$$

where $c_i = 1+\sigma_i$, *and* σ_i *is defined by equation* (4.10), *and the number of successive* *dropouts, $d_k^i \in \mathbb{Z}$, satisfies*

$$d_k^i \leq d_{\text{MANSD}}^i \triangleq \left\lfloor \log_{\left(1+\frac{\delta_i}{c_i}\right)} \left(1 + \frac{\varrho_i}{c_i}\right) - 1 \right\rfloor, \tag{4.35}$$

then the inequality

$$\|x_i(t) - x_i(b_k^i)\|_2 \leq \frac{\xi_i(d_k^i)}{c_i} \|x_i(b_k^i)\|_2 \leq \frac{\varrho_i}{c_i} \|x_i(b_k^i)\|_2 \tag{4.36}$$

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

$$\xi_i(d_k^i) \triangleq c_i \left(1 + \frac{\delta_i}{c_i}\right)^{d_k^i + 1} - c_i \in (0, \varrho_i). \tag{4.37}$$

Proof: The proof is in Appendix A.13. □

Remark 4.3.1 *If all the hypotheses in Lemma 4.3.1 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 (4.36), 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 4.2.3.*

Lemma 4.3.2 *Consider the NCS in equation (4.1) with $w_i = 0$ for all $i \in \mathcal{N}$. Suppose that assumption 4.2.1 holds and α_1^i, α_2^i in equation (4.6) satisfy*

$$\begin{aligned} \alpha_1^i(\|x_i\|_2) &\geq \underline{L}_i \|x_i\|_2^q \\ \alpha_2^i(\|x_i\|_2) &\leq \bar{L}_i \|x_i\|_2^q, \end{aligned} \tag{4.38}$$

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 $i = 1, 2, \cdots, N$ such that*

$$\|g_i\left(x_{\bar{D}_i}(t), \kappa_i\left(\hat{x}_{\bar{Z}_i}(t)\right)\right)\|_2 \leq \theta_i, \tag{4.39}$$

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 $\delta_i \in (0, 1)$, $\varrho_i \in [\delta_i, 1)$ for $i = 1, 2, \cdots, N$, if for any $i \in \mathcal{N}$, the broadcast release time r^i_{j+1} is triggered by the violation of the inequality in equation (4.34), the number of successive dropouts, d^i_k, satisfies equation (4.35), and the delay in the $k + 1$st successful transmission satisfies

$$f^i_{k+1} - b^i_{k+1} \leq \frac{1 - \xi_i(d^i_k)}{c_i \theta_i} \max\left\{\frac{\|x_i(b^i_k)\|_2}{2}, \Delta\right\} \tag{4.40}$$

where ξ_i is defined in equation (4.37), then there exists $T \geq t_0$ such that

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

holds for all $t \geq T$, where

$$\mu = \begin{cases} 1 & p \leq q \\ N^{1-\frac{q}{p}} & p > q \end{cases} \tag{4.41}$$

$$\pi = \left(\frac{\sum_{i \in \mathcal{N}} \left(\zeta_i - |S_i \cup U_i|\beta_i\right)(1 - \delta_i)}{\min_{i \in \mathcal{N}}\{\left(\zeta_i - |S_i \cup U_i|\beta_i\right)(1 - \bar{\varsigma}_i)\}}\right)^{\frac{1}{p}} \tag{4.42}$$

$$\bar{\varsigma}_i = \max\left\{\left(\frac{1 + \varrho_i}{2}\right)^p, \ \varrho_i\right\}. \tag{4.43}$$

Proof: The proof is in Appendix A.14. □

Lemma 4.3.2 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 (4.39) in Lemma 4.3.2. 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 4.2.1 holds and g_i, κ_i are locally Lipschitz for all $i \in \mathcal{N}$. Then we can define a compact set, $\Lambda \subset \mathbb{R}^{nN}$, as

$$\Lambda \triangleq \left\{x \in \mathbb{R}^{nN} \mid V(x) \leq V(x_0)\right\} \tag{4.44}$$

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

$$\|g_i(x_{\bar{D}_i}, \kappa_i(\hat{x}_{\bar{Z}_i}))\|_2 \leq L_i \sum_{i \in \mathcal{N}} (\|x_i\|_2 + \|\hat{x}_i\|_2), \;\; \forall x, \hat{x} \in \Lambda \tag{4.45}$$

$$\alpha_1^i(\|x_i\|_2) \geq \underline{L}_i \|x_i\|_2^q, \;\; \forall x \in \Lambda$$

$$\alpha_2^i(\|x_i\|_2) \leq \bar{L}_i \|x_i\|_2^q, \;\; \forall x \in \Lambda \tag{4.46}$$

$$\theta_i = 2 L_i N^{\frac{q-1}{q}} \left(\frac{V(x_0)}{\min_{i \in \mathcal{N}} \underline{L}_i} \right)^{\frac{1}{q}} \tag{4.47}$$

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 4.3.3 *Consider the NCS in equation* (4.1) *with* $w_i = 0$ *for all* $i \in \mathcal{N}$. *Suppose that assumption 4.2.1 and equation* (4.45), (4.46) *hold with some* $q \geq 1$. *Given positive constants* $\delta_i \in (0, 1)$, $\varrho_i \in [\delta_i, 1)$, $\bar{\theta}_i \in [\theta_i, \infty)$ *for all* $i \in \mathcal{N}$ *and* $\bar{\pi} \in (\pi, \infty)$, *where* θ_i, π *are defined in equation* (4.47) *and* (4.42), *respectively, if for any* $i \in \mathcal{N}$, *the broadcast release time* r_{j+1}^i *is triggered by the violation of the inequality in equation* (4.34), *the number of successive data dropouts,* $d_k^i \in \mathbb{Z}$, *satisfies equation* (4.35), *and the delay in the* $k+1$*st successful transmission* *satisfies*

$$f^i_{k+1} - b^i_{k+1} \leq \eta^i_k \triangleq \max\left\{ \frac{(1-\xi_i(d^i_k))}{2c_i\bar{\theta}_i}\|x_i(b^i_k)\|_2, \frac{(1-\xi_i(d^i_k))\min_i \underline{L}_i^{\frac{1}{q}}}{2c_i L_i \bar{\pi} N^{1-\frac{1}{\max\{p,q\}}} \max_i \bar{L}_i^{\frac{1}{q}}} \right\} \quad (4.48)$$

where L_i, $\bar{L}_i$, $\underline{L}_i$, ξ_i *are defined in equation* (4.45), (4.46), (4.37), *respectively, then* $x(t) \in \Lambda$ *for all* $t \geq t_0$, *where* Λ *is defined in equation* (4.44), .

Proof: The proof is in Appendix A.15. □

With these three lemmas, we now can present the main theorem.

Theorem 4.3.4 *If the hypotheses in Lemma 4.3.3 hold, the NCS is asymptotically stable.*

Proof: The proof is in Appendix A.16. □

Remark 4.3.2 η^i_k *in equation* (4.48) *serves as the deadline for the kth successful broadcast of agent i. With the fact that* $\xi_i(d^i_k) \leq \varrho_i$ *holds, we have* η^i_k *always greater than a positive constant,* τ^i_{SPD}. *In other words,*

$$\eta^i_k \geq \tau^i_{\text{SPD}} = \frac{(1-\varrho_i)\min_i \underline{L}_i^{\frac{1}{q}}}{2c_i L_i \bar{\pi} N^{1-\frac{1}{\max\{p,q\}}} \max_i \bar{L}_i^{\frac{1}{q}}} > 0 \quad (4.49)$$

holds for all $k \in \mathbb{N}$. τ^i_{SPD} *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 4.3.3 *The number of successive dropouts,* d^i_k *determines the deadline* η^i_k. *As* d^i_k *increases, the value of* $\xi_i(d^i_k)$ *increases. It, therefore, results in a short deadline according to equation* (4.48). *There is a trade-off between the number of successive dropouts and the deadline.*

Remark 4.3.4 *Two parameters δ_i, ϱ_i are used in the scheme. The parameter δ_i determines d^i_{MANSD}, τ^i_{SPD}, and the transmission periods, T^i_j. The large δ_i is, the longer T^i_j is and the smaller d^i_{MANSD} is, according to equation* (4.34) *and* (4.35)*. Large δ_i may also result in a small π according to equation* (4.42) *and therefore leads to a larger τ^i_{SPD}. The parameter ϱ_i determines d^i_{MANSD} and τ^i_{SPD}. The large ϱ_i is, the larger d^i_{MANSD} is and the smaller τ^i_{SPD} is. As a "rule of thumb", a reasonable strategy is to choose δ_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 4.3.5 *To design a system with special requirements on d^i_{MANSD} and τ^i_{SPD}, we solve equation* (4.35) *and* (4.48) *for δ_i and ϱ_i. There is no constraint on the selection of d^i_{MANSD}. It can be arbitrarily large. τ^i_{SPD}, however, must be less than some positive constant so that equation* (4.35) *and* (4.48) *have solutions.*

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

Corollary 4.3.5 *If the hypotheses in Lemma 4.3.3 hold, then*

$$T^i_j = r^i_{j+1} - r^i_j \geq \frac{\delta_i}{c_i \theta_i} \|x_i(r^i_j)\|_2$$

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

Proof: By Lemma 4.3.3, we have

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

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

$$\frac{d}{dt}\left\|\varepsilon_i^j(t)\right\|_2 \leq \left\|\dot{\varepsilon}_i^j(t)\right\|_2 = \|\dot{x}_i(t)\|_2 = \|g_i\left(x_{\bar{D}_i}, \kappa_i\left(\hat{x}_{\bar{Z}_i}\right)\right)\|_2 \leq \theta_i$$

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

$$\left\|\varepsilon_i^j(t)\right\|_2 \leq \theta_i(t - r_j^i) \tag{4.50}$$

holds for all $t \in [r_j^i, r_{j+1}^i)$. Since $\left\|\varepsilon_i^j(r_{j+1}^i)\right\|_2 = \frac{\delta_i}{c_i}\|x_i(r_j^i)\|_2$, equation (4.50) implies

$$\left\|\varepsilon_i^j(r_{j+1}^i)\right\|_2 = \frac{\delta_i}{c_i}\|x_i(r_j^i)\|_2 \leq \theta_i(r_{j+1}^i - r_j^i),$$

which means $r_{j+1}^i - r_j^i \geq \frac{\delta_i}{c_i\theta_i}\|x_i(r_j^i)\|_2$. □

Remark 4.3.6 *It is easy to see that in Corollary 4.3.5, 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 section 4.4, the simulation results show that the periods are always greater than a positive constant. It suggests that the bound in Corollary 4.3.5 may be conservative.*

Remark 4.3.7 *With the bound on periods given in Corollary 4.3.5, the MANSD given in equation* (4.35)*, and the deadline given in equation* (4.48)*, 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.*

Based on the preceding results, we are able to present the distributed event design scheme and the distributed event-triggering scheme.

Distributed Event Design Scheme (DEDS)

1. Select positive constants p, q and $\beta_i, \rho_i, \gamma_i \in \mathbb{R}^+$ for $i = 1, \cdots, N$;
2. For agent i,
   (1) Find $V_i : \mathbb{R}^n \to \mathbb{R}$, α_1^i, $\alpha_2^i : \mathbb{R} \to \mathbb{R}$, $\zeta_i \in \mathbb{R}^+$, and $\kappa_i : \mathbb{R}^{n|\bar{Z}_i|} \to \mathbb{R}^m$ satisfying equation (4.6), (4.7), (4.8);
   (2) Compute σ_i according to equation (4.10) and $c_i = 1 + \sigma_i$;
   (3) Select $\delta_i \in (0, 1)$, $\varrho_i \in (\delta_i, 1)$, and identify d^i_{MANSD} by (4.35);
   (4) Compute L_i, $\underline{L}_i$, $\bar{L}_i$ satisfying equation (4.45), (4.46), respectively;
   (5) Select $\bar{\theta}_i$ large enough so that $\bar{\theta}_i \geq \theta_i$, where θ_i is defined in (4.47);
   (6) Compute $\bar{\varsigma}_i$ according to equation (4.43);
   (7) Identify $\bar{\pi} \in (\pi, \infty)$, where π is defined in equation (4.42);
   (8) Identify $\max_{i\in\mathcal{N}} \bar{L}_i$ and $\min_{i\in\mathcal{N}} \underline{L}_i$.

The DEDS is an offline design process. It computes the parameters used in the following online distributed event-triggering scheme.

Distributed Event-Triggering Scheme (DETS) for Agent i

When the violation of the inequality in equation (4.34) occurs at time r^i_j,

1 sample and broadcast $x_i(r^i_j)$,

2 update the triggering event using the newly sampled state $x_i(r^i_j)$,

3 compute the deadline η^i_k according to equation (4.48),

4 determine τ^i_k, $\hat{\tau}^i_k$ such that $\tau^i_k + \hat{\tau}^i_k \leq \eta^i_k$,

5 If receiving confirmations from all agents in U_i within τ^i_k seconds, sends out permission of using $x_i(r^i_j)$ within $\hat{\tau}^i_k$ seconds,

6 If the permission is sent out, update u_i with $x_i(r^i_j)$.

Remark 4.3.8 *It is obvious that the event-triggering scheme is completely decentralized. In the event design scheme, however, global coordination is required identify the value of* L_i, $\bar{L}_i$, $\underline{L}_i$, θ_i, $\bar{\pi}$, $\max_{i\in\mathcal{N}} \bar{L}_i$, *and* $\min_{i\in\mathcal{N}} \underline{L}_i$.

4.4 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 4.3). Both soft spring models (nonlinear) and normal spring models (linear) are considered. The state of the ith subsystem is the vector $x_i = \begin{bmatrix} y_i & \dot{y}_i \end{bmatrix}^T$ where y_i is the ith 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 $i+1$.

This section consists of four subsections. Subsection 4.4.1 shows how to implement our scheme in nonlinear systems. Subsection 4.4.2 investigates the ro-

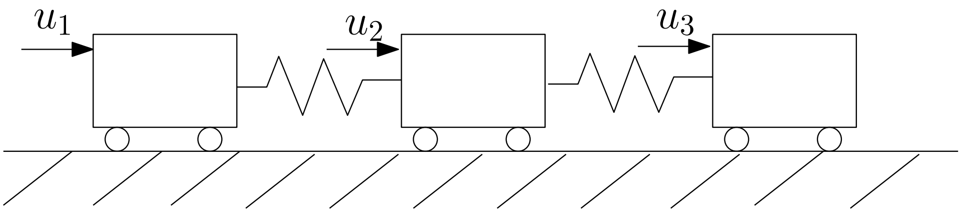

Figure 4.3. Three carts coupled by springs

bustness of the event-triggering scheme to transmission delays, data dropouts, and exogenous disturbances. Subsection 4.4.3 explores how the parameters in the scheme affects transmission periods, the MANSD, and the SPD. Finally, subsection 4.4.4 examines the communication cost and the complexity of our scheme by comparing it against the approach in [37] 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.

4.4.1 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 [18]. The state equation for the ith cart is

$$\dot{x}_i = \begin{bmatrix} \dot{y}_i \\ u_i + \nu_i^1 \tanh(y_{i+1} - y_i) + \nu_i^2 \tanh(y_{i-1} - y_i) \end{bmatrix} \tag{4.51}$$

In the preceding equation, $\nu_1^2 = \nu_N^1 = 0$. Otherwise, $\nu_i^1 = \nu_i^2 = 1$.

The control input of subsystem i is

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

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

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

$$\begin{aligned}
&-0.1\|x_1(r_j^i)\|_2 + 3.6847\|\varepsilon_1^j(t)\|_2 = 0 \\
&-0.1\|x_i(r_j^i)\|_2 + 5.7202\|\varepsilon_i^j(t)\|_2 = 0, \text{ for } i = 2, \cdots, N-1 \\
&-0.1\|x_N(r_j^N)\|_2 + 3.8700\|\varepsilon_N^j(t)\|_2 = 0
\end{aligned}$$

according to equation (4.34) and the MANSDs for agents are all 5 according to equation (4.35).

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 (4.48). The initial state x_0 was randomly generated satisfying $\|x_0\|_\infty \leq 1$. From the top plot of Figure 4.4, 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 4.4 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 the periods are always greater than a positive constant, even when the states are close to the equilibrium. This is important because it shows that our scheme can avoiding

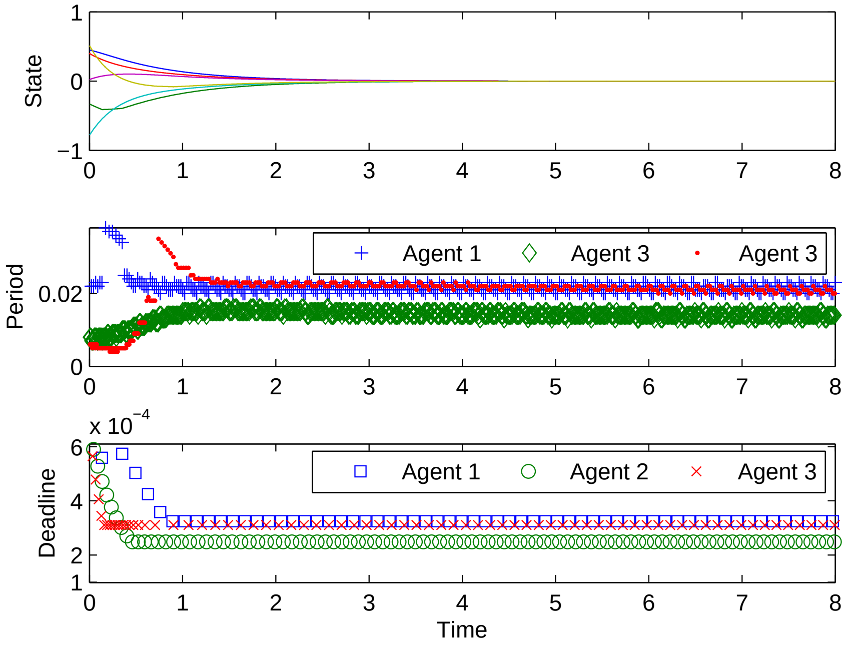

Figure 4.4. State trajectory, broadcast periods, and predicted deadlines in an event-triggered NCS

infinitely fast broadcasting. On the other hand, it means the lower bound on periods obtained in Corollary 4.3.5 might be conservative. The bottom plot in Figure 4.4 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, $\frac{(1-\xi_k^i)\min_i \underline{L}_i^{\frac{1}{q}}}{2c_i L_i \bar{\pi} N^{1-\frac{1}{\max\{p,q\}}} \max_i \bar{L}_i^{\frac{1}{q}}}$ dominates the deadlines. The SPDs of agent 1, 2, 3 are 400 μs, 200 μs, and 300 μs, respectively.

4.4.2 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 4.4.1 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 (400 μs, 200 μs, and 300 μs in agent 1,2,3, respectively) are around 15% of the actual allowable delay.

The next simulation used the model in section 4.4.1 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 $N = 10$. $\delta_i = \varrho_i = 0.9$ for

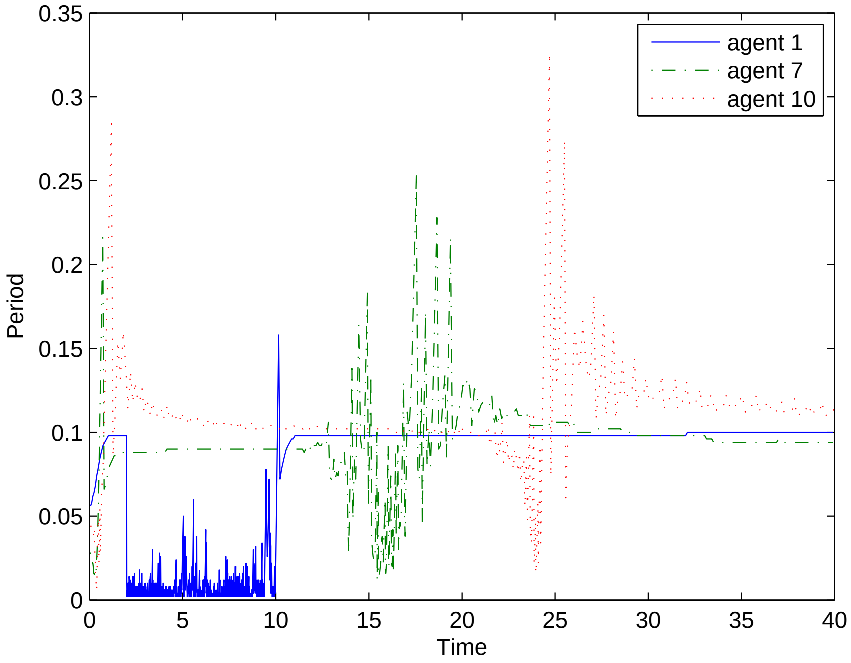

Figure 4.5. Successful broadcast periods versus time in an event-triggered NCS with disturbances in agent 1

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 4.5. 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 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 decease 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.

4.4.3 Selection of Parameters

The simulations in this subsection examined the effect of parameters δ_i and ϱ_i on the broadcast periods, the MANSDs, 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 MANSD. The parameters $\delta_2 = \delta_3 = 0.1$ and $\varrho_2 = \varrho_3 = 0.9$.

We first fixed $\delta_1 = 0.1$ and varied ϱ_1 from 0.1 to 0.9. The simulation results are shown in Figure 4.6. The top plot in Figure 4.6 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 4.6 is the SPD versus ϱ_1. We can see that when ϱ_1 increases, the SPD

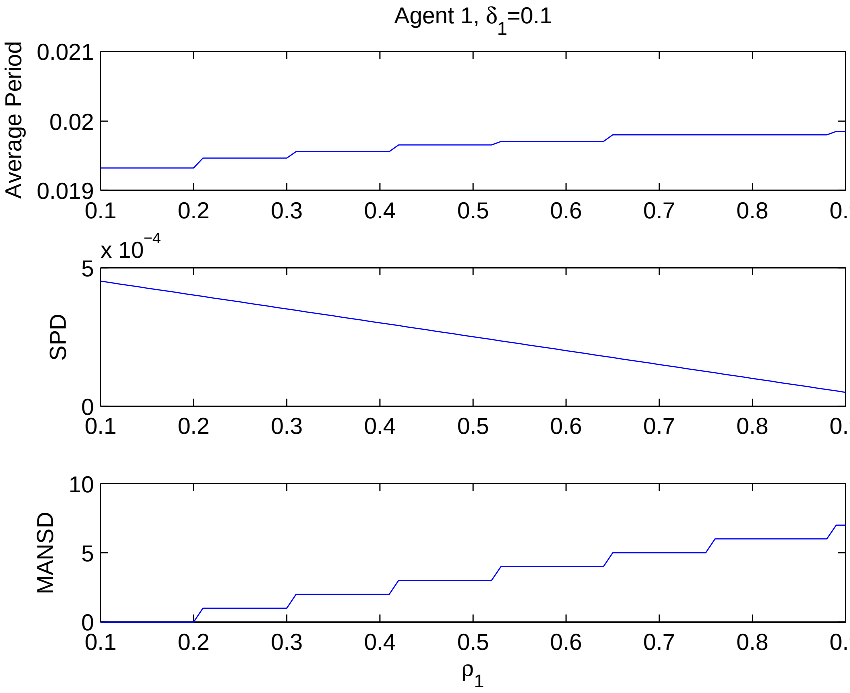

Figure 4.6. The average period, the SPD, and the MANSD in agent 1 versus ϱ_1

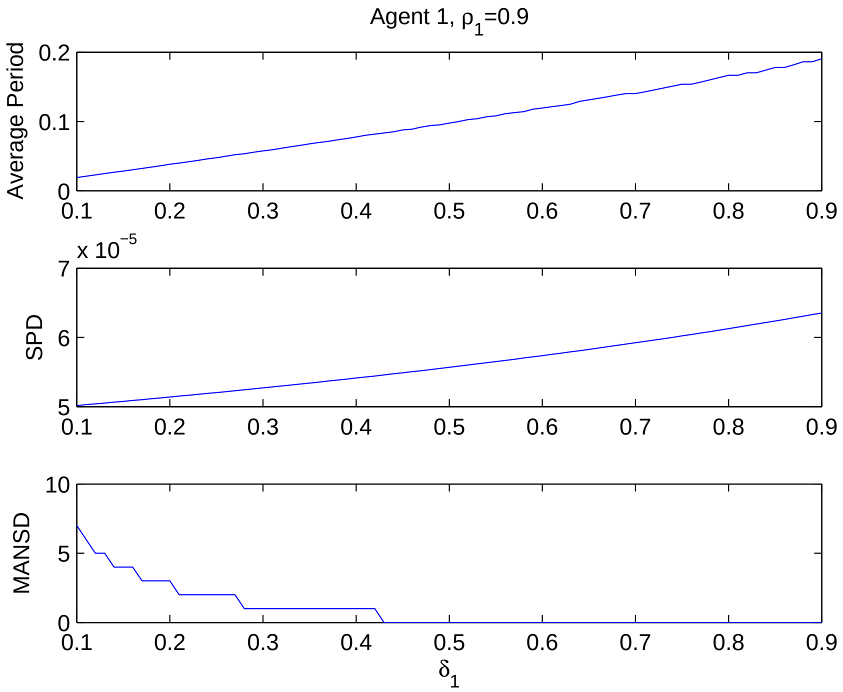

Figure 4.7. The average period, the SPD, and the MANSD in agent 1 versus δ_1

decreases. Therefore, to have a longer SPD, we need a small ϱ_i. The bottom plot in Figure 4.6 is the MANSD versus ϱ_1. As ϱ_1 increases, the MANSD increases, which means ϱ_i has to be large to ensure large MANSD.

We then fixed $\varrho = 0.9$ and varied δ_1 from 0.1 to 0.9. The simulation results are shown in Figure 4.7. The top plot in Figure 4.6 is the average broadcast periods versus δ_1. As δ_1 increases, the average period increases. It implies that to obtain long periods, δ_1 needs to be large. The middle plot in Figure 4.6 is the SPD versus δ_1. We can see that when δ_1 increases, the SPD increases. The bottom plot in Figure 4.6 is the MANSD versus δ_1. As ϱ_1 increases, the MANSD decreases. These simulations verify the comments in Remark 4.3.4. It suggests a tradeoff between the broadcast periods, the MANSD, and the SPD.

4.4.4 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 [37] that derives the bound on the MATI. The simulations were done using normal spring models (linear). The state equation of the ith cart is

$$\begin{aligned} \dot{x}_i &= \begin{bmatrix} \dot{y}_i \\ u_i + \kappa_i^1(y_{i+1} - y_i) + \kappa_i^2(y_{i-1} - y_i) \end{bmatrix} \\ u_i &= K_i\hat{x}_i - \kappa_i^1(\hat{y}_{i+1} - \hat{y}_i), \end{aligned} \tag{4.52}$$

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

We set $\rho = 10$ and $\beta = 1$ and solved local LMI problems 4.2.1 using MATLAB toolbox. With $\delta_i = 0.9$ for $\forall i \in \mathcal{N}$, the events are

$$\begin{aligned} &-0.9\|x_1(r_j^i)\|_2 + 2.5908\|\varepsilon_1^j(t)\|_2 = 0 \\ &-0.9\|x_i(r_j^i)\|_2 + 4.1626\|\varepsilon_i^j(t)\|_2 = 0, \text{ for } i = 2, \cdots, N-1 \\ &-0.9\|x_N(r_j^N)\|_2 + 3.7833\|\varepsilon_N^j(t)\|_2 = 0 \end{aligned}$$

We first compared the average broadcast period generated by our scheme with the MATI in [37]. Recall that for a NCS containing N subsystems, the MATI in

[37], denoted as T^N_{MATI}, is,

$$T^N_{\text{MATI}} = \frac{1}{L} \ln \frac{L+\gamma}{\delta L + \gamma}, \tag{4.53}$$

where, with TOD protocol, $\delta = \sqrt{\frac{N-1}{N}}$, $L = \max(0.5\lambda_{\max}(-BK - K^T B^T), 0)$, 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 (4.15).

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

$$\bar{T}_N = \frac{\text{System Runtime}}{\text{Total Number of Broadcasts}}. \tag{4.54}$$

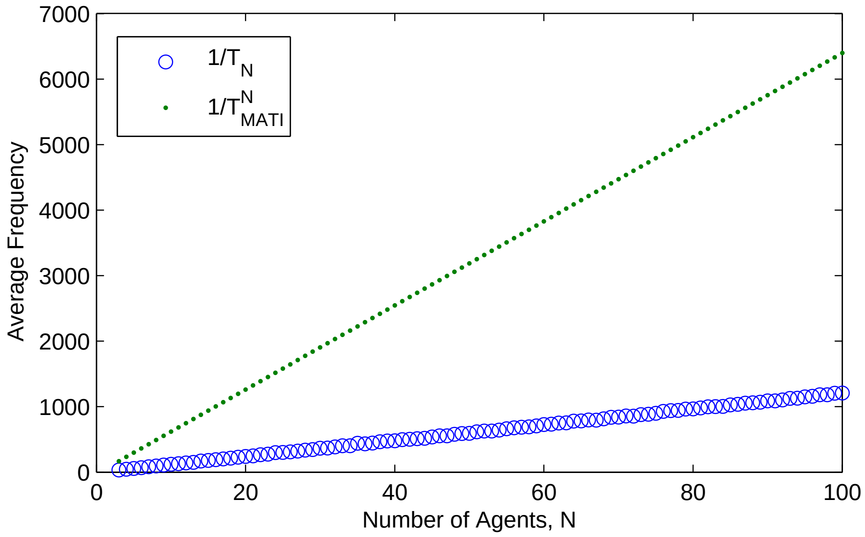

Figure 4.8. The comparison between the bound on the MATI in [37] and the average period generated by our scheme

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^N_{MATI} as N varied. In the simulation, the MATI, T^N_{MATI}, 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. Figure 4.8 plots the number of agents versus the average broadcast frequency generated by our scheme and the approach in [37]. 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 $\frac{1}{T^N_{\mathrm{MATI}}}$ 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 [37]. 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 (4.53), 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.

4.5 Summary

This chapter 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 networks 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.

CHAPTER 5

Future Work

This chapter discusses some research problems in event-triggered networked control systems (NCS) that have not been completely addressed in our work. The first problem is the robustness of event-triggered NCSs. It is discussed in Section 5.1. Section 5.2 studies the effect of quantization on event-triggered NCSs.

5.1 Distributed Event Design for Finite-Gain $\mathcal{L}_p$ Stability

This section discusses the robustness of event-triggered NCSs. This problem has been partially addressed in [51, 57]. This work presents a distributed event design scheme that guarantees finite-gain $\mathcal{L}_p$ stability of the resulting NCS. The analysis is based on the assumption that the transmission delays are zero.

Consider a distributed NCS containing N agents. The state equation of agent i is

$$\begin{aligned} \dot{x}_i(t) &= g_i(x_{\bar{D}_i}(t), u_i(t), w_i(t)) \\ u_i(t) &= \kappa_i(\hat{x}_{\bar{Z}_i}(t)) \\ x_i(t_0) &= x_{i0}. \end{aligned} \tag{5.1}$$

The objective is to develop distributed event-triggering schemes to identify

$\{r^i_j\}_{j=1}^{\infty}$, $\{b^i_k\}_{k=1}^{\infty}$, and $\{f^i_k\}_{k=1}^{\infty}$ such that the NCS defined in equation (5.1) is finite-gain $\mathcal{L}_p$ stability.

5.1.1 Real-Time Constraints in Nonlinear Systems

This section proposes a distributed approach for nonlinear systems to construct local events such that the resulting event-triggered NCS is finite-gain $\mathcal{L}_p$ stable. For notational convenience, we let $L_{g_i}V = \frac{\partial V}{\partial x_i} g_i(x_{\bar{D}_i}, \kappa_i(x_{\bar{Z}_i} - e_{\bar{Z}_i}), w_i)$ with some function $V(x) \in \mathbb{R}$.

Assumption 5.1.1 *Assume that, for any $i \in \mathcal{N}$, there exist a constant $p \geq 1$, a continuous, positive-definite functions $V_i : \mathbb{R}^n \to \mathbb{R}$, class $\mathcal{K}$ functions α^i_1, α^i_2 : $\mathbb{R} \to \mathbb{R}$, positive constants $\zeta_i, \beta_i, \rho_i, \gamma_i \in \mathbb{R}$, and control law $\kappa_i : \mathbb{R}^{n|\bar{Z}_i|} \to \mathbb{R}^{m_i}$ satisfying*

$$\alpha^i_1(\|x_i\|_2) \leq V_i(x_i) \leq \alpha^i_2(\|x_i\|_2) \tag{5.2}$$

$$L_{g_i}V_i \leq -\zeta_i\|x_i\|_2^p + \sum_{j\in D_i\cup Z_i} \beta_j\|x_j\|_2^p + \sum_{j\in\bar{Z}_i} \rho_j\|e_j\|_2^p + \gamma_i^p\|w_i\|_2^p \tag{5.3}$$

$$\zeta_i - |S_i \cup U_i|\beta_i > 0. \tag{5.4}$$

Theorem 5.1.1 *Consider the NCS defined in equation* (5.1)*. Suppose that assumption 5.1.1 holds. If for any $i \in \mathcal{N}$, the inequality*

$$-\delta_i\|x_i(t)\|_2 + \sigma_i\|e_i(t)\|_2 \leq 0 \tag{5.5}$$

holds for all $t \geq 0$, where $\delta_i \in (0, 1)$ and

$$\sigma_i = \left(\frac{|\bar{U}_i|\rho_i}{\zeta_i - |S_i \cup U_i|\beta_i}\right)^{\frac{1}{p}}, \tag{5.6}$$

then the NCS is finite-gain $\mathcal{L}_p$ stable from w to x.

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

$$\begin{aligned}
\dot{V} &= \sum_{i\in\mathcal{N}} L_{g_i} V_i \\
&\le \sum_{i\in\mathcal{N}} \left[-\zeta_i \|x_i\|_2^p + \sum_{j\in D_i\cup Z_i} \beta_j \|x_j\|_2^p + \sum_{j\in \bar{Z}_i} \rho_j \|e_j\|_2^p + \gamma_i^p \|w_i\|_2^p \right] \\
&= \sum_{i\in\mathcal{N}} \left[-\left(\zeta_i - |S_i \cup U_i| \beta_i\right) \|x_i\|_2^p + \rho_i |\bar{U}_i| \|e_i\|_2^p + \gamma_i^p \|w_i\|_2^p \right],
\end{aligned}$$

where the equality is obtained by resorting the items according to index i.

Applying equation (5.5) into the preceding equation yields

$$\dot{V} \le \sum_{i\in\mathcal{N}} [-(1-\delta_i^p)\left(\zeta_i - |S_i \cup U_i|\beta_i\right) \|x_i\|_2^p + \gamma_i^p \|w_i\|_2^p,] \tag{5.7}$$

which is, with equation (5.4), sufficient to show the NCS is finite-gain $\mathcal{L}_p$ stable from w to x. □

5.1.2 Real-Time Constraints in Linear Systems

This section shows how to implement the distributed scheme proposed in Section 5.1.1 in linear systems. For linear systems, we set $p = 2$. The local event design is characterized by a local LMI problem. With the linear structure, agent i has the following state equation,

$$\begin{aligned}
\dot{x}_i(t) &= A_{ii} x_i(t) + B_i u_i(t) + \sum_{j\in D_i} A_{ij} x_j(t) + C_i w_i(t) \\
u_i(t) &= K_{ii} \hat{x}_i(t) + \sum_{j\in Z_i} K_{ij} \hat{x}_j(t)
\end{aligned} \tag{5.8}$$

Therefore, the state equation of the overall NCS is

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + Cw(t) \\ u(t) &= K\hat{x}(t). \end{aligned} \tag{5.9}$$

We first propose a centralized approach to design local events for agents that are used to trigger the broadcast. Linear Matrix Inequalities (LMI) are used to identify the parameters in those events. The resulting event-triggered NCS is finite-gain $\mathcal{L}_2$ stable, shown in Theorem 5.1.2.

Theorem 5.1.2 *Consider the NCS in equation* (5.9). *Assume that there exist positive-definite matrices* $P, Q \in \mathbb{R}^{nN \times nN}$ *and* $W_i, M_i \in \mathbb{R}^{n \times n}$, $i = 1, 2, \cdots, N$ *such that:*

$$\begin{bmatrix} -P(A+BK) - (A+BK)^T P - Q & PC \\ C^T P & \gamma^2 I_{lN \times lN} \end{bmatrix} \geq 0, \tag{5.10}$$

$$\begin{bmatrix} Q - W & PBK \\ K^T B^T P & M \end{bmatrix} \geq 0, \tag{5.11}$$

$$P, Q, M_i, W_i > 0, \tag{5.12}$$

hold, where $M = diag\{M_j\}_{j \in \mathcal{N}}$ *and* $W = diag\{W_j\}_{j \in \mathcal{N}}$. *If for any* $i \in \mathcal{N}$, *the inequality*

$$a_i \|e_i(t)\|_2 \leq \delta_i \|\hat{x}_i(t)\|_2 \tag{5.13}$$

holds for all $t \geq 0$, *where* $a_i = 1 + \sqrt{\frac{\lambda_{\max}(M_i)}{\lambda_{\min}(W_i)}}$ *with some* $\delta_i \in (0, 1)$, *then the NCS is finite-gain* $\mathcal{L}_2$ *stable from* w *to* x.

Proof: The proof is in Appendix A.17. □

Theorem 5.1.2 presents a method to design local events. Equation (5.10), (5.11), (5.12) is a highly centralized approach that may not be suitable for large-scale systems. In the later discussion, we focus on distributed event design for NCS. Each agent is associated with a local LMI problem. The feasibility of those local LMI implies the feasibility of the centralized LMI in equation (5.12), (5.10), (5.11). The solutions to local LMI can be used to construct the solution to the centralized LMI.

Let us look at agent i. Assume that

$$\bar{Z}_i = \{i_1, i_2, \cdots, i_{q_i}\} \subseteq \mathcal{N},$$
$$\bar{Z}_i \cup \bar{D}_i = \{i_1, \cdots, i_{q_i}, i_{q_i+1}, \cdots, i_{s_i}\} \subseteq \mathcal{N}.$$

It is easy to see that $q_i = |\bar{Z}_i|$ and $s_i = |\bar{Z}_i \cup \bar{D}_i|$. Without loss of the generality, we assume $i_1 = i$.

The local LMI problem associated with agent i is defined in the following way:

Problem 5.1.1 (Local LMI) *For given constants $\rho, \beta > 0$, find $P_i, Q_i, W_i \in \mathbb{R}^{n\times n}$ and $\gamma_i \in \mathbb{R}$ such that*

$$\begin{bmatrix} -F_i(P_i) - F_i^T(P_i) - G_i(Q_i;\beta) & H_i \\ H_i^T & \gamma_i^2 I_{l\times l} \end{bmatrix} \geq 0 \tag{5.14}$$

$$\begin{bmatrix} Q_i - |S_i \cup U_i|\beta I_{n\times n} - W_i & P_i B_i \tilde{K}_i \\ \tilde{K}_i^T B_i^T P_i & \rho I_{nq_i \times nq_i} \end{bmatrix} \geq 0 \tag{5.15}$$

$$P_i, W_i > 0, \gamma_i > 0, \tag{5.16}$$

where $H_i = \begin{bmatrix} (P_iC_i)^T & \mathbf{0} \end{bmatrix}^T \in \mathbb{R}^{ns_i \times l}$ and $\tilde{K}_i \in \mathbb{R}^{m\times nq_i}$, $F_i : \mathbb{R}^{n\times n} \to \mathbb{R}^{ns_i \times ns_i}$,

$G : \mathbb{R}^{n\times n} \times \mathbb{R} \to \mathbb{R}^{ns_i \times ns_i}$ are defined in equation (4.23), (4.24), (4.25), respectively.

The following theorem shows that with the distributed event-design approach, the resulting event-triggered NCS is $\mathcal{L}_2$ stable.

Theorem 5.1.3 *Consider the NCS in equation* (5.8). *Assume that for any $i \in \mathcal{N}$, the local LMI in problem 5.1.1 is feasible and P_i, Q_i, $W_i \in \mathbb{R}^{n\times n}$, and $\gamma_i \in \mathbb{R}$ are the solutions. If for any $i \in \mathcal{N}$, the inequality*

$$b_i \|e_i(t)\|_2 \le \delta_i \|\hat{x}_i(t)\|_2 \tag{5.17}$$

holds for all $t \ge 0$, where $b_i = 1 + \sqrt{\frac{\rho|\bar{U}_i|}{\lambda_{\min}(W_i)}}$ and $\delta_i \in (0, 1)$, then the NCS is finite-gain $\mathcal{L}_2$ stable from w to x.

Proof: The proof is in Appendix A.18. □

Remark 5.1.1 *Similar to Corollary 4.2.6, for the NCS in equation* (5.1), *if there exist positive-definite matrices $P_i \in \mathbb{R}^{n\times n}$ such that*

$$F_i(P_i) + F_i^T(P_i) + G_i(|S_i \cup U_i|\beta I_{n\times n}; \beta) < 0 \tag{5.18}$$

then there always exists a positive constant $\rho^* \in \mathbb{R}^+$, such that for any $\rho \ge \rho^*$, the LMI in problem 5.1.1 is feasible.

Remark 5.1.2 *Similar to Corollary 4.2.7, 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* (5.18) *holds, then for any $\hat{\beta} > 0$, the pair $\frac{\hat{\beta}}{\beta}P_i$ and $\hat{\beta}$ also satisfies equation* (5.18).

5.1.3 Discussion

The previous sections discuss how to design local real-time constraints that ensure $\mathcal{L}_p$ stability of NCSs. Although packet loss is not studied in this chapter, it can be addressed using the similar approach proposed in Section 4.3 in Chapter 4. The difficulty lies in how to derive the bounds on the transmission delays. The bounds on delays provided in Chapter 4 are for asymptotic stability. It is derived based on the fact of bounded system dynamics. However, when exogenous disturbances are involved, the system state are not necessarily bounded in general. Novel approaches are expected to solve this delayed case in the future. One possible way is to restrict the class of disturbances such that the system dynamics are still bounded.

5.2 Quantization

Quantization is always an important research topic in NCSs. Although our work did not consider the effect of quantization on NCSs, it seems promising to extend our results to quantized NCSs. Recall that the local real-time constraint derived in Chapter 4 requires the satisfaction of the equation

$$c_i\|e_i(t)\| \leq \varrho_i\|\hat{x}_i(t)\| \tag{5.19}$$

with some $\varrho_i \in (0,1)$. Notice that no matter quantization exists or not, as long as equation (5.19) holds, the overall system will be asymptotically stable.

Let $q : \mathbb{R}^n \to \mathbb{R}^n$ be the quantizer. Then

$$\hat{x}_i(t) = q(x_i(b_k^i))$$

for all $t \in [f_k^i, f_{k+1}^i)$. We can split the time interval $[b_k^i, f_{k+1}^i]$ into three parts: $t = b_k^i$, (b_k^i, b_{k+1}^i), and $[b_{k+1}^i, f_{k+1}^i)$. Quantization happens at $t = b_k^i$. Data dropouts happen over (b_k^i, b_{k+1}^i). Delays happen over $[b_{k+1}^i, f_{k+1}^i)$. To ensure the satisfaction of equation (5.19), we just need to enforce

$$\begin{aligned}
&\|x_i(b_k^i) - qx_i(b_k^i)\| \le \delta_i^1 \|q(x_i(b_k^i))\|, \\
&\|x_i(t) - qx_i(b_k^i)\| \le \delta_i^2 \|q(x_i(b_k^i))\|, \;\; \forall t \in [b_k^i, b_{k+1}^i) \\
&\|x_i(t) - qx_i(b_k^i)\| \le \varrho_i \|q(x_i(b_k^i))\|, \;\; \forall t \in [b_{k+1}^i, f_{k+1}^i)
\end{aligned}$$

where $\delta_i^1 \le \delta_i^2 \le \varrho_i < 1$. The quantizer, the MANSD and the bounds on delays, therefore, might be derived following the similar analysis in Section 4.3, Chapter 4.

APPENDIX A

PROOFS

A.1 Proof of Theorem 2.2.2

Proof: We first show that the sampling period T_k is bounded from below by a positive constant. By equation (2.6),

$$\frac{d}{dt}\|e_k(t)\|_2 \leq \|\dot{x}(t)\|_2 \leq 2L\|e_k(t)\|_2 + L\|x(r_k)\|_2$$

holds for all $t \in [r_k, r_{k+1})$. Solving this differential inequality with the initial condition $e_k(r_k) = 0$ yields

$$\|e_k(t)\|_2 \leq \frac{\|x(r_k)\|_2}{2}\left(e^{2LT_k} - 1\right) \tag{A.1}$$

for all $t \in [r_k, r_{k+1})$. According to equation (2.7) and (2.9),

$$\|x(r_k)\|_2 \leq \alpha_1^{-1}(V(r_k)) \leq L_1 V(r_k) \tag{A.2}$$

holds. Applying equation (A.1) and (A.2) into equation (2.8) leads to the inequality

$$\dot{V}(t) \leq -\bar{a}V(t) + \frac{\bar{b}L_1 V(r_k)}{2}\left(e^{2LT_k} - 1\right)$$

for $t \in [r_k, r_{k+1})$. Solving this differential inequality with the initial condition $V(r_k)$ provides

$$V(t) \;\leq\; V(r_k)e^{-\bar{a}(t-r_k)} - \frac{\bar{b}L_1 V(r_k)}{2\bar{a}}\left(e^{2LT_k} - 1\right)\left(e^{-\bar{a}(t-r_k)} - 1\right) \qquad \text{(A.3)}$$

for all $t \in [r_k, r_{k+1})$. Because r_{k+1} is triggered by the violation of equation (2.14),

$$V(r_{k+1}) = -\delta\bar{a}V(r_k)T_k + V(r_k) \qquad \text{(A.4)}$$

holds. Combining equation (A.3) and (A.4) yields

$$-\delta\alpha V(r_k)T_k + V(r_k) \leq V(r_k)e^{-\alpha T_k} - \frac{\beta L_1 V(r_k)}{2\alpha}\left(e^{2LT_k} - 1\right)\left(e^{-\alpha T_k} - 1\right)$$

which implies

$$p(T_k) \triangleq -\bar{a}\delta T_k + 1 \leq e^{-\bar{a}T_k} - \frac{\bar{b}L_1}{2\bar{a}}\left(e^{2LT_k} - 1\right)\left(e^{-\bar{a}T_k} - 1\right) \triangleq q(T_k).$$

It is obvious that $p(0) = q(0) = 1$ and $\dot{q}(0) = -\bar{a} < \dot{p}(0) = -\bar{a}\delta$. Therefore, using Lemma 2.2.1, we conclude

$$T_k \geq \xi, \qquad \text{(A.5)}$$

where ξ is the smallest positive solution to the equation $p(t) = q(t)$.

We now show asymptotic stability of the resulting system. First, define the function $h(t|x_0)$ as:

$$h(t|x_0) = -\delta\bar{a}V(r_k)(t - r_k) + V(r_k), \forall t \in [r_k, r_{k+1}). \qquad \text{(A.6)}$$

Because r_{k+1} is triggered by the violation of equation (2.14),

$$V(t) \leq -\delta\bar{a}V(r_k)(t - r_k) + V(r_k) = h(t|x_0) \tag{A.7}$$

holds for all $t \in [r_k, r_{k+1})$, $k \in \mathbb{N}$.

To show asymptotic stability of the system, we still need to show $\lim_{t\to\infty} h(t|x_0) = 0$. By the definition of $h(t|x_0)$ in equation (A.6), we know $h(t|x_0)$ is differentiable for $t \in (r_k, r_{k+1})$. By equation (2.7), the derivative of $h(t|x_0)$ is

$$\dot{h}(t|x_0) = -\delta\bar{a}V(r_k) \leq -\delta\bar{a} \cdot \alpha_1(\|x(r_k)\|_2) \tag{A.8}$$

for all $t \in (r_k, r_{k+1})$, which means $h(t|x_0)$ is decreasing over (r_k, r_{k+1}).

Although $h(t|x_0)$ may not be differentiable at $t = r_k$ for some $k \in \mathbb{N}$, it satisfies

$$\begin{aligned}
&\lim_{t\to r_k^-} h(t|x_0) = \lim_{t\to r_k^+} h(t|x_0) = h(r_k|x_0), \\
&\lim_{t\to r_k^-} \dot{h}(t|x_0) \leq -\delta\bar{a} \cdot \alpha_1(\|x(r_{k-1})\|_2), \text{ and} \\
&\lim_{t\to r_k^+} \dot{h}(t|x_0) \leq -\delta\bar{a} \cdot \alpha_1(\|x(r_k)\|_2)
\end{aligned} \tag{A.9}$$

which means for any $k \in \mathbb{N}$, $h(t|x_0)$ is continuous at $t = r_k$ and the left-hand and right-hand sided derivatives of $h(t|x_0)$ at $t = r_k$ are both negative. Since equation (A.5) holds, $r_k \to \infty$ holds. Combining this with equation (A.8) and (A.9) yileds

$$\lim_{t\to\infty} h(t|x_0) = 0. \tag{A.10}$$

Since equation (A.7) and (A.10) are satisfied, we can conclude that the sampled-data system is asymptotically stable. □

A.2 Proof of Lemma 2.3.1

Proof: Consider the derivative of $\|e_{k-1}(t)\|_2$.

$$\frac{d}{dt}\|e_{k-1}(t)\|_2 \leq 2L\|e_{k-1}(t)\|_2 + L\|x(r_{k-1})\|_2$$

holds for all $t \in [r_k, f_k)$. Solving the differential inequality with the initial condition $\|e_{k-1}(r_k)\|_2$, we have

$$\|e_{k-1}(t)\|_2 \leq \|e_{k-1}(r_k)\|_2 e^{2L\tau_k} + \frac{\|x(r_{k-1})\|_2}{2}\left(e^{2L\tau_k} - 1\right) \tag{A.11}$$

for all $t \in [r_k, f_k)$. By equation (2.8), the inequality

$$\dot{V}(t) \leq -\bar{a}V(t) + \bar{b}\|e_{k-1}(t)\|_2$$

holds for $t \in [r_k, f_k)$. Combining this inequality with equation (A.11) yields

$$\dot{V}(t) \;\; \leq \;\; -\bar{a}V(t) + \bar{b}\|e_{k-1}(r_k)\|_2 e^{2L\tau_k} + \bar{b}\frac{\|x(r_{k-1})\|_2}{2}\left(e^{2L\tau_k} - 1\right)$$

for all $t \in [r_k, f_k)$. Solving this differential inequality with the initial condition $V(r_k)$ leads to

$$\begin{aligned} V(t) \;\; \leq \;\; & V(r_k)e^{-\bar{a}(t-r_k)} - \frac{\bar{b}}{\bar{a}}\|e_{k-1}(r_k)\|_2 e^{2L\tau_k}\left(e^{-\bar{a}(t-r_k)} - 1\right) \\ & -\frac{\bar{b}}{\bar{a}}\frac{\|x(r_{k-1})\|_2}{2}\left(e^{2L\tau_k} - 1\right)\left(e^{-\bar{a}(t-r_k)} - 1\right) \end{aligned} \tag{A.12}$$

for all $t \in [r_k, f_k)$. Using $\|x\|_2 \leq \alpha_1^{-1}(V(x)) \leq L_1 V(x)$ and $V(r_{k-1}) \leq \sigma V(r_k)$, equation (A.12) implies

$$\begin{aligned} V(t) &\leq V(r_k) - \frac{\beta L_1 V(r_{k-1})\left(e^{2L\tau_k} - 1\right)\left(e^{-\alpha\tau_k} - 1\right)}{2\alpha} \\ &\quad - \frac{\beta L_1}{\alpha}(V(r_k) + V(r_{k-1}))e^{2L\tau_k}\left(e^{-\alpha\tau_k} - 1\right) \\ &\leq V(r_k) - \frac{\bar{b}L_1}{\bar{a}}(1+\sigma)V(r_k)e^{2L\tau_k}\left(e^{-\bar{a}\tau_k} - 1\right) \\ &\quad - \frac{\bar{b}L_1}{\bar{a}}\frac{\sigma V(r_k)}{2}\left(e^{2L\tau_k} - 1\right)\left(e^{-\bar{a}\tau_k} - 1\right) \end{aligned} \tag{A.13}$$

for all $t \in [r_k, f_k)$.

By Lemma 2.2.1 and $\tau_k < \Delta_1$, we know

$$1 - \frac{\bar{b}L_1}{\bar{a}}(1+\sigma)e^{2L\tau_k}\left(e^{-\bar{a}\tau_k} - 1\right) - \frac{\bar{b}L_1}{\bar{a}}\frac{\sigma}{2}\left(e^{2L\tau_k} - 1\right)\left(e^{-\bar{a}\tau_k} - 1\right) \leq 1 + \epsilon\tau_k \tag{A.14}$$

where $\epsilon > \bar{b}L_1(1+\sigma)$. According to equation (A.13) and (A.14), it is easy to show that $V(t) \leq V(r_k)(1+\epsilon\tau_k)$ holds for all $t \in [r_k, f_k)$. Using the similar technique, we can show that if $\tau_k < \Delta_2$, $\sigma V(t) \geq V(r_k)$ holds for all $t \in [r_k, f_k)$. □

A.3 Proof of Lemma 2.3.2

Proof: Consider the derivative of $\|e_k(t)\|_2$ over the interval $t \in [r_k, f_k)$:

$$\frac{d}{dt}\|e_k(t)\|_2 \leq 2L\|e_k(t)\|_2 + L\|x(r_k)\|_2 + L\|e_{k-1}(r_k)\|_2.$$

Solving this differential inequality with the initial condition $e_k(r_k) = 0$, we have

$$\|e_k(t)\|_2 \leq \frac{\|x(r_k)\|_2 + \|e_{k-1}(r_k)\|_2}{2}\left(e^{2L\tau_k} - 1\right) \tag{A.15}$$

for all $t \in [r_k, f_k)$. The derivative of $e_k(t)$ over $[f_k, t_k^*)$ satisfies

$$\frac{d}{dt}\|e_k(t)\|_2 \leq 2L\|e_k(t)\|_2 + L\|x(r_k)\|_2.$$

Solving this differential inequality with the initial condition given by equation (A.15), we have

$$\begin{aligned} \|e_k(t)\|_2 &\leq \frac{\|x(r_k)\|_2 + \|e_{k-1}(r_k)\|_2}{2}\left(e^{2L\tau_k} - 1\right)e^{2L(t-f_k)} + \frac{\|x(r_k)\|_2}{2}\left(e^{2L(t-f_k)} - 1\right) \\ &\leq \frac{\|x(r_k)\|_2 + \|e_{k-1}(r_k)\|_2}{2}\left(e^{2L\tau_k} - 1\right)e^{2L(t_k^*-f_k)} + \frac{\|x(r_k)\|_2}{2}\left(e^{2L(t_k^*-f_k)} - 1\right) \end{aligned} \tag{A.16}$$

holds for all $t \in [f_k, t_k^*)$. By equation (2.8) and (A.16), we have

$$\begin{aligned} \dot{V}(t) &\leq -\bar{a}V(t) + \bar{b}\frac{\|x(r_k)\|_2}{2}\left(e^{2L(t_k^*-f_k)} - 1\right) \\ &\quad +\bar{b}\frac{\|x(r_k)\|_2 + \|e_{k-1}(r_k)\|_2}{2}\left(e^{2L\tau_k} - 1\right)e^{2L(t_k^*-f_k)} \end{aligned}$$

for all $t \in [f_k, t_k^*)$. Since $\|x(r_k)\|_2 \leq \alpha_1^{-1}(V(r_k)) \leq L_1 V(r_k)$ for all $k \in \mathbb{N}$ and $V(r_{k-1}) \leq \sigma V(r_k)$, the inequality above can be further reduced as

$$\dot{V}(t) \leq -\bar{a}V(t) + \eta(t_k^* - f_k, \tau_k)V(r_k), \tag{A.17}$$

for all $t \in [f_k, t_k^*)$. Solving the differential inequality in equation (A.17) with the initial condition $V(f_k)$ leads to

$$V(t_k^*) \leq V(f_k)e^{-\bar{a}(t_k^*-f_k)} - \frac{\eta(t_k^* - f_k, \tau_k)V(r_k)}{\bar{a}}\left(e^{-\bar{a}(t_k^*-f_k)} - 1\right).$$

Since the hypotheses of Lemma 2.3.1 are satisfied, $V(f_k) \leq (1 + \epsilon\Delta)V(r_k)$

holds. Applying this and equation (2.18) to the inequality above, we have

$$\begin{aligned} p(t_k^* - f_k) &\triangleq 1 + \epsilon\Delta - \delta\bar{a}(t_k^* - f_k) \\ &\leq (1+\epsilon\Delta)e^{-\bar{a}(t_k^*-f_k)} - \frac{\eta(t_k^* - f_k, \Delta)}{\bar{a}}\left(e^{-\bar{a}(t_k^*-f_k)} - 1\right) \triangleq q(t_k^* - f_k) \end{aligned}$$

Notice that $p(0) = q(0)$. Since $\Delta < \Delta_3$ implies $\dot{p}(0) > \dot{q}(0)$, by Lemma 2.2.1, we know $t_k^* - f_k \geq \xi(\Delta) > 0$. □

A.4 Proof of Theorem 2.3.3

Proof: By Lemma 2.3.1, we know E_1 and E_2 always hold when $t = f_k$. It is easy to show that $\xi(0) > 0$ according to equation (2.19). Therefore, $\epsilon\tau_k - \xi(\tau_k)\delta\bar{a} < 0$ holds when $\tau_k = 0$. By lemma 2.2.1 and the definition of Δ_4, we have $\epsilon\tau_k - \xi(\tau_k)\delta\bar{a} < 0$ for all $\tau_k < \Delta_4$. Consequently , there must be a positive constant ε such that

$$\epsilon\Delta - \xi(\Delta)\delta\bar{a} < -\varepsilon, \tag{A.18}$$

holds since $\Delta < \Delta_4$. We construct a piecewise continuous function $h : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^+$ in the following way:

$$h(t|x_0) = \begin{cases} [1 + \epsilon\Delta - \delta\bar{a}(t - f_k)]V(r_k), t \in [f_k, r_{k+1}) \\ (1+\epsilon\Delta)V(r_{k+1}), \quad t \in [r_{k+1}, f_{k+1}) \end{cases} \tag{A.19}$$

Because r_{k+1} is triggered by the violation of E_1 or E_2, it is easy to show $V(t) \leq h(t|x_0)$ for all $t \in [f_k, r_{k+1})$ and $k \in \mathbb{N}$. Since $\tau_k < \Delta \leq \Delta_1$ holds, by Lemma 2.3.1, $V(t) \leq (1+\epsilon\Delta)V(r_k) = h(t|x_0)$ holds for all $t \in [r_k, f_k)$ and $k \in \mathbb{N}$.

Therefore,

$$V(t) \leq h(t|x_0) \tag{A.20}$$

holds for all $t \in \mathbb{R}^+$. We now show that $\lim_{t\to\infty} h(t|x_0) = 0$. Two cases are considered.

Case I: r_{k+1} is triggered by the violation of E_1. Then

$$\begin{aligned} h(f_k|x_0) &= (1+\epsilon\Delta)V(r_k) \\ h(f_{k+1}|x_0) &= (1+\epsilon\Delta)V(r_{k+1}) \end{aligned} \tag{A.21}$$

Since r_{k+1} is triggered by the violation of E_1, we have

$$V(r_{k+1}) = [1+\epsilon\Delta - \delta\bar{a}(r_{k+1}-f_k)]V(r_k).$$

By Lemma 2.3.2, we know $r_{k+1} - f_k \geq \xi(\Delta)$. So

$$V(r_{k+1}) \leq [1+\epsilon\Delta - \delta\bar{a}\xi(\Delta)]V(r_k)$$

holds. Combining this inequality with equation (A.21), we have $h(f_{k+1}|x_0) \leq [1+\epsilon\Delta - \delta\bar{a}\xi(\Delta)]h(f_k|x_0)$, which implies

$$h(f_{k+1}|x_0) \leq (1-\varepsilon)h(f_k|x_0) \tag{A.22}$$

according to equation (A.18).

Case II: r_{k+1} is triggered by the violation of E_2. Following the similar analysis for case I, we have

$$h(f_{k+1}|x_0) = \frac{1}{\sigma} h(f_k|x_0). \tag{A.23}$$

Equation (A.22) and (A.23) implies

$$\lim_{k\to\infty} h(f_k|x_0) = 0 \tag{A.24}$$

since $\sigma \in (1, \infty)$. Notice that $h(t|x_0) \leq h(f_k|x_0)$ holds for all $t \geq f_k$. So equation (A.24) implies

$$\lim_{t\to\infty} h(t|x_0) = 0. \tag{A.25}$$

Equation (A.20) and (A.25) are sufficient to conclude that the sampled-data system is asymptotically stable. □

A.5 Proof of Theorem 3.2.1

Proof: Consider the storage function $V : \mathbb{R}^n \to \mathbb{R}$ given by $V(x) = x^T Px$ for $x \in \mathbb{R}^n$ where P is a symmetric positive semi-definite matrix satisfying the algebraic Riccati equation (equation (3.5)). The directional derivative of V for

$t \in [f_k, f_{k+1})$ is

$$\begin{aligned}
\dot{V} &= \frac{\partial V}{\partial x}\big(Ax(t) - B_1 B_1^T P x(r_k) + B_2 w(t)\big) \\
&= -x(t)^T \left(I - Q + \frac{1}{\gamma^2} P B_2 B_2^T P\right) x(t) \\
&\quad -2x(t)^T Q x(r_k) + 2x(t)^T P B_2 w(t) \\
&= -x(t)^T (I-Q) x(t) - \left\| \gamma w(t) - \frac{1}{\gamma} B_2^T P x(t) \right\|_2^2 \\
&\quad +\gamma^2 \|w(t)\|_2^2 - 2x(t)^T Q x(r_k) \\
&\leq -x(t)^T (I-Q) x(t) + \gamma^2 \|w(t)\|_2^2 - 2x(t)^T Q x(r_k)
\end{aligned}$$

Insert $x(t) = e_k(t) + x(r_k)$ into the above equation to obtain

$$\begin{aligned}
\dot{V} &\leq -\|x(t)\|_2^2 + [e_k(t) + x(r_k)]^T Q \,[e_k(t) + x(r_k)] \\
&\quad -2\,[e_k(t) + x(r_k)]^T Q x(r_k) + \gamma^2 \|w(t)\|_2^2 \\
&= -\|x(t)\|_2^2 + e_k(t)^T Q e_k(t) - x^T(r_k) Q x(r_k) \\
&\quad +\gamma^2 \|w(t)\|_2^2 \qquad \text{(A.26)}
\end{aligned}$$

By the assumption in equation (3.11), we know that equation (A.26) can be rewritten as

$$\dot{V} \leq -\beta^2 \|x(t)\|_2^2 + \gamma^2 \|w(t)\|_2^2, \qquad \text{(A.27)}$$

which holds for all t and is sufficient to ensure the sampled-data system is finite-gain $\mathcal{L}_2$ stable from w to x with a gain less than γ/β. □

A.6 Proof of Corollary 3.2.2

Proof: Equation (3.14) can be rewritten as

$$\begin{aligned} e_k(t)^T M e_k(t) &= (1-\beta^2)\|e_k(t)\|_2^2 + e_k(t)^T Q e_k(t) \\ &\leq \frac{1}{2}(1-\beta^2)\|x(r_k)\|_2^2 + x(r_k)^T Q x(r_k) \end{aligned}$$

This can be rewritten to obtain

$$\begin{aligned} e_k(t)^T Q e_k(t) &\leq (1-\beta^2)\left(\|x(r_k)\|_2^2 + \|e_k(t)\|_2^2\right) + x^T(r_k) Q x(r_k) \\ &\quad -(1-\beta^2)\left(\frac{1}{2}\|x(r_k)\|_2^2 + 2\|e_k(t)\|_2^2\right) \\ &= (1-\beta^2)\left(\|x(r_k)\|_2^2 + \|e_k(t)\|_2^2\right) + x^T(r_k) Q x(r_k) \\ &\quad -(1-\beta^2)\left(\frac{1}{2}\|x(r_k)\|_2^2 + 2\|e_k(t)\|_2^2\right) \\ &\quad -(1-\beta^2)\left(2x^T(r_k)e_k(t) - 2x^T(r_k)e_k(t)\right) \\ &= (1-\beta^2)\|x(r_k)+e_k(t)\|_2^2 + x^T(r_k) Q x(r_k) \\ &\quad -(1-\beta^2)\left\|\frac{1}{\sqrt{2}}x(r_k) + \sqrt{2}e_k(t)\right\|_2^2 \\ &\leq (1-\beta^2)\|x(t)\|_2^2 + x(r_k)^T Q x(r_k). \end{aligned}$$

This inequality is the sufficient condition in Theorem 3.2.1 so we can conclude that the sampled-data system is $\mathcal{L}_2$ stable from w to x with a gain less than γ/β. □

A.7 Proof of Theorem 3.3.1

Proof: Let $\Gamma = \{t \in [r_k, f_{k+1}) \mid \|z_k(t)\|_2 = 0\}$. The time derivative of $\|z_k(t)\|_2$ for $t \in [r_k, f_{k+1})\backslash\Gamma$ satisfies

$$\begin{aligned} \frac{d}{dt}\|z_k(t)\|_2 &\le \left\|\sqrt{M}\dot{e}_k(t)\right\|_2 = \left\|\sqrt{M}\dot{x}(t)\right\|_2 \\ &= \left\|\sqrt{M}\left(Ax(t) - B_1B_1^T Px(r_k) + B_2 w(t)\right)\right\|_2 \\ &\le \left\|\sqrt{M}Ae_k(t)\right\|_2 + \left\|\sqrt{M}A_{\text{cl}}x(r_k)\right\|_2 + \left\|\sqrt{M}B_2\right\| \|w(t)\|_2, \end{aligned} \tag{A.28}$$

where the righthand sided derivative is used when $t = r_k$. Since $\|w(t)\|_2 \le a\|x(t)\|_2$, $x(t) = e_k(t) + x(r_k)$, and $z_k(t) = \sqrt{M}e_k(t)$, we can bound the preceding equation (A.28) as

$$\begin{aligned} \frac{d}{dt}\|z_k(t)\|_2 &\le \left\|\sqrt{M}A\sqrt{M}^{-1}\right\| \|z_k(t)\|_2 + \left\|\sqrt{M}A_{\text{cl}}x(r_k)\right\|_2 \\ &\quad + a\|\sqrt{M}B_2\|\|\sqrt{M}^{-1}z_k(t)\|_2 + a\|\sqrt{M}B_2\|\|x(r_k)\|_2 \\ &\le \left(\left\|\sqrt{M}A\sqrt{M}^{-1}\right\| + a\|\sqrt{M}B_2\|\|\sqrt{M}^{-1}\|\right)\|z_k(t)\|_2 \\ &\quad + \left\|\sqrt{M}A_{cl}x(r_k)\right\|_2 + a\left\|\sqrt{M}B_2\right\|\|x(r_k)\|_2 \\ &= \sigma\|z_k(t)\|_2 + \mu_0(x(r_k)). \end{aligned} \tag{A.29}$$

where σ and $\mu_0 : \mathbb{R}^n \to \mathbb{R}$ are defined in equation (3.20) and (3.21), respectively.

The initial condition is $\|z_k(r_k)\|_2 = 0$. Using this in the differential inequality in equation (A.29) yields,

$$\|z_k(t)\|_2 \le \frac{\mu_0(x(r_k))}{\sigma}\left(e^{\sigma(t-r_k)} - 1\right) \tag{A.30}$$

for all $t \in [r_k, f_{k+1})$ since $\|z_k(t)\|_2 = 0$ for all $t \in \Gamma$.

By assumption $r_{k+1} = f_{k+1}$ (i.e. no task delay) and $\delta\psi(x(r_k)) = \|z_k(r_{k+1})\|_2$, so we can conclude that

$$\delta\psi(x(r_k)) = \|z_k(r_{k+1})\|_2 \leq \frac{\mu_0(x(r_k))}{\sigma}\left(e^{\sigma T_k} - 1\right) \tag{A.31}$$

where $T_k = r_{k+1} - r_k$ is the task sampling period for job k. Solving equation (A.31) for T_k yields equation (3.19). The righthand side of inequality (3.19) is clearly strictly greater than zero, which implies that $r_{k+1} - r_k > 0$. Therefore $r_k = f_k \leq r_{k+1}$ which implies that the sequence of finishing and release times is admissible. Finally we know that $\|z_k(t)\|_2 \leq \delta\psi(x(r_k))$ for all $t \in [r_k, f_{k+1}) = [f_k, f_{k+1})$ and all $k = 0, \ldots, \infty$, which by Corollary 3.2.2 implies that the system is $\mathcal{L}_2$ stable from w to x with a gain less than γ/β. $\square$

A.8 Proof of Lemma 3.3.2

Proof: Let $\Gamma = \{t \in [r_k, f_k) \mid \|z_k(t)\|_2 = 0\}$. For $t \in [r_k, f_k)\backslash\Gamma$, the derivative of $\|z_k(t)\|_2$ satisfies the differential inequality,

$$\begin{aligned}
\frac{d}{dt}\|z_k(t)\|_2 &\leq \|\dot{z}_k(t)\|_2 = \left\|\sqrt{M}\dot{e}_k(t)\right\|_2 = \left\|\sqrt{M}\dot{x}(t)\right\|_2 \\
&= \left\|\sqrt{M}\left(Ax(t) - B_1B_1^TPx(r_{k-1}) + B_2w(t)\right)\right\|_2 \\
&= \left\|\sqrt{M}Ae_k(t) + \sqrt{M}Ax(r_k) - \sqrt{M}B_1B_1^TPx(r_{k-1}) + \sqrt{M}B_2w(t)\right\|_2 \\
&\leq \left(\left\|\sqrt{M}A\sqrt{M}^{-1}\right\| + a\left\|\sqrt{M}B_2\right\|\left\|\sqrt{M}^{-1}\right\|\right)\|z_k(t)\|_2 \\
&\quad + \left\|\sqrt{M}\left(Ax(r_k) - B_1B_1^TPx(r_{k-1})\right)\right\|_2 + a\left\|\sqrt{M}B_2\right\|\|x(r_k)\|_2 \\
&= \lambda\|z_k(t)\|_2 + \mu_1(x(r_k), x(r_{k-1})), \qquad \text{(A.32)}
\end{aligned}$$

where we use the righthand sided derivative when $t = r_k$. The differential inequality in equation (A.32) along with the initial condition $z_k(r_k) = 0$, allows us to conclude that

$$\|z_k(t)\|_2 \leq \phi(x(r_k), x(r_{k-1}); t - r_k) \tag{A.33}$$

for all $t \in [r_k, f_k)$ since $\|z_k(t)\|_2 = 0$ for all $t \in \Gamma$.

The assumption in equation (3.26) can be rewritten as

$$\phi(x(r_k), x(r_{k-1}); \tau_k) \leq \epsilon\psi(x(r_k)) \tag{A.34}$$

$\phi(x(r_k), x(r_{k-1}); t - r_k)$ is a monotone increasing function of $t - r_k$. Combining this fact with equation (A.33) and (A.34) yields

$$\|z_k(t)\|_2 \leq \phi(x(r_k), x(r_{k-1}); t - r_k) \leq \phi(x(r_k), x(r_{k-1}); \tau_k) \leq \epsilon\psi(x(r_k))$$

which leads to equation (3.27) holding for all $t \in [r_k, f_k)$. □

A.9 Proof of Lemma 3.3.3

Proof: The hypotheses of this lemma also satisfy the hypotheses of Lemma 3.3.2 so we know that

$$\|z_k(f_k)\|_2 \leq \phi(x(r_k), x(r_{k-1}); \tau_k) \leq \epsilon\psi(x(r_k)) \leq \pi\psi(x(r_k)). \tag{A.35}$$

By equation (3.29) and (A.35), we have

$$\eta_2(x(r_k), x(r_{k-1}); \tau_k, \pi) > 0$$

which implies

$$d_\pi > f_k$$

Assume the system state $x(t)$ satisfies the differential equation

$$\dot{x}(t) = Ax(t) - B_1 B_1^T P x(r_k) + B_2 w(t)$$

for $t \in [f_k, d_\pi]$. Using an argument similar to that in Lemma 3.3.2, we can show that $\|z_k(t)\|_2$ satisfies the differential inequality

$$\frac{d}{dt}\|z_k(t)\|_2 \leq \lambda \|z_k(t)\|_2 + \mu_0(x(r_k)). \tag{A.36}$$

Equation (A.35) can be viewed as an initial condition on the differential inequality in equation (A.36). Solving the differential inequality, we know for all $t \in [f_k, d_\pi]$,

$$\|z_k(t)\|_2 \leq e^{\lambda(t-f_k)}\phi(x(r_k), x(r_{k-1}); \tau_k) + \frac{\mu_0(x(r_k))}{\lambda}\left(e^{\lambda(t-f_k)} - 1\right). \tag{A.37}$$

Because the right side of equation (A.37) is an increasing function of t, we get

$$\begin{aligned} \|z_k(t)\|_2 &\leq e^{\lambda(d_\pi - f_k)}\phi(x(r_k), x(r_{k-1}); \tau_k) + \frac{\mu_0(x(r_k))}{\lambda}\left(e^{\lambda(d_\pi - f_k)} - 1\right) \\ &= \pi\psi(x(r_k)) \end{aligned} \tag{A.38}$$

for all $t \in [f_k, d_\pi]$, where the equivalence in the right side of equation (A.38) is achieved according to the definition of d_π in equation (3.28). □

A.10 Proof of Lemma 3.3.4

Proof: First note that $x(r_k) = e_{k-1}(r_k) + x(r_{k-1})$ implies that

$$-\|e_{k-1}(r_k)\|_2 \leq \|x(r_k)\|_2 - \|x(r_{k-1})\|_2 \leq \|e_{k-1}(r_k)\|_2.$$

We now use this inequality to bound $\mu_1(x(r_k), x(r_{k-1}))$ and $\psi(x(r_k))$ as a function of $x(r_{k-1})$.

An upper bound on $\mu_1(x(r_k), x(r_{k-1}))$ can be obtained by noting that

$$\begin{aligned}
& \mu_1(x(r_k), x(r_{k-1})) \\
= & \left\|\sqrt{M}\left(Ax(r_k) - B_1 B_1^T P x(r_{k-1})\right)\right\|_2 + a\|\sqrt{M}B_2\|\|x(r_k)\|_2 \\
= & \left\|\sqrt{M}\left(A_{\text{cl}}x(r_{k-1}) + Ae_{k-1}(r_k)\right)\right\|_2 + a\|\sqrt{M}B_2\|\|x(r_{k-1}) + e_{k-1}(r_k)\|_2 \\
\leq & \left\|\sqrt{M}A_{\text{cl}}x(r_{k-1})\right\| + a\|\sqrt{M}B_2\|\|x(r_{k-1})\|_2 + \left\|\sqrt{M}A\sqrt{M}^{-1}z_{k-1}(r_k)\right\|_2 \\
& + a\|\sqrt{M}B_2\|\|\sqrt{M}^{-1}z_{k-1}(r_k)\|_2 \\
\leq & \left\|\sqrt{M}A_{\text{cl}}x(r_{k-1})\right\| + a\|\sqrt{M}B_2\|\|x(r_{k-1})\|_2 \\
& + \left(\left\|\sqrt{M}A\sqrt{M}^{-1}\right\| + a\|\sqrt{M}B_2\|\|\sqrt{M}^{-1}\|\right)\delta\psi(x(r_{k-1})) \\
= & \mu_0(x(r_{k-1})) + \lambda\delta\psi(x(r_{k-1}))
\end{aligned}$$

A lower bound on $\psi(x(r_k))$ is obtained by noting that

$$\begin{aligned}
\psi(x(r_k)) &= \left\|\sqrt{N}x(r_k)\right\|_2 = \left\|\sqrt{N}(e_{k-1}(r_k) + x(r_{k-1}))\right\|_2 \\
&\geq \|\sqrt{N}x(r_{k-1})\|_2 - \left\|\sqrt{N}e_{k-1}(r_k)\right\|_2 \\
&= \|\sqrt{N}x(r_{k-1})\|_2 - \sqrt{e_{k-1}^T(r_k)Ne_{k-1}(r_k)}
\end{aligned}$$

We know $M \geq N$ by the definitions of M and N in equation (3.12) and (3.13), respectively. So the inequality above can be further reduced as

$$\begin{aligned}
\psi(x(r_k)) &\geq \|\sqrt{N}x(r_{k-1})\|_2 - \sqrt{e^T_{k-1}(r_k)Me_{k-1}(r_k)} \\
&\geq \psi(x(r_{k-1})) - \delta\psi(x(r_{k-1})) \\
&= (1-\delta)\psi(x(r_{k-1}))
\end{aligned}$$

Putting both inequalities together we see that

$$\begin{aligned}
\eta_1(x(r_k), x(r_{k-1}); \epsilon) &= \frac{1}{\lambda}\ln\left(1 + \epsilon\lambda\frac{\psi(x(r_k))}{\mu_1(x(r_k), x(r_{k-1}))}\right) \\
&\geq \frac{1}{\lambda}\ln\left(1 + \epsilon\lambda\frac{(1-\delta)\psi(x(r_{k-1}))}{\lambda\delta\psi(x(r_{k-1})) + \mu_0(x(r_{k-1}))}\right) \\
&\equiv \Delta(x(r_{k-1}); \epsilon, \delta) > 0,
\end{aligned}$$

which completes the proof. □

A.11 Proof of Theorem 3.3.5

Proof: From the definition of Δ in equation (3.35), we can easily see that $\Delta(x(r_k); \epsilon, \delta) > 0$ for any non-negative integer k. We can therefore use equation (3.37) to conclude that the interval $[r_{k+1}, r_{k+1} + \Delta(x(r_k); \epsilon, \delta)]$ is nonempty for all k.

Next, we insert equation (3.36) into equation (3.37) to show that

$$\begin{aligned}
f_{k+1} &\leq r_{k+1} + \Delta(x(r_k); \epsilon, \delta) \\
&\leq f_k + \eta_2(x(r_k), x(r_{k-1}); \tau_k, \delta) + \Delta(x(r_k); \epsilon, \delta) \\
&\leq f_k + \eta_2(x(r_k), x(r_{k-1}); \tau_k, 1) \qquad \text{(A.39)}
\end{aligned}$$

for all non-negative integers k.

With the preceding two preliminary results, we now consider the following statement about the kth job. This statement is that

1. $r_k \leq f_k \leq r_{k+1}$,
2. $\|z_k(t)\|_2 \leq \delta\psi(x(r_k))$ for all $t \in [f_k, r_{k+1}]$,
3. and $\|z_k(t)\|_2 \leq \psi(x(r_k))$ for all $t \in [f_k, f_{k+1}]$.

We now use mathematical induction to show that under the theorem's hypotheses, this statement holds for all non-negative integers k.

First consider the base case when $k = 0$. According to the definition of η_2 (equation (3.29)) we know that

$$\eta_2(x_0, x_0; \tau_0, \delta) = \eta_2(x_0, x_0; 0, \delta) > 0$$

We can therefore combine equation (3.37) and (3.36) to obtain

$$r_0 = f_0 \leq f_0 + \eta_2(x_0, x_0; \tau_0, \delta) = r_1 \tag{A.40}$$

which establishes the first part of the inductive statement when $k = 0$.

Next note that

$$\tau_0 = 0 \leq \eta_1(x(r_0), x(r_{-1}); \epsilon). \tag{A.41}$$

If we use the fact that $\delta \in (\epsilon, 1) \subset (0, 1]$ in equation (3.36) and (A.41), we can see that the hypotheses of Lemma 3.3.3 are satisfied. This means that $\|z_0(t)\|_2 \leq \delta\psi(x(r_0))$ for all $t \in [f_0, r_1]$ which completes the second part of the inductive statement for $k = 0$.

Now define the time

$$d_1^0 = f_0 + \eta_2(x(r_0), x(r_{-1}); \tau_0, 1)$$

Equation (A.41) again implies that the hypotheses of Lemma 3.3.3 are satisfied, so that

$$\|z_0(t)\|_2 \leq \psi(x(r_0)) \text{ for all } t \in [f_0, d_1^0]. \tag{A.42}$$

From equation (A.39), we know that $f_1 \leq d_1^0$. We can also combine equation (3.37) and (A.40) to conclude that $f_0 \leq f_1$. We therefore know that $[f_0, f_1] \subseteq [f_0, d_1^0]$ which combined with equation (A.42) implies that

$$\|z_0(t)\|_2 \leq \psi(x(r_0)) \text{ for all } t \in [f_0, f_1]$$

This therefore establishes the last part of the inductive statement for $k = 0$.

We now turn to the general case for any k. For a given k let us assume that the statement holds. This means that

$$\begin{aligned} r_k &\leq f_k \leq r_{k+1} && \text{(A.43)} \\ \|z_k(t)\|_2 &\leq \delta\psi(x(r_k)) \qquad \text{for all } t \in [f_k, r_{k+1}] && \text{(A.44)} \\ \|z_k(t)\|_2 &\leq \psi(x(r_k)) \qquad \text{for all } t \in [f_k, f_{k+1}] && \text{(A.45)} \end{aligned}$$

Now consider the $k+1$st job. Because equation (A.44) is true, the hypothesis of Lemma 3.3.4 is satisfied which means there exists a function Δ (given by equation

(3.35)) such that

$$0 < \Delta(x(r_k); \epsilon, \delta) \leq \eta_1(x(r_{k+1}), x(r_k); \epsilon).$$

We can use this in equation (3.37) to obtain

$$0 \leq \tau_{k+1} \leq \Delta(x(r_k); \epsilon, \delta) \leq \eta_1(x(r_{k+1}), x(r_k); \epsilon). \tag{A.46}$$

From equation (A.46) and the fact that $\delta(0,1)$ we know that the hypotheses of Lemma 3.3.3 hold and we can conclude that

$$f_{k+1} \leq r_{k+2} \quad \text{and} \tag{A.47}$$

$$\|z_{k+1}\|_2 \leq \delta\psi(x(r_{k+1})) \text{ for all } t \in [f_{k+1}, r_{k+2}]. \quad . \tag{A.48}$$

Combining equation (3.37) with equation (A.47) yields $r_{k+1} \leq f_{k+1} \leq r_{k+2}$ which establishes the first part of the statement for the case $k+1$. Equation (A.48) is the second part of the statement.

Finally let

$$d_1^{k+1} = f_{k+1} + \eta_2(x(r_{k+1}), x(r_k); \tau_{k+1}, 1)$$

Following our prior argument for the case when $k = 0$, we know that the validity of equation (A.46) satisfies the hypotheses of Lemma 3.3.3. We can therefore conclude that

$$\|z_{k+1}(t)\|_2 \leq \psi(x(r_{k+1})) \text{ for all } t \in [f_{k+1}, d_1^{k+1}] \tag{A.49}$$

According to equation (A.39), $f_{k+2} \leq d_1^{k+1}$. We can therefore combine equation (3.37) and (A.47) to show that $f_{k+1} \leq f_{k+2}$ and therefore conclude that $[f_{k+1}, f_{k+2}] \subseteq [f_{k+1}, d_1^{k+1}]$. Combining this observation with equation (A.49) yields $\|z_{k+1}(t)\|_2 \leq \psi(x(r_{k+1}))$ for all $t \in [f_{k+1}, f_{k+2}]$ which completes the third part of the inductive statement for case $k+1$.

We may therefore use mathematical induction to conclude that the inductive statement holds for all non-negative integers k. The first part of the statement, of course, simply means that the sequences $\{r_k\}_{k=0}^{\infty}$ and $\{f_k\}_{k=0}^{\infty}$ are admissible. The third part of the inductive statement implies that the hypotheses of Corollary 3.2.2 are satisfied, thereby ensuring that the system's induced $\mathcal{L}_2$ gain is less than γ/β. □

A.12 Proof of Corollary 3.3.6

Proof: From Theorem 3.3.5, we know

$$f_k - r_k \leq \Delta(x(r_{k-1}); \epsilon, \delta) \leq \eta_1(x(r_k), x(r_{k-1}); \epsilon)$$

Therefore, by Lemma 3.3.2,

$$\|z_k(f_k)\|_2 \leq \phi(x(r_k), x(r_{k-1}); \tau_k) \leq \epsilon\psi(x(r_k))$$

Let us first take a look at T_k. From equation (3.36), we have

$$\begin{aligned}
T_k &\geq r_{k+1} - f_k = \eta_2(x(r_k), x(r_{k-1}); \tau_k, \delta) \\
&\geq \frac{1}{\sigma} \ln\left(1 + \sigma \frac{\delta\psi(x(r_k)) - \epsilon\psi(x(r_k))}{\mu_0(x(r_k)) + \sigma\epsilon\psi(x(r_k))}\right) \\
&\geq \frac{1}{\sigma} \ln\left(1 + \frac{\sigma(\delta - \epsilon)\lambda_{\min}(\sqrt{N})}{\|\sqrt{M}A_{\text{cl}}\| + a\|\sqrt{M}B_2\| + \sigma\epsilon\lambda_{\max}(\sqrt{N})}\right) \\
&= \zeta_1 > 0
\end{aligned}$$

It is easy to show that

$$\begin{aligned}
\Delta(x(r_k); \epsilon, \delta) &\geq \frac{1}{\sigma} \ln\left(1 + \frac{\epsilon\sigma(1 - \delta)\lambda_{\min}(\sqrt{N})}{\left\|\sqrt{M}A_{\text{cl}}\right\| + a\|\sqrt{M}B_2\| + \delta\sigma\lambda_{\max}(\sqrt{N})}\right) \\
&= \zeta_2 > 0
\end{aligned}$$

holds. □

A.13 Proof of Lemma 4.3.1

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

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 (4.34) into the preceding equation yields

$$\|x_i(t) - x_i(b_k^i)\|_2 \leq \sum_{l=0}^{j} \frac{\delta_i}{c_i} \|x_i(r_j^i)\|_2 \tag{A.50}$$

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

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

holds for all $t \in [b_k^i, b_{k+1}^i)$.

Because

$$\|\varepsilon_i^j(r_{j+1}^i)\|_2 = \|x_i(r_{j+1}^i) - x_i(r_j^i)\|_2 = \frac{\delta_i}{c_i}\|x_i(r_j^i)\|_2,$$

we have

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

and therefore

$$\|x_i(r_{j+1}^i)\|_2 \leq \left(1 + \frac{\delta_i}{c_i}\right)^{j+1} \|x_i(r_0^i)\|_2 = \left(1 + \frac{\delta_i}{c_i}\right)^{j+1} \|x_i(b_k^i)\|_2 \tag{A.52}$$

for $j = 0, 1, 2, \cdots, d_k^i$.

Applying equation (A.52) into (A.51) yields

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

for all $t \in [b_k^i, b_{k+1}^i)$. Also notice that $d_k^i \leq d_{\mathrm{MANSD}}^i$ holds according to equation

(4.35). It is easy to show that

$$\xi_k^i = c_i \left(1 + \frac{\delta_i}{c_i}\right)^{d_k^i+1} - c_i \le c_i \left(1 + \frac{\delta_i}{c_i}\right)^{d^i_{\text{MANSD}}+1} - c_i \le \varrho_i.$$

□

A.14 Proof of Lemma 4.3.2

Proof: We first consider the behavior of agent i after a successful transmission occurs, say the kth successful transmission of agent i. For notational convenience, we assume $b^i_{k-1} = r^i_0 \le r^i_1 \le \cdots \le r^i_{d^i_k} \le r^i_{d^i_k+1} = b^i_k$. Consider the derivative of $\left\|\varepsilon_i^{d_k^i+1}(t)\right\|_2 \triangleq \left\|x_i(t) - x_i(r^i_{d^i_k+1})\right\|_2$ over the time interval $[b^i_k, f^i_k)$.

$$\frac{d}{dt}\left\|\varepsilon_i^{d_k^i+1}(t)\right\|_2 \le \left\|\dot{\varepsilon}_i^{d_k^i+1}(t)\right\|_2 = \|\dot{x}_i(t)\|_2 = \|g_i\left(x_{\bar{D}_i}, \kappa_i\left(\hat{x}_{\bar{Z}_i}\right)\right)\|_2 \le \theta_i$$

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

$$\left\|\varepsilon_i^{d_k^i+1}(t)\right\|_2 = \|x_i(t) - x_i(b^i_k)\|_2 \le \theta_i(t - b^i_k) \le \frac{1-\xi^i_k}{c_i}\max\left\{\frac{\|x_i(b^i_{k-1})\|_2}{2}, \Delta\right\} \tag{A.53}$$

holds for all $t \in [b^i_k, f^i_k)$, where the inequality on the right side is obtained by applying (4.40).

Since the hypotheses in Lemma 4.3.1 are satisfied, we know

$$\|x_i(t) - x_i(b^i_{k-1})\|_2 \le \frac{\xi^i_k}{c_i}\|x_i(b^i_{k-1})\|_2 \le \frac{\varrho_i}{c_i}\|x_i(b^i_{k-1})\|_2 \tag{A.54}$$

holds for all $t \in [b^i_{k-1}, b^i_k)$ and therefore,

$$\|x_i(b^i_k) - x_i(b^i_{k-1})\|_2 \le \frac{\xi^i_k}{c_i}\|x_i(b^i_{k-1})\|_2. \tag{A.55}$$

Combining equation (A.53) and (A.55) implies that for $t \in [b^i_k, f^i_k)$,

$$\begin{align} \|x_i(t) - x_i(b^i_{k-1})\|_2 &\le \|x_i(t) - x_i(b^i_k)\|_2 + \|x_i(b^i_k) - x_i(b^i_{k-1})\|_2 \tag{A.56} \\ &\le \frac{1-\xi^i_k}{c_i} \max\left\{\frac{\|x_i(b^i_{k-1})\|_2}{2}, \Delta\right\} + \frac{\xi^i_k}{c_i}\|x_i(b^i_{k-1})\|_2. \tag{A.57} \end{align}$$

Let $\varsigma^i_k = \frac{1+\xi^i_k}{2}$. Therefore, equation (A.57), with equation (A.54), implies

$$\|x_i(t) - x_i(b^i_{k-1})\|_2 \le \max\left\{\frac{\varsigma^i_k}{c_i}\|x_i(b^i_{k-1})\|_2, \frac{\Delta(1-\xi^i_k)}{c_i} + \frac{\xi^i_k}{c_i}\|x_i(b^i_{k-1})\|_2\right\} \tag{A.58}$$

for all $t \in [b^i_{k-1}, f^i_k)$. Because $0 < \delta_i \le \xi^i_k \le \varrho_i < 1$, we know

$$\delta_i \le \xi^i_k < \varsigma^i_k \le \frac{1+\varrho_i}{2} < 1. \tag{A.59}$$

Equation (A.58) then suggests that

$$\begin{align} \sigma_i\|e_i(t)\|_2 &= \sigma_i\|x_i(t) - x_i(b^i_{k-1})\|_2 = (c_i - 1)\|x_i(t) - x_i(b^i_{k-1})\|_2 \nonumber \\ &\le \max\left\{\varsigma^i_k\|x_i(b^i_{k-1})\|_2, (1-\xi^i_k)\Delta + \xi^i_k\|x_i(b^i_{k-1})\|_2\right\} - \|x_i(t) - x_i(b^i_{k-1})\|_2 \nonumber \\ &\le \max\left\{\varsigma^i_k\|x_i(t)\|_2, (1-\xi^i_k)\Delta + \xi^i_k\|x_i(b^i_{k-1})\|_2 - \|x_i(t) - x_i(b^i_{k-1})\|_2\right\} \nonumber \\ &\le \max\left\{\varsigma^i_k\|x_i(t)\|_2, (1-\xi^i_k)\Delta + \xi^i_k\|x_i(b^i_{k-1})\|_2 - \xi^i_k\|x_i(t) - x_i(b^i_{k-1})\|_2\right\} \nonumber \\ &\le \max\left\{\varsigma^i_k\|x_i(t)\|_2, (1-\xi^i_k)\Delta + \xi^i_k\|x_i(t)\|_2\right\} \tag{A.60} \end{align}$$

holds for $t \in [f^i_{k-1}, f^i_k)$, where σ_i is defined in equation (4.10) and $c_i = 1 + \sigma_i$.

Therefore,

$$\sigma_i^p \|e_i(t)\|_2^p \leq \max \left\{ \varsigma_k^{i\,p} \|x_i(t)\|_2^p, \left((1-\xi_k^i)\Delta. + \xi_k^i \|x_i(t)\|_2 \right)^p \right\} \tag{A.61}$$

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 (A.61) implies

$$\sigma_i^p \|e_i(t)\|_2^p \leq \max \left\{ \varsigma_k^{i\,p} \|x_i(t)\|_2^p, (1-\xi_k^i)\Delta^p + \xi_k^i \|x_i(t)\|_2^p \right\}. \tag{A.62}$$

We now consider $\dot{V}$ for any $t \geq 0$. Equation (4.7) implies that

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

Because $\zeta_i - |S_i \cup U_i|\beta_i > 0$ holds, applying equation (A.62) into the preceding equation yields

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

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

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

Applying equation (A.59), (4.43), to the preceding equation implies

$$\begin{aligned}\dot{V} \quad \leq \quad & \sum_{i\in\mathcal{N}\setminus\Omega_t} (\zeta_i - |S_i \cup U_i|\beta_i)\,(1-\delta_i)\Delta^p \\ & + \sum_{i\in\mathcal{N}} (\zeta_i - |S_i \cup U_i|\beta_i)\,(\bar{\varsigma}_i - 1)\|x_i(t)\|_2^p \\ \leq \quad & \sum_{i\in\mathcal{N}} (\zeta_i - |S_i \cup U_i|\beta_i)\,(1-\delta_i)\Delta^p \\ & + \sum_{i\in\mathcal{N}} (\zeta_i - |S_i \cup U_i|\beta_i)\,(\bar{\varsigma}_i - 1)\|x_i(t)\|_2^p \\ \leq \quad & \Delta^p \sum_{i\in\mathcal{N}} (\zeta_i - |S_i \cup U_i|\beta_i)\,(1-\delta_i) \\ & - \min_i\{(\zeta_i - |S_i \cup U_i|\beta_i)\,(1-\bar{\varsigma}_i)\} \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^p, \qquad \text{(A.63)}\end{aligned}$$

which means $\dot{V} \leq 0$ if

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

where π is defined in equation (4.42).

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

$$\left(\sum_{i=1}^{N} \|x_i\|_2^q\right)^{\frac{1}{q}} \leq \left(\sum_{i=1}^{N} \|x_i\|_2^p\right)^{\frac{1}{p}} \leq N^{\frac{1}{p}-\frac{1}{q}} \left(\sum_{i=1}^{N} \|x_i\|_2^q\right)^{\frac{1}{q}} \qquad \text{(A.64)}$$

holds. So equation (A.63) implies that if $p \leq q$, then $\dot{V} \leq 0$ when

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

Similarly, if $p \geq q$, then we have $\dot{V} \leq 0$ when

$$\sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \geq N^{1-\frac{q}{p}} \Delta^q \pi^q.$$

Combining these two cases, we have

$$\dot{V} \leq 0 \quad \text{when} \quad \sum_{i \in \mathcal{N}} \|x_i(t)\|_2^q \geq \mu \Delta^q \pi^q, \tag{A.65}$$

where μ is defined in equation (4.41). By equation (4.38),

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

holds, which, together with equation (A.65), is sufficient to show that there exists $T \geq t_0$, such that

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

holds for all $t \geq T$, as shown in [25]. □

A.15 Proof of Lemma 4.3.3

Proof: Consider the set

$$\Gamma = \left\{ x \in \Lambda \mid \sum_{i \in \mathcal{N}} \|x_i\|_2^q \leq \frac{V(x_0)}{\max_{i \in \mathcal{N}} \bar{L}_i} \right\}. \tag{A.67}$$

According to equation (4.46), we have

$$\min_{i\in\mathcal{N}} \underline{L}_i \sum_{i\in\mathcal{N}} \|x_i\|_2^q \leq V(x) = \sum_{i\in\mathcal{N}} V_i(x_i) \leq \max_{i\in\mathcal{N}} \bar{L}_i \sum_{i\in\mathcal{N}} \|x_i\|_2^q, \tag{A.68}$$

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 equation (4.34) is violated, the inequality

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

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

$$V(s) = V(t_0) \tag{A.69}$$

$$\dot{V}(t) \geq 0, \;\; \forall t \in (s - \epsilon, s). \tag{A.70}$$

Assume that s is the first time in $(t_0, \hat{t})$ satisfying equation (A.69), (A.70) with a parameter ϵ_1. Then we have

$$t_0 < \bar{t} < s < \hat{t} \tag{A.71}$$

$$V(t) \leq V(t_0), \;\; \forall t \in [t_0, s). \tag{A.72}$$

Equation (A.72) implies

$$x(t) \in \Lambda \quad \text{and} \quad \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^q \leq \frac{V(t_0)}{\min_{i\in\mathcal{N}} \underline{L}_i} \tag{A.73}$$

for all $t \in [t_0, s)$ according to equation (A.68).

With the fact that

$$\frac{1}{N^{q-1}} \left(\sum_{i\in\mathcal{N}} \|x_i\|_2 \right)^q \leq \sum_{i\in\mathcal{N}} \|x_i\|_2^q, \quad \forall q \geq 1, \forall x_i \in \mathbb{R}^n, \tag{A.74}$$

equation (A.72), (A.73), (A.74) 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 (4.45), we have

$$g_i\left(x_{\bar{D}_i}(t), \kappa_i(x_{\bar{Z}_i}(t))\right) \leq 2L_i N^{\frac{q-1}{q}} \left(\frac{V(t_0)}{\min_{i\in\mathcal{N}} \underline{L}_i} \right)^{\frac{1}{q}} = \theta_i$$

for all $t \in [t_0, s)$. Also equation (4.48) implies

$$f_k^i - b_k^i \leq \max \left\{ \frac{(1-\xi_k^i)}{2c_i\theta_i} \|x_i(b_{k-1}^i)\|_2, \frac{(1-\xi_k^i)}{c_i\theta_i} \Delta \right\} \tag{A.75}$$

with

$$\Delta = \frac{\theta_i}{2L_i\bar{\pi}N^{1-\frac{1}{\max\{p,q\}}} \max_{i,j\in\mathcal{N}} \left\{ \left(\frac{\bar{L}_i}{\underline{L}_j} \right)^{\frac{1}{q}} \right\}}. \tag{A.76}$$

Therefore, following the same reasoning in Lemma 4.3.2, we have

$$\begin{aligned}\dot{V}(t) &\le \Delta^p \sum_{i\in\mathcal{N}} (\zeta_i - |S_i \cup U_i|\beta_i)\, \bar{\mu}_i - \min_i\{(\zeta_i - |S_i \cup U_i|\beta_i)\,(1-\bar{\varsigma}_i)\} \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^p \\ &= \min_i\{(\zeta_i - |S_i \cup U_i|\beta_i)\,(1-\bar{\varsigma}_i)\} \left[\Delta^p \pi^p - \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^p\right] \qquad \text{(A.77)}\end{aligned}$$

for all $t \in [t_0, s)$, where $\bar{\varsigma}_i$ is defined in equation (4.43). Since $\dot{V}(t) \ge 0$ for all $t \in (s-\epsilon_1, s)$, from equation (A.77), we know

$$\Delta^p \pi^p \ge \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^p, \;\; \forall t \in (s-\epsilon_1, s), \qquad \text{(A.78)}$$

which implies

$$\mu \Delta^q \pi^q \ge \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^q, \;\; \forall t \in (s-\epsilon_1, s), \qquad \text{(A.79)}$$

where μ is defined in equation (4.41).

Therefore, implementing equation (A.76) into the preceding equation implies

$$\mu \Delta^q \pi^q = \frac{V(t_0)\pi^q}{\bar{\pi}^q \max_{i\in\mathcal{N}} \bar{L}_i} \ge \sum_{i\in\mathcal{N}} \|x_i(t)\|_2^q, \;\; \forall t \in (s-\epsilon_1, s). \qquad \text{(A.80)}$$

Since $x(t)$ is continuous, equation (A.80) implies

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

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

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

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 (A.69). Therefore, we conclude that $V(t) \leq V(t_0)$ holds for all $t \geq t_0$. □

A.16 Proof of Theorem 4.3.4

Proof: By Lemma 4.3.3, we know the state trajectory $x(t) \in \Lambda$ for all $t \in t_0$. Therefore, by equation (4.46),

$$\min_{i\in\mathcal{N}} \underline{L}_i \frac{1}{N^{q-1}}\left(\sum_{i\in\mathcal{N}}\|x_i(t)\|_2\right)^q \leq \min_{i\in\mathcal{N}} \underline{L}_i \sum_{i\in\mathcal{N}}\|x_i(t)\|_2^q \leq V(t) \leq V(t_0),\ \ \forall t \geq t_0 \tag{A.81}$$

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

$$\sum_{i\in\mathcal{N}}\|x_i(t)\|_2 \leq N^{\frac{q-1}{q}}\left(\frac{V(t_0)}{\min_{i\in\mathcal{N}}\underline{L}_i}\right)^{\frac{1}{q}},\ \ \forall t \geq t_0. \tag{A.82}$$

According to equation (4.45), we have

$$g_i\left(x_{\bar{D}_i}(t), \kappa_i(\hat{x}_{\bar{Z}_i}(t))\right) \leq 2L_i N^{\frac{q-1}{q}}\left(\frac{V(t_0)}{\min_{i\in\mathcal{N}}\underline{L}_i}\right)^{\frac{1}{q}} = \theta_i \tag{A.83}$$

for all $t \geq t_0$.

Since the hypotheses of Lemma 4.3.2 are satisfied with

$$\Delta := \Delta_1 = \frac{\theta_i}{2L_i\bar{\pi}N^{1-\frac{1}{\max\{p,q\}}}\max_{i,j\in\mathcal{N}}\left\{\left(\frac{\bar{L}_i}{\underline{L}_j}\right)^{\frac{1}{q}}\right\}},$$

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

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

where μ is defined in equation (4.41).

Applying the preceding equation into equation (4.45) yields

$$g_i\left(x_{\bar{D}_i}(t), \kappa_i(\hat{x}_{\bar{Z}_i}(t))\right) \le \frac{\pi}{\bar{\pi}}\theta_i, \quad \forall t \ge t_1. \tag{A.84}$$

We now set $\Delta := \Delta_2 = \frac{\pi}{\bar{\pi}}\Delta_1$ and use the preceding equation to bound the behavior of g_i over $[t_1, \infty)$. Then Lemma 4.3.2 suggests that there exists $t_2 \ge t_1$ such that

$$\begin{aligned}\frac{1}{N^{q-1}}\left(\sum_{i\in\mathcal{N}}\|x_i(t)\|_2\right)^q &\le \sum_{i\in\mathcal{N}}\|x_i(t)\|_2^q \\ &\le \max_{i,j\in\mathcal{N}}\left\{\frac{\bar{L}_i}{\underline{L}_j}\right\}\mu\pi^q\Delta_2^q = \left(\frac{\pi}{\bar{\pi}}\right)^{2q}\frac{V(t_0)}{\min_{i\in\mathcal{N}}\underline{L}_i}, \quad \forall t \ge t_2\end{aligned}$$

With the preceding equation, we can re-compute the bound for g_i over $[t_2, \infty)$ and re-apply Lemma 4.3.2 to get a new ultimate bound on $\sum_{i\in\mathcal{N}}\|x_i(t)\|_2^q$, so on and so forth. Then there exists $t_k > t_0$ such that

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

hold for all $t \ge t_k$. Since $\frac{\pi}{\bar{\pi}} \in (0, 1)$, as $k \to \infty$, the preceding equation implies $x(t) \to 0$, which means the NCS is asymptotically stable. □

A.17 Proof of Theorem 5.1.2

Proof: Consider $\dot{V}$ with $V(x) = x^T Px$ at time t.

$$\begin{aligned}
\dot{V} &= x^T(PA + A^T P)x + 2x^T PBK\hat{x} + 2x^T PCw \\
&= x^T(P(A+BK) + (A+BK)^T P + \frac{1}{\gamma^2}PCC^T P)x \\
&\quad +2x^T PBKe + \gamma^2\|w\|_2^2 - \|\gamma w - \frac{1}{\gamma}C^T Px\|_2^2 \\
&\leq x^T(P(A+BK) + (A+BK)^T P + \frac{1}{\gamma^2}PCC^T P)x \\
&\quad +2x^T PBKe + \gamma^2\|w\|_2^2
\end{aligned}$$

Since equation (5.10) holds, the inequality above can be further reduced as:

$$\begin{aligned}
\dot{V} &\leq -x^T Qx + 2x^T PBKe + \gamma^2\|w\|_2^2 \\
&\leq -x^T(Q - PBKM^{-1}K^T B^T P)x \\
&\quad +e^T Me + \gamma^2\|w\|_2^2
\end{aligned}$$

Combining equation (5.11) and the preceding inequality, we have

$$\begin{aligned}
\dot{V} &\leq -x^T Wx + e^T Me + \gamma^2\|w\|_2^2 \\
&= -\sum_{i\in\mathcal{N}} x_i^T W_i x_i + \sum_{i\in\mathcal{N}} e_i^T M_i e_i + \gamma^2\|w\|_2^2 \\
&\leq -\sum_{i\in\mathcal{N}} \lambda_{\min}(W_i)\|x_i\|_2^2 + \sum_{i\in\mathcal{N}} \lambda_{\max}(M_i)\|e_i\|_2^2 + \gamma^2\|w\|_2^2. \qquad \text{(A.85)}
\end{aligned}$$

Equation (5.13) implies

$$\lambda_{\max}(M_i)\|e_i\|_2^2 \leq \delta_i^2 \lambda_{\min}(W_i)\|x_i\|_2^2 \qquad \text{(A.86)}$$

holds. Applying this inequality into equation (A.85) yields

$$\begin{aligned}\dot{V} &\leq -\sum_{i\in\mathcal{N}}(1-\delta_i^2)\lambda_{\min}(W_i)\|x_i\|_2^2+\gamma^2\|w\|_2^2\\ &\leq -\min_{i\in\mathcal{N}}\left\{(1-\delta_i^2)\lambda_{\min}(W_i)\right\}\|x\|_2^2+\gamma^2\|w\|_2^2\end{aligned}$$

for any $t \geq 0$, which is sufficient to show that the NCS in equation (5.1) is finite-gain $\mathcal{L}_2$ stable with an induced gain less than $\frac{\gamma}{\sqrt{\min_{i\in\mathcal{N}}\left\{(1-\delta_i^2)\lambda_{\min}(W_i)\right\}}}$. □

A.18 Proof of Theorem 5.1.3

Proof: Notice that the inequality still holds when we expand the matrices in equation (5.14) into $nN \times nN$ dimension by appropriately adding zero. Summing both sides of the expanded matrix inequalities yields the satisfaction of equation (5.10) with

$$\begin{aligned}&P = \text{diag}\{P_i\}_{i=1}^N\\ &Q = \text{diag}\{Q_i - |S_i \cup U_i|\beta I_{n\times n}\}_{i=1}^N\\ &\gamma = \max_i\{\gamma_i\},\end{aligned}$$

where $Q_i - |S_i \cup U_i|\beta I_{n\times n} > 0$ holds due to equation (5.15).

Similarly, we can show the satisfaction of equation (5.11) with

$$\begin{aligned}&W = \text{diag}\{W_i\}_{i=1}^N\\ &M = \text{diag}\{\rho(|\bar{U}_i|)I_{n\times n}\}_{i=1}^N\end{aligned}$$

Since the hypotheses in Theorem 5.1.2 are satisfied, we conclude that the NCS is finite-gain $\mathcal{L}_2$ stable with an induced gain less than $\frac{\max_i\{\gamma_i\}}{\sqrt{\min_i\left\{(1-\delta_i^2)\lambda_{\min}(W_i)\right\}}}$. □

BIBLIOGRAPHY

1. A. Anta and P. Tabuada. Self-triggered stabilization of homogeneous control systems. In *American Control Conference*, pages 4129–4134, 2008.

2. A. Anta and P. Tabuada. Space-time scaling laws for self-triggered control. In *47th IEEE Conference on Decision and Control*, pages 4420–4425, 2008.

3. K.E. Arzen. A simple event-based PID controller. In *Proceedings of the 14th IFAC World Congress*, 1999.

4. K.J. Astrom and B.M. Bernhardsson. Comparison of Riemann and Lebesgue sampling for first order stochastic systems. In *Proceedings of IEEE Conference on Decision and Control*, 1999.

5. K.J. Astrom and B. Wittenmark. *Computer-Controlled Systems: theory and design*. Prentice-Hall, third edition, 1997.

6. B. Bamieh. Intersample and finite wordlength effects in sampled-data problems. *IEEE Transactions on Automatic Control*, 48(4):639–643, 2003.

7. B. Bamieh, F. Paganini, and M.A. Dahleh. Distributed control of spatially invariant systems. *IEEE Transactions on Automatic Control*, 47(7):1091–1107, 2002.

8. T. Basar and P. Bernhard. *H_∞-optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*. Birkhauser, 1995.

9. Roger Brockett. Stabilization of motor networks. In *Proceedings of IEEE Conference on Decision and Control*, 1995.

10. Roger Brockett. Minimum attention control. In *Proceedings of IEEE Conference on Decision and Control*, 1997.

11. G. Buttazzo, G. Lipari, M. Caccamo, and L. Abeni. Elastic scheduling for flexible workload management. *IEEE Transactions on Computers*, 51(3):289–302, 2002.

12. E. Camponogara, D. Jia, B.H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Systems*, 22(1):44–52, 2002.

13. Daniele Carnevale, Andrew R. Teel, and Dragan Nesic. Further results on stability of networked control systems: a lyapunov approach. *IEEE Transactions on Automatic Control*, 52:892–897, 2007.

14. A. Cervin, J. Eker, B. Bernhardsson, and K-E Arzen. Feedback-feedforward scheduling of control tasks. *Real-time Systems*, 23(1-2):25–53, 2002.

15. T. Chantem, X. Hu, and M.D. Lemmon. Generalized elastic scheduling. In *IEEE Real Time Systems Symposium*, 2006.

16. T. Chantem, X.S. Hu, and M.D. Lemmon. Period and deadline selection problem for real-time systems. In *Real Time Systems Symposium (work-in-progress track)*, 2007.

17. T. Chantem, X. Wang, M.D. Lemmon, and X.S. Hu. Period and deadline selection for schedulability in real-time systems. In *Euromicro Technical Committee on Real-Time Systems*, 2008.

18. X. Chen, K.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*, 20(3):195–204, 1995.

19. R. D'Andrea and G.E. Dullerud. Distributed control design for spatially interconnected systems. *IEEE Transactions on Automatic Control*, 48(9):1478–1495, 2003.

20. W.B. Dunbar. A distributed receding horizon control algorithm for dynamically coupled nonlinear systems. In *Proceedings of IEEE Conference on Decision and Control*, 2005.

21. P. Gupta and P.R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.

22. D. Hristu-Varsakelis and P.R. Kumar. Interrupt-based feedback control over a shared communication medium. In *Proceedings of IEEE Conference on Decision and Control*, 2002.

23. M.R. Jovanovic and B. Bamieh. Lyapunov-based distributed control of systems on lattices. *IEEE Transactions on Automatic Control*, 50(4):422–433, 2005.

24. PA Kawka and AG Alleyne. Stability and performance of packet-based feedback control over a Markov channel. In *American Control Conference, 2006*, page 6, 2006.

25. H.K. Khalil. *Nonlinear systems*. Prentice Hall Upper Saddle River, NJ, 2002.

26. R. Krtolica, U. Ozguner, H. Chan, H. Goktas, J. Winckelman, and M. Liubakka. Stability of linear feedback systems with random communication delays. In *American Control Conference*, 1991.

27. C. Langbort, R.S. Chandra, and R. D'Andrea. Distributed control design for systems interconnected over an arbitrary graph. *IEEE Transactions on Automatic Control*, 49(9):1502–1519, 2004.

28. E.A. Lee. Computing Foundations and Practice for Cyber-Physical Systems: A Preliminary Report.

29. M.D. Lemmon, T. Chantem, X. Hu, and M. Zyskowski. On self-triggered full information h-infinity controllers. In *Hybrid Systems: computation and control*, 2007.

30. F-L Lian, J. Moyne, and D. Tilbury. Network design consdieration for distributed control systems. *IEEE Transactions on Control Systems Technology*, 10(2):297–307, 2002.

31. Q. Ling and M.D. Lemmon. Robust performance of soft real-time networked control systems with data dropouts. In *Proceedings of IEEE Conference on Decision and Control*, 2002.

32. Q. Ling and MD Lemmon. Optimal dropout compensation in networked control systems. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 1, 2003.

33. Q. Ling and MD Lemmon. Soft real-time scheduling of networked control systems with dropouts governed by a Markov chain. In *American Control Conference, 2003. Proceedings of the 2003*, volume 6, 2003.

34. C.L. Liu and J.W. Layland. Scheduling for multiprogramming in a hard-real-time environment. *Journal of the Association for Computing Machinery*, 20(1):46–61, 1973.

35. P. Marti, C. Lin, S. Brandt, M. Velasco, and J. Fuertes. Optimal state feedback resource allocation for resource-constrained control tasks. In *IEEE Real-Time Systems Symposium (RTSS 2004)*, pages 161–172, 2004.

36. M. Mazo Jr and P. Tabuada. On event-triggered and self-triggered control over sensor/actuator networks. In *47th IEEE Conference on Decision and Control*, pages 435–440, 2008.

37. D. Nesic and A.R. Teel. Input-output stability properties of networked control systems. *IEEE Transactions on Automatic Control*, 49:1650–1667, 2004.

38. D. Nesic, A.R. Teel, and E.D. Sontag. Formulas relating $\mathcal{KL}$ stability estimates of discrete-time and sampled-data nonlinear systems. *Systems and Control Letters*, 38:49–60, 1999.

39. J. Nilsson. Real-Time Control Systems with Delays. *Lund, Sweden: Lund Institute of Technology*, 1998.

40. Luigi Palopoli, Claudio Pinello, Antonio Bicchi, and Alberto Sangiovanni-Vincentelli. Maximizing the stability radius of a set of systems under real-time scheduling constraints. *IEEE Transactions on Automatic Control*, 50:1790–1795, 2005.

41. D. Seto, J.P. Lehoczky, L. Sha, and K.G. Shin. On task schedulability in real-time control systems. In *IEEE Real-time Technology and Applications Symposium (RTAS)*, pages 13–21, 1996.

42. K. G. Shin. Real-time communications in a computer-controlled workcell. *IEEE Transactions on Robotics and Automation*, 7(1):105–113, 1991.

43. M. Tabbara, D. Nesic, and A.R. Teel. Stability of wireless and wireline networked control systems. *IEEE Transactions on Automatic Control*, 52:1615–1630, 2007.

44. P. Tabuada. Event-Triggered Real-Time Scheduling of Stabilizing Control Tasks. *Automatic Control, IEEE Transactions on*, 52(9):1680–1685, 2007.

45. P. Tabuada and X. Wang. Preliminary results on state-triggered scheduling of stabilizing control tasks. In *Proceedings of IEEE Conference on Decision and Control*, 2006.

46. M. Velasco, P. Martí, and E. Bini. Control-Driven Tasks: Modeling and Analysis. In *Proceedings of the 2008 Real-Time Systems Symposium-Volume 00*, pages 280–290. IEEE Computer Society Washington, DC, USA, 2008.

47. M. Velasco, P. Marti, and J.M. Fuertes. The self triggered task model for real-time control systems. In *Work-in-Progress Session of the 24th IEEE Real-Time Systems Symposium (RTSS03)*, 2003.

48. G.C. Walsh, H. Ye, and L.G. Bushnell. Stability analysis of networked control systems. *IEEE Transactions on Control Systems Technology*, 10(3):438–446, 2002.

49. X. Wang and M. Lemmon. Self-triggered feedback control systems with finite-gain $\mathcal{L}_2$ stability. *IEEE Transactions on Automatic Control*, 54(452 – 467):1680–1685, 2008.

50. X. Wang and M.D. Lemmon. Asymptotic stability in distributed event-triggered networked control systems with delays. Submitted to *48th IEEE Conference on Decision and Control*.

51. X. Wang and M.D. Lemmon. Event-triggering in distributed networked control systems. Submitted to *IEEE Transactions on Automatic Control*.

52. X. Wang and M.D. Lemmon. Decentralized event-triggering broadcast over networked systems. In *Hybrid Systems: Computation and Control*, 2008.

53. X. Wang and M.D. Lemmon. Event Design in Event-Triggered Feedback Control Systems. In *47th IEEE Conference on Decision and Control, 2008. CDC 2008*, pages 2105–2110, 2008.

54. X. Wang and M.D. Lemmon. Event-triggered broadcasting across distributed networked control systems. In *American Control Conference*, 2008.

55. X. Wang and M.D. Lemmon. State based self-triggered feedback control systems with $\mathcal{L}_2$ stability. In *Proceedings of the 17th IFAC World Congress*, 2008.

56. X. Wang and M.D. Lemmon. Event-triggering in distributed networked systems with data dropouts and delays. In *Hybrid Systems: Computation and Control*, 2009.

57. X. Wang and M.D. Lemmon. Finite gain $\mathcal{L}_2$ stability in distributed event-triggered networked control systems with data dropouts. Accepted to *European Control Conference*, 2009.

58. X. Wang and M.D. Lemmon. Self-triggered Feedback Systems with State-Independent Disturbances. In *American Control Conference*, 2009.

59. W. S. Wong and R. W. Brockett. Systems with finite communication bandwidth constraints – Part I: State estimation problems. *IEEE Transactions on Automatic Control*, 42(9):1294–1299, 1997.

60. 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*, 44(5):1049–1053, 1999.

61. H. Ye, AN Michel, and L. Hou. Stability theory for hybrid dynamical systems. *Automatic Control, IEEE Transactions on*, 43(4):461–474, 1998.

62. L. Zaccarian, A.R. Teel, and D. Nesic. On finite gain $\mathcal{L}_p$ stability of nonlinear sampled-data systems. *System and Control Letters*, 49:201–212, 2003.

63. W. Zhang, M.S. Branicky, and S.M. Phillips. Stability of networked control systems. *IEEE Control Systems Magazine*, 21:84–99, 2001.

64. Y. Zheng, D.H. Owens, and S.A. Billings. Fast sampling and stability of nonlinear sampled-data systems: Part 2. sampling rate estimations. *IMA Journal of Mathematical Control and Information*, 7:13–33, 1990.

65. K. M. Zuberi and K. G. Shin. Scheduling messages on controller area network for real-time CIM applications. *IEEE Transactions on Robotics and Automation*, 13(2):310–316, 1997.