UNIVERSITY of CALIFORNIA

Santa Barbara

Delay Impulsive Systems: A Framework For Modeling Networked Control Systems

A Dissertation submitted in partial satisfaction of the

requirements for the degree

Doctor of Philosophy

in

Electrical and Computer Engineering

by

Payam Naghshtabrizi

Committee in charge:

Professor João P. Hespanha, Chair

Professor Brad E. Paden

Professor Roy S. Smith

Professor Andrew R. Teel

September 2007

The dissertation of **Payam Naghshtabrizi** is approved.

Professor Brad E. Paden

Professor Roy S. Smith

Professor Andrew R. Teel

Professor João P. Hespanha, Committee Chair

September 2007

Delay Impulsive Systems: A Framework For Modeling Networked Control Systems

by

Payam Naghshtabrizi

To my parents *Azar* and *Mansour*

for their unconditional love and support

Acknowledgements

My warmest appreciation goes to my advisor Professor João Hespanha for having given me the utmost freedom. He provided me a great academic environment and he taught me to be rigorous and ambitious. I am doubtful that I would have successfully completed my Ph.D without João's patience and support. It was an absolute pleasure to work with him.

I would like to thank Professor Brad Paden, Professor Roy Smith, and Professor Andrew Teel for being in my committee. I would also like to thank Professor Petar Kokotović, Professor Michael Safonov, and Professor Mohammad-Ali Masoumnia who inspired me and helped me to deepen my knowledge in the field of control.

I would like to use this opportunity to acknowledge Professor Andrew Teel and Professor Nathan van de Wouw who had great impact on this work. Chapter 3 was motivate by discussions I had with Andy. Nathan introduce me to the tracking problem discussed in Chapter 7 during his sabbatical at UCSB.

I enjoyed every moment of living in Santa Barbara. This town is beautiful and its mountains and beaches are inspiring. I had the opportunity to think about my life during long hikes, to touch with my feelings, and to make important decisions that changed my life forever. Perhaps, my experience would not be the same without meeting great people who later became my dearest friends. It is really hard to have a complete list but to name a few: my dear friend and brother Makan Fardad and his wife Anna Chernobi, Paolo Santesso and his wife Anna colleselli, my roommates Edgar Torres and Timothy Yothers, Arash Naragi, Scott Weatherman, Joseph Topping, John Gonzales, Michael Rogers, Peter Russell, and

Rooin Roshandel. I am also indebted to my officemates and fellow CCDC students for my pleasant days and years in the lab.

My deepest appreciation goes to my parents Azar and Mansour. I am always grateful for all they have done for me and for providing a loving environment. I would like to thank my sisters, Sahar, Sadaf, and Spedieh and my bother in laws, Alireza and Arta. I want to acknowledge my aunt Nosrat and her family who helped me adapt to the life in the USA and welcoming me in their home for more than a year. Also I want to acknowledge my uncle Mohsen who has been a great inspiration for me.

Curriculum Vitæ

Payam Naghshtabrizi

Education

Ph.D.	University of California at Santa Barbara, Sept. 2007.
M.S.	University of Southern California, Los Angeles, Jan. 2002.
B.S.	Sharif University of Technology, Iran, Jan. 1997.

Experience

Graduate Researcher	University of California at Santa Barbara, 2002 - 2007.
Teacher Assistant	University of Southern California, 2001 - 2002.
R&D Engineer	Heating Company, Iran, 1998 - 2000.
R&D Engineer	Sazeh Pouyesh, Iran, 1997 - 1998.

Jurnal Papers and Book Chapters

1. P. Naghshtabrizi, J. P. Hespanha and A. R. Teel, "On the exponential stability of impulsive systems with application to uncertain sampled-data systems". *To be published in Systems & Control Letters.*

2. J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "Network control systems: analysis and design," *The Proceedings of the IEEE*. Vol. 95, No. 1, Jan. 2007, 138-162.

3. P. Naghshtabrizi and J. P. Hespanha, "Anticipative and non-anticipative controller design for network control systems," *Network Embedded sensing and control*, Springer, 203-218.

Work in Progress

1. N. van de Wouw, P. Naghshtabrizi, M. Cloosterman, and J. P. Hespanha, " Tracking control for network control systems with uncertain samplings and delays". *To be submitted for publication in Int. Journal of Robust and Nonlinear Control.*

2. P. Naghshtabrizi, J. P. Hespanha and A. R. Teel, "Stability Theory of Delay Impulsive Systems with Application to Network Control Systems". *To be submitted for publication in Transaction on Automatic Control.*

3. P. Naghshtabrizi and J. P. Hespanha "Communication protocol design for deterministic networks". *To be submitted for publication in The Proceedings of 2008 American Control Conference.*

Refereed Proceedings

1. N. van de Wouw, P. Naghshtabrizi, M. Cloosterman, and J. P. Hespanha, " Tracking Control for Networked Control Systems",*To appear 46th Conf. on Decision and Control, Dec. 2007.*

2. P. Naghshtabrizi, J. P. Hespanha and A. R. Teel, "Stability Theory of Delay Impulsive Systems with Application to Network Control Systems," *The Proceedings of 2007 American Control Conference*, 4499-4504.

3. P. Naghshtabrizi and J. P. Hespanha, "Stability of network control systems with variable sampling and delays," *44th Allerton Conf. On Comm., Cont., and Computing*, Sept. 2006.

4. P. Naghshtabrizi and J. P. Hespanha, "Designing transparent stabilizing controllers for haptic systems," *The Proceedings of 2006 American Control Conference*, 2475-2480.

5. P. Naghshtabrizi, J. P. Hespanha and A. R. Teel, "On the robust stability and stabilization of sampled-data systems: A hybrid system approach," *45th Conf. on Decision and Control, Dec. 2006*, 4873-4878.

6. P. Naghshtabrizi and J. P. Hespanha, " Designing observer-type controllers for network control systems," in *44th Conf. on Decision and Control, Dec. 2005*, 848-853.

Abstract

Delay Impulsive Systems: A Framework For Modeling Networked Control Systems

by

Payam Naghshtabrizi

We model Networked Control Systems (NCSs) with variable delay, sampling intervals and packet dropouts as delay impulsive systems which exhibit continuous evolutions described by ODEs and state jumps or impulses that experience delay. We develop theorems for the exponential stability of nonlinear time-varying delay impulsive systems which can be viewed as extensions of the Lyapunov-Krasovskii Theorem for time-delay systems. For linear plants and controllers, exponential stability conditions can be formulated as Linear Matrix Inequalities (LMIs), which can be solved numerically. By solving these LMIs, one can find classes of delay-sampling sequences for the different sample-hold pairs in a NCS such that exponential stability is guaranteed.

The timing requirements of delay-sampling sequences can be met by deterministic networks for which delivery of packets can be guaranteed with bounded delay. Scheduling theory provides conditions to check whether all the timing requirements can be met. If appropriate scheduling conditions are satisfied, the network will in fact be capable of delivering all the packets on time, and stability of all systems connected to the network is guaranteed. Our analysis leads to the

design of communication protocols to determine which nodes gain access to the network and an algorithm to select sampling sequences.

We also consider the tracking problem over the network. A feedforward structure is used to force the state of the plant to follow a desired trajectory and feedback structure is used to obtain the desired performance and robustness. Since the feedback and the feedforward control commands are sampled and experience variable delays, exact trajectory tracking is not possible and there is an error between the desired trajectory and the real trajectory of the system. The error dynamics can be modeled as an impulsive system driven by an external input corresponding to feedforward signal mismatch. Sufficient condition for the Input-to-State Stability (ISS) of the tracking error dynamics with respect to this input is given. These results also provide classes of sampling-delay sequences for which the steady-state tracking error is guaranteed to be smaller than a desired level.

Contents

List of Figures

Chapter 1

Introduction

Network Control Systems (NCSs) are spatially distributed systems in which the communication between sensors, actuators, and controllers occurs through a shared band-limited digital communication network, as shown in Fig. 1.1.

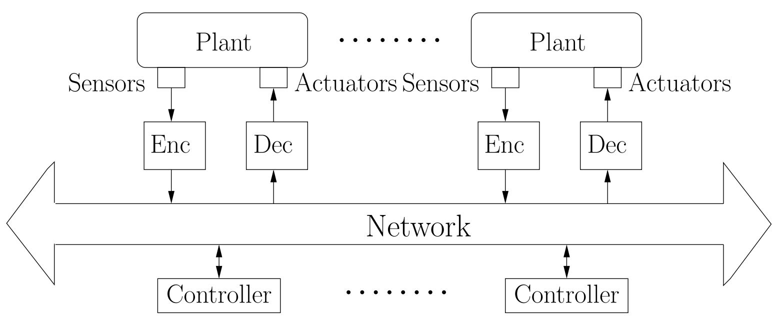


Figure 1.1. General NCS architecture.

The use of a multi-purpose shared network to connect spatially distributed elements results in flexible architectures and generally reduces installation and maintenance costs. Consequently, NCSs have been finding application in a broad range of areas such as mobile sensor networks [54], remote surgery [34], haptics collaboration over the Internet [19, 22, 59], and automated highway systems and unmanned aerial vehicles [57, 58]. However, the use of a shared network—in contrast to using several dedicated independent connections—introduces new challenges and Murray et al. [40] identify *control over networks* as one of the key future directions for control.

NCSs lie at the intersection of control and communication theories. Traditionally, control theory focuses on the study of interconnected dynamical systems linked through “ideal channels”, whereas communication theory studies the transmission of information over “imperfect channels”. A combination of these two frameworks is needed to model NCSs. This dissertation is primarily written from a controls perspective and attempts to address several key issues that make NCSs distinct from other control systems.

Sampling and delay To transmit a continuous-time signal over a network, the signal must be sampled, encoded in a proper digital format called *packet*, transmitted over the network, and finally the data must be decoded at the receiver side. This process is significantly different from the usual periodic sampling in digital control. A significant number of results have attempted to characterize a maximum upper-bound on the sampling interval for which stability can be guaranteed. These results implicitly attempt to minimize the *packet-rate* that is needed to stabilize a system through feedback [37, 72].

The overall *delay* between sampling and eventual decoding at the receiver can be highly variable because both the network access delays (i.e., the time it takes for a shared network to accept data) and the transmission delays (i.e., the time during which data are in transit inside the network) depend on highly variable network conditions such as congestion and channel quality. In some NCSs, the data transmitted are time-stamped, which means that the receiver may have an estimate of the delay's duration and take appropriate corrective action.

Packet dropout Another significant difference between NCSs and standard digital control is the possibility that data may be lost while in transit through the network. Typically, *Packet dropouts* result from transmission errors in physical network links (which is far more common in wireless than in wired networks) or from buffer overflows due to congestion. Long transmission delays sometimes result in packet re-ordering, which essentially amounts to a packet dropout if the receiver discards "outdated" arrivals. Reliable transmission protocols, such as TCP, guarantee the eventual delivery of packets. However, these protocols are not appropriate for NCSs since the re-transmission of old data is generally not very useful.

Communication protocol In standard digital control systems samples are sent periodically and in a cyclic manner. However, in NCSs a communication protocol grants the access to the network. The communication protocol should be capable of supporting time-critical requirements for control systems. Static communication protocols such as Round-Robin [68] grant the access to the network in a periodic and pre-specified fashion inspired by time multiplexing Medium Access Control (MAC) protocols. Dynamic communication protocols such as Try-Once-Discard [68, 45, 63, 7] grant network access to a node based on the dynamics of the control system. These communication protocols are specifically designed for NCSs to improve the closed-loop system stability and performance.

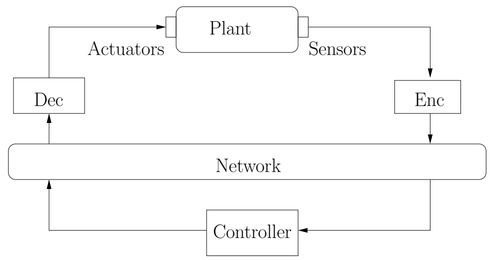


Figure 1.2. Single-loop NCS.

Systems architecture Figure 1.1 shows the general architecture of a NCS. In this figure, *encoder* blocks map measurements into streams of bits grouped into packets. Encoders serve two purposes: they decide *when* to sample a continuous-time signal for transmission and *what* to send through the network. Conversely, *decoder* blocks perform the task of mapping the streams of bits received from the network into continuous actuation signals. One could also include in Fig. 1.1 encoding/decoding blocks to mediate the controllers' access to the network. We do not explicitly represent these blocks because the boundaries between a digital controller and encoder/decoder blocks are often blurry. Throughout this thesis the encoder is simply a sampler and the decoder is a hold. However, in Chapter 5 we will consider more sophisticated encoder/decoder pairs. We will generally include the sensor and actuator into the plant model.

Most of the research on NCSs considers structures simpler than the general one depicted in Fig. 1.1. For example, some controllers may be co-located (and therefore can communicate directly) with the corresponding actuators. It is also often common to consider a single feedback loop as in Fig. 1.2. Although considerably simpler than the general system shown in Fig. 1.1, we will consider this architecture in Chapters 3, 4, and 7 to capture many important characteristics of NCSs such as bandwidth limitations, variable communication delays, and packet dropouts. However, in Chapters 5 and 6 we consider the general structure depicted in Fig. 1.1 to study systems with distributed sensors and actuators and to design communication protocols.

1.1 Main contributions of the dissertation

The main contributions of this dissertation can be summarized as follows:

Stability of nonlinear time-varying delay impulsive systems Motivated by NCSs with delay, we study delay impulsive systems. Delay impulsive dynamical systems exhibit continuous evolutions described by Ordinary Differential Equations (ODEs) and state jumps or impulses that experience delays. We develop general theorems for stability, asymptotic stability, and exponential stability of nonlinear time-varying delay impulsive systems. Our theorems can be viewed as extensions of the Lyapunov-Krasovskii Theorem for time-delay systems. A distinguishing feature of the stability conditions in this thesis is that, when specialized to linear impulsive systems, the stability tests can be formulated as Linear Matrix Inequalities (LMIs) that can be solved efficiently.

Stability of NCSs with variable sampling, delays, and packet dropouts We model NCSs with variable sampling intervals, variable delays and packet dropouts as delay impulsive systems. We assume that the plants and controllers in the network are linear and we present stability conditions in terms of LMIs. By solving these LMIs, one can find classes of delay-sampling sequences for each sample-hold pair in the NCS such that exponential stability of all systems connected to the network is guaranteed. The advantages of our approach are as follows: (i) different network effects, i.e., variable sampling, delay and packet dropout can be modeled in a unified framework, (ii) unlike most of the results in the literature, we allow delays to be significantly larger than the sampling intervals, and (iii) the stability conditions take into account the minimum and maximum values for the delay.

Communication protocol design and sampling selection The results in this thesis are applicable to deterministic networks for which packet delivery can be guaranteed with bounded delay. Examples of such networks include token-passing networks, CAN, FlexRay, and switched networks. Based on the scheduling theory of real-time systems, we provide conditions to guarantee that the network will be capable of delivering all the packets on time such that stability of all systems connected to the network is guaranteed. Our analysis leads to the design of communication protocols to determine when each node should gain access to the network and also to an algorithm to select the sampling times.

Tracking over the network We consider the tracking problem over the network, in which a controller sends commands over the network to cause the output or the state of the plant to follow a reference path. A feedforward control structure is used to improve tracking performance and feedback is used for convergence to the desired solution and favorable robustness and disturbance attenuation. Since the feedback and the feedforward control commands are sampled and experience variable delays, exact trajectory tracking is not possible, resulting in an error between the desired and the real trajectories of the system. The error dynamics can be modeled as an impulsive system driven by an external input corresponding to feedforward signal mismatch. Sufficient conditions for the input-to-state stability

(ISS) of the tracking error dynamics with respect to this input are given. These results also provide classes of sampling-delay sequences such that the steady-state tracking error is guaranteed to be smaller than a desired level.

1.2 Organizational Outline

The thesis is organized as follows:

Chapter 2: Review Of Related Work This chapter summarizes some of the previous work on NCSs with variable sampling, delays, and packet dropouts. We also briefly review previous work on the Round-Robin and Try-Once-Discard communication protocols.

Chapter 3: Impulsive Systems: A Model For NCSs With Variable Sampling, SISO Case Impulsive dynamical systems exhibit continuous evolutions described by Ordinary Differential Equations (ODEs) and instantaneous state jumps or impulses. In this chapter, we establish exponential stability of nonlinear time-varying impulsive systems by employing Lyapunov functions with discontinuities at a countable set of times.

Then we consider NCSs that can be modeled as a SISO impulsive system. We consider a linear plant with a static feedback controller where the sampling intervals are uncertain and variable. We assume that the delays in the network are negligible, but this framework is general enough to capture packet dropout effects.

We apply our theorems to the analysis and the state-feedback stabilization of such systems. Our stability and stabilization results are presented as Linear Matrix Inequalities (LMIs). By solving these LMIs, one can find an upper bound on the sampling intervals such that the stability of the closed-loop is guaranteed. The control design LMIs also provide a controller gain to stabilize the system. For the sake of brevity, we consider control designs only in chapter 3, but the same procedure can be used to design state-feedback controllers for other models in this thesis.

Chapter 4: Delay Impulsive Systems: A Model For NCSs With Variable Sampling And Delay, SISO Case In this chapter we present stability, asymptotic stability, and exponential stability theorems for delay impulsive systems by employing functionals with discontinuities at a countable set of times.

These results allow us to study stability of the NCSs considered in Chapter 3, but with potentially large delays. By solving appropriate stability LMIs, one can find a positive constant that determines an upper bound between the sampling time and the next input update time for which the stability of the closed-loop system is guaranteed, for given lower and upper bounds on the total loop delay. When the delay in the feedback loop is small, these LMIs reduce to the ones presented in Chapter 3. This observation shows that the results in Chapter 3 are robust to small delays.

Chapter 5: Delay Impulsive Systems: A Model For NCSs With Variable Sampling And Delay, MIMO Case To enlarge the class of NCSs, we consider MIMO delay impulsive systems. This framework is general enough to model both one-channel NCSs and multi-channel NCSs with dynamic output-feedback controllers that may or may not be "anticipative". For LTI plants and controllers, we present two stability tests in terms of LMIs. The first one is less conservative but the number of LMIs grows exponentially with the number of sample-hold pairs. The second stability condition is based on the feasibility of a single LMI with dimension that grows linearly with the number of sample-hold pairs. For a smaller number of sample-hold pairs the first stability test is preferable, because it leads to less conservative results, but the second stability test is more adequate for systems with many sample-hold pairs. By solving these LMIs, one obtains positive constants that determine the upper bound between a sampling time and the next input update time for each sample-hold pair for given lower and upper bounds on the total delay in each loop.

Chapter 6: Communication Protocol Design For Deterministic Networks This chapter proposes design of communication protocols that guarantee stability of all subsystems connected to a deterministic network. Based on the scheduling theory of real-time systems, we provide conditions to guarantee that the network will be capable of delivering all the packets on time such that stability of all systems connected to the network is guaranteed. Our analysis leads to the design of communication protocols to determine when each node should gain access to the network and also to an algorithm for selecting the sampling times. Then we implement the proposed communication protocol on Control Area Networks (CAN) in a distributed fashion.

Chapter 7: Input-To-State Stability Of Delay Impulsive System With Application To Tracking Over Network We consider the tracking problem over the network, in which a controller sends commands over the network to cause the output or the state of the plant to follow a reference path. A feedforward control structure is used to improve tracking performance and feedback is used for convergence to the desired solution and favorable robustness and disturbance attenuation. Since the feedback and the feedforward control commands are sampled and experience variable delays, exact trajectory tracking is not possible, resulting in an error between the desired and the real trajectories of the system. The error dynamics can be modeled as an impulsive system driven by an external input corresponding to feedforward signal mismatch. Sufficient conditions for the input-to-state stability (ISS) of the tracking error dynamics with respect to this input are given. These results also provide classes of sampling-delays such that the steady-state tracking error is guaranteed to be smaller than a desired level.

Chapter 8: Conclusion and future directions The last chapter is dedicated to the conclusions and directions for future research.

1.3 Notation and basic definition

We denote the transpose of a matrix P by P' and the smallest and the largest eigenvalue of a matrix P by $\lambda_{\min}(P)$ and $\lambda_{\max}(P)$. We write $P > 0$ (or $P < 0$) when P is a symmetric positive (or negative) definite matrix and we write a symmetric matrix $\begin{bmatrix} A & B \\ B' & C \end{bmatrix}$ as $\begin{bmatrix} A & B \\ * & C \end{bmatrix}$. The notations $0_{ij}, I_k$ are used to denote a $i \times j$ matrix with zero entries and a $k \times k$ identity matrix. When there is no confusion about the size of such matrices we drop the subscript with the dimensions.

We denote the limit from below and above of a signal $x(t)$ by $x^-(t)$ and $x^+(t)$ respectively, i.e., $x^-(t) := \lim_{\tau \uparrow t} x(\tau)$, and $x^+(t) := \lim_{\tau \downarrow t} x(\tau)$. The left-hand side derivative of x with respect to t, i.e., $\dot{x}(t) := \lim_{\tau \uparrow t} \frac{x(\tau)-x(t)}{\tau - t}$ is denoted by $\dot{x}(t)$.

A function $\alpha \in [0,\infty) \rightarrow [0,\infty)$ is of class $\mathcal{K}$, and we write $\alpha \in \mathcal{K}$ when α is continuous, strictly increasing, and $\alpha(0) = 0$. If α is also unbounded, then we say it is of class $\mathcal{K}_\infty$ and we write $\alpha \in \mathcal{K}_\infty$.

A (continuous) function $\beta : [0,\infty) \times [0,\infty) \rightarrow [0,\infty)$ is of class $\mathcal{KL}$, and we write $\beta \in \mathcal{KL}$ when, $\beta(.,t)$ is of class $\mathcal{K}$ for each fixed $t \geq 0$ and $\beta(s,t)$ decreases to 0 as $t \rightarrow \infty$ for each fixed s.

Notation that is specific to a chapter will be introduced as needed.

Chapter 2

Review of related work

In this chapter we review some of the related work in stability and controller synthesis when the feedback loops are closed over a network. This problem is motivated by scenarios in which sensors, controllers, and actuators are not co-located and use a shared network to communicate. We ignore quantization and focus our attention on the effects of data sampling, network delay, and packet dropouts on the stability of the resulting NCSs.

2.1 Stability

First we present a collection of results on the sampling and delay effect on the stability of NCSs. Then we consider the effect of packet dropout where the underlying assumption is that the sampling interval is constant. In section 2.1.3 we review the results that consider the effect of sampling, delay and packet dropout in a unified framework which are highly related to the results in this thesis.

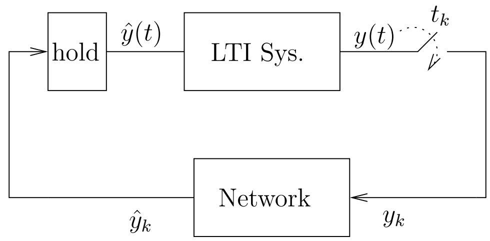


Figure 2.1. One-channel feedback NCS with LTI plant/controller

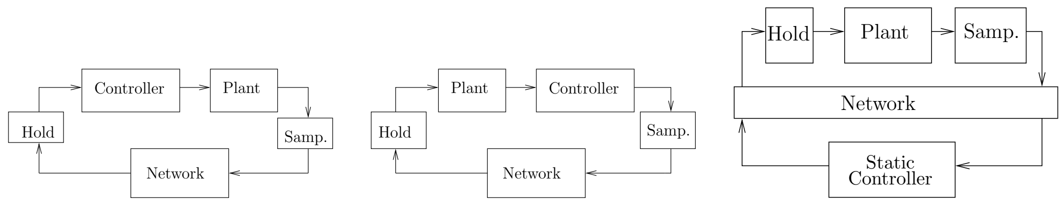
2.1.1 Sampling and delay

The *one-channel feedback NCS* in Fig. 2.1 has been commonly used to investigate the effects of sampling and delay in the stability of NCSs. The LTI system encapsulates a linear time-invariant plant/controller pair modeled by the following continuous-time system

$$\dot{x} = Ax + B\hat{y}, \qquad\qquad y = Cx. \tag{2.1}$$

This one-channel feedback NCS can capture several NCS configurations. The signal y can be regarded as a vector of sensor measurements and $\hat{y}$ as the input to a continuous-time controller collocated with the actuators, as in Fig. 2.2(a) [57, 37]. Alternatively, $\hat{y}$ can be viewed as the input to the actuators and y as the desired control signal computed by a controller collocated with the sensors, as in Fig. 2.2(b). In either case, x would include the states of the plant and the controller. The block diagram in Fig. 2.1 also captures the case of a static controller that is not collocated with the sensors nor with the actuators as in Fig. 2.2(c), because a memoryless controller could be moved next to the actuators, without affecting the stability of the closed loop [3, 76].

In the one-channel feedback NCS in Fig. 2.1, the signal $y(t)$ is sampled at times $\{t_k : k \in \mathbb{N}\}$ and the samples $y_k := y(t_k)$, $\forall k \in \mathbb{N}$ are sent through the



(a) (b) (c)

Figure 2.2. NCS control architectures captured by the one-channel feedback NCS in Fig. 2.1

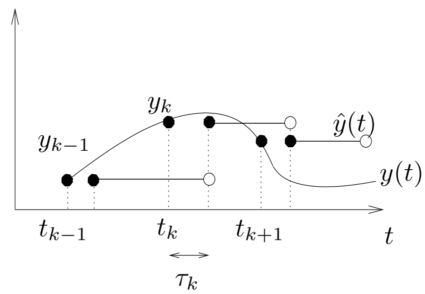


Figure 2.3. The piecewise constant signal $\hat{y}(t)$ defined by (2.3).

network. In a lossless network, we have

$$\hat{y}_k = y_k, \quad \forall k \in \mathbb{N}, \tag{2.2}$$

but the samples only arrive at the destination after a (possibly variable) delay of $\tau_k \geq 0$. At these times $\hat{y}(t)$ is updated, leading to

$$\hat{y}(t) = \begin{cases} y_{k-1} & t \in [t_k, t_k + \tau_k) \\ y_k & t \in [t_k + \tau_k, t_{k+1}) \end{cases} \tag{2.3}$$

where we assume that the network delays are always smaller than one sampling interval, i.e., that $t_k + \tau_k < t_{k+1}$, $\forall k \in \mathbb{N}$ (cf. Fig 2.3). Defining $x_k := x(t_k)$, $\forall k \in \mathbb{N}$ and applying the variation of constants formula to (2.1) and (2.3) we conclude that

$$x_{k+1} = e^{A(t_{k+1}-t_k)}x_k + e^{A(t_{k+1}-t_k-\tau_k)}\Gamma(\tau_k)B\hat{y}_{k-1} + \Gamma(t_{k+1}-t_k-\tau_k)B\hat{y}_k, \quad (2.4)$$

where $\Gamma(s) := \int_0^s e^{Az}dz$, $\forall s \in \mathbb{R}$. To analyze this NCS, one can define an augmented discrete-time state $z'_k := \begin{bmatrix} x(t_k)' & \hat{y}'_{k-1} \end{bmatrix}$, $\forall k \in \mathbb{N}$ and conclude from (2.2) and (2.4) that it evolves according to

$$z_{k+1} = \Phi(t_{k+1}-t_k, \tau_k)z_k, \quad \forall k \in \mathbb{N}, \quad (2.5)$$

where

$$\Phi(a,b) := \begin{bmatrix} e^{Aa} + \Gamma(a-b)BC & e^{A(a-b)}\Gamma(b)B \\ C & 0 \end{bmatrix}, \quad \forall a, b \in \mathbb{R}.$$

In the absence of delay (i.e., when $\tau_k = 0$, $\forall k \in \mathbb{N}$), the state z_k does not need to include $\hat{y}_{k-1}$ and therefore the function $\Phi(\cdot)$ in (2.5) can simply be defined by

$$\Phi(a,0) := e^{Aa} + \Gamma(a)BC, \quad \forall a > 0. \quad (2.6)$$

When the plant (2.1) is open-loop unstable, (2.5) will generally be unstable if the interval between sampling times becomes large. In view of this, significant work has been devoted to finding upper bounds on $t_{k+1} - t_k$, $\forall k \in \mathbb{N}$ for which stability can be guaranteed. These upper bounds are sometimes called the *maximum allowable transfer interval* (MATI) [67].

Delays longer than one sampling interval may result in more than one $\hat{y}_k$ (or none) arriving during a single sampling interval, making the derivation of recursive formulas like (2.5) difficult. All results reviewed in this section are based on a reduction of the NCS to some form of discrete-time system such as (2.5), for simplicity we will therefore implicitly assume delays smaller than one sampling interval. This restriction will be lifted in Section 2.1.3.

Periodic sampling and constant delay When $y(t)$ is sampled periodically and the delay is constant, the discrete-time system (2.5) is time-invariant and it is straightforward to establish its stability:

Theorem 1 ([3]). *Assuming that there exist constants $h > \tau \geq 0$ such that*

$$t_{k+1} - t_k = h, \quad \tau_k = \tau, \quad \forall k \in \mathbb{N},$$

the NCS (2.1)–(2.3) *in Fig. 2.1 is exponentially stable if and only if* $\Phi(h, \tau)$ *is Schur (i.e., all its eigenvalues have magnitude strictly less than one).* □

Remark 1. Defining the alternative augmented state $\bar{z}_k' := \begin{bmatrix} x(t_k + \tau_k)' & x(t_k)' \end{bmatrix}$, $\forall k \in \mathbb{N}$ we obtain

$$\bar{z}_{k+1} = \bar{\Phi}(t_{k+1} - t_k - \tau_k, \tau_{k+1})\bar{z}_k, \quad \forall k \in \mathbb{N}, \tag{2.7}$$

$$\bar{\Phi}(a, b) := \begin{bmatrix} e^{A(a+b)} & \Gamma(a+b)BC \\ e^{Aa} & \Gamma(a)BC \end{bmatrix}, \quad \forall a, b \in \mathbb{R}, \tag{2.8}$$

from which stability of the NCS can also be deduced. Zhang et al. [76] use results from [2] on the stability of nonlinear hybrid systems to conclude that Schurness of $\bar{\Phi}(h, \tau)$ is a sufficient condition for stability of the NCS in the time-invariant case. From (2.7) one can see that this condition is also necessary. □

While some network protocols guarantee constant delay, such as the Controller Area Network (CAN) protocol [26], most protocols introduce delays that can vary significantly from message to message. Variable delays can be equalized by introducing a buffer at the receiver, where data packets can be held so that all packets appear to have the same delay from the perspective of the NCS [33]. However, the downside of delay equalization is that all packets will appear to have a delay as large as the worst-case delay that the network can introduce.

Periodic sampling and variable delay Suppose that the sampling intervals are constant and equal to h and delay takes values equal to $\kappa\frac{h}{N}$ where $\kappa \in 0, 1, \cdots, D_{\max}$ and $D_{\max} \leq N \in \mathbb{N}$. This situation happens when computation and transmission delays are negligible and access delays serve as the main source of delays in NCS [30, 29, 31]. Under these assumptions the closed-loop system (2.5) can be written as a discrete-time switched system with $D_{\max} + 1$ modes as follows

$$z_{k+1} = A_{\sigma_k} z_k, \qquad \forall k \in \mathbb{N},$$

where the *switching signal* σ_k takes values from $\{0, 1, \cdots D_{\max}\}$ at each time step and when $\sigma_k = \kappa$

$$A_\kappa := \Phi\big(h, \kappa\frac{h}{N}\big) \qquad \kappa \in \{0, 1, \cdots D_{\max}\}$$

Lin et al. [30] assume that for the case of no delay or small delays ($\kappa \leq N_0$), the corresponding state matrix, A_κ, is Schur stable while for the case of large delay ($\kappa > N_0$), A_κ is not Schur stable. Using *average dwell time* results for discrete switched systems [74] provides conditions such that NCS stability is guaranteed. Also the authors consider robust disturbance attenuation analysis for this class of NCSs.

Remark 2. One Packet dropout can be modeled as an extra mode where $\kappa = N$. The authors extended the results for the case of consecutive packet dropouts in [29].

Variable sampling and delay When the network delay is not constant or when the signal $y(t)$ is sampled in a non-periodic fashion, (2.5) is not time-invariant and eigenvalue argument does not suffice and one can employ Lyapunov-based argument to prove its stability. The following result is adapted from [75][1] and expresses a sufficient condition for $V(z) := z'Pz$ to be a Lyapunov function for (2.5), from which stability of the NCS can be deduced:

Theorem 2. *Assume that there exist constants $h_{\min}, h_{\max}, \tau_{\min}, \tau_{\max}$ such that*

$$0 \le h_{\min} \le t_{k+1} - t_k \le h_{\max}, \qquad 0 \le \tau_{\min} \le \tau_k \le \tau_{\max}, \quad \forall k \in \mathbb{N}, \tag{2.9}$$

and that $\tau_{\max} < h_{\min}$. The NCS (2.1)–(2.3) *in Fig. 2.1 is exponentially stable if there exists a symmetric matrix P such that*

$$P > 0, \quad \Phi(h,\tau)' P \Phi(h,\tau) - P < 0, \quad \forall h \in [h_{\min}, h_{\max}],\ \tau \in [\tau_{\min}, \tau_{\max}]. \tag{2.10}$$

□

From a numerical perspective, it is generally not simple to find a matrix P that satisfies (2.10) for *all* values of h and τ in the given intervals. However, testing the existence of a matrix P that satisfies (2.10) for values of h and τ on a *finite grid* leads to a finite set of Linear Matrix Inequalities (LMIs) that is easy to solve. Zhang and Branicky [75] propose a randomized algorithm to find the largest value of $h_{\max}$ for which stability can be guaranteed when $h_{\min} = \tau_{\min} = \tau_{\max} = 0$.

Model-based controller Montestruque and Antsaklis [35, 36, 37, 38, 39] consider the model-based one-channel feedback NCS in Fig. 2.4, in which the signal y transmitted across the network is the state of an LTI plant

$$\dot{x}_P = A_P x_P + B_P u, \qquad\qquad y = x_P \tag{2.11}$$

[1] A special case of Theorem 2 with $h_{\min} = \tau_{\min} = \tau_{\max} = 0$ and the matrix $\Phi(\cdot)$ given by (2.6) can be found in [75].

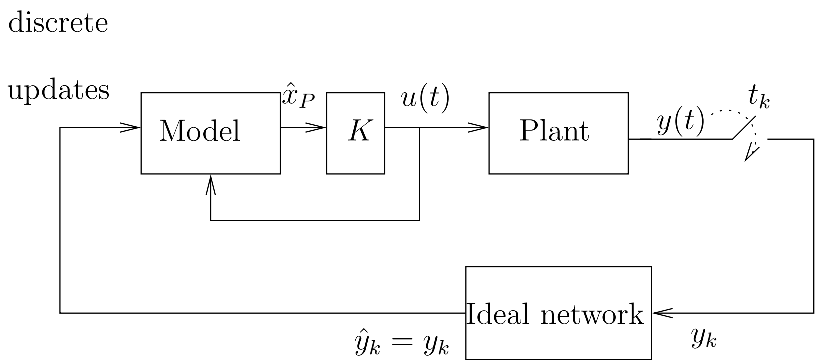


Figure 2.4. State feedback model-based NCS.

whose input u is generated by an estimator-based controller collocated with the actuators. In general, an exact model of the plant is not available and therefore the controller must construct an estimate $\hat{x}_P$ of x_P based on the following approximate plant model

$$\dot{\hat{x}}_P = \hat{A}_P \hat{x}_P + \hat{B}_P u, \tag{2.12}$$

which is instantaneously updated at the sampling times $\{t_k : k \in \mathbb{N}\}$ using the true value of x_P coming from the network. The key difference between the NCSs in Figures 2.1 and 2.4 is that in the former the data coming from the network is held constant between sampling times, whereas in the later this data is used to instantaneously update the state of the controller. Assuming that the network delay is negligible, the controller updates its state estimate according to

$$\hat{x}_P(t_k) = y(t_k) = x_P(t_k), \quad \forall k \in \mathbb{N}. \tag{2.13}$$

It then uses a certainty equivalence control law of the form

$$u = K\hat{x}_P, \tag{2.14}$$

with the matrix K chosen so that $\hat{A}_P + \hat{B}_P K$ is Hurwitz (i.e., all its eigenvalues have strictly negative real part). Note that the matrices in the plant model

(2.11) and the estimator model (2.12) generally do not match due to parametric uncertainty.

Because of (2.13), we conclude that the state estimation error $e := x_P - \hat{x}_P$ is reset to zero at sampling times and therefore its dynamics are defined by

$$\dot{e} = (\tilde{A}_P + \tilde{B}_P K)x_P + (\hat{A}_P - \tilde{B}_P K)e, \qquad e(t_k) = 0,$$

where the matrices $\tilde{A}_P := A_P - \hat{A}_P$ and $\tilde{B}_P := B_P - \hat{B}_P$ represent the difference between the actual plant and the model used to build the estimator. Defining $z' := \begin{bmatrix} x'_P & e' \end{bmatrix}$, we conclude that the overall closed-loop evolves according to the following impulsive system

$$\dot{z} = \Lambda z, \quad z(t_k) = \begin{bmatrix} x_P^-(t_k) \\ 0 \end{bmatrix}, \tag{2.15}$$

where

$$\Lambda := \begin{bmatrix} A_P + B_P K & -B_P K \\ \tilde{A}_P + \tilde{B}_P K & A_P - \tilde{B}_P K \end{bmatrix}. \tag{2.16}$$

Defining the discrete-time state $z_k := z(t_k)$, $\forall k \in \mathbb{N}$, we obtain the following model for its evolution

$$z_{k+1} = M(t_{k+1} - t_k)z_k, \ \ \forall k \in \mathbb{N}, \qquad M(a) := \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} e^{\Lambda a} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \ \ \forall a > 0.$$

The following result is adapted from [37] and follows from standard results on the stability of discrete-time systems:

Theorem 3. *The following two results hold for the NCS* (2.11)–(2.14) *in Fig. 2.4 (there is no delay and packet dropout because the network is ideal):*

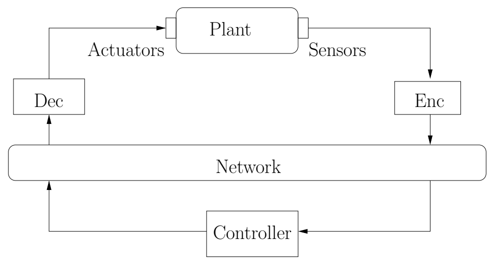


Figure 2.5. Single-loop NCS.

1. *When there exists a constant* h *such that* $t_{k+1} - t_k = h$, $\forall k \in \mathbb{N}$, *the NCS is globally exponentially stable if and only if* $M(h)$ *is Schur.*

2. *When there exist constants* $h_{\min}$ *and* $h_{\max}$ *such that* $0 \leq h_{\min} \leq t_{k+1} - t_k \leq h_{\max}$, $\forall k \in \mathbb{N}$, *the NCS is exponentially stable if there exists a symmetric matrix* P *such that*

$$P > 0, \qquad M(h)PM(h) - P < 0, \qquad \forall h \in [h_{\min}, h_{\max}]. \qquad \square$$

For periodic sampling, Montestruque and Antsaklis [35, 36, 37] use a similar approach to determine the maximum value of $h := t_k - t_{k-1}$, $\forall k \in \mathbb{N}$ for which the NCS is stable, both under state and output feedback.

General nonlinear case Consider a nonlinear plant and remote controller with exogenous disturbances of the following form:

$$\dot{x}_P = f_P(x_P, \hat{u}, w), \qquad y = g_P(x_P), \tag{2.17a}$$

$$\dot{x}_C = f_C(x_C, \hat{y}, w), \qquad u = g_C(x_C), \tag{2.17b}$$

where x_P and x_C are the states of the plant and the controller; $\hat{u}$ and y the plant's input and output; $\hat{y}$ and u the controller's input and output; and w an exogenous disturbance. The plant and the controller are connected through a *two-channel* *feedback NCS* as in Fig. 2.5. Ignoring network delay, between the sampling times $\{t_k : k \in \mathbb{N}\}$ both $\hat{u}$ and $\hat{y}$ are held constant:

$$\hat{u}(t) = \hat{u}^+(t_k), \qquad \hat{y}(t) = \hat{y}^+(t_k), \qquad \forall t \in (t_k, t_{k+1}],\ k \in \mathbb{N}. \tag{2.18}$$

The signals $u(t)$ and $y(t)$ are not necessarily both sampled and sent to the network at every sampling time and therefore

$$\hat{u}^+(t_k) = \begin{cases} u(t_k) & \text{if } u \text{ sampled at time } t_k \\ \hat{u}(t_k) & \text{if } u \text{ not sampled at time } t_k \end{cases} \qquad \forall k \in \mathbb{N}. \tag{2.19}$$

$$\hat{y}^+(t_k) = \begin{cases} y(t_k) & \text{if } y \text{ sampled at time } t_k \\ \hat{y}(t_k) & \text{if } y \text{ not sampled at time } t_k \end{cases} \qquad \forall k \in \mathbb{N}. \tag{2.20}$$

The sampling model (2.19) can be written compactly as

$$\hat{u}^+(t_k) = u(t_k) + h_u\big(k, e(t_k)\big), \quad \hat{y}^+(t_k) = y(t_k) + h_y\big(k, e(t_k)\big), \quad \forall k \in \mathbb{N}, \tag{2.21}$$

where

$$e = \begin{bmatrix} e_y \\ e_u \end{bmatrix} := \begin{bmatrix} \hat{y} - y \\ \hat{u} - u \end{bmatrix} \in \mathbb{R}^{n_e},$$

and

$$h_u(k, e) := \begin{cases} 0 & \text{if } u \text{ sampled at time } t_k \\ e_u & \text{if } u \text{ not sampled at time } t_k \end{cases} \qquad \forall k \in \mathbb{N},\ e \in \mathbb{R}^{n_e}$$

$$h_y(k, e) := \begin{cases} 0 & \text{if } y \text{ sampled at time } t_k \\ e_y & \text{if } y \text{ not sampled at time } t_k \end{cases} \qquad \forall k \in \mathbb{N},\ e \in \mathbb{R}^{n_e}.$$

These definitions allow us to write the following "reset-map" for the error e at the sampling times:

$$e^+(t_k) = \begin{bmatrix} \hat{y}^+(t_k) - y(t_k) \\ \hat{u}^+(t_k) - u(t_k) \end{bmatrix} = \begin{bmatrix} h_u(k, e) \\ h_y(k, e) \end{bmatrix}, \tag{2.22}$$

where we used (2.21) and the fact that both y and u are continuous functions of time. The fact that the neither the state of the process nor the state of the controller appear in (2.22) will be very convenient for the analysis. It is also interesting to observe that the error reset map in (2.22) does not depend on the process/controller dynamics, but only on the "protocol" used to decide which variables should be sampled at each sampling time.

Walsh et al. [68], Nesic and Teel [45] actually consider a sampling model more general than (2.19), as they allow for only some entries of u and y to be transmitted through the network at each sampling time. In practice, this means that only some entries of $h_u(\cdot)$ and $h_y(\cdot)$ may be equal to zero at each sampling time. To capture this, Nesic and Teel [45] generalize (2.22) to

$$e^+(t_k) = h\big(k, e(t_k)\big), \qquad \forall t \in (t_k, t_{k+1}],\ k \in \mathbb{N}, \tag{2.23}$$

where $h(k, \cdot)$ specifies which entries of the error are reset to zero at the kth sampling time. This function can be regarded as implementing a *network access protocol* that decides which input/output signals should be sampled at each time t_k, $k \in \mathbb{N}$. When this decision is based on the current mismatches between u and $\hat{u}$ and/or between y and $\hat{y}$, we have a *dynamic protocol*, such as the Try-Once-Discard protocol in [68, 45]. Otherwise, we have a *static protocol*, such as the Round-robin protocol in [76, 68, 45]. Since the current mismatches may not always be available, Tabbara et al. [63] propose an alternative formulation in which the decision to sample a signal is based on an estimate of the mismatch for that signal.

Defining $x := \begin{bmatrix} x_P' & x_C' \end{bmatrix}'$, the NCS described by (2.17), (2.18), and (2.23) can be modeled by an impulsive system of the form

$$\dot{x} = f(x, e, w), \qquad \forall t \geq 0, \qquad x \in \mathbb{R}^{n_x},\ w \in \mathbb{R}^{n_w}, \tag{2.24a}$$

$$\dot{e} = g(x, e, w), \qquad \forall t \in (t_k, t_{k+1}], \qquad e \in \mathbb{R}^{n_e}, \tag{2.24b}$$

$$e^+(t_k) = h\big(k, e(t_k)\big), \qquad \forall k \in \mathbb{N}. \tag{2.24c}$$

This is a generalization of the NCS model in (2.15), in which signals were assumed continuous from above instead of continuous from below.

The following result is adapted from [45] and can be used to establish the stability of (2.24):

Theorem 4. *Suppose that the following conditions hold:*

1. *There exists a function* $W : \mathbb{N} \times \mathbb{R}^{n_e} \to [0, \infty)$ *and constants* $\rho \in [0, 1)$, $a_1, a_2 > 0$ *such that*

   $$a_1\|e\| \leq W(k, e) \leq a_2\|e\|, \qquad W(k+1, h(k, e)) \leq \rho\, W(k, e),$$

   for $\forall k \in \mathbb{N},\ e \in \mathbb{R}^{n_e}$.

2. *There exists a function* $H : \mathbb{R}^{n_x} \to [0, \infty)$ *and a constant* L *such that*

   $$\frac{\partial W(k, e)}{\partial e} \cdot g(x, e, w) \leq L\, W(k, e) + H(x) + \|w\|,$$

   for $\forall k \in \mathbb{N},\ x \in \mathbb{R}^{n_x},\ e \in \mathbb{R}^{n_e},\ w \in \mathbb{R}^{n_w}$.

3. *There exists a class* $\mathcal{KL}$ *function* β_1 *and a positive constant* $\gamma_1 > 0$ *such that*

   $$H\big(x(t)\big) \leq \beta_1\big(\|x(t_0)\|, t - t_0\big) + \gamma_1 \operatorname*{ess\,sup}_{\tau \in (t_0, t)} \Big(\|e(\tau)\| + \|w(\tau)\|\Big),$$

   for $\forall t \geq t_0 \geq 0$ *along solutions to* (2.24a).

4. *There exists a class $\mathcal{KL}$ function β_2 and a class $\mathcal{K}$ function γ_2 such that*

$$\|x(t)\| \leq \beta_2\big(\|x(t_0)\|, t - t_0\big) + \operatorname*{ess\,sup}_{\tau \in (t_0, t)} \gamma_2\Big(H\big(x(\tau)\big) + \|e(\tau)\| + \|w(\tau)\|\Big),$$

for $\forall t \geq t_0 \geq 0$ along solutions of (2.24a)*.*

5. *There exists a positive constant ξ (called the* maximum allowable transfer interval*, MATI) such that*

$$0 < t_{k+1} - t_k \leq \xi < \frac{1}{L} \ln\Big(\frac{L + \gamma_1}{\rho L + \gamma_1}\Big), \qquad \forall k \in \mathbb{N}.$$

Then the NCS modeled by (2.24) *is input-to-state stable from the disturbance input w to its state (x, e).* □

Condition 1 should be viewed as a requirement on the network access protocol specified by the function $h(\cdot)$. In practice, this condition requires the protocol to define an exponentially stable auxiliary discrete-time system

$$z_{k+1} = h(k, z_k)$$

with a decay rate of $\rho < 1$. In view of this, Nesic and Teel [45] introduce the terminology "uniformly exponentially stable protocol" to denote any protocol that satisfies condition 1.

For linear systems, the remaining assumptions of Theorem 4 are fairly mild. They basically require a growth for the error dynamics no faster than exponential and appropriate disturbance rejection properties of the "closed-loop" system (2.24), with respect to the inputs e and w. However, for nonlinear systems these assumptions may be difficult to verify. In either case, Nesic and Teel [45, 46, 47] show that the MATI condition 5 in Theorem 4 is less conservative than the ones in [67, 68, 66].

2.1.2 Packet dropouts

Packet dropouts can be modeled either as stochastic or deterministic phenomena. The simplest stochastic model assumes that dropouts are realizations of a Bernoulli process [60, 64]. Finite-state Markov chains can be used to model correlated dropouts [61] and Poisson processes can be used to model stochastic dropouts in continuous time [69]. Deterministic models for dropouts have also been proposed, either specified in terms of time-averages [76] or in terms of worst-case bounds on the number of consecutive dropouts [73, 42]. We defer the study of worst-case dropout models to Section 2.1.3.

Consider again the one-channel feedback NCS in Fig. 2.1, with a plant/controller pair (2.1), for which the signal y is sampled at times $\{t_k : k \in \mathbb{N}\}$ and the samples $y_k := y(t_k)$ are sent through the network. When packets are dropped, the network model in (2.2) must be changed. It is often assumed that when the packet containing the sample y_k is dropped, the NCS utilizes the previous value of $\hat{y}_k$ [76, 57]. This corresponds to replacing the lossless network model (2.2) by

$$\hat{y}_k = \theta_k y_k + (1-\theta_k)\hat{y}_{k-1} = \begin{cases} y_k & \theta_k = 1 \text{ (no packet dropout)} \\ \hat{y}_{k-1} & \theta_k = 0 \text{ (packet dropout)} \end{cases} \qquad \forall k \in \mathbb{N}, \tag{2.25}$$

where $\theta_k = 0$ when there is a packet dropout at time k and $\theta_k = 1$ otherwise. Hadjicostis and Touri [17] assume instead that $\hat{y}_k$ is set to zero when the packet containing y_k is dropped, i.e., $\hat{y}_k = \theta_k y_k$, $\forall k \in \mathbb{N}$.

Assuming that the delay[2] τ_k experienced by the kth packet is smaller than the

[2]When the kth packet is dropped the value of τ_k is of no consequence and can be assumed zero.

corresponding sampling interval, the continuous-time signal $\hat{y}(t)$ is still updated according to (2.3). For simplicity, we assume periodic sampling and constant network delay, i.e., $t_{k+1} - t_k = h$, $\tau_k = \tau$, $\forall k \in \mathbb{N}$. To analyze this NCS, once again we define an augmented discrete-time state $z_k' := \begin{bmatrix} x(t_k)' & \hat{y}_{k-1}' \end{bmatrix}$. From (2.4) and (2.25) we now conclude that

$$z_{k+1} = \Phi_{\theta_k} z_k, \tag{2.26}$$

where

$$\Phi_\theta := \begin{bmatrix} e^{Ah} + \theta\Gamma(h-\tau)BC & e^{A(h-\tau)}\Gamma(\tau)B + (1-\theta)\Gamma(h-\tau)B \\ \theta C & (1-\theta)I \end{bmatrix}, \ \forall \theta \in \{0, 1\}. \tag{2.27}$$

Deterministic dropouts Zhang et al. [76] consider a deterministic dropout model, with packet dropouts occurring at an asymptotic rate defined by the following time-average

$$r := \lim_{T\to\infty} \frac{1}{T} \sum_{k=k_0}^{k_0+T-1} (1-\theta_k), \quad \forall k_0 \in \mathbb{N}, \tag{2.28}$$

which implicitly assumes that the limit exists. Under this dropout model, the system (2.26) falls under the class of *asynchronous dynamical systems* (ADSs). These are hybrid systems whose continuous dynamics are governed by differential or difference equations and the discrete dynamics are governed by finite automata. In ADSs, the finite automata are driven asynchronously by external events that occur at pre-specified rates. The ADSs of interest to us are defined by a difference equation such as (2.26), where θ_k takes values in some index set $\{0, 1, \dots, N\}$ and the rate at which the event $\theta_k = j$ occurs is defined by the following time average

$$r_j := \lim_{T\to\infty} \frac{1}{T} \sum_{k=k_0}^{k_0+T-1} \delta_{j\,\theta_k}, \qquad \forall k_0 \in \mathbb{N},\ j \in \{0, 1, \dots, N\}, \tag{2.29}$$

where $\delta_{j\,\theta_k} = 1$ when $\theta_k = j$ and zero otherwise. When all the limits exist, we have $\sum_{j=0}^{N} r_j = 1$. The following result is adapted from Hassibi et al. [18] and uses a quadratic Lyapunov function of the form $V(z) := z'Pz$ to establish the asymptotic stability of the ADS system (2.26) with rates (2.29).

Theorem 5 ([18]). *Assume that there exist a symmetric matrix $P > 0$ and scalars $\alpha_0, \ldots, \alpha_N$ and α such that*

$$\alpha_0^{r_0}\alpha_1^{r_1}\cdots\alpha_N^{r_N} > \alpha > 1, \qquad \Phi_j' P \Phi_j \leq \alpha_j^{-2} P, \quad \forall j \in \{0, 1, \ldots, N\},$$

then the ADS (2.26) *is exponentially stable in the sense that* $\lim_{k\to\infty} \alpha^k \|z_k\| = 0$ *for every sequence $\delta_{j\,\theta_k}$ for which* (2.29) *holds.* □

The following result is obtained by applying Theorem 5 to our NCS with constant sampling interval h and constant delay τ:

Corollary 1. *Assuming that there exist a symmetric matrix $P > 0$ and scalars $\alpha, \alpha_0, \alpha_1$ such that*

$$\alpha_0^r\,\alpha_1^{1-r} > \alpha > 1, \quad \Phi_0(h,\tau)' P \Phi_0(h,\tau) \leq \alpha_0^{-2} P, \quad \Phi_1(h,\tau)' P \Phi_1(h,\tau) \leq \alpha_1^{-2} P, \tag{2.30}$$

then the NCS (2.26) *is exponentially stable in the sense that* $\lim_{k\to\infty} \alpha^k \|z_k\| = 0$ *for every sequence θ_k for which* (2.28) *holds.* □

The main difficulty in applying this result is that the set of matrix inequalities that appears in (2.30) is bilinear in the unknowns P, α_j and therefore generally non-convex. However, one can use a "line-search" procedure over the two scalars α_0, α_1 to determine the feasibility of (2.30).

Remark 3. One can also express (2.25) as $\hat{y}_k = C\hat{x}_k$, $\forall k \in \mathbb{N}$, with $\hat{x}_k$ defined by

$$\hat{x}_k := \theta_k x_k + (1-\theta_k)\hat{x}_{k-1} = \begin{cases} x_k & \theta_k = 1 \text{ (no packet dropout)} \\ \hat{x}_{k-1} & \theta_k = 0 \text{ (packet dropout)} \end{cases} \qquad \forall k \in \mathbb{N}.$$

When $\tau = 0$ one can analyze the system's stability using the discrete-time state $\bar{z}'_k := \begin{bmatrix} x'_k & \hat{x}'_k \end{bmatrix}$, which evolves according to

$$\bar{z}_{k+1} = \bar{\Phi}_{\theta_{k+1}} \bar{z}_k, \qquad \bar{\Phi}_\theta := \begin{bmatrix} e^{Ah} & \Gamma(h)BC \\ \theta e^{Ah} & \theta\Gamma(h)BC + (1-\theta)I \end{bmatrix}, \qquad \forall k \in \mathbb{N}.$$

In their work, Zhang et al. [76] considered this discrete-time system instead of (2.26). $\square$

Stochastic dropouts Seiler and Sengupta [57, 58] consider stochastic losses. In their formulation, $\boldsymbol{\theta}_k$ is a Bernoulli process[3] with probability of dropout (i.e., $\boldsymbol{\theta}_k = 0$) equal to $p \in [0, 1)$. Under this dropout model, the system (2.26) is a special case of a discrete-time *Markovian jump linear system* (MJLS). In general MJLSs, the index $\boldsymbol{\theta}_k$ in (2.26) would be the state of a discrete-time Markov chain with a finite number of states and a given transition probability matrix. For Bernoulli drops, the Markov chain only has two states and the transition probability from any state to the *dropout-state* $\boldsymbol{\theta}_k = 0$ is equal to p and the transition probability from any state to the state $\boldsymbol{\theta}_k = 1$ is equal to $1 - p$, as shown in Fig 2.6. The stability of discrete-time MJLSs can be established using results from [5] (cf. [10] for continuous-time MJLs), leading to the following theorem:

[3] To emphasize the fact that now the $\boldsymbol{\theta}_k$, $k \in \mathbb{N}$ are random variables, we denote them in boldface.

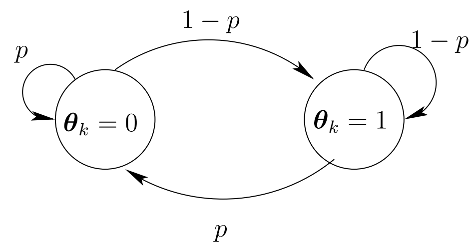


Figure 2.6. Bernoulli drops with probability p modeled as a two-state discrete-time Markov chain.

Theorem 6 ([57])**.** *The NCS* (2.26) *with dropout probability* p *(Bernoulli) is exponentially mean-square stable if there exists a symmetric matrix* $Z > 0$ *such that*

$$\begin{bmatrix} Z & \sqrt{p}(\Phi_0 Z)' & \sqrt{1-p}(\Phi_1 Z)' \\ * & Z & 0 \\ * & * & Z \end{bmatrix} > 0. \tag{2.31}$$

□

When the controller is collocated with the actuators, Seiler and Sengupta [57, 58] suggest that the control law can adapt to the occurrence of dropouts by allowing the controller to use different gains at different time instants k, based on the value of $\theta_k \in \{0, 1\}$. In this case, the matrices A, B, C in (2.27) depend on θ, but one still gets a system of the form (2.26) and one can use similar tools to analyze its stability. We will return to this issue in Section 2.2 when we discuss the design of NCS controllers.

2.1.3 NCSs with sampling, delays, and dropouts

Once again we go back to the one-channel feedback NCS in Fig. 2.1, with a plant/controller pair (2.1), for which the signal $y(t)$ is sampled at times $\{t_k : k \in \mathbb{N}\}$. In a lossless network, all the samples $\hat{y}_k = y_k = Cx(t_k)$ arrive at the destination with a (possibly variable) delay of $\tau_k \geq 0$, which leads to

$$\hat{y}(t) = Cx(t_k), \quad \forall t \in [t_k + \tau_k, t_{k+1} + \tau_{k+1}),\ k \in \mathbb{N}. \tag{2.32}$$

In the previous sections, we proceeded by deriving discrete-time models for the evolution of the state of the NCS at sampling times. Instead, Yu et al. [70] propose to regard (2.32) as a delayed equation

$$\hat{y}(t) = Cx\big(t - \tau(t)\big), \quad \forall t \geq 0, \tag{2.33}$$

in which the delay $\tau(t)$ is time-varying and evolves according to

$$\tau(t) = t - t_k, \qquad\qquad \forall t \in [t_k + \tau_k, t_{k+1} + \tau_{k+1}),\ k \in \mathbb{N}.$$

Fig. 2.7(a) shows the evolution of the time-varying delay $\tau(t)$ in (2.33) for the case of periodic sampling with period $h = t_{k+1} - t_k$, $\forall k \in \mathbb{N}$ and constant network delay $\tau_0 = \tau_k$, $\forall k \in \mathbb{N}$. In view of (2.33), this approach regards the overall NCS as a continuous-time delayed differential equation (DDE) of the form

$$\dot{x}(t) = Ax(t) + BCx\big(t - \tau(t)\big), \quad \forall t \geq 0, \tag{2.34}$$

where the time-varying delay $\tau(t)$ satisfies

$$\tau(t) \in [\tau_{\min}, \tau_{\max}), \quad \dot{\tau} = 1, \quad \forall t \geq 0,\ a.e. \tag{2.35}$$

where

$$\tau_{\min} := \min_{k\in\mathbb{N}}\{\tau_k\}, \qquad\qquad \tau_{\max} := \max_{k\in\mathbb{N}}\{t_{k+1} - t_k + \tau_{k+1}\}. \tag{2.36}$$

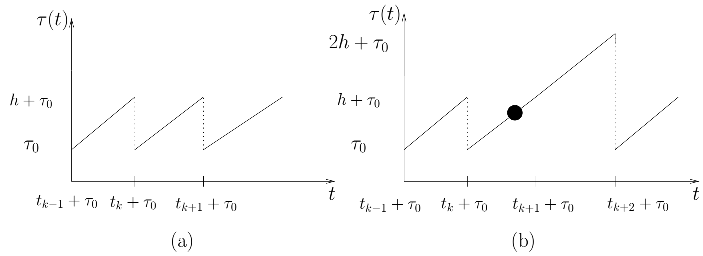


Figure 2.7. Variable delay in an NCS for constant network delay $\tau_k = \tau_0$, $\forall k \in \mathbb{N}$ and periodic sampling with period $t_{k+1} - t_k = h$, $\forall k \in \mathbb{N}$ for (a) no packet dropout and (b) one packet dropout shown by a black dot.

An important advantage of characterizing an NCS as in (2.34)–(2.36) is that these equations are valid even when the delay exceeds the sampling interval. So in this section we shall not restrict our attention to delays smaller than one sampling interval.

As illustrated in Fig. 2.7(b), we can also view packet dropouts as a delay $\tau(t)$ that grows beyond the maximum in (2.36). This means that an NCS with a maximum number of consecutive dropouts equal to m is still a DDE like (2.34)–(2.35), but with

$$\tau_{\min} := \min_{k\in\mathbb{N}}\{\tau_k\}, \qquad \tau_{\max} := \max_{k\in\mathbb{N}}\{t_{k+1+m} - t_k + \tau_{k+1+m}\}.$$

The Lyapunov-Krasovskii and the Razumikhin Theorems [16, 56, 13] are the two main tools available to study the stability of DDEs of the form (2.34)–(2.35). However, the Lyapunov-Krasovskii Theorem generally leads to less conservative results. To formulate this theorem we need following notation: Given a constant $\tau_{\max} > 0$, a continuous signal $x : (-\tau_{\max}, \infty) \to \mathbb{R}^n$, and some time $t \in \mathbb{R}$, we denote by $x_t : [-\tau_{\max}, 0] \to \mathbb{R}^n$ the restriction of x to the interval $[t - \tau_{\max}, t]$ translated to $[-\tau_{\max}, 0]$, i.e., $x_t(s) = x(t+s)$, $\forall s \in [-\tau_{\max}, 0]$. The function x_t is an element of the Banach space $\mathcal{C}([-\tau_{\max}, 0], \mathbb{R}^n)$ of continuous functions from

$[-\tau_{\max}, 0]$ to $\mathbb{R}^n$.

Theorem 7 (Lyapunov-Krasovskii [16])**.** *The DDE*

$$\dot{x}(t) = Ax(t) + A_d x\big(t - \tau(t)\big), \quad \forall t \geq 0, \tag{2.37}$$

with $\tau(t) \in [0, \tau_{\max}]$*,* $\forall t \geq 0$ *and initial condition* $x(t) = \psi(t)$*,* $\forall t \in [-\tau_{\max}, 0]$ *is asymptotically stable if there exists a bounded quadratic* Lyapunov-Krasovskii functional $V : \mathcal{C}([-\tau_{\max}, 0], \mathbb{R}^n) \to \mathbb{R}$ *and a positive constant* ϵ *for which*

$$V(\phi) \geq \epsilon \|\phi(0)\|^2, \qquad \frac{\mathrm{d}V(x_t)}{\mathrm{d}t}\Big|_{x_t = \phi} \leq -\epsilon \|\phi(0)\|^2, \qquad \forall \phi \in \mathcal{C}([-\tau_{\max}, 0], \mathbb{R}^n).$$

where the (total) derivative is taken along solutions to (2.37)*.* □

To study the stability of (2.34), Yue et al. [73] use the following Lyapunov-Krasovskii functional

$$V(x_t) = x(t)' P x(t) + \int_{t-\tau_{\max}}^{t} \int_{s}^{t} \dot{x}(v)' T \dot{x}(v) dv ds,$$

with symmetric matrices $P > 0$ and $T > 0$. They show that the derivative of $V(x_t)$ is negative along solutions to (2.34) if

$$\begin{bmatrix} N_1 + N_1' - M_1 A - A' M_1' & N_2' - N_1 - A' M_2' - M_1 BC & N_3' - A' M_3' + M_1 + P & \tau_{\max} N_1 \\ * & -N_2 - N_2' - M_2 BC - C' B' M_2' & -N_3' + M_2 - C' B' M_3' & \tau_{\max} N_2 \\ * & * & M_3 + M_3' + \tau_{\max} T & \tau_{\max} N_3 \\ * & * & * & -\tau_{\max} T \end{bmatrix} < 0, \tag{2.38}$$

where N_i, M_i, $i \in \{1, 2, 3\}$ are slack matrix variables. This leads to the following result:

Theorem 8 ([73])**.** *For a given scalar* $\tau_{\max} > 0$*, suppose that there exist square matrices* N_i, M_i*,* $i \in \{1, 2, 3\}$ *and symmetric matrices* $P, T > 0$ *such that* (2.38) *holds. Then the NCS* (2.34)–(2.35) *in Fig 2.1 is asymptotically stable as long as*

$$t_{k+1+m} - t_k + \tau_{k+1+m} \leq \tau_{\max}, \quad \forall k \in \mathbb{N},$$

where m *denotes the maximum number of consecutive dropouts.* □

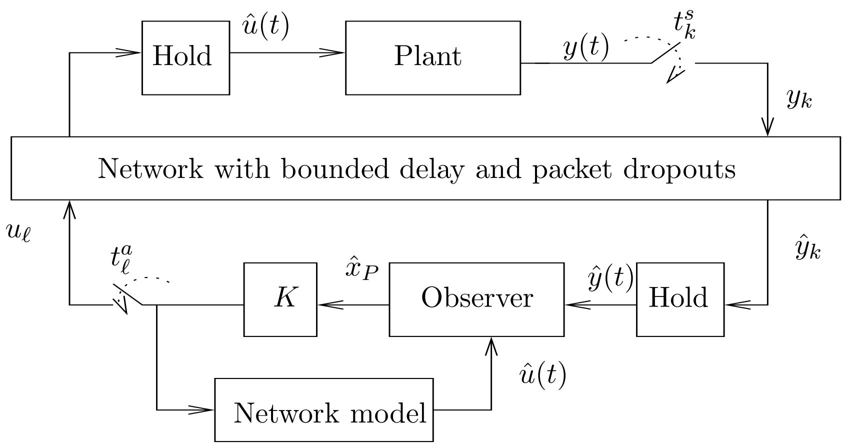


Figure 2.8. Two channel feedback NCS with observer-based controller.

Yu et al. [70] also model one-channel feedback NCSs as DDEs, but they study their stability using the Razumikhin Theorem, which generally leads to more conservative results.

Naghshtabrizi and Hespanha [42] consider the two-channel feedback NCS in Fig. 2.8, in which a known plant

$$\dot{x}_P = A_P x_P + B_P \hat{u}, \qquad\qquad y = C_P x_P \tag{2.39}$$

is controlled by a remote observer-based controller that receives sensor data through a *sensor channel* and sends control signals to the actuators through an *actuation channel*. The output signal $y(t)$ is sampled at times t_k^s, $k \in \mathbb{N}$ and the samples $y_k := y(t_k^s)$, $k \in \mathbb{N}$ are sent through the sensor channel suffering a (possibly variable) delay of $\tau_k^s \geq 0$. In a lossless network, $\hat{y}(t)$ is therefore updated according to

$$\hat{y}(t) = y_k = C_P x_P(t_k^s), \quad \forall t \in [t_k^s + \tau_k^s, t_{k+1}^s + \tau_{k+1}^s),\ k \in \mathbb{N},$$

and if $m^s > 0$ sensor channel packets are dropped after the kth packet, the above equation holds $\forall t \in [t_k^s + \tau_k^s, t_{k+1+m^s}^s + \tau_{k+1+m^s}^s)$.

Two alternative observer-based controllers are proposed: non-anticipative and anticipative. *Non-anticipative controllers* construct an estimate $\hat{x}_P$ of the plant state x_P using

$$\dot{\hat{x}}_P(t) = A_P\hat{x}_P(t) + B_P\hat{u}(t) + L\big(\hat{y}(t) - C_P\hat{x}_P(t_k^s)\big),\ \forall t \in [t_k^s + \tau_k^s, t_{k+1}^s + \tau_{k+1}^s), \tag{2.40}$$

for which the innovation term $\hat{y} - C_P\hat{x}_P(t_k^s)$ was chosen so that it is equal to zero whenever $x_P(t_k^s) = \hat{x}_P(t_k^s)$. Control updates $u_\ell := K\hat{x}_P(t_\ell^a)$ are sent through the actuation channel to the actuators at times t_ℓ^a, $\ell \in \mathbb{N}$ suffering a (possibly variable) delay of $\tau_\ell^a \geq 0$. In a lossless network, the control signal $\hat{u}(t)$ is therefore updated according to

$$\hat{u}(t) = u_\ell = K\hat{x}_P(t_\ell^a), \quad \forall t \in [t_\ell^a + \tau_\ell^a, t_{\ell+1}^a + \tau_{\ell+1}^a),\ \ell \in \mathbb{N},$$

which would hold over a longer interval if actuation channels packets were dropped.

Anticipative controllers attempt to compensate the sampling and delay introduced by the actuation channel. For simplicity, we assume that the actuation channel is sampled with period $h^a = t_{\ell+1}^a - t_\ell^a$, $\forall \ell \in \mathbb{N}$ and that its delay is constant and equal to $\tau^a = \tau_\ell^a$, $\forall \ell \in \mathbb{N}$. At each sampling time $t_\ell^a = \ell h^a$, $\ell \in \mathbb{N}$ the controller sends a time-varying control signal $u_\ell(\cdot)$ that should be used from the time $\ell h^a + \tau^a$ at which it arrives until the time $(\ell+1)h^a + \tau^a$ at which the next control update will arrive. This leads to

$$\hat{u}(t) = u_\ell(t), \quad \forall t \in [\ell h^a + \tau^a, (\ell+1)h^a + \tau^a),\ \ell \in \mathbb{N}.$$

To stabilize (2.39), $u_\ell(t)$ should be equal to $-K\hat{x}_P(t)$, where $\hat{x}_P(t)$ is an estimate of $x_P(t)$. However, the estimates $\hat{x}_P(\cdot)$ needed in the interval $[\ell h^a + \tau^a, (\ell+1)h^a + \tau^a)$ must be available at the transmission time ℓh^a, which requires the control unit to estimate the plant state up to $h^a + \tau^a$ time units into the future. In this case, the estimator (2.40) is of no use. Instead, an estimate $z(t)$ of $x_P(t + h^a + \tau^a)$ is constructed as follows:

$$\dot{z}(t) = A_P z(t) + B_P \hat{u}(t + h^a + \tau^a) + L\big(\hat{y}(t) - C_P z(t_k^s - h^a - \tau^a)\big),$$

for $\forall t \in [t_k^s + \tau_k^s, t_{k+1}^s + \tau_{k+1}^s),\ k \in \mathbb{N}$, for which the innovation term $\hat{y} - C_P z(t_k^s - h^a - \tau^a)$ was chosen so that it is equal to zero whenever $x_P(t_k^s) = z\big(t_k^s - h^a - \tau^a\big)$. The signal $u_\ell(\cdot)$ sent at time ℓh^a and to be used during the interval $[\ell h^a + \tau^a, (\ell + 1)h^a + \tau^a)$, is given by

$$u_\ell(t) = -Kz(t - h^a - \tau^a), \quad \forall t \in [\ell h^a + \tau^a, (\ell + 1)h^a + \tau^a),\ \ell \in \mathbb{N},$$

which only requires knowledge of $z(\cdot)$ in $[(\ell - 1)h^a, \ell h^a)$ and is therefore available at the transmission time ℓh^a. For anticipative controllers to be able to compensate for packet dropouts in the actuation channel, z would have to estimate x_P further into the future. Anticipative controllers send through the actuation channel actuation signals to be used during time intervals of duration h^a, therefore for these controllers the sample and hold blocks in Fig. 2.8 should be understood in a broad sense. In practice, the sample block would send over the network some parametric form of the control signal $u_\ell(\cdot)$ (e.g., the coefficients of a polynomial approximation to this signal).

Naghshtabrizi and Hespanha [42] write the closed-loop NCSs as DDEs for both the anticipative and the non-anticipative controllers. For an anticipative controller with no dropouts, this leads to

$$\dot{x}(t) = \begin{bmatrix} A_P - B_P K & 0 \\ 0 & A_P \end{bmatrix} x(t) + \begin{bmatrix} 0 & LC_P \\ 0 & -LC_P \end{bmatrix} x(t - \tau), \quad \forall t \geq 0, \qquad (2.41)$$

where $x'(t) := \begin{bmatrix} z(t) & x_P(t + h^a + \tau^a) - z(t) \end{bmatrix}$, $\forall t \geq 0$ and

$$\tau(t) := t - t_k^s + h^a + \tau^a, \quad \forall t \in [t_k^s + \tau_k^s, t_{k+1+m}^s + \tau_{k+1+m}^s),\ k \in \mathbb{N}.$$

Moreover, if $m^s > 0$ sensor channel packets are dropped after the kth packet, this equation holds over the interval $[t_k^s + \tau_k^s, t_{k+1+m^s}^s + \tau_{k+1+m^s}^s)$.

The "triangular" structure of (2.41) is unique to the anticipate controller. With this type of controller, if we choose K so that $A_P - B_P K$ is Hurwitz, asymptotic stability of the NCS is equivalent to the asymptotic stability of the (decoupled) dynamics of the error $e(t) := x_P(t + h^a + \tau^a) - z(t)$, $\forall t \geq 0$, which is given by the following DDE

$$\dot{e}(t) = A_P\, e(t) - LC_P\, e\big(t - \tau(t)\big), \quad t \geq 0,$$

with $\tau(t) \in [\tau_{\min}, \tau_{\max})$, $\dot{\tau} = 1$, $\forall t \geq 0$, a.e., where

$$\tau_{\min} := \min_{k \in \mathbb{N}} \{\tau_k^s + h^a + \tau^a\}, \quad \tau_{\max} := \max_{k \in \mathbb{N}} \{t_{k+1+m^s}^s - t_k^s + \tau_{k+1+m^s}^s + h^a + \tau^a\},$$

where m^s denotes the maximum number of consecutive packet dropouts in the sensor channel. Naghshtabrizi and Hespanha [42] use the following Lyapunov-Krasovskii functional to analyze this system

$$V(e_t) = e(t)' P_1 e(t) + \int_{t-\tau_{\max}}^{t} \int_{s}^{t} \dot{e}'(v) R \dot{e}(v) dv ds + \int_{t-\tau_{\min}}^{t} e'(s) S e(s) ds,$$

where $P_1 > 0$, $R > 0$, $S > 0$. This leads to the following result:

Theorem 9 ([42])**.** *Suppose that there exist symmetric matrices $P_1, S, R > 0$,* *square matrices P_2, P_3, Z_1, Z_2, and a (non-square) matrix T such that*

$$\begin{bmatrix} \Psi & P' \begin{bmatrix} 0 \\ -LC_P \end{bmatrix} - T' \\ * & -S \end{bmatrix} < 0, \quad \begin{bmatrix} R & \begin{bmatrix} 0 & -(LC_P)' \end{bmatrix} P \\ * & Z_2 \end{bmatrix} > 0, \quad \begin{bmatrix} R & T \\ * & Z_1 \end{bmatrix} > 0, \tag{2.42}$$

where

$$P := \begin{bmatrix} P_1 & 0 \\ P_2 & P_3 \end{bmatrix}, \quad \Psi := P' \begin{bmatrix} 0 & I \\ A_P & -I \end{bmatrix} + \begin{bmatrix} 0 & I \\ A_P & -I \end{bmatrix}' P + \begin{bmatrix} S & 0 \\ 0 & \tau_{\max} R_i \end{bmatrix}$$
$$+\tau_{\min} Z_1 + (\tau_{\max} - \tau_{\min}) Z_2 + \begin{bmatrix} T \\ 0 \end{bmatrix} + \begin{bmatrix} T \\ 0 \end{bmatrix}'. \tag{2.43}$$

Then the NCS with the anticipative controller (2.41), (2.35) *is asymptotically stable as long as there are no dropouts in the actuation channel and*

$$\tau_k^s + h^a + \tau^a \geq \tau_{\min}, \quad t^s_{k+1+m^s} - t^s_k + \tau^s_{k+1+m^s} + h^a + \tau^a \leq \tau_{\max}, \quad \forall k \in \mathbb{N},$$

where m^s denotes the maximum number of consecutive dropouts in the sensor channel. □

The reader is referred to [42] for an analogous result with a non-anticipative controller.

2.2 Controller Synthesis

In this section, we discuss the design of feedback controllers for NCSs. Some of these results stem directly from the analysis methods presented in Section 2.1.

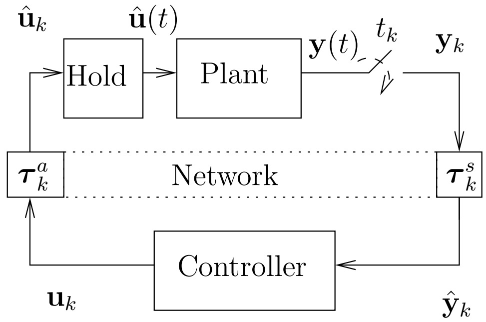


Figure 2.9. Two-channel feedback NCS considered by Nilsson [48].

2.2.1 Sampling and delay

Nilsson et al. [49, 51, 50, 53, 52, 48] consider the two-channel feedback NCS in Fig. 2.9. The plant is an LTI system with the following state-space model

$$\dot{\mathbf{x}} = A\mathbf{x} + B\hat{\mathbf{u}} + \mathbf{w}, \qquad \mathbf{y} = C\mathbf{x} + \mathbf{v}, \tag{2.44}$$

where $\mathbf{v}$ and $\mathbf{w}$ are zero-mean uncorrelated white noise processes. The output signal $\mathbf{y}(t)$ is sampled periodically at times $t_k := kh$, $\forall k \in \mathbb{N}$ and the samples $\mathbf{y}_k := \mathbf{y}(t_k), \forall k \in \mathbb{N}$ are sent through the network. After a (possibly varying) delay of $\boldsymbol{\tau}_k^s \geq 0$, these samples reach a remote controller that immediately computes control updates $\mathbf{u}_k$ and sends them to the network. These updates reach the actuators after a (possibly varying) delay of $\boldsymbol{\tau}_k^a \geq 0$. Assuming that there are no packet dropouts in either of the network channels, this leads to

$$\hat{\mathbf{u}}(t) = \begin{cases} \mathbf{u}_{k-1} & t \in [t_k, t_k + \boldsymbol{\tau}_k^s + \boldsymbol{\tau}_k^a) \\ \mathbf{u}_k & t \in [t_k + \boldsymbol{\tau}_k^s + \boldsymbol{\tau}_k^a, t_{k+1}) \end{cases}, \tag{2.45}$$

where we assumed that the total delay is smaller than one sampling interval, i.e., that $\boldsymbol{\tau}_k^s + \boldsymbol{\tau}_k^a < h$, $\forall k \in \mathbb{N}$. Any computation time needed by the remote controller can be incorporated in the network delay $\boldsymbol{\tau}_k^a$.

The delays $\boldsymbol{\tau}_k^s$ and $\boldsymbol{\tau}_k^a$ are assumed to be independent random variables with known probability distributions. Moreover, it is assumed that all data sent through the network is time-stamped, which means that the controller knows the value of the delay $\boldsymbol{\tau}_k^s$ when the kth measurement $\hat{\mathbf{y}}_k$ arrives.

Defining $\mathbf{x}_k := \mathbf{x}(t_k)$, $\forall k \in \mathbb{N}$ and applying the variation of constants formula to (2.44)–(2.45), we conclude that

$$\begin{aligned}
\mathbf{x}_{k+1} &= e^{Ah}\mathbf{x}_k + e^{A(h-\boldsymbol{\tau}_k^s-\boldsymbol{\tau}_k^a)}\Gamma(\boldsymbol{\tau}_k^s+\boldsymbol{\tau}_k^a)B\mathbf{u}_{k-1} + \Gamma(h-\boldsymbol{\tau}_k^s-\boldsymbol{\tau}_k^a)B\mathbf{u}_k + \boldsymbol{v}_k,\\
\mathbf{y}_k &= C\mathbf{x}_k + \mathbf{w}_k,
\end{aligned}$$

where $\Gamma(s) := \int_0^s e^{Az}dz$, $\forall s \in \mathbb{R}$ and $\mathbf{v}_k$, $\mathbf{w}_k$ are uncorrelated zero-mean white noise processes.

Assuming state-feedback (i.e., $\mathbf{y}_k = \mathbf{x}_k$, $\forall k \in \mathbb{N}$), Nilsson et al. [53, 51] show that the optimal control $\mathbf{u}_k$ that minimizes

$$J_k = \mathrm{E}\left\{\mathbf{x}_N' Q_N \mathbf{x}_N + \sum_{j=k}^{N-1} \begin{bmatrix} \mathbf{x}_j \\ \mathbf{u}_j \end{bmatrix}' Q \begin{bmatrix} \mathbf{x}_j \\ \mathbf{u}_j \end{bmatrix}\right\},$$

with

$$Q_N \geq 0, \qquad Q := \begin{bmatrix} Q_{11} & Q_{12} \\ * & Q_{22} \end{bmatrix} \geq 0, \quad Q_{22} > 0,$$

is of the form

$$\mathbf{u}_k = -L_k(\boldsymbol{\tau}_k^s)\begin{bmatrix} \mathbf{x}_k \\ \mathbf{u}_{k-1} \end{bmatrix}, \quad \forall k \in \mathbb{N}. \tag{2.46}$$

Hence the optimal controller with full state information is a linear but $\boldsymbol{\tau}_k^s$-dependent function of the current state and previous control signal. The computation of the matrix gain $L_k(\boldsymbol{\tau}_k^s)$ requires the solution of a backwards-in-time Riccati equation that involves the computation of expectations with respect to the random variables $\boldsymbol{\tau}_k^s$ and $\boldsymbol{\tau}_k^a$.

In practice, the delays $\boldsymbol{\tau}_k^s, \boldsymbol{\tau}_k^a$, $\forall k \in \mathbb{N}$ are often correlated because they depend on the network load, which typically varies at time scales slower than the sampling interval h. To account for this, Nilsson and Bernhardsson [50] consider three alternative distributions for the delay and model the transitions between the distributions using a three-state Markov chain. Each state of the Markov chain would correspond to a particular network load (low, medium, or high). In this case the optimal control strategy is of the form

$$\mathbf{u}_k = -L_k(\boldsymbol{\tau}_k^s, \mathbf{r}_k) \begin{bmatrix} \mathbf{x}_k \\ \mathbf{u}_{k-1} \end{bmatrix}, \tag{2.47}$$

where now the matrix gain $L_k(\boldsymbol{\tau}_k^s, \mathbf{r}_k)$ depends both on the delay $\boldsymbol{\tau}_k^s$ and the current state $\mathbf{r}_k$ of the Markov chain. To implement this control law, the remote controller must know the current value of $\mathbf{r}_k$.

The main difficulty in using the optimal controllers (2.46), (2.47) is the computation of the matrix gains $L_k(\cdot)$. However, when stationary values for these gains exist, they can be computed offline and stored in a table, which is indexed in real time by the current value of the delay $\boldsymbol{\tau}_k^s$ and network state $\mathbf{r}_k$. Nilsson et al. [53, 51] also propose to use suboptimal controllers that are more attractive from a computational perspective. The same authors [53, 51, 50] extended this work for the output feedback case. They showed that the separation principle holds and that the optimal control can be obtained by replacing $\mathbf{x}_k$ in (2.46) and (2.47) by an estimate $\hat{\mathbf{x}}$ computed using a time-varying Kalman filter.

Nilsson et al. [52] further extended this work by considering non-periodic sam-

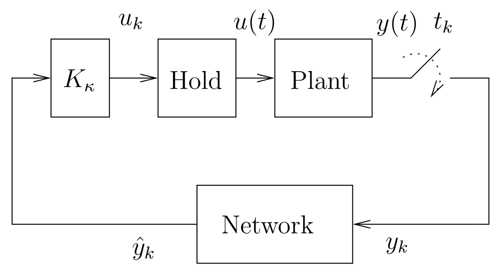


Figure 2.10. One-channel feedback NCS with switching controller.

pling, timeouts on the period during which a controller waits for new measurements before sending a new control command, and asynchronous loops in which the clocks used for time-stamping are not synchronized and run at different speeds. Lincoln and Bernhardsson [32] also extended these results to situations in which the sum of the delays exceeds one sampling interval but remains bounded.

2.2.2 Packet dropouts

Deterministic dropout rates Yu et al. [71] consider the one-channel feedback NCS shown in Fig. 2.10 with a LTI plant

$$\dot{x} = Ax + Bu, \qquad\qquad y = Cx,$$

whose output is sampled periodically at times $\{t_k := kh : k \in \mathbb{N}\}$ and the samples $y_k := y(t_k)$ are sent through the network. It is assumed that the delay introduced by the network is negligible but packets may be dropped. The network output $\hat{y}_k$ is kept equal to its previous value when the packet containing the sample y_k is dropped as in (2.25). Denoting by κ_j, $j \in \mathbb{N}$ the indexes of the packets that are *not dropped*, $\hat{y}_k$ remains equal to y_k from $k = \kappa_j$ until $k = \kappa_{j+1} - 1$, i.e.,

$$\hat{y}_k = y_{\kappa_j}, \quad \forall k \in \{\kappa_j, \kappa_j + 1, \dots, \kappa_{j+1} - 1\}.$$

Yu et al. [71] use a static output-feedback controller, whose gain changes depending on whether or not a packet is dropped. More precisely, they use

$$u_k = K_{k-\kappa_j}\hat{y}_k = K_{k-\kappa_j} y_{\kappa_j}, \quad \forall k \in \{\kappa_j, \kappa_j+1, \ldots, \kappa_{j+1}-1\},$$

where the matrix gain K_0 is used when the sample y_k has not been dropped, K_1 is used when y_k has been dropped but y_{k-1} has not, K_2 is used when y_k and y_{k-1} have been dropped but y_{k-2} has not, and so on. The control signal $u(t)$ is kept constant between samples:

$$u(t) = u_k, \quad t \in [t_k, t_{k+1}), \quad \forall k \in \mathbb{N}.$$

Defining $z_j := x(t_{\kappa_j})$, $\forall j \in \mathbb{N}$ and using the variation of constants formula, we conclude that

$$z_{j+1} = A_{\kappa_{j+1}-\kappa_j-1} z_j, \quad A_d := e^{Ah(d+1)} + \sum_{i=0}^{d} e^{Ah(d-i)}\Gamma(h)BK_iC, \quad \forall d \in \mathbb{N}, \quad (2.48)$$

where $\Gamma(s) := \int_0^s e^{Az}dz$, $\forall s \in \mathbb{R}$. Assuming that the maximum number of consecutive dropouts is equal to m, we have $0 \le \kappa_{j+1} - \kappa_j - 1 \le m$, $\forall j \in \mathbb{N}$ and we can view (2.48) as a linear system that switches among the matrices $A_0, \ldots, A_m$. The stability of such system can be established using a common quadratic Lyapunov function $V(z) := z'S^{-1}z$, leading to the following theorem:

Theorem 10 ([71])**.** *Suppose that there exists matrices M, Y_i, $i \in \{0, 1, \ldots, m\}$ and a symmetric matrix $S > 0$ such that*[4] $MC = CS$ *and*

$$\begin{bmatrix} -S & * \\ e^{Ah(d+1)}S + \sum_{i=0}^{d} e^{Ah(d-i)}\Gamma(h)BY_iC & -S \end{bmatrix} < 0, \quad \forall d \in \{0, 1, \ldots, m\}.$$

Then the NCS (2.48) *in Fig. 2.10 is exponentially stable for the controller gain* $K_i = Y_iM^{-1}$, $\forall i \in \{0, 1, \ldots, m\}$. □

[4] *In the state feedback case, C is the identity matrix and we simply have $M = S$.*

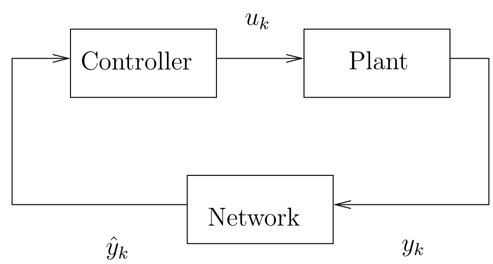


Figure 2.11. One-channel feedback NCS with a discrete-time plant and controller.

2.2.3 NCSs with sampling, delays, and dropouts

In Section 2.1.3 we presented matrix inequalities from which one could conclude stability for NCSs modeled as delay differential equations. For given controllers gains C and L, the inequalities (2.38) and (4.19) in Theorems 8 and 9, respectively, are linear on all the matrices that do not depend on the (known) plant model. The stability of the NCS can therefore be verified by studying the feasibility of a (convex) LMI. However, if the controllers gains are also taken as unknowns, we obtain a BMI and therefore these matrix inequalities are not directly suitable for controller synthesis.

Yue et al. [73] utilize Theorems 8 as the basis for a numerically tractable controller synthesis procedure. They require the matrices M_1, M_2, M_3 in (2.38) to further satisfy

$$M_1 = M_1' > 0, \qquad M_2 = \rho_2 M_1, \qquad M_3 = \rho_3 M_1, \tag{2.49}$$

for some constants $\rho_2, \rho_3 \in \mathbb{R}$. They then make the (bijective) change of variables

$$X := M^{-1}, \qquad Y := CX', \qquad \tilde{P} := XPX',$$
$$\tilde{T} := XTX', \qquad \tilde{N}_i := XN_iX', \qquad \tilde{M}_i := XM_iX',$$

for $\forall i \in \{1,2,3\}$. Pre- and post-multiplying (2.38) by $\mathrm{diag}(X\,X\,X\,X)$ and its transpose, respectively, yields

$$\begin{bmatrix} \tilde{N}_1+\tilde{N}_1'-AX'-XA' & \tilde{N}_2'-\tilde{N}_1-\rho_2 XA'-BY & \tilde{N}_3'-\rho_3 XA'+X'+\tilde{P} & \tau_{\max}\tilde{N}_1 \\ * & -\tilde{N}_2-\tilde{N}_2'-\rho_2 BY-\rho_2 Y'B' & -\tilde{N}_3'+\rho_2 X'-\rho_3 Y'B' & \tau_{\max}\tilde{N}_2 \\ * & * & \rho_3 X+\rho_3 X'+\tau_{\max}\tilde{T} & \tau_{\max}\tilde{N}_3 \\ * & * & * & -\tau_{\max}\tilde{T} \end{bmatrix} < 0,$$

which is an LMI on the unknown matrices $X, Y, \tilde{P}, \tilde{T}$ and $\tilde{N}_i$, $i \in \{1,2,3\}$ and the values of ρ_2, ρ_3 are determined by line search such that $\tau_{\max}$ is maximized. Moreover, the controller gain can be recovered using $C = YX^{-1}$. This procedure introduces some conservativeness because it will not find controller gains C for which (2.38) holds for matrices M_2 and M_3 that are not scalar multiples of M_1, as in (2.49).

A simple but conservative way to make the matrix inequalities in Theorem 9 suitable for controller synthesis consists of requiring that

$$P_2 > 0, \qquad\qquad P_3 = \rho P_2,$$

for some positive constant $\rho > 0$ and making the (bijective) change of variables $Y = P_2 L$, which transforms (4.19) into

$$\begin{bmatrix} \Psi & -\begin{bmatrix} YC_P \\ \rho Y C_P \end{bmatrix} - T' \\ * & -S \end{bmatrix} < 0, \quad \begin{bmatrix} R & -\begin{bmatrix} C_P' Y' & \rho C_P' Y' \end{bmatrix} \\ * & Z_2 \end{bmatrix} > 0, \quad \begin{bmatrix} R & T \\ * & Z_1 \end{bmatrix} > 0,$$

with Ψ given by (2.43). This inequality is linear in the unknowns P_1, P_2, S, R, Z_1, Z_2, T, Y and can therefore be solved using efficient numerical algorithms while the value for ρ is found by line search such that $\tau_{\max}$ is maximized. The observer gain is found using $L = P_2^{-1} Y$. This procedure introduces some conservativeness because it restricts P_3 to be a scalar multiple of P_2. Naghshtabrizi and Hespanha [42] use the linear cone complementarity algorithm introduced by Ghaoui et al.

[15] to design the controller gains L and C for the anticipative or non-anticipative controllers in Section 2.1.3. The use of the cone complementarity algorithm avoids introducing additional conservativeness in going from a matrix inequality that is only appropriate for analysis to another matrix inequality that is appropriate for controller synthesis.

Chapter 3

Impulsive Systems: A Model For NCSs With Variable Sampling, SISO Case

This chapter starts off by considering an abstract SISO system of the form

$$\dot{x}(t) = Ax(t) + Bx(s_k), \qquad t \in [s_k, s_{k+1}),\ k \in \mathbb{N}, \tag{3.1}$$

which represents the closed-loop system in Fig. 3.1, where s_k denotes the k-th sampling time instance. We model this system as an impulsive system. The system in Fig. 3.1 represented by (3.1) can be viewed as an NCS in which a linear plant is in feedback with a static state-feedback remote controller.

Then we establish exponential stability of nonlinear time-varying impulsive systems. Lyapunov Theory provides the main tool in the time domain to test the stability of an impulsive system by employing a Lyapunov function (or a family of Lyapunov functions) [9, 28]. We employ a Lyapunov function, which is discon-

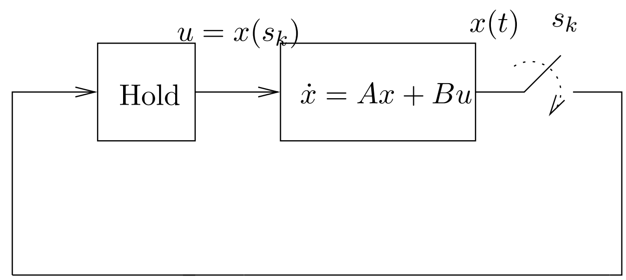


Figure 3.1. An abstract SISO system with variable sampling intervals where $u(t) = x(s_k)$ for $t \in [s_k, s_{k+1})$. The delays in the control loop are very small and hence negligible.

tinuous at a countable set of times, to establish the exponential stability of the system. Then we apply our theorems to the analysis and state-feedback stabilization for NCSs. We find a positive constant which determines an upper bound on the sampling intervals for which the stability of the closed-loop is guaranteed. The control design LMIs also provide controller gains that can be used to stabilize the plant.

3.1 Impulsive system model

Consider the SISO system in Fig. 3.1. The LTI process has a state space model of the form

$$\dot{x}(t) = Ax(t) + Bu(t),$$

where x, u are the state and input of the process. At the sampling time s_k, $k \in \mathbb{N}$ the process's state, $x(s_k)$, is sent to update the process input to be used as soon as it arrives until the next control command update. We find a set $\mathcal{S}$ of admissible sampling sequences $\{s_k\}$ such that

$$\epsilon \leq s_{k+1} - s_k \leq \tau_{MATI}, \tag{3.2}$$

so that if every sampling sequence $\{s_k\}$ belongs to $\mathcal{S}$, exponential stability of the closed-loop system (3.1) is guaranteed. To do so, we write the resulting closed-loop system (3.1) as an impulsive system of the form

$$\dot{\xi}(t) = F\xi(t), \qquad t \neq s_k, \forall k \in \mathbb{N} \tag{3.3a}$$

$$\xi(s_k) = \begin{bmatrix} x^-(s_k) \\ x^-(s_k) \end{bmatrix}, \qquad t = s_k, \forall k \in \mathbb{N}, \tag{3.3b}$$

where

$$F := \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \qquad \xi(t) := \begin{bmatrix} x(t) \\ z(t) \end{bmatrix}.$$

The overall state of the system ξ is composed of the process state, x, and the *hold state*, z where $z(t) := x(s_k),\ t \in [s_k, s_{k+1})$.

Remark 4. Note that we only index the samples that get to the destination, which enables us to capture sample drops [73]. Consequently, even if the sampling intervals are constant, because of the sample drops the system can be seen as a system with variable sampling intervals.

NCSs modeled by system (3.3) Equation (3.3) or (3.1) can model NCSs in which a linear plant with state-space

$$\dot{x}_p(t) = A_p x_p(t) + B_p u_p(t),$$

where $x_p \in \mathbb{R}^n, u_p \in \mathbb{R}^m$ are the state and the input of the plant, is in feedback with a static feedback gain K. At time s_k, $k \in \mathbb{N}$ the plant's state, $x(s_k)$, is sent to the controller and the control command $Kx(s_k)$ is sent back to the plant to be used as soon as it arrives until the next control command update. Then the closed-loop system can be written as (3.3) with

$$x := x_p, \quad A := A_p, \quad B := B_p K.$$

Since we only index the samples that get to the destination, we can capture packet dropout effect. So our model captures variable sampling and packet dropout; however, we assume the delays in the control loop are very small and hence negligible.

3.2 Exponential stability of impulsive systems

In the previous section we modeled the abstract system in Fig. 3.1 which represents a specific class of NCSs discussed in Section 3.1 as a linear impulsive system. In this section we present theorems for exponential stability of nonlinear time-varying impulsive systems which are more general than the linear impulsive system (3.3). The novelty of the conditions is that when specialized to the linear case, the conditions can be formulated as LMIs which can be solved effectively.

The nonlinear time-varying impulsive system has the following form

$$\dot{x}(t) = f_k(x(t), t), \qquad t \neq s_k,\ \forall k \in \mathbb{N}, \tag{3.4a}$$

$$x(s_k) = g_k(x^-(s_k), s_k), \qquad t = s_k,\ \forall k \in \mathbb{N}, \tag{3.4b}$$

where f_k and g_k are locally Lipschitz functions [25] from $\mathbb{R}^n \times \mathbb{R}$ to $\mathbb{R}^n$ such that $f_k(0,t) = 0, g_k(0,t) = 0,\ \forall t \geq 0$. The impulse time sequence $\{s_k\}$ forms a strictly increasing sequence in $[s_0, \infty)$ for some initial sampling time $s_0 \geq 0$.

Suppose that a sequence of impulse times $\{s_k\}$ is given. We say that the impulsive system (3.4) is Globally Exponentially Stable (GES) if

$$|x(t)| \leq c|x(s_0)|e^{-\lambda(t-s_0)}, \quad \forall t \geq s_0, \tag{3.5}$$

for some $c, \lambda > 0$. The constant λ provides an estimate for the decay rate and c an estimate for the overshoot of the solution. This definition depends on the choice of the sequence; however, it is often of interest to characterize GES over classes of impulse sequence. We say that the system (3.4) is Globally Uniformly Exponentially Stable (GUES) over the class $\mathcal{S}$ (of impulse times) if for any $\{s_k\} \in \mathcal{S}$ the condition (3.5) is satisfied with the same c, λ for every $\{s_k\} \in \mathcal{S}$. We define $\rho(t) := t - s_k,\ t \in [s_k, s_{k+1})$, for $\forall k \in \mathbb{N}$, which indicates the amount of time has passed since the last impulse time. As a result $\rho(s_k) = 0, \forall k \in \mathbb{N}$, $\rho^-(s_k) = s_k - s_{k-1}, \forall k \in \mathbb{N}$, and $0 \leq \rho(t) \leq \tau_{MATI}, \forall t \geq 0$.

We state a theorem to guarantee that the system (3.4) is GUES over the class $\mathcal{S}$ by employing a Lyapunov function with discontinuities at the impulse times.

Theorem 11. *Assume that there exist positive scalars c_1, c_2, c_3, b and a Lyapunov function $V : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$, such that for any impulse sequence $\{s_k\} \in \mathcal{S}$ and any $t \geq s_0$ the corresponding solution $x(.)$ to (3.4) satisfies:*

$$c_1|x|^b \leq V(x, \rho) \leq c_2|x|^b, \qquad \forall x, \forall \rho \in [0, \tau_{MATI}], \tag{3.6}$$

$$\frac{\mathrm{d}V(x(t), \rho(t))}{\mathrm{d}t} \leq -c_3 V(x(t), \rho(t)), \qquad \forall t \neq s_k, \forall k \in \mathbb{N}, \tag{3.7}$$

and that

$$V(x(s_k), 0) \leq \lim_{t \uparrow s_k} V(x(t), \rho(t)), \qquad \forall k \in \mathbb{N}. \tag{3.8}$$

Then the system (3.4) is GUES over the class $\mathcal{S}$ of impulse times with the following *estimates for the overshoot and the decay rate*

$$c := \big(\frac{c_2}{c_1}\big)^{\frac{1}{b}}, \qquad\qquad \lambda := \frac{c_3}{b}. \tag{3.9}$$

Moreover if we assume that every $f_k, k \in \mathbb{N}$ is globally Lipschitz with the Lipschitz constant $L > 0$ and instead of (3.6) *the following condition holds*

$$c_1|x(s_k)|^b \leq V(x(s_k), 0) \leq c_2|x(s_k)|^b, \qquad\qquad \forall k \in \mathbb{N}, \tag{3.10}$$

then the overshoot can be estimated as

$$c := \big(\frac{c_2}{c_1}\big)^{\frac{1}{b}} e^{(L+\frac{c_3}{b})\tau_{MATI}}. \qquad \square$$

Proof of Theorem 11. By the Comparison Lemma [25] and inequalities (3.7) and (3.8) we have

$$V(x(t), \rho(t)) \leq V(x(s_0), 0)e^{-c_3(t-s_0)}, \qquad \forall t \geq s_0. \tag{3.11}$$

Also from the condition (3.6) and the equation (3.11) we have

$$\begin{aligned} |x(t)| &\leq \Big(\frac{V(x(t), \rho(t))}{c_1}\Big)^{\frac{1}{b}} \leq \Big(\frac{V(x(s_0), 0)e^{-c_3(t-s_0)}}{c_1}\Big)^{\frac{1}{b}} \\ &\leq \Big(\frac{c_2|x(s_0)|^b e^{-c_3(t-s_0)}}{c_1}\Big)^{1/b} = \big(\frac{c_2}{c_1}\big)^{\frac{1}{b}}|x(s_0)|e^{-\frac{c_3}{b}(t-s_0)}. \end{aligned}$$

Thus the system (3.4) is GUES over the class $\mathcal{S}$ of impulse sequences with the decay rate and the overshoot estimate given by (3.9). Moreover from the equation (3.11) and the condition (3.10) we conclude that for any $\{s_k\} \in \mathcal{S}$

$$\begin{aligned} |x(s_k)| &\leq \Big(\frac{V(x(s_k), 0)}{c_1}\Big)^{\frac{1}{b}} \leq \Big(\frac{V(x(s_0), 0)e^{-c_3(s_k-s_0)}}{c_1}\Big)^{\frac{1}{b}} \\ &\leq \Big(\frac{c_2|x(s_0)|^b e^{-c_3(s_k-s_0)}}{c_1}\Big)^{1/b} = \big(\frac{c_2}{c_1}\big)^{\frac{1}{b}} e^{-\frac{c_3}{b}(s_k-s_0)}|x(s_0)|. \end{aligned} \tag{3.12}$$

Since every $f_k, k \in \mathbb{N}$ is globally Lipschitz with the Lipschitz constant $L > 0$, we have that

$$|x(t)| \leq e^{L(t-s_k)}|x(s_k)|, \qquad t \in [s_k, s_{k+1}), \tag{3.13}$$

[25] and from (3.12) and (3.13) we have that

$$|x(t)| \leq e^{L(t-s_k)}\Big(\frac{c_2}{c_1}\Big)^{\frac{1}{b}} e^{-\frac{c_3}{b}(s_k-s_0)}|x(s_0)| \leq \Big(\frac{c_2}{c_1}\Big)^{\frac{1}{b}} e^{(L+\frac{c_3}{b})\tau_{MATI}} e^{-\frac{c_3}{b}(t-s_0)}|x(s_0)|.$$

So the system is GUES over the class $\mathcal{S}$ with the decay rate given by (3.9) and the overshoot estimate given by $c := \big(\frac{c_2}{c_1}\big)^{\frac{1}{b}} e^{(L+\frac{c_3}{b})\tau_{MATI}}$. ■

The condition (3.6) requires that the candidate Lyapunov function to be positive for all times. This condition is relaxed in (3.10) by requiring the Lyapunov function to be positive only at the impulse times with the expense that we get a worse estimate for the overshoot of the system.

3.3 Exponential stability of NCSs

In this section we provide exponential stability conditions for the linear impulsive system in (3.3) which models the system in Fig. 3.1 or the NCSs described in Section 3.1. We will not focus on finding the overshoot and the decay rate, but instead we will find the largest sampling interval τ_{MATI} that the system is GUES over the class $\mathcal{S}$ defined in (3.2).

We now construct a finite-dimensional Lyapunov function for the system (3.3). Consider the candidate Lyapunov function

$$V(\xi, \rho) := V_1(x) + V_2(\xi, \rho) + V_3(\xi, \rho), \tag{3.14}$$

where

$$V_1(x) := x'Px, \quad V_2(\xi,\rho) := \xi'\Big(\int_{-\rho}^{0}(s+\tau_{MATI})(F\exp(Fs))'\tilde{R}\,F\exp(Fs)ds\Big)\xi,$$

$$V_3(\xi,\rho) := (\tau_{MATI}-\rho)(x-z)'X_1(x-z),$$

and $\tilde{R} := \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix}$ with R, P, X_1 symmetric positive definite matrices. The requirements $P, R, X_1 > 0$, guarantee the existence of positive c_1, c_2 such that (3.6) holds. Note that $V_2(\xi,\rho)$ can be written as

$$\int_{-\rho}^{0}\int_{t+\theta}^{t}\dot{x}'(s)R\dot{x}(s)dsd\theta. \tag{3.15}$$

This type of functional (or closely related forms) appeared in the delay differential equation and NCSs literature extensively, e.g., in [12, 56] but with the ρ replaced by τ_{MATI} in (3.15) and without the term $V_3(\xi,\rho)$.

Along jumps the Lyapunov function in (3.14) does not increase since $V_2(\xi,\rho)$ and $V_3(\xi,\rho)$ are non-negative before the jumps and they become zero right after the jumps so the condition (3.8) of Theorem 11 also holds. The next theorem provides a sufficient condition for (3.7) to hold. Consequently, if the conditions in the next theorem hold, based on Theorem 11 we conclude that the system (3.3) is GUES over the class $\mathcal{S}$ of impulse times.

Theorem 12. *The system (3.3) is GUES over the class $\mathcal{S}$ of impulse sequences, if there exist symmetric positive definite matrices P, R, X_1 and a (not necessarily symmetric) matrix N that satisfy the following LMIs:*

$$M_1 + \tau_{MATI}M_2 < 0, \tag{3.16a}$$

$$\begin{bmatrix} M_1 & \tau_{MATI}N \\ * & -\tau_{MATI}R \end{bmatrix} < 0, \tag{3.16b}$$

where

$$
\begin{aligned}
\bar{F} &:= \begin{bmatrix} A & B \end{bmatrix}, \\
M_1 &:= \begin{bmatrix} P \\ 0 \end{bmatrix} \bar{F} + \bar{F}' \begin{bmatrix} P & 0 \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} X_1 \begin{bmatrix} I & -I \end{bmatrix} \\
&\quad - N \begin{bmatrix} I & -I \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} N' + \tau_{MATI} \bar{F}' R \bar{F}, \\
M_2 &:= \begin{bmatrix} I \\ -I \end{bmatrix} X_1 \bar{F} + \bar{F}' X_1' \begin{bmatrix} I & -I \end{bmatrix}.
\end{aligned} \tag{3.17}
$$

□

See Chapter 3.6 for the proof of Theorem 12. When the sampling intervals approach zero (guarantee that $\tau_{MATI} \to 0$) the conditions (3.16a) and (3.16b) reduce to

$$
\begin{bmatrix} P \\ 0 \end{bmatrix} \bar{F} + \bar{F}' \begin{bmatrix} P & 0 \end{bmatrix} - N \begin{bmatrix} I & -I \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} N' < 0 \tag{3.18}
$$

(since M_2 does not appear in (3.16) when $\tau_{MATI} \to 0$ and the only remaining term in equation (3.16) that contains X_1 is negative semi-definite, one may simply take $X_1 = 0$). A sufficient condition for (3.18) to be satisfied is

$$
(A+B)'P + P(A+B) < 0, \qquad P = P' > 0, \tag{3.19}
$$

because if (3.19) holds, then (3.18) holds with the choice $N = \left[\begin{smallmatrix} -PB+I \\ -I \end{smallmatrix}\right]$. The condition (3.19) is the necessary and sufficient condition for the stability of the closed-loop system $\dot{x} = (A+B)x$. Hence the system in Fig. 3.1 is stable for small enough sampling intervals if the corresponding closed-loop continuous system is stable. By the Matrix Elimination Lemma it turns out that (3.19) is also a necessary condition for (3.18). Therefore as the sampling intervals approach zero, Theorem 12 recovers exactly the continuous-time stability condition. This does not happen for the conditions that appeared in [23]. Moreover using a Lyapunov function instead of a Lyapunov functional facilitates proving the exponential stability (instead of just asymptotic stability) of the system (3.3).

Remark 5. Suppose that there exist matrices $P_{1f} > 0$, P_{2f}, P_{3f}, Z_f and $R_f > 0$ satisfying the following stability conditions from Lemma 1 of [12]:

$$\Psi_{1f} < 0, \qquad -Z_f + P_f' \begin{bmatrix} 0 \\ B \end{bmatrix} R_f^{-1} \begin{bmatrix} 0 \\ B \end{bmatrix}' P_f < 0,$$

where

$$P_f := \begin{bmatrix} P_{1f} & 0 \\ P_{2f} & P_{3f} \end{bmatrix}, \qquad \Psi_{1f} := \Psi_{0f} + \tau_{MATI} Z_f + \tau_{MATI} \begin{bmatrix} 0 & 0 \\ 0 & R_f \end{bmatrix},$$

$$\Psi_{0f} := P' \begin{bmatrix} 0 & I \\ A+B & -I \end{bmatrix} + \begin{bmatrix} 0 & I \\ A+B & -I \end{bmatrix}' P.$$

Then we have

$$\Psi_{0f} + \tau_{MATI} \begin{bmatrix} 0 & 0 \\ 0 & R_f \end{bmatrix} + \tau_{MATI} P_f' \begin{bmatrix} 0 \\ B \end{bmatrix} R_f^{-1} \begin{bmatrix} 0 \\ B \end{bmatrix}' P_f < 0. \tag{3.20}$$

Multiplying (3.20) from the right and left by $\begin{bmatrix} I & 0 \\ A & B \end{bmatrix}$ and its transpose we obtain (3.28) with

$$\rho = \tau_{MATI} = h_f, \quad P = P_{1f}, \quad R = R_f, \quad N' = -\begin{bmatrix} B'P_{2f} + B'P_{3f}A & B'P_{3f}B \end{bmatrix}.$$

This means that if there are matrix variables satisfying the conditions of Lemma 1 in [12] then the conditions in Theorem 12 will necessarily also be satisfied. It is also possible to show that when the stability condition in [73] holds (given by equation (12) in [73]), then the condition in Theorem 12 must also necessarily hold with

$$\rho = \tau_{MATI} = \eta_y \quad R = T_y, \quad P = P_y, \quad N = \begin{bmatrix} N'_{1y} + N'_{3y}A & N'_{2y} + N'_{3y}B \end{bmatrix}.$$

Hence our Lyapunov function leads to conditions in Theorem 12 that are less conservative than the stability conditions in [12, 73] using a Lyapunov functional. From this perspective, considering an infinite dimensional delay differential equation model and using a Lyapunov functional to prove its stability offers no advantage for this class of finite dimensional systems. $\square$

Remark 6. Suppose that the system matrices $\Omega := \begin{bmatrix} A & B \end{bmatrix}$ are not exactly known and instead they are specified through the following polytopic condition:

$$\Omega \in \Big\{ \sum_{j=1}^{\kappa} f_j \Omega_j, \quad 0 \leq f_j \leq 1, \quad \sum_{j=1}^{\kappa} f_j = 1 \Big\},$$

where the κ vertices of the polytope are described by $\Omega_j := \begin{bmatrix} A_j & B_j \end{bmatrix}$. Stability of the system can be checked by solving the LMIs in Theorem 12 for each of the individual vertices with the same matrix variables P, X_1, R, N. $\square$

3.4 Exponential stability of NCSs with constant sampling

Now we consider the case where the sampling intervals are constant. This case may appear uninteresting since there are classical results giving necessary and sufficient conditions for stability and stabilization of such systems because the system is a sampled-data system with constant sampling. However, the LMI conditions presented here can be used for problems related to stability and stabilization of sampled-data system with polytopic uncertainty in the system matrices (Remark 6 and Example 2). In this case the classical results based on discretization method are not applicable or, in the case of lifting approach, they generally lead to a conservative results[1]. Also following the same steps as in [12], we can consider the stability and stabilization of sampled-data system with input saturation. Since our LMIs are less conservative (Remark 5 and that the results for constant sampling are less conservative than the variable sampling) we get a larger region of attraction than [12].

For constant sampling instead of (3.14) we use

$$V(\xi,\rho) := V_1(x) + V_2(\xi,\rho) + \bar{V}_3(\xi,\rho) \tag{3.21}$$

where $V_1(x)$ and $V_2(\xi,\rho)$ are as in (3.14) and

$$\bar{V}_3(\xi,\rho) :=(\tau_{MATI} - \rho)\big((x-z)'X_1(x-z) + 2z'X_2(x-z)\big). \tag{3.22}$$

Note that $\bar{V}_3(\xi,\rho)$ is not necessarily a positive function. However, right after the jumps this Lyapunov function is positive ($V(\xi,0) = x'Px$) and it satisfies

[1]In the lifting approach, stability of an uncertain sampled-data system is formulated as an H_∞ problem of the lifted system where the polytopic uncertainty is treated as a norm bounded uncertainty which may result in conservativeness.

(3.10). Along the jumps this Lyapunov function does not increase since $V_1(x)$ does not change at the sampling instance, $V_2(\zeta)$ is non-negative before the jumps and it becomes zero right after the jumps, and $\bar{V}_3(\xi, \rho)$ is zero before and after the jumps. Note that $\bar{V}_3(\xi^-(s_k), \rho^-(s_k))$ is zero before the jumps because $\rho = \tau_{MATI}$ and after the jumps it is zero because $x = z$ and consequently the condition (3.8) is satisfied. The next theorem provides a sufficient condition for (3.7) to hold. In conclusion, if the LMIs in the next theorem are feasible, all the conditions of Theorem 11 hold and the system (3.3) is GUES over the class $\mathcal{S}$ in (3.2) with $\epsilon = \tau_{MATI}$.

Theorem 13. *The system (3.3) is GUES over the class $\mathcal{S}$ with $\epsilon = \tau_{MATI}$ (i.e., constant sampling) if there exist symmetric positive definite matrices P, R and (not necessarily symmetric) matrices N, X_1, X_2 that satisfy the following LMIs:*

$$\bar{M}_1 + \tau_{MATI}\bar{M}_2 < 0,$$
$$\begin{bmatrix} \bar{M}_1 & \tau_{MATI}N \\ * & -\tau_{MATI}R \end{bmatrix} < 0,$$

where

$$\bar{M}_1 := M_1 - \begin{bmatrix} 0 \\ I \end{bmatrix} X_2 \begin{bmatrix} I & -I \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} X_2' \begin{bmatrix} 0 & I \end{bmatrix},$$
$$\bar{M}_2 := M_2 + \begin{bmatrix} 0 \\ I \end{bmatrix} X_2\bar{F} + \bar{F}'X_2' \begin{bmatrix} 0 & I \end{bmatrix},$$

and M_1, M_2 are defined in (3.17). □

See Chapter 3.6 for the proof of Theorem 13.

In table 3.1 we compare the number of scalar unknowns that appear in the LMIs of the different papers assuming that the dimension of the process is n.

Th. 12	Th. 13	[12]	[73]	[72]
$3.5n^2 + 1.5n$	$5n^2 + n$	$5n^2 + 2n$	$7n^2 + n$	$16n^2 + 3n$
1.1137	1.3277	0.8696	0.8696	0.8871

Table 3.1. The second row shows the number of variables in the LMIs used to test stability and the third row shows the value of τ_{MATI} for Example 1.

Notice that for an $n \times n$ symmetric matrix variable $\frac{n(n+1)}{2}$ scalar variables are needed whereas for an $m \times n$ matrix variable mn scalar variables are required. We can see that our results use fewer variables, but this is not at the expense of degrading the value obtained for τ_{MATI}. It is not fair to compare the number of variables in [72] to the others in Table 3.1 because this paper considered NCSs system with delays (although we presented the number of variables in [72] in Table 3.1).

Example 1. Consider the plant model from [3]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} u, \tag{3.24}$$

with the state feedback gain $K = -\begin{bmatrix} 3.75 & 11.5 \end{bmatrix}$. In our notation, this corresponds to

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -0.1 \end{bmatrix}, \qquad B = -\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \times \begin{bmatrix} 3.75 & 11.5 \end{bmatrix}.$$

Constant sampling Using standard techniques from digital control one can show that the maximum constant sampling interval for which the closed-loop system remains stable is $1.7s$. The maximum constant sampling interval given by Theorem 13 is 1.3277.

Variable sampling The stability results in [12, 73, 42] provide an upper bound on the sampling interval for which stability is guaranteed equal to 0.8696. This bound is improved to 0.8871 in [72]. Theorem 12 gives the upper bound equal to 1.1137. When compared to the constant-sampling bound given by Theorem 13, we now obtain a more conservative value, which is reasonable because we are now guaranteeing stability for every sequence of sampling times, with consecutive samples separated by no more than 1.1137, but potentially with different sampling intervals from one sample to the next. □

Example 2. Consider the plant model from [12] with

$$A = \begin{bmatrix} 1 & 0.5 \\ g_1 & -1 \end{bmatrix}, \qquad B_u = \begin{bmatrix} 1 + g_2 \\ -1 \end{bmatrix}, \qquad |g_1| \leq 0.1,\ |g_2| \leq 0.3$$

With the state feedback gain $K = -\begin{bmatrix} 2.6884 & 0.6649 \end{bmatrix}$, Fridman et al. [12] verified that the system is stable for any sampling interval smaller than 0.35. By solving the LMIs in Theorem 12 (see Remark 6) for each combination of A_j and $B_j, 1 \leq j \leq 2$ defined by

$$A_1 := \begin{bmatrix} 1 & 0.5 \\ -0.1 & -1 \end{bmatrix}, \quad A_2 := \begin{bmatrix} 1 & 0.5 \\ 0.1 & -1 \end{bmatrix}, \quad B_1 := \begin{bmatrix} 0.7 \\ -1 \end{bmatrix}, \quad B_2 := \begin{bmatrix} 1.3 \\ -1 \end{bmatrix},$$

we can verify that the system is stable for any variable sampling interval up to 0.4476. Applying Theorem 13 the system is stable for constant sampling up to 0.4610. □

3.5 State feedback controller design

In the design problem, when we want to find a feedback gain K that stabilizes the closed-loop system, the LMIs presented in the previous section become Bilinear Matrix Inequalities (BMIs) since there are cross terms between $B = KB_p$ and P. The next theorem provides LMI conditions that enable us to find a stabilizing feedback gain for variable sampling intervals. Following the same steps, one could find the state feedback for the constant sampling case.

Theorem 14. *There exists a state feedback gain K that makes the system (3.3) GUES over the class $\mathcal{S}$ if there exist positive scalars ϵ_1, ϵ_2, a symmetric positive definite matrix Q, (not necessarily symmetric) matrices N_d, Y, that satisfy the following LMIs:*

$$\begin{bmatrix} M_{1d} + \tau_{MATI} M_{2d} & \tau_{MATI} F_d' \\ * & -\tau_{MATI} \epsilon_1^{-1} Q \end{bmatrix} < 0, \tag{3.25a}$$

$$\begin{bmatrix} M_{1d} & \tau_{MATI} F_d' & \tau_{MATI} N_d \\ * & -\tau_{MATI} \epsilon_1^{-1} Q & 0 \\ * & * & -\tau_{MATI} \epsilon_1 Q \end{bmatrix} < 0, \tag{3.25b}$$

where

$$\begin{aligned} F_d &:= \begin{bmatrix} AQ & B_p Y \end{bmatrix}, \\ M_{1d} &:= \begin{bmatrix} I \\ 0 \end{bmatrix} F_d + F_d' \begin{bmatrix} I & 0 \end{bmatrix} - \epsilon_2 \begin{bmatrix} I \\ -I \end{bmatrix} Q \begin{bmatrix} I & -I \end{bmatrix} - N_d \begin{bmatrix} I & -I \end{bmatrix} - \begin{bmatrix} I \\ -I \end{bmatrix} N_d', \\ M_{2d} &:= \epsilon_2 \begin{bmatrix} I \\ -I \end{bmatrix} F_d + \epsilon_2 F_d' \begin{bmatrix} I & -I \end{bmatrix}. \end{aligned}$$

When these LMIs are feasible, the stabilizing state feedback gain is given by $K = YQ^{-1}$. □

Proof. Suppose the conditions (3.25a) and (3.25b) hold. We define $P := Q^{-1}$ and we multiply (3.25a) by $\bar{P} := \text{diag}(P, P)$ and (3.25b) by $\text{diag}(P, P, P)$ from the right and the left. We can show that the conditions (3.16a) and (3.16b) hold by using Schur lemma and choosing

$$N := \bar{P} N_d P, \qquad B := B_p K = B_p Y P, \qquad X_1 := \epsilon_2 P, \qquad R := \epsilon_1 P.$$

Since all the conditions of Theorem 12 are satisfied, the system (3.3) with state feedback gain K is GUES for any sampling sequence belongs to set $\mathcal{S}$. ■

The choice $R = \epsilon_1 P$ may lead to having more conservative results. This conservativeness could be reduced by using the cone complementarity algorithm [15]. Now instead of (3.25) the matrix inequalities to be considered are

$$\begin{bmatrix} M_{1d} + \tau_{MATI} M_{2d} & \tau_{MATI} F_d' \\ * & -\tau_{MATI} R^{-1} \end{bmatrix} < 0,$$
$$\begin{bmatrix} M_{1d} & \tau_{MATI} F_d' & \tau_{MATI} N_d \\ * & -\tau_{MATI} R^{-1} & 0 \\ * & * & -\tau_{MATI} Z \end{bmatrix} < 0,$$
$$\begin{bmatrix} R & Q^{-1} \\ Q^{-1} & Z \end{bmatrix} > 0.$$

These matrix inequalities are not LMIs because the inverse of the matrix variables appear. However, the cone complementarity algorithm transforms this problem into a sequence of linear optimizations subject to a set of LMIs that can be solved numerically. The improvement of using the cone complementarity algorithm has been investigated in [14]. If these LMIs are satisfied then $Z > Q^{-1}R^{-1}Q^{-1}$ (by Schur lemma), so the second inequality still holds when Z is replaced by $Q^{-1}R^{-1}Q^{-1}$. We can show that the conditions (3.16a) and (3.16b) hold by using Schur lemma and choosing

$$N := \bar{P}N_dP, \qquad K := YP, \qquad X_1 := \epsilon_2 P.$$

Example 3. Now we consider the state feedback controller design for the plant (3.24). We would like to find a feedback gain K that maximizes the upper bound of the variable sampling intervals. Yue et al. [73] found a stabilizing controller that guarantees stability up to a sampling interval equal to 402s. Our results provide the controller $K = \begin{bmatrix} 5 \times 10^{-5} & -0.0348 \end{bmatrix}$, which improves this upper bound to 820s. This upper bound on the variable sampling intervals is very large because the plant (3.24) is marginally stable and little control action is needed to exponentially stabilize the plant. In fact by choosing $u = -\alpha(0.1x_1 + x_2)$ with a small α, we can obtain a relatively large τ_{MATI}. For example with $\alpha = 0.001$ we get $\tau_{MATI} = 10^6$. The reason why this input leads to very large τ_{MATI} is the following. Defining $z := 0.1x_1 + x_2$ then the system dynamics are $\dot{z} = 0.1u$ and $\dot{x}_2 = -0.1x_2 + 0.1u$. This system with the input $u = -\alpha z$ is stable for large sampling intervals with small enough α because we only need to stabilize the z state and because of the cascade structure the state x_2 would be stable. Although our LMIs improve upon the previous results, probably because of numerical errors they are unable to give the controller that previous observation suggests. □

Example 4. Consider the system in Example 2. We would like to find a feedback gain K that maximizes the upper bound of the variable sampling intervals.

Fridman et al. [12] found $K = -\begin{bmatrix} 2.6884 & 0.6649 \end{bmatrix}$ which guarantees stability up to a sampling interval equal to 0.35. Our results provide the controller $K = -\begin{bmatrix} 2.5824 & 0.6419 \end{bmatrix}$, which improves this upper bound to 0.4550. $\square$

3.6 Appendix

Proof of Theorem 12. We write $V_2(\xi, \rho)$ in a form that is more convenient for our calculation

$$V_2(\xi, \rho) = \int_{t-\rho}^{t} (\tau_{MATI} - t + s)\dot{x}'(s)R\dot{x}(s)ds.$$

This simplifies the calculation (in comparison to [43]). Along the trajectory of the system (3.3) we have

$$\begin{aligned} \frac{\mathrm{d}V(\xi,\rho)}{\mathrm{d}t} &= 2x'P\dot{x} + 2(\tau_{MATI} - \rho)(x-z)'X_1\dot{x} \\ &\quad - (x-z)'X_1(x-z) + \tau_{MATI}\dot{x}'R\dot{x} - \int_{t-\rho}^{t} \dot{x}(s)'R\dot{x}(s)ds. \end{aligned} \tag{3.26}$$

Since $x(t) - z(t) = x(t) - x(t-\rho)$, for any matrix N, we have

$$\begin{aligned} 2\xi'N(x-z) = 2\xi'N\int_{t-\rho}^{t}\dot{x}(s)ds &\le \int_{t-\rho}^{t}\big(\xi'NR^{-1}N'\xi + \dot{x}(s)R\dot{x}(s)\big)ds \\ &\le \rho\xi'NR^{-1}N'\xi + \int_{t-\rho}^{t}\dot{x}(s)R\dot{x}(s)ds. \end{aligned} \tag{3.27}$$

The matrix variable N represents a degree of freedom that can be exploited to minimize conservativeness and we call it a slack matrix. Combining (3.26) and (3.27) we get

$$\begin{aligned} \frac{\mathrm{d}V(\xi,\rho)}{\mathrm{d}t} &\le 2x'P\dot{x} + 2(\tau_{MATI}-\rho)(x-z)'X_1\dot{x} - (x-z)'X_1(x-z) \\ &\quad + \tau_{MATI}\dot{x}'R\dot{x} - 2\xi'N(x-z) + \rho\xi'NR^{-1}N'\xi. \end{aligned}$$

We have $\frac{\mathrm{d}V(\xi,\rho)}{\mathrm{d}t} < 0$ if

$$M_1 + (\tau_{MATI} - \rho)M_2 + \rho M_3 < 0, \quad \forall \rho \in [0, \tau_{MATI}], \tag{3.28}$$

where M_1, M_2 are defined in (3.17) and $M_3 := NR^{-1}N'$. A necessary and sufficient condition to satisfy (3.28) is

$$M_1 + \tau_{MATI} M_2 < 0, \tag{3.29}$$

$$M_1 + \tau_{MATI} M_3 < 0. \tag{3.30}$$

To see that these matrix inequalities are sufficient, consider $\alpha \in [0, 1]$ and note that

$$\alpha\big(M_1 + \tau_{MATI} M_3\big) + (1-\alpha)\big(M_1 + \tau_{MATI} M_2\big) = M_1 + (\tau_{MATI} - \rho)M_2 + \rho M_3.$$

Setting $\alpha = \rho/\tau_{MATI}$ we conclude that (3.28) holds. Now suppose that (3.28) holds for every $\rho \leq \tau_{MATI}$. Hence it should hold when $\rho = 0$ and $\rho = \tau_{MATI}$ which gives (3.29) and (3.30) respectively. By Schur complement, the matrix inequalities in (3.29) and (3.30) can be written as the LMIs given in Theorem 12. Finally, when the LMIs in Theorem 12 are feasible, then $\frac{\mathrm{d}V(\xi,\rho)}{\mathrm{d}t} \leq -\bar{c}_3|\xi|^2$ for some $\bar{c}_3 \geq 0$ and (3.7) holds for $c_3 := \bar{c}_3/c_2$ where

$$\begin{aligned}
c_2 &:= \lambda_{\max}(P) + \tau_{MATI}(\gamma_1 + \gamma_2),\\
\gamma_1 &:= 1/2 \max_{-\tau_{MATI} \leq s \leq 0} \lambda_{\max}\big((Fe^{Fs})'\tilde{R}Fe^{Fs}\big),\\
\gamma_2 &:= \lambda_{\max}\Big(\begin{bmatrix} I \\ -I \end{bmatrix} X_1 \begin{bmatrix} I & -I \end{bmatrix} \Big).
\end{aligned} \tag{3.31}$$

We can apply Theorem 11 to prove exponential stability of the system; however, finding c_1 such that $c_1|\xi|^2 \leq V(\xi,\rho)$ is not easy. Note that we have $c_1|x|^2 \leq V(\xi,\rho)$

whereas we need $c_1|\xi|^2 \leq V(\xi,\rho)$. Instead, we follow the same steps of the proof of Theorem 11 with some modifications. From (3.7), $V(\xi,\rho)$ decreases to zero exponentially fast according to (3.11). Consequently $V_1(x) \leq V(x(s_0),0)e^{-c_3(t-s_0)}$. Given that $c_1|x|^2 \leq V_1(x)$ with $c_1 := \lambda_{\min}(P)$, the rest of the proof follows exactly the proof of Theorem (11). Moreover the system (3.3) is GUES over the class $\mathcal{S}$ with the decay rate and overshoot estimate defined in Theorem 11 with $b = 2$, and c_1, c_2, c_3 defined earlier. ■

Proof of Theorem 13. Along the flow

$$\begin{aligned}\frac{\mathrm{d}\bar{V}_3(\xi)}{\mathrm{d}t} &= 2(\tau_{MATI} - \rho)(x-z)'X_1\dot{x} - (x-z)'X_1(x-z) \\ &\quad + 2(\tau_{MATI} - \rho)z'X_2\dot{x} - 2z'X_2(x-z)\end{aligned}$$

and rest of the LMIs derivation is the same as Theorem 12. When the LMIs in Theorem 13 are feasible, then $\frac{\mathrm{d}V(\xi,\rho)}{\mathrm{d}t} \leq -\bar{c}_3|\xi|^2$ for some $\bar{c}_3 \geq 0$ and (3.7) holds for

$$\begin{aligned}\bar{c}_2 &:= \lambda_{\max}(P) + \tau_{MATI}(\gamma_1 + \gamma_2 + \gamma_3), \\ \gamma_3 &:= 2\lambda_{\max}\Big(\begin{bmatrix}0 \\ I\end{bmatrix} X_2 \begin{bmatrix} I & -I\end{bmatrix}\Big),\end{aligned}$$

and γ_1, γ_2 are defined by (3.31). The condition (3.10) holds with $c_1 := \lambda_{\min}(P)$ and $c_2 := \lambda_{\max}(P)$. So based on Theorem 11 the system (3.3) is GUES over the class $\mathcal{S}$ with the decay rate and overshoot estimate defined in Theorem 11 with $b = 2, L = |F|$ and c_1, c_2, c_3 defined earlier. ■

Chapter 4

Delay Impulsive Systems: A Model For NCSs With Variable Sampling And Delay, SISO Case

In the previous chapter, we considered NCSs in which the effect of the network appears as variable sampling. For some NCSs, the delay in the control loop is negligible. However, for most systems there might be large delays due to the shared communication medium (or computational resources). This motivates us to study a SISO system in Fig. 4.1, which can be expressed by

$$\dot{x}(t) = Ax(t) + Bx(s_k), \qquad t \in [s_k + \tau_k, s_{k+1} + \tau_{k+1}),\ k \in \mathbb{N}, \tag{4.1}$$

where s_k denotes the *k-th sampling time* and t_k the so called *k-th input update time*, which is the time instant at which the k-th sample arrives to the destination. In particular, denoting by τ_k the total delay that the k-th sample experiences in the loop, then $t_k := s_k + \tau_k$. The system in Fig. 4.1 and equation (4.1) can be viewed as an NCS in which a linear plant is in feedback with a static state-feedback remote controller. This class of NCSs is basically the same as the one considered in Chapter 3 and the only difference is that the samples experience delay in the control loop.

Motivated by NCSs with variable sampling intervals, delays and packet dropouts, we are interested in studying delay impulsive systems. We establish stability, asymptotic stability, and exponential stability theorems for delay impulsive systems by employing functionals with discontinuities at a countable set of times. A distinguishing feature of the stability conditions in this thesis is that, when specialized to linear impulsive systems, the stability tests can be formulated as LMIs, which can be solved efficiently.

We introduce a new *discontinuous Lyapunov functional* to establish the stability of (4.1) based on the theorems developed here for general nonlinear time-varying delay impulsive systems. The Lyapunov functional is discontinuous at the input update times, but a decrease is guaranteed by construction. We provide an inequality that guarantees the decrease of the Lyapunov functional between the discontinuities, from which stability follows. This inequality is expressed as a set of LMIs that can be solved numerically using software packages such as MATLAB. By solving these LMIs, one can find a positive constant that determines an upper bound between the sampling time s_k and the next input update time t_{k+1}, for which the stability of the closed-loop system is guaranteed for given lower and upper bounds on the total delay τ_k. When there is no delay, this upper bound corresponds to the maximum sampling interval, which is often called τ_{MATI} in the NCS literature. We use the τ_{MATI} terminology also for the case when there are delays in the system, which allows us to state our result in the form: the

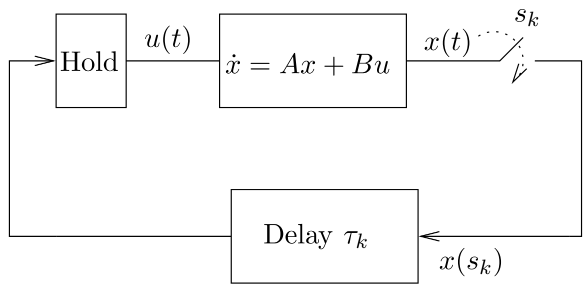


Figure 4.1. An abstract system with delay τ_k in the feedback loop where $u(t) = x(s_k)$ for $t \in [s_k + \tau_k, s_{k+1} + \tau_{k+1})$

.

system (4.1) is exponentially stable for any sampling-delay sequence satisfying $s_{k+1} + \tau_{k+1} - s_k \leq \tau_{MATI}$ and $\tau_{\min} \leq \tau_k \leq \tau_{\max}$ for $\forall k \in \mathbb{N}$, where $\tau_{\min}$, $\tau_{\max}$, and τ_{MATI} appear in our LMIs.

Our stability conditions depends both on the lower bound $\tau_{\min}$ and on an upper bound $\tau_{\max}$ on the loop delay, which can be estimated (perhaps conservatively) for most networks [62]. Through an example we show that considering $\tau_{\max}$ can significantly reduce conservativeness. When the delay in the feedback loop is small ($\tau_{\min}, \tau_{\max} \to 0$), our LMIs reduce to the ones presented in Chapter 3. This observation shows that the results in Chapter 3 are robust to small delays.

Notation Given an interval $I \subset \mathbb{R}$, $B(I, \mathbb{R}^n)$ denotes the space of real functions from I to $\mathbb{R}^n$ with norm $\|\phi\| := \sup_{t \in I} |\phi(t)|, \phi \in B(I, \mathbb{R}^n)$ where $|.|$ denotes any one of the equivalent norms in $\mathbb{R}^n$. x_t denotes the function $x_t : [-r, 0] \to \mathbb{R}^n$ defined by $x_t(\theta) = x(t + \theta)$, and r is a fixed positive constant.

4.1 Delay impulsive system model

Consider the system depicted in Fig. 4.1. The LTI process has a state space model of the form

$$\dot{x}(t) = Ax(t) + Bu(t),$$

where x, u are the state and input of the process. At the sampling time s_k, $k \in \mathbb{N}$ the process state, $x(s_k)$, is sent to update the process input to be used as soon as it arrives until the next control command update. We denote the *k-th input update time* with t_k, which is the time instant at which the k-th sample arrives at the destination. In particular, denoting by τ_k the total delay that the k-th sample experiences in the loop, then $t_k := s_k + \tau_k$. The resulting closed-loop system can be written as (4.1).

To be consistent with the results in [42], [72] and [11] we characterize the admissible set $\mathcal{S}$ of sampling-delay sequences $(\{s_k\}, \{\tau_k\})$ such that

$$s_{k+1} + \tau_{k+1} - s_k \leq \tau_{MATI}, \qquad \tau_{\min} \leq \tau_k \leq \tau_{\max}. \tag{4.2}$$

Although we adopt the above characterization in this chapter, (4.2) is not in a convenient form to provide the sampling rule. Another characterization is the admissible set $\bar{\mathcal{S}}$ of sampling-delay sequences $(\{s_k\}, \{\tau_k\})$ such that

$$s_{k+1} - s_k \leq \tau_{MATI} - \tau_{\max}, \qquad \tau_{\min} \leq \tau_k \leq \tau_{\max}, \tag{4.3}$$

which provides an explicit bound on the sampling intervals. Note that if any sampling-delay sequence belongs to $\bar{\mathcal{S}}$, it necessarily belongs to $\mathcal{S}$. We will find $\tau_{\min}, \tau_{\max}, \tau_{MATI}$ such that for any sampling-delay sequence belongs to set $\mathcal{S}$, exponential stability of the closed-loop system (4.1) is guaranteed. This necessarily means that any sampling-delay sequence belongs to set $\bar{\mathcal{S}}$ guarantees exponential stability of the closed-loop system (4.1).

We write the resulting closed-loop system (4.1) as an impulsive system of the form

$$\dot{\xi}(t) = F\xi(t), \qquad t \neq t_k, \forall k \in \mathbb{N} \tag{4.4a}$$

$$\xi(t_k) = \begin{bmatrix} x^-(t_k) \\ x(s_k) \end{bmatrix}, \qquad t = t_k, \forall k \in \mathbb{N}, \tag{4.4b}$$

where

$$F := \begin{bmatrix} A & B \\ 0 & 0 \end{bmatrix}, \qquad \xi(t) := \begin{bmatrix} x(t) \\ z(t) \end{bmatrix}.$$

The overall state of the system ξ is composed of the process state, x, and the *hold state*, z where $z(t) := x(s_k),\ t \in [t_k, t_{k+1})$.

NCSs modeled by system (4.4) Equation (4.4) or (4.1) models NCSs in which a linear plant with state-space

$$\dot{x}_p(t) = A_p x_p(t) + B_p u_p(t)$$

where $x_p \in \mathbb{R}^n, u_p \in \mathbb{R}^m$ are the state and the input of the plant, is in feedback with a static feedback gain K. At time s_k, $k \in \mathbb{N}$ the plant's state, $x(s_k)$, is sent to the controller and the control command $Kx(s_k)$ is sent back to the plant to be used as soon as it arrives until the next control command update. In particular, denoting by τ_k the total delay that the k-th sample experiences in the loop, then $t_k := s_k + \tau_k$. Then the closed-loop system can be written as (4.4)

$$x := x_p, \quad A := A_p, \quad B := B_p K.$$

Since we only index the samples that get to the destination (Remark 4), we can capture packet dropout effect .

4.2 Stability of delay impulsive systems

Consider the following delay impulsive system

$$\dot{x}(t) = f_k(x(t), t), \qquad t \neq t_k,\ \forall k \in \mathbb{N}, \tag{4.5a}$$

$$x(t_k) = g_k(x^-(t_k), x(s_k), t_k), \qquad t = t_k,\ \forall k \in \mathbb{N}, \tag{4.5b}$$

where f_k and g_k are locally Lipschitz functions [25] such that $f_k(0,t) = 0$, $g_k(0,0,t) = 0,\ \forall t \in \mathbb{R}_{[0,\infty)}$. For a given sampling-delay sequence $\{s_k, \tau_k : k \in \mathbb{N}\}$, the sequence of input update times $\{t_k, k \in \mathbb{N}\}$ is defined as $t_k := s_k + \tau_k, k \in \mathbb{N}$. We call the system (4.5) a delay impulsive system since the reset map (4.5b) depends on the past value of state.

The sampling times $\{s_0, s_1, \cdots\}$ and the input update times $\{t_0, t_1, \cdots\}$ form unbounded strictly increasing sequences. We allow the delays τ_k to grow larger than the sampling intervals $s_k - s_{k-1}$, provided that the sequence of input update times $\{t_0, t_1, \cdots\}$ remains strictly increasing. In essence, this means that if a sample gets to the destination out of order (i.e., an old sample gets to the destination after the most recent one), it should be dropped.

We can view (4.5) as an infinite dimensional system whose state contains the past history of $x(\cdot)$ so that $x(s_k)$ can be recovered from the state $x_{t_k} := x(t_k + s), -\tau_{MATI} \leq s \leq 0$ in order to apply the reset map in (4.5b). This allow us to apply Lyapunov-Krasovskii tools in the analysis of (4.5). In this framework, it is straightforward to analyze (4.5) even when the delays grow much larger than the sampling intervals, which is not easy in methods based on a discretization of (4.5) between update times [76, 4].

We assume that the impulse-delay sequences $(\{s_k\}, \{\tau_k\})$ belong to a given set $\mathcal{S}$ and we consider different stability definitions for (4.5) over $\mathcal{S}$:

(a) The system (4.5) is said to be *Globally Uniformly Stable* (GUS) over $\mathcal{S}$, if there exists some $\alpha \in \mathcal{K}$ such that for every $(\{s_k\}, \{\tau_k\}) \in \mathcal{S}$ and every initial condition x_{t_0} the solution to (4.5) is globally defined and satisfies $|x(t)| \leq \alpha(\|x_{t_0}\|)$, $\forall t \geq t_0$.

(b) The system (4.5) is said to be *Globally Asymptotically Stable* (GAS) over $\mathcal{S}$, if in addition to the conditions in (a), every solution converges to zero as $t \to \infty$.

(c) The system (4.5) is said to be *Globally Uniformly Asymptotically Stable* (GUAS) over $\mathcal{S}$, if there exists some $\beta \in \mathcal{KL}$ such that for every $(\{s_k\}, \{\tau_k\}) \in \mathcal{S}$ and every initial condition x_{t_0} the solution to (4.5) is globally defined and satisfies $|x(t)| \leq \beta(\|x_{t_0}\|, t - t_0)$, $\forall t \geq t_0$.

(d) The system (4.5) is said to be *Globally Uniformly Exponentially Stable* (GUES) over $\mathcal{S}$, when the function β in (c) is of the form $\beta(s, r) = ce^{-\lambda r}s$ for some $c, \lambda > 0$.

Theorem 15. *Suppose that there exist $\psi_1, \psi_2 \in \mathcal{K}_\infty$, $\psi_3 \in \mathcal{K}$ and a functional $V : B([-r, 0], \mathbb{R}^n) \times \mathbb{R}_{[0,\infty)} \to \mathbb{R}_{[0,\infty)}$, absolutely continuous between input update times, such that*

$$\psi_1(|\phi(0)|) \leq V(\phi, t) \leq \psi_2(\|\phi\|), \ \forall \, \phi \in B(I, \mathbb{R}^n), \, t \geq 0, \tag{4.6}$$

and, for every $(\{s_k\}, \{\tau_k\}) \in \mathcal{S}$, any solution x to (4.5) is globally defined for

$t \geq t_0$ and satisfies

$$\frac{\mathrm{d}V(x_t, t)}{\mathrm{d}t} \leq -\psi_3(|x(t)|), \qquad t \neq t_k,\ k \in \mathbb{N}, \tag{4.7}$$

$$V(x_{t_k}, t_k) \leq \lim_{t \uparrow t_k} V(x_t, t), \qquad \forall k \in \mathbb{N}. \tag{4.8}$$

Then the system (4.5) is GUS over $\mathcal{S}$. In addition, the following statements hold:

(a) The system (4.5) is GUAS over $\mathcal{S}$ if there is a constant $h_{\min} > 0$ for which $t_{k+1} - t_k \geq h_{\min}$, $\forall k \in \mathbb{N}$ for every $(\{s_k\}, \{\tau_k\}) \in \mathcal{S}$.

(b) The system (4.5) is GUES over $\mathcal{S}$, when the functions ψ_1, ψ_2 are of the following forms:

$$\psi_1(|\phi(0)|)) := c_1|\phi(0)|^b, \quad \psi_2(\|\phi\|) := c_2\|\phi\|^b, \tag{4.9}$$

and instead of (4.7), the following condition hold

$$\frac{\mathrm{d}V(x_t, t)}{\mathrm{d}t} \leq -c_3\|x_t\|^b, \qquad \forall t_k \leq t < t_{k+1},\ k \in \mathbb{N} \tag{4.10}$$

for some positive constants c_1, c_2, c_3, and b.

(c) The system (4.5) is GUES over $\mathcal{S}$, when the functions ψ_1, ψ_3 are of the following forms:

$$\psi_1(|\phi(0)|) := d_1|\phi(0)|^b, \quad \psi_3(|x(t)|) := d_3|x(t)|^b,$$

and in (4.6) the upper bound $\psi_2(\|\phi\|)$ is replaced by

$$d_2|\phi(0)|^b + \bar{d}_2 \int_{t-r}^{t} |\phi(s)|^b ds,$$

for some positive constants $d_1, d_2, \bar{d}_2, d_3$ and b. □

Items (b) and (c) in Theorem 15 both provide alternative conditions to guarantee GUES over $\mathcal{S}$. The former poses milder conditions on the Lyapunov functional than the latter, but it poses a more strict condition on the time derivative of the functional. We shall see shortly that the latter statement will lead to sufficient conditions in terms of LMIs for linear impulsive systems.

Proof of Theorem 15. For every $(\{s_k\}, \{\tau_k\}) \in \mathcal{S}$, we have $\frac{\mathrm{d}V(x_t,t)}{\mathrm{d}t} \leq 0$ for $\forall t \in [t_k, t_{k+1})$, $k \in \mathbb{N}$, therefore

$$\psi_1(|x(t)|) \leq V(x_t, t) \leq V(x_{t_k}, t_k), \quad t \in [t_k, t_{k+1}). \tag{4.11}$$

Based on the conditions (4.7) and (4.8), we also have

$$V(x^-_{t_{k+1}}, t^-_{k+1}) \leq V(x_{t_k}, t_k) \leq V(x^-_{t_k}, t^-_k), \quad \forall k \in \mathbb{N}. \tag{4.12}$$

Combining (4.11) and (4.12), we conclude that

$$\psi_1(|x(t)|) \leq V(x_t, t) \leq V(x^-_{t_k}, t^-_k) \leq \cdots \leq V(x_{t_0}, t_0) \leq \psi_2(\|x_{t_0}\|). \tag{4.13}$$

From (4.13), Lyapunov stability follows since $|x(t)| \leq \alpha(\|x_{t_0}\|)$, $\forall t \geq t_0$ for $\alpha(.) := \psi_1^{-1}(\psi_2(.))$.

(a) for every $\epsilon > 0$ let $\delta_1 > 0$ be such that $\psi_2(\delta_1) \leq \psi_1(\epsilon)$. Then $\|x_{t_0}\| \leq \delta_1$ implies that $|x(t)| < \epsilon$, $t \geq t_0$ because of (4.13). For this δ_1 and any $\eta > 0$, we show that there exists a $T = T(\delta_1, \eta)$ such that $|x(t)| \leq \eta$ for $\forall t \geq t_0 + T$. Choose $\delta_2 > 0$ such that $\psi_2(\delta_2) \leq \psi_1(\eta)$ for $t \geq t_0 + T$. Then it suffices to show that $\|x_{t_0+T}\| < \delta_2$ which implies $|x(t)| < \eta$, $\forall t \geq 0$. By contradiction we assume that such a T does not exist therefore there exists a sequence $c_k, k \in \mathbb{N}$ such that $\|x_{c_k}\| > \delta_2$. Each c_k is in an interval $[t_{k_i}, t_{k_{i+1}})$ where t_{k_i} is a subsequence of t_k.

Since $t_{k+1} - t_k \geq h_{\min}$, $\forall k \in \mathbb{N}$ then either $c_k - t_{k_i} \geq \frac{h_{\min}}{2}$ or $t_{k_{i+1}} - c_k \geq \frac{h_{\min}}{2}$. We define intervals

$$I_k := \begin{cases} [c_k - \frac{\delta_2}{2L_1}, c_k] & \text{if } c_k - t_{k_i} \geq \frac{h_{\min}}{2} \\ [c_k, c_k + \frac{\delta_2}{2L_1}] & \text{if } t_{k_{i+1}} - c_k \geq \frac{h_{\min}}{2} \end{cases},$$

where $L_1 > \max(L, \frac{\delta_2}{h_{\min}})$ and $|f_k(x,t)| < L$ for $\forall k \in \mathbb{N}$ (since f_k is Lipshitz, there exists $L > 0$ such that $|f_k(x,t)| < L$). By construction, $x(t)$ is continuous for any $t \in I_k$ and we can use the Mean Value Theorem. So for any $t \in I_k$ there exists a $\theta \in [0,1]$ such that

$$\begin{aligned} |x(t)| &= |x(c_k) + \dot{x}(c_k + \theta(t - c_k))(t - c_k)| \\ &\geq |x(c_k)| - |\dot{x}(c_k + \theta(t - c_k))| \left(|t - c_k|\right) \geq \delta_2 - L\frac{\delta_2}{2L_1} \geq \frac{\delta_2}{2}. \end{aligned}$$

Therefore $\frac{\mathrm{d}V(x_t,t)}{\mathrm{d}t} \leq -\psi_3(\frac{\delta_2}{2})$ for any $t \in I_k$ and otherwise $\frac{\mathrm{d}V(x_t,t)}{\mathrm{d}t} \leq 0$. By integration we have

$$V(x_{c_k}, c_k) \leq V(x_{t_0}, t_0) - \psi_3\Big(\frac{\delta_2}{2}\Big)\ \frac{k\delta_2}{2L_1},$$

but this would imply that $V(x_{c_k}, c_k) < 0$ for a sufficiently large k. By contradiction, we conclude that the system is GUAS over $\mathcal{S}$.

(b) Inequalities (4.6) with the choice of (4.9) and (4.10) implies

$$\frac{\mathrm{d}V(x_t,t)}{\mathrm{d}t} \leq -\frac{c_3}{c_2}V(x_t,t).$$

By the Comparison Lemma [25] we have $V(x_t,t) \leq V(x_{t_0},t_0)e^{-\frac{c_3}{c_2}(t-t_0)}$. Hence

$$\begin{aligned} |x(t)| &\leq \Big(\frac{V(x_t,t)}{c_1}\Big)^{1/b} \leq \Big(\frac{V(x_{t_0},t_0)e^{-\frac{c_3}{c_2}(t-t_0)}}{c_1}\Big)^{1/b} \\ &\leq \Big(\frac{c_2\|x_{t_0}\|^b e^{\frac{-c_3}{c_2}(t-t_0)}}{c_1}\Big)^{1/b} = (\frac{c_2}{c_1})^{1/b}\|x_{t_0}\|e^{-\frac{c_3}{c_2 b}(t-t_0)}. \end{aligned}$$

Thus, the origin is GUES over $\mathcal{S}$.

(c) Defining $W(x_t,t) := e^{\epsilon(t-t_0)}V(x_t,t)$, we conclude that

$$\begin{aligned}\frac{\mathrm{d}W(x_t,t)}{\mathrm{d}t} &= \epsilon e^{\epsilon(t-t_0)}V(x_t,t) + e^{\epsilon(t-t_0)}\frac{\mathrm{d}V(x_t,t)}{\mathrm{d}t} \\ &\leq \epsilon e^{\epsilon(t-t_0)}\Big(d_2|x(t)|^b + \bar{d}_2\int_{t-r}^{t}|x(v)|^b dv\Big) - d_3 e^{\epsilon(t-t_0)}|x(t)|^b. \end{aligned} \tag{4.14}$$

By integration of (4.14), we have

$$\begin{aligned}W(x_t,t) - W(x_{t_0},t_0) \leq &\epsilon d_2\int_{t_0}^{t}e^{\epsilon(s-t_0)}|x(s)|^b ds + \epsilon\bar{d}_2\int_{t_0}^{t}\int_{s-r}^{s}e^{\epsilon(s-t_0)}|x(v)|^b dvds \\ &- d_3\int_{t_0}^{t}e^{\epsilon(s-t_0)}|x(s)|^b ds.\end{aligned} \tag{4.15}$$

One can show that

$$\begin{aligned}&\int_{t_0}^{t}\int_{s-r}^{s}e^{\epsilon(s-t_0)}|x(v)|^b dvds \leq \int_{t_0-r}^{t_0}\int_{t_0}^{v+r}e^{\epsilon(s-t_0)}|x(v)|^b dsdv + \\ &\int_{t_0}^{t}\int_{v}^{v+r}e^{\epsilon(s-t_0)}|x(v)|^b dsdv \leq re^{\epsilon r}\int_{t_0-r}^{t_0}|x(v)|^b dv + re^{\epsilon r}\int_{t_0}^{t}e^{\epsilon(v-t_0)}|x(v)|^b dv.\end{aligned} \tag{4.16}$$

Combining (4.15), (4.16) and the fact that

$$W(x_{t_0},t_0) \leq d_2|x(t_0)|^b + \bar{d}_2\int_{t_0-r}^{t_0}|x(s)|^b ds$$

we get

$$\begin{aligned}W(x_t,t) \leq d_2|x(t_0)|^b &+ \bar{d}_2(1+\epsilon re^{\epsilon r})\int_{t_0-r}^{t_0}|x(v)|^b dv \\ &+ (\epsilon\bar{d}_2 re^{\epsilon r} - d_3)\int_{t_0}^{t}e^{\epsilon(v-t_0)}|x(v)|^b dv.\end{aligned}$$

For small enough ϵ,

$$W(x_t,t) \leq d_2|x(0)|^b + \bar{d}_2(1+\epsilon re^{\epsilon r})\int_{t_0-r}^{t_0}|x(v)|^b dv. \tag{4.17}$$

If (4.17) holds, there exists a $d_4 > 0$ such that $W(x_t, t) \leq d_4 \|x_{t_0}\|^b$ or $V(x_t, t) \leq d_4 e^{-\epsilon(t-t_0)} \|x_{t_0}\|^b$ and consequently $x(t) \leq (\frac{d_4}{d_1})^{1/b} e^{-\frac{\epsilon}{b}(t-t_0)} \|x_{t_0}\|$ for every $\{s_k, \tau_k\}$ $\in \mathcal{S}$. ■

4.3 Exponential stability of NCSs

In this section we provide conditions in terms of LMIs to guarantee exponential stability of the linear delay impulsive system in (4.4) which models the NCS described in section 4.1. Consider the Lyapunov functional

$$\begin{aligned}
& V := x'Px + \int_{t-\rho_1}^{t} (\rho_{1\max} - t + s)\dot{x}'(s)R_1\dot{x}(s)ds + \\
& \int_{t-\rho_2}^{t} (\rho_{2\max} - t + s)\dot{x}'(s)R_2\dot{x}(s)ds + \int_{t-\tau_{\min}}^{t} (\tau_{\min} - t + s)\dot{x}'(s)R_3\dot{x}(s)ds + \\
& \int_{t-\rho_1}^{t-\tau_{\min}} (\rho_{1\max} - t + s)\dot{x}'(s)R_4\dot{x}(s)ds + (\rho_{1\max} - \tau_{\min}) \int_{t-\tau_{\min}}^{t} \dot{x}'(s)R_4\dot{x}(s)ds + \\
& \int_{t-\tau_{\min}}^{t} x'(s)Zx(s)ds + (\rho_{1\max} - \rho_1)(x-w)'X(x-w), \qquad (4.18)
\end{aligned}$$

with $P, , X, Z, R_i, i = 1, .., 4$ positive definite matrices and

$$w(t) := x(t_k), \qquad \rho_1(t) := t - s_k, \qquad \rho_2(t) := t - t_k, \quad t_k \leq t < t_{k+1},$$

$$\rho_{1\max} := \sup_{t\geq 0} \rho_1(t), \qquad \rho_{2\max} := \sup_{t\geq 0} \rho_2(t).$$

If the LMIs in the next theorem are feasible for given $\tau_{MATI}, \tau_{\min}$, and $\tau_{\max}$, then there exists a $d_3 > 0$ such that $\frac{\mathrm{d}V(x_t,t)}{\mathrm{d}t} \leq -d_3|x(t)|^2$. It is straightforward to show that the Lyapunov functional (4.18) satisfies the remaining conditions in Theorem (15). Hence the NCS modeled by the delay impulsive system (4.4) is GUES over $\mathcal{S}$ given by (4.2).

Theorem 16. *The system (4.4) is GUES over $\mathcal{S}$ defined by* (4.2)*, if there exist positive definite matrices $P, X, Z, R_i, i = 1, .., 4$ and (not necessarily symmetric) matrices $N_i, i = 1, .., 4$ that satisfy the following LMIs:*

$$\begin{bmatrix} M_1+\beta_{\max}(M_2+M_3) & \tau_{\max}N_1 & \tau_{\min}N_3 \\ * & -\tau_{\max}R_1 & 0 \\ * & * & -\tau_{\min}R_3 \end{bmatrix} < 0, \tag{4.19a}$$

$$\begin{bmatrix} M_1+\beta_{\max}M_2 & \tau_{\max}N_1 & \tau_{\min}N_3 & \beta_{\max}(N_1+N_2) & \beta_{\max}N_4 \\ * & -\tau_{\max}R_1 & 0 & 0 & 0 \\ * & * & -\tau_{\min}R_3 & 0 & 0 \\ * & * & * & -\beta_{\max}(R_1+R_2) & 0 \\ * & * & * & * & -\beta_{\max}R_4 \end{bmatrix} < 0, \tag{4.19b}$$

where

$$\begin{aligned} M_1 :=& \bar{F}' \begin{bmatrix} P & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} P \\ 0 \\ 0 \\ 0 \end{bmatrix} \bar{F} + \tau_{\min}F'(R_1 + R_3)F - \\ & \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} X \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix}' + \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \end{bmatrix} Z \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \end{bmatrix}' - \begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix} Z \begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix}' \\ & - N_1 \begin{bmatrix} I & -I & 0 & 0 \end{bmatrix} - \begin{bmatrix} I \\ -I \\ 0 \\ 0 \end{bmatrix} N_1' - N_2 \begin{bmatrix} I & 0 & -I & 0 \end{bmatrix} - \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} N_2' \\ & - N_3 \begin{bmatrix} I & 0 & 0 & -I \end{bmatrix} - \begin{bmatrix} I \\ 0 \\ 0 \\ -I \end{bmatrix} N_3' - N_4 \begin{bmatrix} 0 & -I & 0 & I \end{bmatrix} - \begin{bmatrix} 0 \\ -I \\ 0 \\ I \end{bmatrix} N_4', \\ M_2 :=& \bar{F}'(R_1 + R_2 + R_4)\bar{F}, \\ M_3 :=& \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} X\bar{F} + \bar{F}'X \begin{bmatrix} I & 0 & -I & 0 \end{bmatrix}. \end{aligned} \tag{4.20}$$

with $\bar{F} := \begin{bmatrix} A & B & 0 & 0 \end{bmatrix}.$ □

Proof is given in Chapter 4.4.

Remark 7. When the delays are small, i.e., $\tau_{\min}, \tau_{\max} \to 0$ the LMIs (4.19a) and (4.19b) are equivalent to

$$\begin{aligned} & M_1 + \beta_{\max}(M_2 + M3) < 0, \\ & M_1 + \beta_{\max}M_2 + \beta_{max}(N_1 + N_2)(R_1 + R_2)^{-1}(N_1 + N_2)' < 0 \end{aligned} \tag{4.21}$$

(after using Schur Lemma). Making $R_3, R_4, Z, N_3, N_4 \to 0$ and

$$N_1 = \begin{bmatrix} N_{11} & N_{12} & N_{13} & 0 \end{bmatrix}', \quad N_2 = \begin{bmatrix} N_{21} & N_{22} & N_{23} & 0 \end{bmatrix}',$$

makes the last row and column of the LMIs in (4.21) approach zero and we can omit them. After multiplying the LMIs in (7) by $\begin{bmatrix} I & 0 & 0 \\ 0 & I & I \end{bmatrix}$ and its transpose from the left and the right, respectively, choosing

$$N = \begin{bmatrix} N_{11} + N_{21} & N_{12} + N_{13} + N_{22} + N_{23} \end{bmatrix}', \qquad R_1 + R_2 = R,$$

we obtain the LMIs in Theorem 12. So the results in the previous chapter are robust with respect to small delays, namely the results in Theorem 12 still guarantee stability for arbitrary small delays. □

It is often the case that the lower bound on the delay in the network is very close to zero. When the load in the network is low and the computation delays are small, the total end to end delay in the loop is equal to transmission and propagation delays which typically are small. Next we present the conditions for the case when $\tau_{\min} = 0$. The conditions can be derived from the conditions in Theorem 16 for the case when $\tau_{\min} \to 0$ or they can be directly derived employing the following Lyapunov functional

$$V := x'Px + \int_{t-\rho_1}^{t} (\rho_{1\max} - t + s)\dot{x}'(s)R_1\dot{x}(s)ds +$$
$$\int_{t-\rho_2}^{t} (\rho_{2\max} - t + s)\dot{x}'(s)R_2\dot{x}(s)ds + (\rho_{1\max} - \rho_1)(x-w)'X(x-w).$$

Theorem 17. *The system (4.4) is GUES over $\mathcal{S}$ with $\tau_{\min} = 0$ defined by* (4.2)*, if there exist positive definite matrices P, X, R_1, R_2 and (not necessarily symmetric)* *matrices N_1, N_2 that satisfy the following LMIs:*

$$\begin{bmatrix} M_1+\beta_{\max}(M_2+M_3) & \tau_{\max}N_1 \\ * & -\tau_{\max}R_1 \end{bmatrix} < 0, \tag{4.22a}$$

$$\begin{bmatrix} M_1+\beta_{\max}M_2 & \tau_{\max}N_1 & \beta_{\max}(N_1+N_2) \\ * & -\tau_{\max}R_1 & 0 \\ * & * & -\beta_{\max}(R_1+R_2) \end{bmatrix} < 0, \tag{4.22b}$$

where

$$\begin{aligned} M_1 :=& \bar{F}' \begin{bmatrix} P & 0 & 0 \end{bmatrix} + \begin{bmatrix} P \\ 0 \\ 0 \end{bmatrix} \bar{F} - \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} X \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix}' - N_1 \begin{bmatrix} I & -I & 0 & 0 \end{bmatrix} - \begin{bmatrix} I \\ -I \\ 0 \\ 0 \end{bmatrix} N_1' \\ & - N_2 \begin{bmatrix} I & 0 & -I & 0 \end{bmatrix} - \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} N_2' \\ M_2 :=& \bar{F}'(R_1 + R_2 + R_4)\bar{F}, \\ M_3 :=& \begin{bmatrix} I \\ 0 \\ -I \end{bmatrix} X\bar{F} + \bar{F}'X \begin{bmatrix} I & 0 & -I \end{bmatrix}. \end{aligned} \tag{4.23}$$

with $\bar{F} := \begin{bmatrix} A & B & 0 \end{bmatrix}$. □

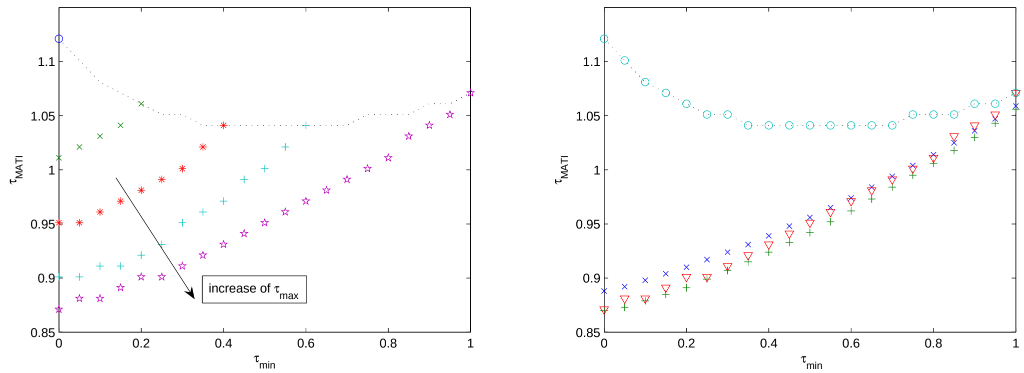
Example 5. Consider the state space plant model [3]

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -0.1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.1 \end{bmatrix} u,$$

with state feedback gain $K = -\begin{bmatrix} 3.75 & 11.5 \end{bmatrix}$, for which we have

$$A = \begin{bmatrix} 0 & 1 \\ 0 & -0.1 \end{bmatrix}, \qquad B = -\begin{bmatrix} 0 \\ 0.1 \end{bmatrix} \times \begin{bmatrix} 3.75 & 11.5 \end{bmatrix}.$$

By checking the condition $eig(\left[\begin{smallmatrix} I & 0 \\ I & 0 \end{smallmatrix}\right] e^{Fh}) < 0$ on a tight grid of h, we can show that the closed-loop system remains stable for any constant sampling interval smaller than 1.7, and becomes unstable for larger constant sampling intervals. On the other hand, when the sampling interval approaches zero, the system is described



(a) (b)

Figure 4.2. Fig. 4.2(a) shows the plot of τ_{MATI} versus $\tau_{\min}$ for $\tau_{\max}$ equal to $0, .2, .4, .6, 1$ based on Theorem 16. The dashed line is the same as the one in Fig.4.2(b). Fig. 4.2(b) shows the plot τ_{MATI} versus $\tau_{\min}$ where $\tau_{\max} = \tau_{\min}$ from [42] ('+') and [72] ('$\times$'), the worse case where $\tau_{\max} = \tau_{MATI}$ ('∇') and the best case where $\tau_{\max} = \tau_{\min}$ ('o') from Theorem 16.

by a DDE and we can find the maximum constant delay for which stability is guaranteed by looking at the roots of the characteristic function $\det(sI - A - Be^{-\tau_0 s})$. We use the Pade approximation $e^{-\tau_0 s} = \frac{1-s\tau_0/2}{1+s\tau_0/2}$ to compute the determinant polynomial and conclude by the Routh-Hurwitz test that the system is stable for any constant delay smaller than 1.36. Comparing these numbers with the maximum variable sampling interval 1.1137 and the maximum variable delay 1.0744 both obtained using Theorem 16 (see below) reveals the conservativeness of our method:

No-delay and variable sampling When there is no delay but the sampling intervals are variable, τ_{MATI} determines an upper bound on the variable sampling intervals $s_{k+1} - s_k$. The upper bound given by [12, 73, 42] (when $\tau_{\min} = 0$) is 0.8696 which is improved to 0.8871 in [72]. Theorem 16 and [43] gives the upper bound equal to 1.1137.

Variable-delay and sampling Fig. 4.2(a) shows the value of τ_{MATI} obtained from Theorem 16, as a function of $\tau_{\min}$ for different values of τ_{max}. The dashed curves in Fig. 4.2(a) and Fig. 4.2(b) are the same which are comprised of the largest τ_{MATI} for different values of $\tau_{\max}$. Fig. 4.2(b) shows τ_{MATI} with respect to τ_{min} where the results from [42], [72] are shown by $+$, $\times$ respectively. The values of τ_{MATI} given by [11] lie between the "$+$" and "$\times$" in Fig. 4.2(b) and we do not show them. In Theorem 16, τ_{MATI} is a function of $\tau_{\min}$ and $\tau_{\max}$. To be able to compare our result to the others we consider two values for $\tau_{\max}$ and we obtain τ_{MATI} as a function of $\tau_{\min}$ based on Theorem 16. Fist we consider $\tau_{\max} = \tau_{\min}$, which is the case that the delay is constant and equal to the value of $\tau_{\min}$. The largest τ_{MATI} for a given $\tau_{\min}$ provided by Theorem 16 is shown using an "o" in Fig. 4.2(b). The second case is when $\tau_{\max} = \tau_{MATI}$, which is the case where there can be very large delays in the loop in comparison to the sampling intervals. The largest τ_{MATI} for a given $\tau_{\min}$ for this case provided by Theorem 16 is shown using a "∇" in Fig. 4.2(b). One can observe that when the delays in the control loop are small, our method shows a good improvement in comparison to the other results in the literature. $\square$

4.4 Appendix

Proof of Theorem 16 . It is easy to show that the Lyapunov functional (4.18) satisfies the condition (4.6) with

$$\psi_1(s) := d_1 s^2, \qquad \psi_2(\|\phi\|) := d_2|\phi(0)|^2 + \bar{d}_2 \int_{t-r}^{t} |\phi(s)|^2 ds,$$

for $d_1, d_2, \bar{d}_2 > 0$. Also the condition (4.8) is guaranteed by construction of the Lyapunov functional. The only remaining condition of Theorem 15 (part c) to guarantee GUES is the condition (4.7) to hold. In the following we derive conditions that if those hold, the condition (4.7) is necessarily satisfied. Along the trajectory of the system (4.5)

$$\begin{aligned}
\dot{V} =& 2x'(t)P(Ax(t) + Bz) + \rho_{1\max}\dot{x}'(t)R_1\dot{x}(t) \\
&- \int_{t-\rho_1}^{t} \dot{x}'(s)R_1\dot{x}(s)ds + \rho_{2\max}\dot{x}'(t)R_2\dot{x}(t) \\
&- \int_{t-\rho_2}^{t} \dot{x}'(s)R_2\dot{x}(s)ds + \tau_{\min}\dot{x}'(t)R_3\dot{x}(t) \\
&- \int_{t-\tau_{\min}}^{t} \dot{x}'(s)R_3\dot{x}(s)ds \\
&+ (\rho_{1\max} - \tau_{\min})\big(\dot{x}'(t-\tau_{\min})R_4\dot{x}(t-\tau_{\min})\big) \\
&- \int_{t-\rho_1}^{t-\tau_{\min}} \dot{x}'(s)R_4\dot{x}(s)ds + (\rho_{1\max} - \tau_{\min})\big(\dot{x}'(t)R_4\dot{x}(t) \\
&- \dot{x}'(t-\tau_{\min})R_4\dot{x}(t-\tau_{\min})\big) + x'(t)Zx(t) \\
&- x'(t-\tau_{\min})Zx(t-\tau_{\min}) - (x(t)-w)'X(x(t)-w) \\
&+ 2(\rho_{1\max} - \rho_1)(x-w)'X(Ax+Bz). \qquad (4.24)
\end{aligned}$$

Defining $\bar{\xi}(t) := \begin{bmatrix} x'(t) & z' & w' & x'(t-\tau_{\min}) \end{bmatrix}'$, for any matrices $N_i, i = 1, .., 4$ we have

$$
\begin{aligned}
&2\bar{\xi}'N_1\begin{bmatrix} I & -I & 0 & 0 \end{bmatrix}\bar{\xi} + 2\bar{\xi}'N_2\begin{bmatrix} I & 0 & -I & 0 \end{bmatrix}\bar{\xi} \\
&= 2\bar{\xi}'(N_1+N_2)\Big(\int_{t-\rho_2}^{t}\dot{x}(s)ds\Big) + 2\bar{\xi}'N_1\Big(\int_{t-\rho_1}^{t-\rho_2}\dot{x}(s)ds\Big) \\
&\leq \rho_2\bar{\xi}'(N_1+N_2)(R_1+R_2)^{-1}(N_1+N_2)'\bar{\xi} \\
&+ \int_{t-\rho_2}^{t}\dot{x}'(s)(R_1+R_2)\dot{x}(s)ds \\
&+ (\rho_1-\rho_2)\bar{\xi}'N_1R_1^{-1}N_1'\bar{\xi} + \int_{t-\rho_1}^{t-\rho_2}\dot{x}'(s)R_1\dot{x}(s)ds, \qquad (4.25) \\
&2\bar{\xi}'N_3\begin{bmatrix} I & 0 & 0 & -I \end{bmatrix}\bar{\xi} = 2\bar{\xi}'N_3\Big(\int_{t-\tau_{\min}}^{t}\dot{x}(s)ds\Big) \\
&\leq \tau_{\min}\bar{\xi}'N_3R_3^{-1}N_3'\bar{\xi} + \int_{t-\tau_{\min}}^{t}\dot{x}'(s)R_3\dot{x}(s)ds, \qquad (4.26) \\
&2\bar{\xi}'N_4\begin{bmatrix} 0 & -I & 0 & I \end{bmatrix}\bar{\xi} = 2\bar{\xi}'N_4\Big(\int_{t-\rho_1}^{t-\tau_{\min}}\dot{x}(s)ds\Big) \\
&\leq (\rho_1-\tau_{\min})\bar{\xi}'N_4R_4^{-1}N_4'\bar{\xi} + \int_{t-\rho_1}^{t-\tau_{\min}}\dot{x}'(s)R_4\dot{x}(s)ds, \qquad (4.27)
\end{aligned}
$$

which relies on the fact that $x(t)-z(t) = x(t)-x(t-\rho_1)$ and $x(t)-w(t) = x(t)-x(t-\rho_2)$. The matrix variables N_1, N_2, N_3, N_4 represent degrees of freedom that can be exploited to minimize conservativeness and we call them slack matrices. Let us define $\beta := \rho_1 - \tau_{\min}$ and $\beta_{\max} := \rho_{1\max} - \tau_{\min}$. Note that $\tau_{\min} \leq \rho_1 - \rho_2 \leq \tau_{\max}$ and

$$
\begin{aligned}
\rho_{1\max} &= \sup_k\left(s_{k+1}+\tau_{k+1}-s_k+\tau_k-\tau_k\right) \leq \rho_{2\max}+\tau_{\max}, \\
\rho_{2\max}+\tau_{\min} &= \sup_k\left(s_{k+1}+\tau_{k+1}-s_k-\tau_k+\tau_{\min}\right) \leq \rho_{1\max},
\end{aligned}
$$

so we conclude that $\tau_{\min} \leq \rho_{1\max} - \rho_{2\max} \leq \tau_{\max}$, $\rho_{2\max} \leq \beta_{\max}$, and $\rho_2 \leq \beta$. After combining (4.24), (4.25),(4.26), and (4.27) and replacing $\rho_{2\max}, \rho_2, \rho_1 - \rho_2$

with $\beta_{\max}, \beta, \tau_{\max}$ we get

$$\begin{aligned}\dot{V}(\bar{\xi}) \leq& \bar{\xi}'\big(\Psi + \beta_{\max}(M_2 + M_3) + \beta(M_4 - M_3)\big)\bar{\xi}, \\ \Psi :=& M_1 + \tau_{\max} N_1 R_1^{-1} N_1' + \tau_{\min} N_3 R_3^{-1} N_3', \\ M_4 :=& (N_1 + N_2)(R_1 + R_2)^{-1}(N_1 + N_2)' + N_4 R_4^{-1} N_4',\end{aligned} \tag{4.28}$$

and M_1, M_2, M_3 are defined in (4.20). The necessary and sufficient condition to satisfy (4.28) is

$$\Psi + \tau_{MATI}(M_2 + M_3) < 0,\ \Psi + \tau_{MATI}(M_2 + M_4) < 0 \tag{4.29}$$

The proof is similar to the proof given in Section 3.6 for Theorem 12. By Schur complement, the matrix inequalities in (4.29), can be written as the LMIs in Theorem 16. If the LMIs in Theorem 16 are feasible, then there exists a $d_3 > 0$ such that condition (4.7) is satisfied with $\psi_3(s) := d_3 s^2$ for $d_3 > 0$. Hence all the conditions in Theorem 15 are satisfied so the system (4.4) is GUES over $\mathcal{S}$. ■

Chapter 5

Delay Impulsive Systems: A Model For NCSs With Variable Sampling And Delay, MIMO Case

In chapters 3 and 4 we study stability of NCSs in which a linear plant is in feedback with a static state-feedback remote controller. To enlarge the class of NCSs we consider an abstract system depicted in the Fig. 5.1 which consists of a Multi-Input Multi-Output (MIMO) process of the form

$$\dot{x}(t) = Ax(t) + Bu(t), \qquad y(t) = Cx(t), \tag{5.1}$$

where $x \in \mathbb{R}^n$ is the state of the process, $u \in \mathbb{R}^q$ is the input of the process, and $y \in \mathbb{R}^q$ is the output of the process. The input and the output are partitioned as $y := \left[\begin{smallmatrix} y_1' & \cdots & y_m' \end{smallmatrix}\right]'$ and $u := \left[\begin{smallmatrix} u_1' & \cdots & u_m' \end{smallmatrix}\right]'$ and the partition of u matches the partition of y. We denote the $k-$th sampling time of the $i-th$ output by s_k^i. At time s_k^i, $k \in \mathbb{N}$ the $i-$th output of the system, $y_i(t)$, $1 \leq i \leq m$ is sampled and $y_i(s_k^i)$ is sent to update the input u_i, to be used as soon as it arrives until the next update arrives.[1]

In NCSs each y_i represents the output of nodes that can send their information to the network in a single packet. For example sensors spatially close to each other, send their measurements in a single packet or controller outputs sent to actuators spatially close to each other, can be sent in a single packet. The abstract system in Fig. 5.1 is general enough to model both one-channel NCSs and two-channel NCSs with dynamic output-feedback controllers that may or may not be anticipative [42] which each will be discussed with more details in Section 5.2. The closed loop system in Fig. 5.1 can be modeled as a linear impulsive system where the reset maps are time varying because each time a new measurement arrives, only a particular partition of the input is updated. Theorem 15 is general enough to establish exponential stability of this type of impulsive system because the flow and reset maps of system (4.5) are time-varying.

For LTI plants and controllers, we present two stability tests in terms of LMIs based on Theorem 15. The first one is less conservative but the number of LMIs grows exponentially with m. The second stability condition is based on the feasibility of a single LMI with dimension that grows linearly with m. For small m the first stability test is more desirable, because it leads to less conservative results, but the second stability test is more adequate for large m.

By solving the LMIs that guarantee stability, one obtains positive constants that determine the upper bound between the sampling time and the next input

[1] Unlike the previous chapters we do not write the closed-loop system such as equation (4.1) in Chapter 4 because the equation is rather complicated to the point that it is not beneficial.

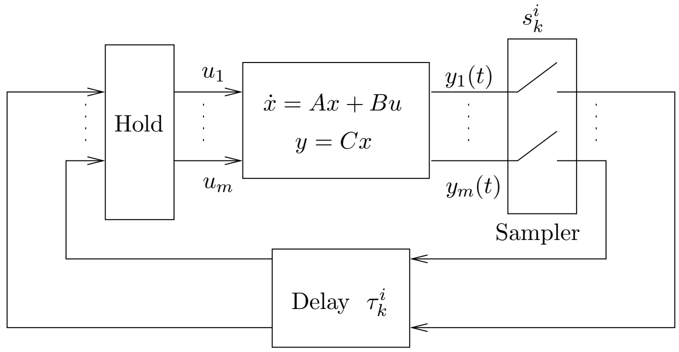


Figure 5.1. MIMO system with variable sampling intervals and delays where $u_i(t) = y_i(s_k^i)$ for $t \in [t_k^i, t_{k+1}^i), \forall i \in \{1, \cdots, m\}$ where $t_k^i := s_k^i + \tau_k^i$.

update time for each output of the process for which the stability of the closed-loop system is guaranteed for a given lower and upper bound on the total delay in each loop.

5.1 Delay impulsive system model

We consider a MIMO system depicted in Fig. 5.1. The input is partitioned as $u := \begin{bmatrix} u_1' & \cdots & u_m' \end{bmatrix}'$ where $u_i \in \mathbb{R}^{q_i}$, $i \in \{1, \cdots, m\}$ and $\sum_{i=1}^m q_i = q$ and the output is partitioned as $y := \begin{bmatrix} y_1' & \cdots & y_m' \end{bmatrix}'$ where $y_i \in \mathbb{R}^{q_i}$, $i \in \{1, \cdots, m\}$ and $\sum_{i=1}^m q_i = q$ and the partitions of u and y are matched. At time s_k^i, $i \in \{1, \cdots, m\}, k \in \mathbb{N}$ the $i-$th output of the system, $y_i(t)$ is sampled and $y_i(s_k^i)$ is sent through the network to update u_i, to be used as soon as it arrives until the next update arrives. The total delay in the control loop that the k-th sample of y_i experiences is denoted by τ_k^i where $\tau_{i\,\min} \leq \tau_k^i \leq \tau_{i\,\max}$, $\forall k \in \mathbb{N}, i \in \{1, \cdots, m\}$. We define $t_k^i := s_k^i + \tau_k^i$ which is the time instant that the value of u_i is updated. The overall system can be written as an impulsive system of the form

$$\dot{\xi}(t) = F\xi(t), \qquad t \neq t_k^i, \forall k \in \mathbb{N}, i \in \{1, \cdots, m\} \tag{5.2a}$$

$$\xi(t_k) = \begin{bmatrix} x^-(t_k^i) \\ \cdots\cdots \\ z_1^-(t_k^i) \\ \vdots \\ y_i(s_k^i) \\ \vdots \\ z_m^-(t_k^i) \end{bmatrix}, \qquad t = t_k^i, \forall k \in \mathbb{N}, i \in \{1, \cdots, m\}, \tag{5.2b}$$

where

$$F := \begin{bmatrix} A & \vdots & B \\ \cdots & \cdots & \cdots \\ 0 & \vdots & 0 \end{bmatrix}, \qquad \xi(t) := \begin{bmatrix} x(t) \\ \cdots\cdots \\ z(t) \end{bmatrix}, \qquad z(t) := \begin{bmatrix} z_1(t) \\ \vdots \\ z_m(t) \end{bmatrix},$$

so we have $z_i(t) := y_i(s_k^i), t \in [t_k^i, t_{k+1}^i)$.

5.2 NCSs modeled by MIMO system (5.2)

The impulsive system (5.2) can be used to represent the distributed sensors/actuators configurations shown in Figs. 5.2, and 5.3. The LTI plant has the state space of the form

$$\dot{x}_p(t) = A_p x_p(t) + B_p u_p(t), \qquad y_p(t) = C_p x_p(t), \tag{5.3}$$

where $x_p \in \mathbb{R}^{n_p}, u_p := \begin{bmatrix} u'_{p1} & \cdots & u'_{pm_c} \end{bmatrix}' \in \mathbb{R}^{m_c}$, and $y_p := \begin{bmatrix} y'_{p1} & \cdots & y'_{pm_p} \end{bmatrix} \in \mathbb{R}^{m_p}$ are the state, input and output of the plant and matrices A_p, B_p, C_p, D_p have the proper dimensions. At time $s_k^i, i \in \{1, \cdots, m_p\}$, sensor i sends $y_{pi}(s_k^i)$ to the controller, which arrives at the destination at time t_k^i. When a new measurement of the sensor i arrives at the controller side, the corresponding value at the hold block, z_i, is updated and held constant until another measurement of the sensor i arrives (all other hold values remain unchanged when the value of hold i is updated). Hence $u_{ci}(t) = z_i(t) = y_{pi}(s_k^i)$, $t \in [t_k^i, t_k^{i+1})$ for $\forall i \in \{1, \cdots, m_p\}$. An output feedback

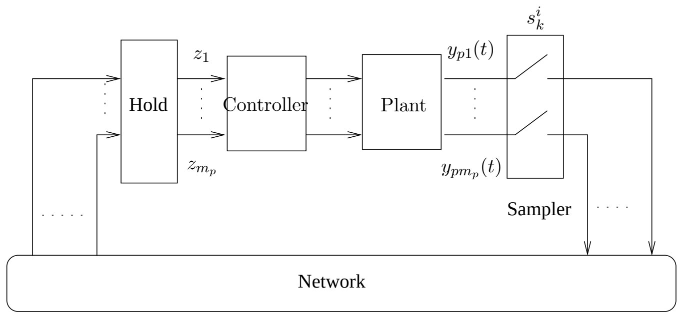


Figure 5.2. One channel NCSs with the plant (5.3) and the controller (5.4).

controller (or a state feedback controller) uses the measurements to construct the control signal. The controller has the state space of the form

$$\dot{x}_c(t) = A_c x_c(t) + B_c u_c(t), \qquad y_c(t) = C_c x_c(t) + D_c u_c(t), \tag{5.4}$$

where $x_c \in \mathbb{R}^{n_c}, u_c := \left[\begin{smallmatrix} u'_{c1} & \cdots & u'_{cm_p}\end{smallmatrix}\right]' \in \mathbb{R}^{m_p}, y_c := \left[\begin{smallmatrix} y'_{c1} & \cdots & y'_{cm_c}\end{smallmatrix}\right] \in \mathbb{R}^{m_p \times 1} \in \mathbb{R}^{m_c \times 1}$ are the state, input and output of the controller and matrices A_c, B_c, C_c, D_c have the proper dimensions. The main difference between the different NCS configurations in this section is the control signal construction.

5.2.1 One-channel NCS with dynamic feedback controller

Fig. 5.2 shows a one-channel NCS consisting of a plant with the state-space (5.3) in feedback with a dynamic output controller with state-space (5.4). Note that in one-channel NCS the controller is directly connected to the plant and only the measurements of the plant are sent through the network. To match the system in Fig. 5.2 with the system in Fig. 5.1 $y_i(t) := y_{pi}(t)$, $m := m_p$, and $x := \left[\begin{smallmatrix} x_p \\ x_c \end{smallmatrix}\right]$. The closed-loop system can be written as the impulsive system (5.2) where

$$A := \begin{bmatrix} A_p & B_p C_c \\ 0 & A_c \end{bmatrix}, \quad B := \begin{bmatrix} B_p D_c \\ B_c \end{bmatrix}, \quad C := \begin{bmatrix} C_p & 0 \end{bmatrix}. \tag{5.5}$$

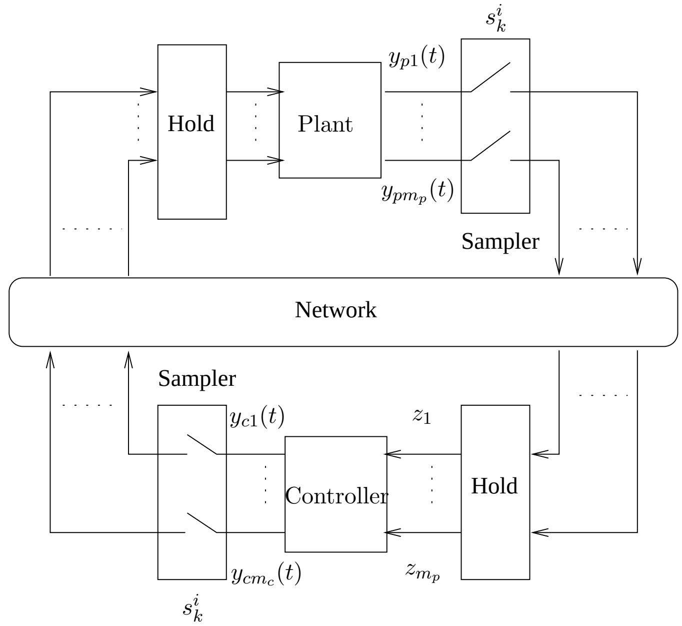


Figure 5.3. Two channel NCS with the plant (5.3) and the anticipative or non-anticipative controller (5.4).

One-channel NCSs may look artificial since the controller and the plant are on one side. One may suggest that there is no need to send the output of the plant through the network since the plant and the controller are physically close to each other and on one side of the network. In fact there are interesting cases of NCSs that can be modeled as a one-channel NCSs. One example is controlling a robot using cameras that are not mounted on the robot which provide the global image of the field. In this case position of the robot is "measured" by cameras and the measurements are sent through the network to the robot to be used to compute the control commands by the local controller of the robot.

5.2.2 Two-channel NCS with non-anticipative controller

Fig. 5.3 shows a two-channel NCS consisting of a plant with the state-space (5.3) in feedback with a *non-anticipative* controller with state-space (5.4) where $D_c = 0$. Non-anticipative controllers are simply output-feedback controllers which a single value control command is calculated. Now the controller is located away from the actuators and the control commands should be sent through the network. The control signal for the actuator i, $y_{ci}(t)$, is sampled at s_k^i, , $i \in [m_p+1, \cdots, m_p+m_c]$, and samples get to the actuator i at $t_k^i := s_k^i + \tau_k^i$. Note that a non-anticipative control unit sends a single-value control command to be applied to the actuator i of the plant and held until the next control update of the actuator i arrives (all other actuator values remain unchanged while the value of actuator i is updated). Hence $u_{pi}(t) = z_i(t) = y_{ci}(s_k^i)$, $t \in [t_k^i, t_{k+1}^i)$ where $i \in [m_p+1, \cdots, m_p+m_c]$. So in this case for the system (5.2) in Fig. 5.1

$$y_i := \begin{cases} y_{pi}, & 1 \leq i \leq m_p \\ y_{ci}, & m_p + 1 \leq i \leq m_p + m_c \end{cases},$$

$m := m_p + m_c$ and $x := \left[\begin{smallmatrix} x_p \\ x_c \end{smallmatrix}\right]$. The closed-loop system with state can be written as the impulsive system (5.2) where

$$A := \begin{bmatrix} A_p & 0 \\ 0 & A_c \end{bmatrix}, \qquad B := \begin{bmatrix} 0 & B_p \\ B_c & 0 \end{bmatrix}, \qquad C := \begin{bmatrix} C_p & 0 \\ 0 & C_c \end{bmatrix}. \tag{5.6}$$

Two-channel NCS is the most common NCSs where the plant and the controller are spatially distributed and connected through a network.

5.2.3 Two-channel NCS with anticipative controller

Fig. 5.3 can represent a two-channel NCS consisting of a plant with the state-space (5.3) in feedback with an *anticipative* controller with state-space (5.4). Anticipative controller attempts to compensate the sampling and delay introduced by the actuation channels. For simplicity, we assume that the actuation channels are sampled with constant sampling intervals $h = s^i_{k+1} - s^i_k$, and that the delay in the actuation channels is constant and equal to $\tau = \tau^i_k \forall k \in \mathbb{N}, i \in \{m_p + 1, \cdots, m_p + m_c\}$. At each sampling time s^i_k, $i \in [m_p + 1, \cdots, m_p + m_c]$ the controller sends a time-varying control signal $y_{ci}(\cdot)$ to the actuator i. This control signal should be used from the time $s^i_k + \tau$ at which it arrives until the time $s^i_k + h + \tau$ at which the next control update of the actuator i will arrive. This leads to

$$u_{pi}(t) = y_{ci}(t), \quad \forall t \in [s^i_k + \tau, s^i_k + h + \tau).$$

However, the prediction of control signal $y_{ci}(t)$ needed in the interval $[s^i_k + \tau, s^i_k + h + \tau)$ must be available at the transmission time s^i_k, which requires the control unit to calculate the control signal up to $h + \tau$ time units into the future.

Remark 8. Anticipative controllers send actuation signals to be used during time intervals of duration h, therefore the sample and hold blocks in Fig. 5.3 should be understood in a broad sense. In practice, the sample block would send over the network some parametric form of the control signal $u_{ci}(\cdot)$ (e.g., the coefficients of a polynomial approximation to this signal).

Remark 9. Anticipative controllers are similar to predictive controllers in the sense that both calculate the future control actions. However in predictive controllers only the most recent control prediction is applied until the new control commands arrive. Anticipative controllers are predictive controllers that send a control prediction for a certain duration. At the expense of sending more information to the actuators in each packet, one expects that fewer packets need to be transmitted to stabilize the system.

An estimate $\hat{x}_c(t)$ of $x_c(t+h+\tau)$ is constructed as follows:

$$\dot{\hat{x}}_c(t) = A_c\hat{x}_c(t) + B_c u_c(t), \tag{5.7}$$

where

$$u_{ci}(t) = y_{pi}(s_k^i),\ \forall t \in [s_k^i + \tau, s_k^i + \tau + h), \tag{5.8}$$

for $i \in \{1, \cdots, m_p\}$. To compensate for the time varying delay and sampling intervals in the actuation channels, $\hat{x}_c$ would have to be calculated further into the future. Hence the assumptions of constant delay and sampling interval for actuation channel can be relaxed by predicting x_c further into the future. With such a controller state prediction available, the signal $y_{ci}(t)$ sent at times s_k^i, to be used in $\left[s_k^i + \tau, s_k^i + h + \tau\right)$, is then given by

$$y_{ci}(t) = C_{ci}\hat{x}_c(t-h-\tau) + D_c u_{ci}(t),\ \forall t \in [s_k^i + \tau, s_k^i + h + \tau), \tag{5.9}$$

which only requires the knowledge of $\hat{x}_c(.)$ in the interval $t \in \left[s_k^i - h, s_k^i\right)$, and therefore is available at the transmission times s_k^i. The closed-loop system can be written as

$$\begin{bmatrix} \dot{\hat{x}}_p(t) \\ \dot{\hat{x}}_c(t) \end{bmatrix} = \begin{bmatrix} A_p & B_pC_c \\ 0 & A_c \end{bmatrix} \begin{bmatrix} \hat{x}_p(t) \\ \hat{x}_c(t) \end{bmatrix} + \begin{bmatrix} B_pD_c \\ B_c \end{bmatrix} u_c(t), \tag{5.10}$$

where $\hat{x}_p(t) := x_p(t+h+\tau)$ and the elements of $u_c(t)$ are defined by (5.8). Hence in this case $y_i(t) := y_{pi}(t), \quad m := m_p$ and the closed-loop system with state $x := \left[\begin{smallmatrix}\hat{x}_p \\ \hat{x}_c\end{smallmatrix}\right]$ can be written as the impulsive system (5.2) where

$$A := \begin{bmatrix} A_p & B_p C_c \\ 0 & A_c \end{bmatrix}, \quad B := \begin{bmatrix} B_p D_c \\ B_c \end{bmatrix}, \quad C := \begin{bmatrix} C_p & 0 \end{bmatrix}. \tag{5.11}$$

Remark 10. Equation (5.5) that represents one-channel NCS with dynamic output feedback is similar to equation (5.11) that represents two-cannel NCS with anticipative controller. Consequently for analysis purpose one can model a two-cannel NCS with anticipative controller as a one-channel NCS with *"fictitious"* delays equal to the sum of the delay in the sensor to actuator channels, the delay in the actuator to sensor channels, and the sampling of the actuator channel.

5.3 Exponential stability of NCSs

In this section we provide exponential stability conditions for the linear impulsive system in (5.2) which can model one-channel NCSs and two-channel NCSs with anticipative or non-anticipative controller as described in Section 5.2.

Since the minimum of delay in the network is typically small and for simplicity of derivations, we assume that $\tau_{i\,\min} = 0$, $1 \le i \le m$. We now present two theorems for the stability of the system (5.2). The stability tests are based on the Theorem (15) which provides stability conditions for system (4.5). Note that system (5.2) is a special case of (4.5). The first theorem is less conservative; however, the number of LMIs grows exponentially with m. The second stability condition is based on the feasibility of a single LMI, but its size grows linearly with m. For small m the first stability test is more adequate because it leads to less conservative results, but the second stability test is more desirable for large m. We present our results for $m = 2$, but deriving the stability conditions for other values of m is straightforward by following the same steps.

Inspired by the Lyapunov functional (4.18), We employ the Lyapunov functional

$$V := V_1 + V_2 + V_3 + V_4, \tag{5.12}$$

where

$$\begin{aligned}
V_1 &:= x'Px, \\
V_2 &:= \sum_{i=1}^{2} \int_{t-\rho_i}^{t} (\rho_{i\max} - t + s)\dot{y}_i'(s) R_{1i} \dot{y}_i(s) ds, \\
V_3 &:= \sum_{i=1}^{2} \int_{t-\sigma_i}^{t} (\sigma_{i\max} - t + s)\dot{y}_i'(s) R_{2i} \dot{y}_i(s) ds, \\
V_4 &:= \sum_{i=1}^{2} (\rho_{i\max} - \rho_i)(y_i - w_i)' X_i (y_i - w_i),
\end{aligned}$$

with P, R_{1i}, R_{2i}, X_i symmetric positive definite matrices and

$$\begin{aligned}
&\rho_i(t) := t - s_k^i,\ \sigma_i(t) := t - t_k^i,\ w_i(t) := y_i(t_k^i), \quad t \in [t_k^i, t_{k+1}^i), i \in \{1, \cdots, m\} \\
&\rho_{i\max} := \sup_{t \geq 0}\ \rho_i(t), \quad \sigma_{i\max} := \sup_{t \geq 0}\ \sigma_i(t).
\end{aligned}$$

The next theorem guarantees that the Lyapunov functional (5.12) decreases along the solution to the system (5.2).

Theorem 18. *The system* (5.2) *is exponentially stable for any delay and sampling interval satisfying* $0 \leq \tau_k^i \leq \tau_{i\max}$ *and* $s_{k+1}^i + \tau_{k+1}^i - s_k^i \leq \rho_{i\max}$*,* $\forall k \in \mathbb{N}, i \in \{1, \cdots, m\}$ *if there exist symmetric positive definite matrices* P, R_{1i}, R_{2i}, X_i *and* *(not necessarily symmetric) matrices N_{1i}, N_{2i} that satisfy the following LMIs:*

$$\begin{bmatrix} M_1+\rho_{1\max}(M_{21}+M_{31})+\rho_{2\max}(M_{22}+M_{32}) & \tau_{1\max}N_{11} & \tau_{2\max}N_{12} \\ * & -\tau_{1\max}R_{11} & 0_{q_1q_2} \\ * & * & -\tau_{2\max}R_{12} \end{bmatrix} < 0, \tag{5.13a}$$

$$\begin{bmatrix} M_1+\rho_{1\max}M_{21}+\rho_{2\max}(M_{22}+M_{32}) & \tau_{1\max}N_{11} & \tau_{2\max}N_{12} & G_{11} \\ * & -\tau_{1\max}R_{11} & 0_{q_1q_2} & 0_{q_1q_1} \\ * & * & -\tau_{2\max}R_{12} & 0_{q_2q_1} \\ * & * & * & G_{21} \end{bmatrix} < 0, \tag{5.13b}$$

$$\begin{bmatrix} M_1+\rho_{1\max}(M_{21}+M_{31})+\rho_{2\max}M_{22} & \tau_{1\max}N_{11} & \tau_{2\max}N_{12} & G_{12} \\ * & -\tau_{1\max}R_{11} & 0_{q_1q_2} & 0_{q_1q_2} \\ * & * & -\tau_{2\max}R_{12} & 0_{q_2q_2} \\ * & * & * & G_{22} \end{bmatrix} < 0, \tag{5.13c}$$

$$\begin{bmatrix} M_1+\rho_{1\max}M_{21}+\rho_{2\max}M_{22} & \tau_{1\max}N_{11} & \tau_{2\max}N_{12} & G_{11} & G_{12} \\ * & -\tau_{1\max}R_{11} & 0_{q_1q_2} & 0_{q_1q_1} & 0_{q_1q_2} \\ * & * & -\tau_{2\max}R_{12} & 0_{q_2q_1} & 0_{q_2q_2} \\ * & * & * & G_{21} & 0_{q_1q_2} \\ * & * & * & * & G_{22} \end{bmatrix} < 0, \tag{5.13d}$$

where $\bar{F} := \begin{bmatrix} A & B & 0_{nq} \end{bmatrix}$ *and*

$$M_1 := \bar{F}' \begin{bmatrix} P & 0_{nq} & 0_{nq} \end{bmatrix} + \begin{bmatrix} P \\ 0_{qn} \\ 0_{qn} \end{bmatrix} \bar{F} - T_1'X_1T_1 - T_2'X_2T_2 - N_{11}T_3$$

$$- T_3'N_{11}' - N_{21}T_1 - T_1'N_{21}' - N_{12}T_4 - T_4'N_{12}' - N_{22}T_2 - T_2'N_{22}',$$

$$M_{2i} := \bar{F}'C_i'(R_{1i}+R_{2i})C_i\bar{F}, \qquad M_{3i} := T_i'X_iC_i\bar{F} + \bar{F}'C_i'X_iT_i,$$

$$G_{1i} := \rho_{i\max}(N_{1i}+N_{2i}), \qquad G_{2i} := \rho_{i\max}(R_{1i}+R_{2i}) \tag{5.14}$$

with

$$C_1 := \begin{bmatrix} I_{q_1} & 0_{q_1q_2} \end{bmatrix} C, \qquad C_2 := \begin{bmatrix} 0_{q_2q_1} & I_{q_2} \end{bmatrix} C,$$

$$T_1 := \begin{bmatrix} C_1 & 0_{q_1q} & -\bar{I}_1 \end{bmatrix}, \qquad T_2 := \begin{bmatrix} C_2 & 0_{q_2q} & -\bar{I}_2 \end{bmatrix},$$

$$T_3 := \begin{bmatrix} C_1 & -\bar{I}_1 & 0_{q_1q} \end{bmatrix}, \qquad T_4 := \begin{bmatrix} C_2 & -\bar{I}_2 & 0_{q_2q} \end{bmatrix},$$

$$\bar{I}_1 := \begin{bmatrix} I_{q_1} & 0_{q_1q_2} \end{bmatrix} \qquad \bar{I}_2 := \begin{bmatrix} 0_{q_2q_1} & I_{q_2} \end{bmatrix} \tag{5.15}$$

□

Proof of Theorem 18. After taking the derivative of the Lyapunov functional (5.12) and following the same steps as in the proof of Theorem 16 we conclude that the derivative is negative as long as

$$M_1 + \sum_{i=1}^{2} \rho_{i\,\max}(M_{2i} + M_{3i}) + \rho_i(M_{4i} - M_{3i}) < 0, \quad \forall \rho_i \in [0, \rho_{i\,\max}]. \tag{5.16}$$

Then we can prove that (5.16) is equivalent to (5.13). ■

It is possible to generalize Theorem 18 for an arbitrary m. However, the number of LMIs is 2^m and the size of LMIs and the number of scalar variables increases linearly. For complex systems with large number of sending nodes, to have a numerically tractable test, it is crucial that the number of LMIs grows linearly too. The next theorem presents another stability test which is more conservative; however, the stability test is based on the feasibility of a single LMI.

Theorem 19. *The system* (5.2) *is exponentially stable for any delay and sampling interval satisfying* $0 \le \tau_k^i \le \tau_{i\,\max}$ *and* $s_{k+1}^i + \tau_{k+1}^i - s_k^i \le \rho_{i\,\max}$, $\forall k \in \mathbb{N}, i \in \{1, \cdots, m\}$ *provided that there exist symmetric positive definite matrices* P, R_{1i}, R_{2i} *and (not necessarily symmetric) matrices* N_{1i}, N_{2i} *that satisfy the following LMIs:*

$$\begin{bmatrix} \bar{M}_1 + \rho_{1\,\max}\bar{M}_{21} + \rho_{2\,\max}\bar{M}_{22} & \tau_{1\,\max}N_{11} & \tau_{2\,\max}N_{12} & G_{11} & G_{12} \\ * & -\tau_{1\,\max}R_{11} & 0_{q_1q_2} & 0_{q_1q_1} & 0_{q_1q_2} \\ * & * & -\tau_{2\,\max}R_{12} & 0_{q_2q_1} & 0_{q_2q_2} \\ * & * & * & G_{21} & 0_{q_1q_2} \\ * & * & * & * & G_{22} \end{bmatrix} < 0, \tag{5.17}$$

where $\bar{F} := \begin{bmatrix} A & B & 0_{nq} \end{bmatrix}$ *and*

$$\begin{aligned} \bar{M}_1 :=& \bar{F}' \begin{bmatrix} P & 0_{nq} & 0_{nq} \end{bmatrix} + \begin{bmatrix} P \\ 0_{qn} \\ 0_{qn} \end{bmatrix} \bar{F} - N_{11}T_3 - T_3'N_{11}' - N_{21}T_1 \\ & -T_1'N_{21}' - N_{12}T_4 - T_4'N_{12}' - N_{22}T_2 - T_2'N_{22}', \\ \bar{M}_{2i} :=& \bar{F}'C_i'(R_{1i} + R_{2i})C_i\bar{F}, \\ G_{1i} :=& \rho_{i\,\max}(N_{1i} + N_{2i}), \qquad G_{2i} := \rho_{i\,\max}(R_{1i} + R_{2i}). \end{aligned}$$

with the matrix variables defined in (5.15). □

Proof of Theorem 19. After taking the time derivative of the Lyapunov functional and following the steps in Theorem 16, we conclude that the derivative is negative if (5.16) holds. Equation (5.16) with $\rho_i = \rho_{i\max}$ is a sufficient condition for (5.16). So we obtain

$$M_1 + \sum_{i=1}^{2} \rho_{i\max}(M_{2i} + M_{4i}) < 0,$$

and by applying Schur Lemma we get (5.17). ∎

By solving the LMIs in Theorems 18 or 19 for general m one can find positive constants $\rho_{i\max}, 1 \leq i \leq m$ which determines the upper bound between the *consecutive samples of channel* i for which the stability of the closed-loop system is guaranteed for a given lower and upper bound on the total delay in the loop i.

However, most of the work in the literature has been devoted to finding τ_{MATI} ([67, 68, 45, 46, 20] and references therein) which determines the upper bound between *any consecutive sampling instances* which the stability of the closed-loop system is guaranteed while the usual assumption for MIMO case is that delays in all loops are small and hence negligible. One can expect that having m constants $\rho_{i\max}$, $1 \leq i \leq m$ instead of one single constant τ_{MATI}, reveals more information about the system as we explore in the following examples.

	no drop
Maximum deterministic time interval between samples from [63]	0.0123
Maximum stochastic arbitrary inter-sampling time distribution from [21]	0.0279
Maximum stochastic uniform inter-sampling time distribution from [21]	0.0517
Maximum deterministic time interval between samples from Theorem 18	0.0405
Maximum deterministic time interval between samples from Theorem 19	0.029

Table 5.1. Comparison of τ_{MATI} for Example 6 when there is no delay.

5.4 Example

Example 6. This example appeared in [68, 45, 21, 63] and considers a linearized model of the form (5.3) for a two-input, two-output batch reactor where

$$A_p := \begin{bmatrix} 1.38 & -0.2077 & 6.715 & -5.676 \\ -0.5814 & -4.29 & 0 & 0.675 \\ 1.067 & 4.273 & -6.654 & 5.893 \\ 0.048 & 4.273 & 1.343 & -2.104 \end{bmatrix}, \quad B_p := \begin{bmatrix} 0 & 0 \\ 5.679 & 0 \\ 1.136 & -3.146 \\ 1.136 & 0 \end{bmatrix},$$

$$C_p := \begin{bmatrix} 1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

This system is controlled by a PI controller of the form (5.4) where

$$A_c := \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \qquad B_c := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

$$C_c := \begin{bmatrix} -2 & 0 \\ 0 & 8 \end{bmatrix}, \qquad D_C := \begin{bmatrix} 0 & -2 \\ 5 & 0 \end{bmatrix}.$$

Following the assumptions of [68, 45, 21, 63], we assume that only the outputs of the plant are transmitted over the network, there are no dropouts and the outputs are sent in a round-robin fashion and consecutively. We compare τ_{MATI} of this example given by the stability results in [68, 45, 21, 63], where in all the references the effect of the delay is ignored. From Theorem 18 we compute $\rho_{1\max} = 0.081, \rho_{2\max} = 0.113$ when there is no delay. We can show that if the upper bound between any consecutive sampling, τ_{MATI}, is smaller than $\frac{1}{2}\min(\rho_{1\max}, \rho_{2\max})$, then the upper bound between the samples of y_{p1} or y_{p2}

	no drop
Maximum deterministic time interval between samples from [63]	$2.81 * 10^{-4}$
Maximum stochastic arbitrary inter-sampling time distribution from [21]	$8.02 * 10^{-4}$
Maximum stochastic uniform inter-sampling time distribution from [21]	$1.48 * 10^{-3}$
Maximum deterministic time interval between samples from Theorem 18	$1.2 * 10^{-3}$

Table 5.2. Comparison of τ_{MATI} for Example 7 when there is no delay.

are smaller than $\min(\rho_{1\max}, \rho_{2\max})$ and the system is stable, so we get $\tau_{MATI} = \frac{1}{2}\min(\rho_{1\max}, \rho_{2\max}) = 0.0405$. Table 5.1 shows the less conservative results in the literature and our τ_{MATI} for comparison. τ_{MATI} for a (stochastic) uniform inter-sampling time distribution given by [21] is less conservative than τ_{MATI} given by (18). However, for a fair comparison our result should be compared to the stochastic arbitrary inter-sampling time distribution given by [21]. If we can send the measurements of y_{1p} and y_{2p} in one packet then $\tau_{MATI} = \min(\rho_{1\max}, \rho_{2\max}) = 0.081$, because the requirement for the stability that the consecutive samplings of y_{1p} and y_{2p} are smaller than $\rho_{1\max}$ and $\rho_{2\max}$ respectively. When the maximum delay is 0.03 then $\rho_{1\max} = 0.058$ and $\rho_{2\max} = 0.087$. □

In general if m nodes send their measurements in different packets then $\tau_{MATI} = \frac{1}{m}\min_i \rho_{i\max}$ (given that there is no delay).

Example 7. This example appeared in [21, 63] and considers the control of a CH-47 tandem-rotor helicopter in the horizontal plane, around a nominal air speed of 40 knots which can be modeled by (5.3) where

$$A_p := \begin{bmatrix} -0.02 & 0.005 & 2.4 & -32 \\ -0.14 & 0.44 & -1.3 & -30 \\ 0 & 0.018 & -1.6 & 1.2 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad B_p := \begin{bmatrix} 0.14 & -0.12 \\ 0.36 & -8.6 \\ 0.35 & 0.009 \\ 0 & 0 \end{bmatrix}, \quad C_p := \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 57.3 \end{bmatrix}.$$

This system is controlled by a static controller where

$$D_c := \begin{bmatrix} -12.7177 & -45.0824 \\ 63.5163 & 25.9144 \end{bmatrix}$$

From Theorem 18 we compute $\rho_{1\max} = 0.0024, \rho_{2\max} = 0.0028$ when there is no delay. Then we compute τ_{MATI} as described in the previous example to compare our results to the ones in the literature. Table 5.2 summarizes the results obtained from [21, 63] and Theorem 18. $\square$

Chapter 6

Communication Protocol Design For Deterministic Networks

In Chapter 4 we considered NCSs that can be modeled as SISO delay impulsive systems and in Chapter 5 we considered NCSs that can be modeled as MIMO delay impulsive systems. In both cases, we found the upper bound on the delays and the sampling intervals corresponding to the outputs of the system sent through the network, such that exponential stability of the closed-loop system is guaranteed.

The framework we have developed can be employed to analyze the stability of NCSs in which a *deterministic network* is used to connect different elements of the system. In deterministic networks, the access to the network may not be predetermined but the latest delivery time of a packet can be computed. Examples of such networks include token-passing networks, Control Area Networks (CAN), switched networks, and RayFlex. However, Ethernet or wireless networks are not deterministic networks since the delays in the network are stochastic and can be

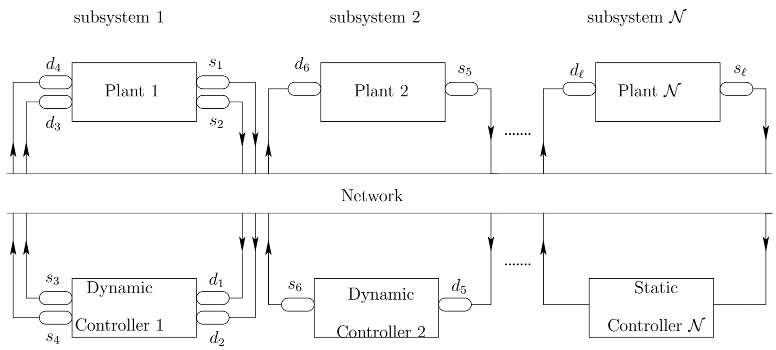


Figure 6.1. Schematic description of an NCS with $\mathcal{N}$ different subsystems consisting of a plant and a controller. subsystem 1 consists of a MIMO plant and a MIMO dynamic controller, subsystem 2 consists of a SISO plant and a dynamic controller, and subsystem $\mathcal{N}$ consists of a SISO plant with a static controller. We refer to a path between a sampler-hold pair as a connection and we assume that there are ℓ connections. A subsystem with one connection, e.g. subsystem $\mathcal{N}$, is modeled by a SISO delay impulsive system and a subsystem with more than one connection, e.g. subsystem 1 is modeled by a MIMO delay impulsive system. Note that the controller of subsystem 2 is a dynamic controller so subsystem 2 consists of two connections but the controller of subsystem $\mathcal{N}$ is a static controller so subsystem $\mathcal{N}$ consists of one connection.

unbounded (for instance due to packet collision).

The schematic description of the NCS we consider in this chapter is given in Fig. 6.1. We consider $\mathcal{N}$ subsystems, each consisting of a plant and a controller connected through the network. The overall system consists of ℓ connections, where a *connection* is a path between a sampler and its corresponding hold. Through the analysis in previous chapters, we can find sets $\mathcal{S}_i, i \in \{1, \cdots, \ell\}$ of admissible sampling-delay sequences $(\{s_k\},\{\tau_k\})$ such that [1]

$$s^i_{k+1} - s^i_k \leq \rho_{i\max}, \qquad\qquad \tau^i_k \leq \tau_{i\max}, \tag{6.1}$$

so that if every sampling-delay sequence $(\{s^i_k\},\{\tau^i_k\})$ of every connection i belongs to $\mathcal{S}_i$, exponential stability of all subsystems is guaranteed. Suppose now that the upper bounds on the sampling intervals of all connections, $\rho_{i\max}, i \in \{1,\cdots \ell\}$, are given, then one can find $\tau_{i\max}$ (from Theorem 16 for the SISO case and Theorem 18 or 19 for the MIMO case with $\tau_{i\min} = 0$) such that if the total delay in each connection i is smaller than $\tau_{i\max}$ for all $i \in \{1,\cdots,\ell\}$, then exponential stability of all subsystems is guaranteed. The main question addressed in this chapter is whether or not the network can deliver all packets for all connections before their *deadlines* $\tau_{i\max}$.

To answer this question we will employ results in real-time scheduling [6]. In real-time scheduling, different jobs are released periodically or the lower bounds between release times are given. In the most basic setting, one shared resource services different jobs and servicing a job takes a certain amount of time. Each job should be completed before a deadline and if all the timing requirements can be met, then the set of jobs is *schedulable*. In the context of real-time computation, typically a processor is a shared computation resource and jobs are computing desired tasks.

In the context of NCSs, the shared resource typically is a network shared between different nodes[2]. Job i refers to transmitting a packet from the source to the destination of connection i and the time required to service a job is the

[1]The definition of sets $\mathcal{S}_i$ is the same as (4.3) defined for one connection and we assume the lower bound on the delay is not available and hence it is zero.

[2]Nodes may share computation resources also. For example a processor can be used to implement two or more controllers. We consider this case in Sections 6.4 and 6.5

time needed to transfer a packet from the source to the destination corresponding to the job, which we call the transmission time. Suppose that $\rho_{i\max}, \tau_{i\max}$ are given such that the stability LMIs are satisfied for these bounds. If the set of "jobs" are schedulable with deadlines $\tau_{i\max}$ and release times smaller than $\rho_{i\max}$ for every $i \in \{1, \cdots, \ell\}$, then the completion of the job i is guaranteed before $\tau_{i\max}$. Hence any sampling-delay sequence, in fact, belongs to set $\mathcal{S}_i$ defined in (6.1) and consequently stability of all subsystems connected to the network are satisfied.

The right choice of the sampling intervals is not trivial. For stability, faster sampling is desirable so that $s^i_{k+1} - s^i_k \leq \rho_{i\max}$. However, faster sampling means higher traffic in the network which may lead to larger network delays τ^i_k to the point that delays in connections become larger than their allowable upper bound $\tau_{i\max}$ required for stability.

The goal of this chapter is two folded. Based on the analysis we provided in the previous chapter we will design the following:

- An algorithm to select sampling bounds such that the sampling-delay sequence of connection i belongs to the set S_i, so stability of all subsystems connected to the network is guaranteed.
- Communication protocols based on priorities determined by "deadlines", so that a node with a closer deadline gains the access to the network.

In Section 6.1 we review some scheduling algorithms with different message priority assignments. We mainly focus on non-preemptive scheduling because a higher priority packet cannot preempt transferring a lower priority packet in CAN networks (after arbitration phase in CAN networks). Then we focus on an Earlier

Deadline First (EDF) algorithm which has some advantages over other scheduling algorithms. The deadlines are determined by $\tau_{i\,\max}$, and based on how close a node is to its deadline, priorities are assigned. Then we provide scheduling tests for EDF scheduling and we propose a convex algorithm to choose the "right" sampling interval bounds in section 6.2.

In Section 6.3 we provide a brief review of CAN specifications, arbitration and message prioritization, and some comments about how to implement EDF scheduling on CAN networks. In Sections 6.1 and 6.2 we assume that the computation delays are very small, so we can ignore them. In Sections 6.4 and 6.5 we lift this assumption to study systems with shared computation and communication resources.

6.1 Communication protocol design and real time scheduling

Real-time scheduling theory studies the time response of servicing different jobs by shared resources. In the real-time computation, a processor is a resource that needs to be shared between different jobs. In NCSs, the shared resource is the network which is shared between different nodes. Computation resources can also be shared in NCSs, for example a processor can be used to implement two or more controllers [27]. However, we will not pursue that here and we assume that the computation delay is negligible. Moreover, we assume there is only one communication resource, so for instance we exclude nested CAN networks, which consists of CAN networks connected through *gateway nodes*. In the last section, we include some remarks that would help to study the effect of sharing many computation and communication resources. Since delay has an important role in our analysis and implementation, we identify the sources of delay in each connection.

6.1.1 Elements of delay in a connection of NCS

The total delay is the time between the moment that data is ready to be sent at the source until the data is ready to be used at the destination and is equal to the sum of the wait time to access the network and the transmission time as shown in Fig. 6.2 (we assume that the computation times at the source and the destination is zero).

- Wait time to access the network of connection i, acs_i, is the time that a node waits until it gets permission to send a packet.
- Transmission time of connection i, $trns_i$, is the time from putting the first bit to the network at the source to receive the last bit from the network at the destination i. The transmission time consists of transmission delay and propagation delay which shown by a and b in Fig. 6.2. Transmission delay is the time to put a packet on the network and propagation delay is the time that a signal travels from the source to the destination.

We can compute $trns_i$ given the network specifications in terms of data rate, packet size, and length of network. Our goal thus is to design a network access protocol so that the network access times acs_i are small enough so that

$$acs_i + trns_i \leq \tau_{i\,\max} \tag{6.2}$$

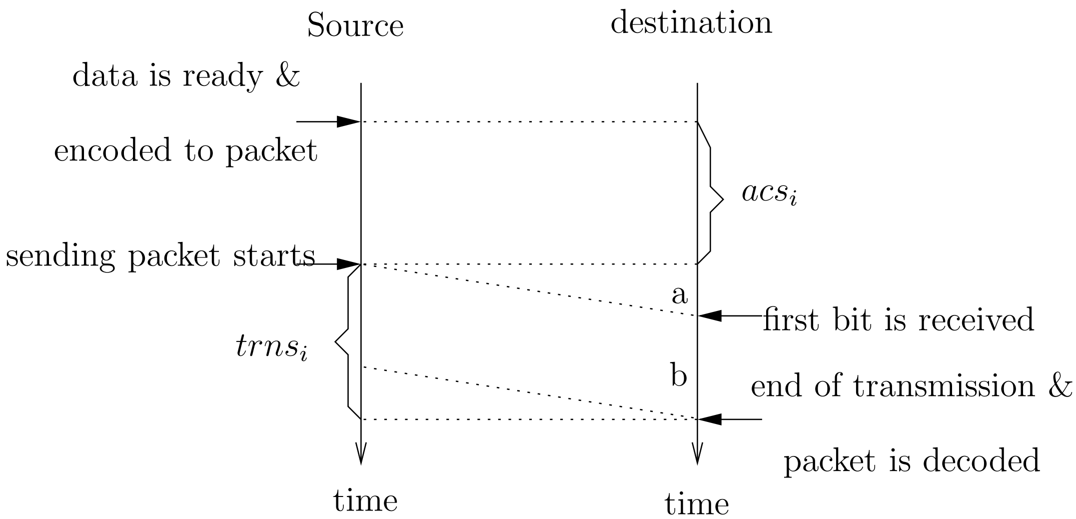


Figure 6.2. Timing diagram of connection i.

holds for every connection $i \in \{1, \cdots, \ell\}$. If, in addition, $s^i_{k+1} - s^i_k \leq \rho_{i\,\max}$ (stability LMIs are satisfied for $\tau_{i\,\max}$ and $\rho_{i\,\max}$) then the resulting delay-sampling sequence belongs to the set $\mathcal{S}_i$ and all the subsystems connected to the network are exponentially stable. The wait time, acs_i, depends on scheduling policy and priority assignment. In the next section we review scheduling policies. Then we provide conditions to determine whether or not it is possible for (6.2) to hold for a particular scheduling policy.

6.1.2 Real-time scheduling

We assume there are at most ℓ jobs each generated by only one of the ℓ source nodes. In real-time scheduling it is assumed that the lower bound between consecutive release times of all source nodes are given. In our problem we denote this lower bound by $\rho_{i\,\min}$ which corresponds to the smallest sampling interval that the source i will ever generate. Each connection $i \in \{1, \cdots, \ell\}$ is characterized by a triple $(\rho_{i\,\min}, trns_i, \tau_{i\,\max})$ consisting of the lower bound on the two consecutive release times of job i, the service time of job i (time that take the network "service" a packet sent by source i), and the deadline of job i, respectively.

Two types of scheduling can be found in the literature: non-preemptive and preemptive. In non-preemptive scheduling, when servicing a job starts it will not be interrupted to service a higher priority job. On the contrary, in preemptive scheduling, as soon as a job with a higher priority is released, the shared resource is allocated to the higher priority job and the current job with lower priority is interrupted. Preemptive scheduling is suitable for computation resource sharing, but in communication networks such as CAN, access to the network will not be granted to any other node until the current transmission is completed. Therefore we only consider non-preemptive scheduling.

There are two main priority assignments to jobs: static and dynamic. A static priority is fixed and a priori known, so it can be stored in a table. Static scheduling is simple, yet it is very inflexible to network changes, failures, and often it under-utilizes the shared resources [6]. If scheduling decisions are based on the current decision variables, it is called a dynamic scheduling. The priority assignment is harder because priorities change over time and they should be computed online; however, a dynamic priority assignment usually does not have the disadvantages of a static priority assignment. In the following we explain most common scheduling policies briefly and we refer the readers to [6] for more details.

First-come First-serve scheduling This policy serves the oldest request first so that resource allocation is based on the order of request arrivals. This policy is not suitable for NCSs application because it may serve a packet with longer deadline over a packet with shorter deadline, and consequently instability may happen. In addition, it is not implementable based on arbitration as we will explain in Chapter 6.3.

Round-Robin scheduling This is a static algorithm in which a fixed time slot is dedicated to each node. For example in every one second slot a node i is allowed to send packets through the network in the first 0.1 seconds. This policy is simple and effective when:

- All nodes have data most of the time.
- All nodes are synchronized.
- Network structure is fixed so no new node joins after the slot time is assigned to the nodes.

If, for any reason, a node loses its turn to send data, no matter how close its deadline is, it should wait until its next allocated time slot.

Deadline Monotonic (DM) scheduling This policy allocates the network resource to nodes according to their deadlines. A task with the *shortest deadline*, (smallest $\tau_{i\max}$) is assigned the highest priority and it is a static policy. For example if $\tau_{1\max} = 3$ and $\tau_{2\max} = 4$ then the packets of connection one will always have higher priority over packets of connection two.

Earliest Deadline First (EDF) scheduling EDF is a dynamic algorithm that assigns priorities to tasks according to their *absolute deadlines*, which are the times remaining to miss the deadline. A packet with the earliest absolute deadline, $1/(t_{il} + \tau_{i\max} - t)$ will have the highest priority, where t_{il} is the last sampling time of connection i and t is the current time. Again consider node one with $\tau_{1\max} = 3, t_{1l} = 2$ and $\tau_{2\max} = 4, t_{2l} = 0$. If both nodes have a packet ready at time $t = 3$, node two gains access to the network because a packet must be received by the destination of the connection 2 before time 4 and by the destination of connection 1 before time 5 so the node 2 has a closer deadline.

For control applications, DM and EDF are the most desirable and, between these two, EDF has the advantages of being a dynamic algorithm. The disadvantage of EDF is that the priority of the packet is a function of time and should be updated periodically, which requires spending more computation power (in order to "save" the "communication power"). We only consider EDF in this Chapter; however, modifying the results to make them suitable for DM is fairly straightforward.

If the conditions in the next theorem hold for a given set of jobs, then the set of job is schedulable under EDF policy, and by schedulable under policy X we mean that (6.2) holds for every connection $i \in \{1, \cdots, \ell\}$ when the priorities are assigned according to policy X. This theorem is based on [77] which considers packet-switched networks.

Theorem 20. *A set of connections* $(\rho_{i\min}, trns_i, \tau_{i\max}), i = \{1, \cdots, \ell\}$ *is schedulable over a network under the (non-preemptive) EDF scheduling policy if and only if the following hold:*

$$\sum_{i=1}^{\ell} \frac{trns_i}{\rho_{i\min}} \leq 1, \tag{6.3}$$

$$\sum_{i=1}^{\ell} \lfloor \frac{t - \tau_{i\max}}{\rho_{i\min}} \rfloor^{+} trns_i + trns_{\max} \leq t, \qquad \forall t \in \cup_{i=1}^{\ell} S_i, \tag{6.4}$$

where $trns_{\max} := \max_i trns_i$

$$S_i := \Big\{\tau_{i\max} + n\rho_{i\min} : n = 0, 1, \cdots, \lfloor \frac{d_{\max} - \tau_{i\max}}{\rho_{i\min}} \rfloor \Big\},$$

$$d_{\max} := \max\Big\{\tau_{1\max}, \cdots, \tau_{\ell\max}, \frac{\sum_{i=1}^{\ell}(1 - \tau_{i\max}/\rho_{i\min})trns_i + trns_{\max}}{1 - \sum_{i=1}^{\ell} trns_i/\rho_{i\min}}\Big\},$$

$\lfloor . \rfloor$ *is a floor function,* $\lfloor x \rfloor^+ := \lfloor x + 1 \rfloor$ *for* $x \geq 0$ *and zero otherwise.* □

The condition (6.3) ensures that the maximum utilization does not exceed the capacity of the network and the condition (6.4) ensures that all the deadlines are met. The deadlines $\tau_{i\max}$ can be smaller or larger than $\rho_{i\min}$ but if $\rho_{i\min} = \tau_{i\max}$ for $i \in \{1, \cdots, \ell\}$ then the condition (6.4) is always satisfied.

The set S_i can become large and consequently checking the condition (6.4) requires a lot of effort. The next theorem inspired by [77] overcomes this burden but it only provides a sufficient condition for scheduling. In addition the next result is also more convenient for the numerical search for sampling intervals considered in the next section. We assume that the triples $(\rho_{i\min}, trns_i, \tau_{i\max}), i = \{1, \cdots, \ell\}$ are ordered based on deadlines such that $\tau_{j\max} \leq \tau_{k\max}, \forall j \leq k$.

Theorem 21. *A set of connections* $(\rho_{i\min}, trns_i, \tau_{i\max})$, $i = \{1, \cdots, \ell\}$ *is schedulable over a network under the (non-preemptive) EDF scheduling policy if the following hold:*

$$\sum_{i=1}^{\ell} \frac{trns_i}{\rho_{i\min}} < 1, \tag{6.5}$$

$$\sum_{i=1}^{k}(1 + \frac{\tau_{k\max} - \tau_{i\max}}{\rho_{i\min}})trns_i + trns_{\max} \leq \tau_{k\max}, \qquad \forall k \in \mathcal{K} \tag{6.6}$$

where $\mathcal{K} := \{1, \cdots \ell\} - \{k : \tau_{k\max} = \tau_{(k+1)\max}\}$. □

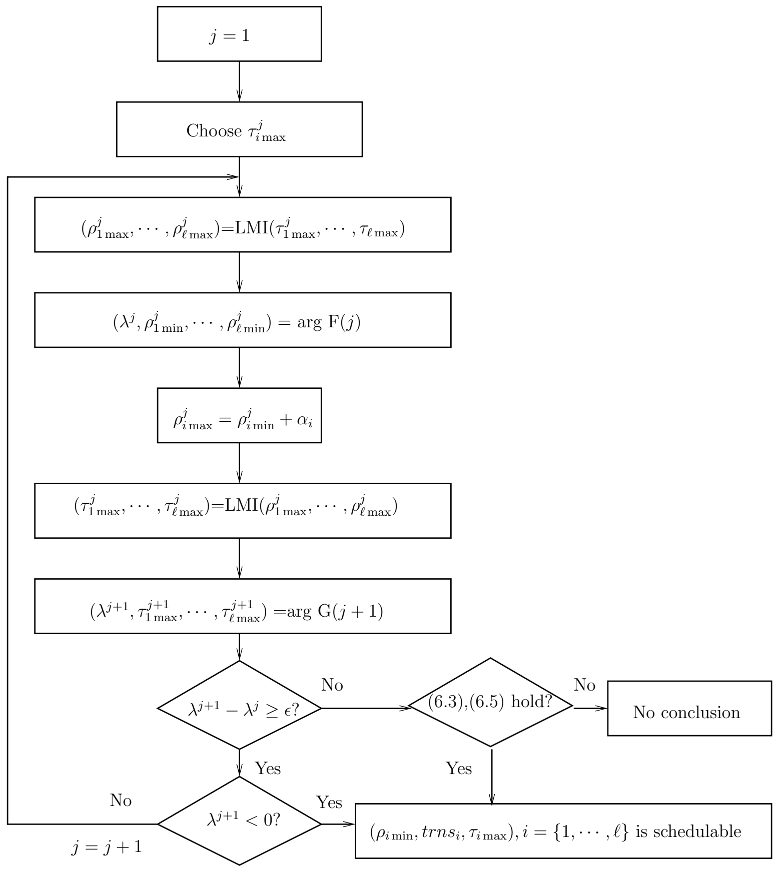


Figure 6.3. Algorithm to find sampling and delay bounds to guarantee stability

where F(j) and G(j) are the minimizations given in (6.7) and (6.8). We denote the largest $\rho^j_{i\,\max}$ such that the corresponding stability LMIs in (4.19), (5.13), or (5.17) are satisfied for given $\tau^j_{i\,\max}$, by $(\rho^j_{1\,\max},\cdots,\rho^j_{\ell\,\max})=\text{LMI}(\tau^j_{1\,\max},\cdots,\tau^j_{\ell\,\max})$ and the largest $\tau^j_{i\,\max}$ such that the corresponding stability LMIs are satisfied for given $\rho^j_{i\,\max}$, by $(\tau^j_{1\,\max},\cdots,\tau^j_{\ell\,\max})=\text{LMI}(\rho^j_{1\,\max},\cdots,\rho^j_{\ell\,\max})$ for $i\in\{1,\cdots,\ell\}$.

6.2 Choice of Sampling periods

For the scheduling tests in Theorems 20 and 21, we assumed that the bounds on the sampling intervals and the delays are given. However, when the bounds on the sampling intervals (specifically $\rho_{i\min}$) are not given, one can use these as design variables to make the problem schedulable. There are two stability requirements given in (6.1), but making $s_{k+1}^i - s_k^i$ small to guarantee that $s_{k+1}^i - s_k^i \leq \rho_{i\max}$ will generally lead to larger network delays, which may violate $\tau_k^i \leq \tau_{i\max}$. So there is a trade off between sampling intervals and delays that must be resolved.

We propose the algorithm in Fig. 6.3 to select bounds $\rho_{i\min}, \rho_{i\max}$ and $\tau_{i\max}$ for the sampling intervals and delays, to guarantee stability of all subsystems under EDF scheduling. For flexibility, we allow the user to specify a desired gap $\alpha_i := \rho_{i\max} - \rho_{i\min}$ between the lower and upper bound of sampling times. When $\alpha_i = 0$, we have periodic sampling with no dropouts, but $\alpha_i > 0$ allows for sampling interval variability or packet dropouts. For instance to allow one packet dropout (due to noise) with a constant sampling equal to $\rho_{i\min}$ we choose $\alpha_i = \rho_{i\min}$ (see Remark 4).

This algorithm is based on a conservative linearization of the condition (6.6) and Theorem 21. Although Theorem 21 is more conservative than Theorem 20, it has the advantage that the conditions in Theorem 21 can be linearized, making them more suitable for the numerical optimization in the proposed algorithm.

This algorithm uses the slack variable λ which measures how much the conditions (6.5) and (6.6) are not satisfied. At each iteration of the algorithm we solve a linear optimization to decrease the value of λ. We denote λ^j the minimum value of λ obtained from the $j-$th iteration. The algorithm starts with an arbitrary choice of $\tau^1_{i\max}, i \in \{1, \cdots, \ell\}$, small enough such that stability LMIs are satisfied for some $\rho^1_{i\max}$. Then the algorithm tries to find the best choices of the sampling interval and delay bounds such that the scheduling conditions are "closest" to being satisfied. The sequence λ^j is decreasing since the best choices of the last iteration are in the search space of the current minimization.

Given $\tau^j_{i\max}, i \in \{1, \cdots, \ell\}$ at the $j-$th iteration of the algorithm, we find the largest $\rho^j_{i\max},\ i \in \{1, \cdots, \ell\}$ such that the corresponding stability LMIs in (4.19), (5.13), or (5.17)are satisfied and we denote this step by $(\rho^j_{1\max}, \cdots, \rho^j_{\ell\max}) =$ LMI$(\tau^j_{1\max}, \cdots, \tau^j_{\ell\max})$.

Given $\rho^j_{i\max}, i \in \{1, \cdots, \ell\}$ at the $j-$th iteration of the algorithm, we find the largest $\tau^j_{i\max}, i \in \{1, \cdots, \ell\}$ such that the corresponding stability LMIs in (4.19), (5.13), or (5.17) are satisfied and we denote this step by $(\tau^j_{1\max}, \cdots, \tau^j_{\ell\max})$ =LMI$(\rho^j_{1\max}, \cdots, \rho^j_{\ell\max})$. In the algorithm, F$(j)$ and G(j) are the following minimizations

$$
\begin{aligned}
\mathrm{F}(j) : \min_{\frac{1}{\rho_{i\min}}} \lambda & \\
& \text{subject to:} \\
& \quad \frac{1}{\rho_{i\min}} \leq \frac{1}{\rho^j_{i\max} - \alpha_i}, \\
& \quad \sum_{i=1}^{n} \frac{C_i}{\rho_{i\min}} < 1 + \lambda, \\
& \quad \sum_{i=1}^{k} \big(1 + \frac{\tau^j_{k\max} - \tau^j_{i\max}}{\rho_{i\min}}\big) trns_i + trns_{\max} \leq \tau^j_{k\max} + \lambda,
\end{aligned} \tag{6.7}
$$

$$
\begin{aligned}
\mathrm{G}(j): \min_{\tau_{i\max}} & \lambda \\
& \text{subject to}: \\
& \tau_{i\max} \le \tau^{j}_{i\max}, \\
& \sum_{i=1}^{k} \big(1 + \frac{\tau_{k\max} - \tau_{i\max}}{\rho^{j}_{i\min}}\big) trns_i + trns_{\max} \le \tau_{k\max} + \lambda,
\end{aligned} \tag{6.8}
$$

where j denotes the $j-$th iteration. If at some iteration j, the value of λ^j becomes negative then both conditions in Theorem 21 are satisfied with the current choices of sampling interval and delay bounds. We stop the algorithm when, either $\lambda^j < 0$ and we find suitable solutions or when $\lambda^j - \lambda^{j-1} < \epsilon$ for some $\epsilon > 0$. When this occurs, the choice of sampling and delay bounds may not satisfy the conditions of Theorem 21 but they may satisfy the conditions of Theorem 20, which guarantees stability, otherwise the algorithm fails and no conclusion can be drawn.

6.3 CAN networks

In this section we focus on CAN networks since they are widely used in the automotive industry, production machinery, medical equipment and building automation [24]. A CAN network is a serial bus used to connect different elements of control systems such as sensors, actuators, and controllers. CAN networks are suitable for short and time critical messages. The total overhead of a message is 47 bits, which includes start of frame, identifier, control, CRC, acknowledgment, end of frame, and inter-mission field. The size of data is 0 to 8 bytes. More details of the data format can be found in [8]. Here we explain the identifier field in more detail because we will use it to assign priorities to packets.

Each message contains an identifier field which is *unique* for each node transmitting a message. The identifier field has 11 bits in standard CAN (version 2.0A) and 29 bits in extended CAN (version 2.0B). There is no destination address field and all nodes receive the packets sent to the network. Then they filter out the packets based on the source of the packet that can be extracted from the identifier field. The identifier field has another important role: it is used to assign a priority to a packet. Any conflict over accessing the network is resolved by bitwise arbitration of the identifier field. The logic value zero has priority over the logic value one and a node with a higher priority gains the access to the network [24, 27]. When a node wants to send a packet, it waits until the bus is free and then (after sending the *start of frame* bit used to synchronize the nodes) starts sending the identifier bit by bit. After sending each bit, the node "listens" to the network. If it senses the same logic as it just sent, the node sends the next bit of the identifier. Otherwise the node realizes that there is another node trying to send a packet with higher priority and backs off until the bus is free gain. If a node succeeds in sending all bits of its identifier field, that node gains access to the network and it continues to transfer its data. CAN network protocols belong to the class of Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA), which avoid collision by bit arbitration.

The arbitration mechanism limits the maximum data rate and physical length of CAN networks because the duration of each bit is required to be large enough so it can propagates the length of the network. For example the data rate of a 40m network can be up to 1Mb/s and for longer networks lower data rates are required.

Priority assignment and implementation As mentioned in previous section, the data from the source i should be received by its destination before a

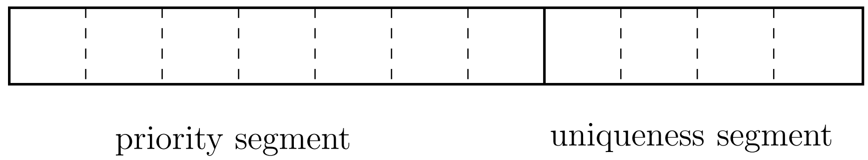


Figure 6.4. Identification field with 4 bits of uniqueness and 7 bits of priority.

deadline determined by $\tau_{i\max}$. We will use these deadlines to assign priorities to messages and encode the priorities in the identification field of the packets in order to implement EDF or DM scheduling policies. The priority assignment will be distributed, meaning that each node by itself assigns the priority of its packets.

In DM scheduling policy roughly speaking $1/\rho_{i\max}$ is encoded in the identifier field. In EDF scheduling policy roughly speaking $1/(t_{il} + \tau_{i\max} - t)$ is encoded in the identifier field, where t_{il} is the last sampling time of connection i and t is the current time. So the node with the earliest deadline gains the access to the network.

To implement any of these scheduling policies, the set of priorities used by each source node needs to be unique so the destination nodes can filter out the packets sent for them. In static priority assignment, all the priorities are known a priori so each destination can identify the messages sent from its source. However, in dynamic priority assignment, priority changes over time and it is priori unknown. To be able to uniquely determine the identification field we divide it to two segments: uniqueness segment and priority segment. At first we uniquely assign a number to each node e.g. 0000 to 1111. Less significant bits of identification field contains the uniqueness field. The most significant bits of the identifier are used for priority assignment.

The ID field only has 11 bits (to reduce the overhead and network load, one would prefer an 11 bit over a 29 bit identifier field) and part of it is used for uniqueness and priority assignment. To effectively assign priorities [78, 79] suggest Mixed Traffic Scheduler (MTS) which divides messages to small relative deadline ($\tau_{i\,\max}$) and large relative deadline real-time messages. Then the large relative deadline messages are scheduled based on DM and the small relative deadline messages are scheduled based on EDF, while they always have priority over the large relative deadline real-time messages. This way the absolute deadline of a packet is always a small number so it can be encoded in the ID field with small number of bits.

In general, each method has its own advantages and disadvantages. A static priority assignment is easy, but usually it does not take advantage of the network capacity fully and a low priority packet may suffer a large delay [6]. On the other hand, a dynamic priority assignment uses the network capacity more effectively, and it is more robust because the priorities are assigned based on the current situation. However, the downside is that it is computationally more demanding. In a sense, a dynamic priority assignment uses communication resources more efficiently at the expenses of using computation resources less efficiently compared to a static one. As the CPUs become faster and cheaper, adding more computation resources is more desirable. Besides the cost, if we have to add another bus, it should be connected through a gateway node [24] to share the packets with the rest of the network. However, a gateway node is not desirable because the whole system would fail if a failure happens in gateway node. So if adding more computation power helps to reduce the number of networks, it is definitely more desirable.

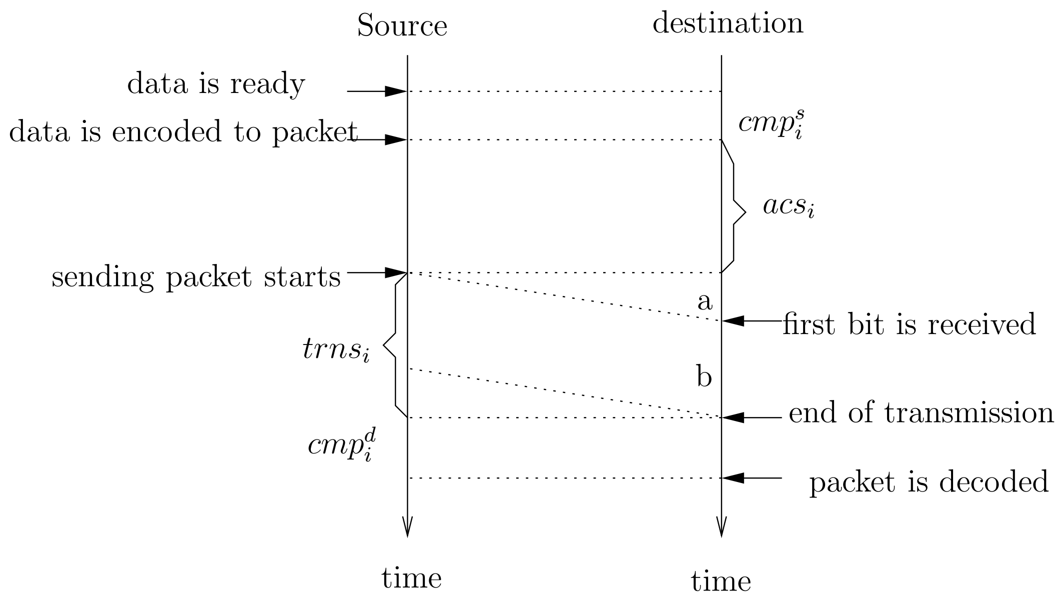


Figure 6.5. Timing diagram of connection i.

6.4 Computation delay at the source and the destination

Now we assume that the computation delays at the sources and the destinations are no longer negligible. Let's assume that the computation delays are constant. The total delay is equal to the sum of the computation time at the source, the wait time to access the network, the transmission time, and the computation time at the destination as shown in Fig. 6.5 where acs_i and $trns_i$ defined in Section 6.1.1 and:

- Computation time at the source of connection i, cmp_i^s, is the time needed to encode data to a packet with a proper format.

- Computation time at the destination of the connection i, cmp_i^d, is the time needed to decode a received packet to data.

Now our goal is to design a network access protocol so that the network access times, acs_i, are small enough so that instead of (6.2), the following holds for every connection $i \in \{1, \cdots \ell\}$,

$$comp_i^s + acs_i + trns_i + comp_i^d \leq \tau_{i\max}. \tag{6.9}$$

To ensure that the above inequality holds with non-preemptive EDF scheduling, the conditions in Theorem 20 or 21 must be satisfied with the new deadline

$$\bar{\tau}_{i\max} := \tau_{i\max} - cmp_i^s - cmp_i^d. \tag{6.10}$$

If the computation times at the source and destination $i \in \{1, \cdots, \ell\}$ are variable but bounded by cmp_i^s, cmp_i^d the situation is exactly the same as the constant case, with the only difference being that the deadlines $\bar{\tau}_{i\max}$ are based on the worse case according to (6.10). Although we cannot improve the scheduling conditions because they must be satisfied even under the worse case, in the implementation under EDF scheduling a packet that suffered from a large delay in the computation will gain the access to the network faster because the actual priority encoded to the ID field is higher.

6.5 Concluding remarks

In this Chapter we considered the case that a communication resource is shared between many nodes and there is a "competition" over accessing the shared resource. However, in general there might be competition over several communication and computation resources. For example, consider the NCS (perhaps an embedded system with shared communication and computation resources is a more descriptive term) in Fig. 6.6. In this scenario there is no competition over

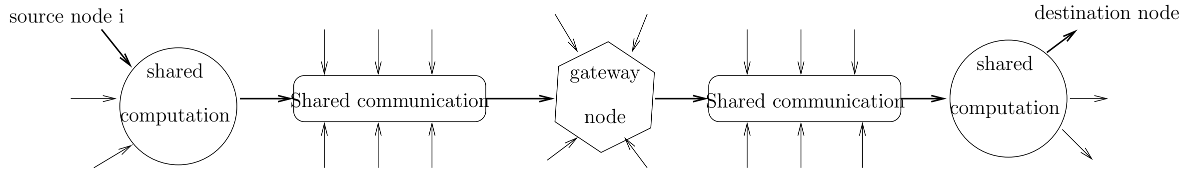


Figure 6.6. A NCS with shared computation and communication resources. For instance, one of the shared communication resources is a CAN network and the other one is a FlexRay network and a gateway connects the networks.

the FlexRay network [44] because the access to the network is based on time slots a priori assigned to the source nodes (similar to Round-Robin scheduling) and the delay produced by this component of the system is almost constant. However, there are competitions over accessing the CAN network and computation resources. Specially there is a *high* competition over the gateway node because it connects a low speed network (CAN data-rate is between 125 Kb/s and 1 Mb/s) and a high speed network (FlexRay data-rate is in the range of 10 Mb/s). To guarantee that there is no "explosion" of information and to find the upper bound of delay (consisting of wait time and service time) produced by each component of the system, proper scheduling conditions should be satisfied for each of shared resources. In general, the scheduling can be preemptive (for computation resources) or non-preemptive scheduling with EDF or DM policies. We only provided scheduling conditions for non-preemptive EDF scheduling and we refer to [6, 77, 8] for other cases.

To guarantee stability of all subsystems the summation of the deadlines of each component in the i-th connection should be smaller that $\tau_{i\max}$ for all $i \in \{1, \cdots, \ell\}$. Perhaps the main challenge is to choose *proper* deadlines so that the summation of deadlines is smaller that $\tau_{i\max}$. For this purpose a numerical procedure similar to the one we proposed in Section 6.2 for choice of deadlines is required. The whole procedure may shed light to identify the "bottlenecks" of the system which produce large delays in order to replace those components with faster ones, or to change the structure of the system.

Chapter 7

Input-To-State Stability Of Delay Impulsive System With Application To Tracking Over Network

In this chapter, we consider a system of the form

$$\dot{x}(t) = Ax(t) + B_1 x(s_k) + B_2 v(t), \qquad t \in [s_k + \tau_k, s_{k+1} + \tau_{k+1}), \tag{7.1}$$

which represents the closed-loop system in Fig. 7.1, where s_k denotes the *k-th sampling time* and t_k the so called *k-th input update time*, which is the time instant at which the k-th sample arrives to the destination and $v(t)$ is an external input. If we denote the delay in the loop that the $k-$th sample experiences by τ_k then $t_k := s_k + \tau_k$. The signal $v(t)$ represents an external input and we would like to find a class of sampling-delay sequences such that the system is Input-to-State

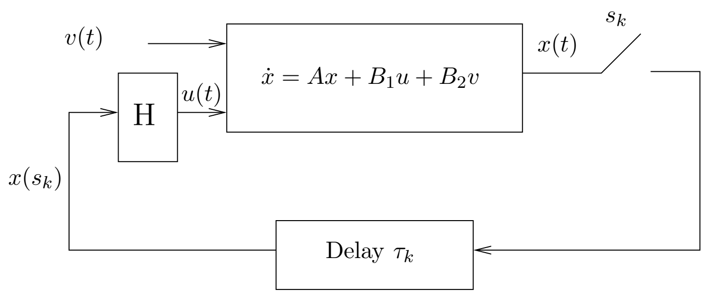


Figure 7.1. An abstract SISO system with external input $v(t)$ and delay τ_k in the feedback loop where $u(t) = x(s_k)$ for $t \in [s_k + \tau_k, s_{k+1} + \tau_{k+1})$,.

Stable (ISS): for a given bound on the input, $v(t)$, the states of the system, $x(t)$, remains bounded. We will formally define ISS stability in Section 7.1. We model the system (7.1) as an impulsive system driven by an external input and we show that tracking problem over the network can be modeled as system (7.1).

For tracking a desired output or state trajectory, the controller sends control commands over the network that force the state or output of the process to follow a reference path. For tracking a feedforward control structure is needed to generate the desired trajectory. Moreover, for convergence to the desired solution and favorable robustness and disturbance attenuation the feedback structure is employed.

Since the feedback and feedforward control commands are discrete and they experience variable delay, exact trajectory tracking is not possible and there is a mismatch or error between the desired trajectory and the real trajectory of the system. We drive the error dynamics which is in the form of (7.1) where the effect of feedforward signal mismatch on the error dynamics shows up as the external input $v(t)$.

In Section 7.1 we model the closed-loop as a linear impulsive system driven

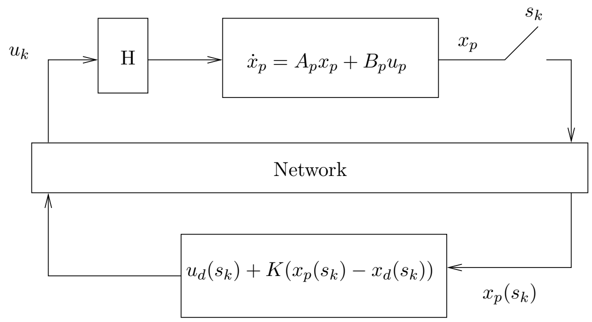


Figure 7.2. NCS for tracking. The desired trajectory evolves according to equation (7.4). The input $u_p = u_k, s_k + \tau_k \leq t < s_{k+1} + \tau_{k+1}$ where τ_k is the total delay in the network that the $k-$th packet experiences and $u_k := u_d(s_k) + K(x_p(s_k) - x_d(s_k))$.

by the feedforward signal mismatch. The effect of feedforward signal mismatch on the error dynamics is related to the ISS property of the linear impulsive system. Hence in Section 7.2 we establish the ISS property of nonlinear time-varying impulsive systems by Razumikhin-type results. There are ISS results based on Lyapunov-Krasovskii theorems in the literature; however, these conditions cannot be expressed in terms of LMIs for linear case and hence generally lead to conservative results [55]. In Section 7.3 we provide ISS bounds in terms of LMIs for a given class of sampling-delay sequences. This analysis will lead to an admissible sampling-delay sequences to guarantee a given tracking mismatch bound. In the last section we consider an illustrative example.

7.1 Delay Impulsive Model for Tracking

We consider the NCS depicted in Fig. 7.2 which can represent a two-channel NCS where the delay in the sensor and the actuator channels are lumped into one delay represented by τ_k. It consists of a continuous-time plant and a static feedback controller connected through a network. Similar to previous chapters, plant has the following state-space

$$\dot{x}_p(t) = A_p x_p(t) + B_p u_p(t), \tag{7.2}$$

where $x_p \in \mathbb{R}^n, u_p \in \mathbb{R}^m$ are the state and the input of the plant. At time s_k, $k \in \mathbb{N}$ the plant's state, $x_p(s_k)$, is sent to the controller. Based on the new measurement received, the control command u_k is calculated and sent back to the plant to be used as soon as it arrives until the next control command update. The total delay in the control loop that the $k-$th sample experiences is denoted by τ_k and $0 \leq \tau_k \leq \tau_{\max}$ and the update time is denoted by $t_k := s_k + \tau_k$. The sampling times $\{s_1, s_2, s_3, \cdots\}$ and the input update times $\{t_1, t_2, t_3, \cdots\}$ form strictly increasing sequences in $[s_0, \infty)$ for some initial time s_0. Out of order packets will be dropped and consequently the input update times form a strictly increasing sequence. The control command u_k is used as soon as it arrives at t_k until the next control command update t_{k+1} so

$$u_p(t) = u_k, \quad t_k \leq t \leq t_{k+1}. \tag{7.3}$$

In what follows we introduce the tracking problem, control signal construction, the tracking error dynamics, and we argue that the ISS property of the error dynamics is the relevant notion to study the effect of the network on the tracking problem.

Control signal construction We would like the system to asymptotically track a desired trajectory $x_d(t)$. The proposed control law consists of a feedforward part and a feedback part. The exact feedforward $u_d(t)$ should be selected such that the desired state trajectory $x_d(t)$ is a solution to the continuous-time system

$$\dot{x}_d(t) = A_p x_d(t) + B_p u_d(t). \tag{7.4}$$

We propose the following tracking control law for use in (7.3):

$$u_k := u_d(s_k) - K\left(x_p(s_k) - x_d(s_k)\right), \tag{7.5}$$

that consists of a feedforward component $u_d(s_k)$ and a feedback component $K\left(x_p(s_k) - x_d(s_k)\right)$.

Closed-loop system Applying the control law (7.5) to system (7.2) yields the following closed-loop dynamics:

$$\dot{x}_p(t) = A_p x_p(t) - B_p K\left(x_p(s_k) - x_d(s_k)\right) + B_p u_d(s_k), \qquad t_k \leq t < t_{k+1}. \tag{7.6}$$

The initial condition $\bar{x}_p(0) := \begin{bmatrix} {x_p}'(0) & {x_p}'(s_0) \end{bmatrix}'$ for this system consists of both the initial state at time 0, i.e., $x_p(0)$, and the *hold* state $x_p(s_0)$ at time $s_0 < 0$.

Tracking error dynamics We define the tracking error by $e = x_p - x_d$. By combining (7.4) and (7.6) we get the tracking error dynamics

$$\dot{e}(t) = A_p e(t) - B_p K e(s_k) + B_p\big(u_d(s_k) - u_d(t)\big), \qquad t_k \leq t < t_{k+1}, \tag{7.7}$$

and with $\bar{e}(0) := \begin{bmatrix} e'(0) & e'(s_0) \end{bmatrix}'$.

Next, we aim to find the bound of the tracking error, $|e(t)|$. If the bound is small or eventually it becomes small after the effect of initial error vanishes, then the plant trajectory becomes close to the desired trajectory. In the next section, we will propose sufficient conditions for the ISS of the system (7.7) to determines the ultimate tracking error bounds.

Since the ISS property of such systems with time-varying delays and sampling times are of interest in a wider context, we consider systems of the form

$$\dot{x}(t) = Ax(t) + B_1 x(s_k) + B_2 v(t), \qquad t_k \le t < t_{k+1}, \tag{7.8}$$

with initial condition $\bar{x}(0) := \begin{bmatrix} x'(0) & x'(s_0) \end{bmatrix}'$. The tracking error dynamic error (7.7) can be recovered from (7.8), by defining

$$\begin{aligned} &A := A_p, \qquad B_1 := -B_p K, \qquad B_2 := B_p, \\ &v(t) := u_d(s_k) - u_d(t), \qquad t_k \le t < t_{k+1}. \end{aligned} \tag{7.9}$$

Alternatively, in the scope of the disturbance rejection problem one may consider equation (7.8) to represent external perturbations or, in the scope of the design of observer-based output-feedback schemes, equation (7.8) may represent the observer error perturbing the closed-loop system.

We would like to have the ISS property for any set of admissible sampling-delays sequences $(\{s_k\}, \{\tau_k\})$ belong to a set $\mathcal{S}$ such that

$$s_{k+1} + \tau_{k+1} - s_k \le \tau_{MATI}, \qquad 0 \le \tau_k \le \tau_{\max}. \tag{7.10}$$

The tracking error dynamics (7.8) can be modeled as a delay impulsive system of the form

$$\dot{\xi}(t) = F\xi(t) + \bar{B}_2 v(t) \qquad t \neq t_k, \forall k \in \mathbb{N} \tag{7.11a}$$

$$\xi(t_k) = \begin{bmatrix} x^-(t_k) \\ x(s_k) \end{bmatrix}, \qquad t = t_k, \forall k \in \mathbb{N}, \tag{7.11b}$$

with the initial condition $\xi(0) := \begin{bmatrix} x'(0) & x'(s_0) \end{bmatrix}'$, where $\xi(t) := \begin{bmatrix} x'(t) & z'(t) \end{bmatrix}'$, $z(t) := x(s_k)$, $t_k \le t < t_{k+1}$ and

$$F := \begin{bmatrix} A & B_1 \\ 0 & 0 \end{bmatrix}, \qquad \bar{B}_2 := \begin{bmatrix} B_2 \\ 0 \end{bmatrix}.$$

In the next chapter we provide the formal definition of ISS stability for delay impulsive systems and we provide conditions for ISS stability of nonlinear time-varying delay impulsive systems over a given class of sampling-delay sequences.

7.2 ISS property of delay impulsive system

In this section, we consider a more general delay impulsive system of the form

$$\dot{x}(t) = f_k(x(t), v(t), t), \qquad t \neq t_k, \forall k \in \mathbb{N}, \tag{7.12a}$$

$$x(t_k) = g_k(x(t_k), x^-(s_k), t_k), \qquad t = t_k, \forall k \in \mathbb{N}, \tag{7.12b}$$

where f_k, g_k are locally Lipschitz functions such that $f_k(0,0,t) = 0, g_k(0,,0,t) = 0,\ \forall t \in \mathbb{R}_{\geq 0}$.

The sampling times $\{s_0, s_1, \cdots\}$ and the input update times $\{t_0, t_1, \cdots\}$ form unbounded strictly increasing sequences. We allow the delays τ_k to grow larger than the sampling intervals $s_k - s_{k-1}$, provided that the sequence of input update times $\{t_0, t_1, \cdots\}$ remains strictly increasing. In essence, this means that if a sample gets to the destination out of order (i.e., an old sample gets to the destination after the most recent one), it should be dropped.

We say that the system (7.12) is *uniformly* ISS over a given class $\mathcal{S}$ of admissible sequences of sampling times and delays $(\{s_k\}, \{\tau_k\})$ if there exist a $\mathcal{KL}$-function $\beta(r, s)$ and a $\mathcal{K}$-function $\gamma(r)$ such that, for any initial condition x_0 and any bounded input $v(t)$, the solution to (7.12) satisfies

$$|x(t)| \leq \beta(\|x_0\|, t) + \gamma(\sup_{t\geq 0} |v(t)|), \tag{7.13}$$

with functions β and γ that are independent of the choice of the particular sequence $(\{s_k\}, \{\tau_k\})$.

System (7.11a) is a special case of system (7.12) such that for $\forall t \geq 0, k \in \mathbb{N}$,

$$x(t) := \xi(t), \qquad f_k(x,t) := F\xi(t),$$
$$g_k(x^-(t_k), x(s_k), t_k) := \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \xi^-(t_k) + \begin{bmatrix} 0 & 0 \\ I & 0 \end{bmatrix} \xi(s_k).$$

We can view (7.12) as an infinite dimensional system whose state contains the past history of $x(\cdot)$ so that $x(s_k)$ can be recovered from the state x_{t_k} in order to apply the reset map in (7.12b). This allow us to apply Razumikhin tools in the analysis of (7.12). In this framework, it is straightforward to analyze (7.12) even when the delays grow much larger than the sampling intervals, which is not easy in methods based on a discretization.

For the system (7.12), we assess the ISS property over the set $\mathcal{S}$ of impulse-delay sequences defined in (7.10) using the tools developed for delay differential equations in [65]. Given a Lyapunov function $V : \mathbb{R}^n \times [-\tau_{MATI}, \infty) \to [0, \infty)$, we use the shorthand notation $V(t) := V(x(t), t)$. We define

$$|x_m(t)| := \max_{-\tau_{MATI} \leq \theta \leq 0} |x(t+\theta)|,$$
$$\|x_m\|_{t_0} := \sup_{s \geq t_0} |x_m(s)| = \sup_{s \geq t_0 - \tau_{MATI}} |x(s)|.$$

Theorem 22. *Assume that there exist $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$, $\gamma_1, \gamma_2 \in \mathcal{G}$, a scalar $\alpha_3 > 0$, and a* function *$V(t)$, such that for any impulse-delay sequence $(\{s_k\}, \{\tau_k\}) \in \mathcal{S}$*

the corresponding solution x to (7.12) *satisfies:*

$$\alpha_1(|x(t)|) \leq V(t) \leq \alpha_2(|x(t)|), \qquad \forall t \geq 0 \tag{7.14}$$

$$V(t) \geq \max\left\{\gamma_1(V_m(t)), \gamma_2(\|v\|_{t_0})\right\} \Rightarrow \frac{\mathrm{d}V(t)}{\mathrm{d}t} \leq -\alpha_3 V(t), \qquad \forall t \neq t_k, k \in \mathbb{N} \tag{7.15}$$

$$\gamma_1(s) < s, \ \forall s > 0, \tag{7.16}$$

and that

$$V(t_k) \leq \lim_{t \uparrow t_k} V(t), \qquad \forall t = t_k, k \in \mathbb{N}. \tag{7.17}$$

Then, the system (7.12) *is uniformly ISS over the class $\mathcal{S}$ of impulse-delay sequences with $\gamma(s) := \alpha_1^{-1}(\gamma_2(s))$, $\beta(s,t) := \alpha_1^{-1}(e^{-\lfloor \frac{t+\tau_{MATI}}{T+\tau_{MATI}} \rfloor \alpha_3 T} \alpha_2(s))$, where $T > 0$ is small enough such that $\gamma_1(s) \leq s e^{-\alpha_3 T}$ for $\forall s \leq V_m(t_0)$.* □

Before the proof of Theorem 22, we state the following lemma. The proof of the lemma is analogous to the proof of Theorem 4.18 in [25].

Lemma 1. *Let $\mu > 0$ and $\alpha_3 \in \mathcal{K}$. If $V(t) \geq \mu$ implies $\frac{\mathrm{d}V(t)}{\mathrm{d}t} \leq -\alpha_3 V(t)$, then we have $V(t) \leq \max\{V(t_0)e^{-\alpha_3(t-t_0)}, \mu\}$.*

Proof of Theorem 22. The proof of this theorem closely follows the proof of Theorem 1 of [65]. From the definitions for any $t \geq t_0$ we have

$$|V_m(t)| \leq \max\{|V_m(t_0)|\phi(t-t_0), \|V\|_{t_0}\}, \tag{7.18}$$

where $\phi(s) := 0.5(1 - \mathrm{sgn}(s - \tau_{MATI}))$ and from Lemma 1 and equation (7.15), we conclude

$$V(t) \leq \max\{V(t_0)e^{-\alpha_3(t-t_0)}, \gamma_1(\|V_m\|_{t_0}), \gamma_2(\|v\|_{t_0})\}. \tag{7.19}$$

From (7.18), we have

$$\|V_m\|_{t_0} \leq \max\{|V_m(t_0)|\phi(t-t_0), \|V\|_{t_0}\}, \tag{7.20}$$

and from (7.19) that

$$\|V\|_{t_0} \leq \max\{V(t_0), \gamma_1(\|V_m\|_{t_0}), \gamma_2(\|v\|_{t_0})\}. \tag{7.21}$$

By combining (7.20) and (7.21) we have

$$\begin{aligned}\|V_m\|_{t_0} &\leq \max\{|V_m(t_0)|\phi(t-t_0), V(t_0), \gamma_1(\|V_m\|_{t_0}), \gamma_2(\|v\|_{t_0})\}, \\ &\leq \max\{|V_m(t_0)|\phi(t-t_0), V(t_0), \gamma_2(\|v\|_{t_0})\} \leq \max\{|V_m(t_0)|, \gamma_2(\|v\|_{t_0})\},\end{aligned}$$

in which we used the fact that for all $a, b \geq 0$, if $a \leq \max\{b, \gamma_1(a)\}$ then $a \leq b$ given that $\gamma_1(a) < a$. Then, we conclude boundedness of the solution,

$$\|x_m\|_{t_0} \leq \max\{\alpha_1^{-1}(\alpha_2(|x_m(t_0)|)), \alpha_1^{-1}(\gamma_2(\|v\|_{t_0}))\}. \tag{7.22}$$

For the proof of convergence, we choose T such that $\gamma_1(s) \leq se^{-\alpha_3 T}$ for $\forall s \leq V_m(t_0)$, and from (7.19) we have

$$\begin{aligned}\|V_m\|_{t_0+\tau_{MATI}+T} &\leq \|V\|_{t_0+T} \leq \max\{V(t_0)e^{-\alpha_3 T}, \gamma_1(\|V_m\|_{t_0}), \gamma_2(\|v\|_{t_0})\} \\ &\leq \max\{\|V_m\|_{t_0}e^{-\alpha_3 T}, \gamma_2(\|v\|_{t_0})\}, \|V_m\|_{t_0+2(\tau_{MATI}+T)} \leq \|V\|_{t_0+2T+\tau_{MATI}} \\ &\leq \max\{\|V_m\|_{t_0+T+\tau_{MATI}}e^{-\alpha_3 T}, \gamma_2(\|v\|_{t_0})\} \leq \max\{\|V_m\|_{t_0}e^{-\alpha_3 2T}, \gamma_2(\|v\|_{t_0})\},\end{aligned}$$

and following the same steps, we conclude that for $\forall n \in \mathbb{N}$

$$\|V_m\|_{t_0+n(\tau_{MATI}+T)} \leq \max\{\|V_m\|_{t_0}e^{-\alpha_3 nT}, \gamma_2(\|v\|_{t_0})\},$$

and from (7.20),

$$\|V_m\|_{t_0+n(\tau_{MATI}+T)} \leq \max\{V_m(t_0)e^{-\alpha_3 nT}, \gamma_2(\|v\|_{t_0})\}. \tag{7.23}$$

Note that

$$\alpha_1(\|x_m\|_{t_0+n(T+\tau_{MATI})}) \leq \|V_m\|_{t_0+n(T+\tau_{MATI})} \qquad V_m(t_0) \leq \alpha_2(|x_m(t_0)|).$$

We can show that for $t_0 + nT + (n-1)\tau_{MATI} \leq t \leq t_0 + (n+1)T + n\tau_{MATI}$,

$$\begin{aligned}|x(t)| \leq \|x_m\|_{t_0+n(T+\tau_{MATI})} &\leq \max\{\alpha_1^{-1}(e^{-\alpha_3 nT}\alpha_2(|x_m(t_0)|)), \alpha^{-1}(\gamma_2(|v|))\} \\ &\leq \alpha_1^{-1}(e^{-\alpha_3 nT}\alpha_2(|x_m(t_0)|)) + \alpha^{-1}(\gamma_2(|v|)),\end{aligned}$$

Based on the above equation, we can find $\beta(s,t)$ and $\gamma(s)$ given in Theorem 22.

■

Note that $\beta(s,t)$ satisfies all the conditions of a class-$\mathcal{KL}$ function except that for fixed s is only non-increasing and not continuous everywhere because for $n(T+\tau_{MATI}) \leq t < (n+1)(T+\tau_{MATI}), \forall n \in \mathbb{N}$ the function $\beta(s,t)$ is flat and it reduces at $t = n(t+T), \forall n \in \mathbb{N}$. However, it is easy to construct a $\bar{\beta}(s,t) \in \mathcal{KL}$ from $\beta(s,t)$.

7.3 Tracking control performance

In this section we find tracking error bounds for the system (7.11) for a given class of sampling-delay sequences based on Theorem 22. This analysis will lead to estimation of tracking error for a given class of sampling-delay sequences. We employ a Lyapunov candidate function of the form

$$V(t) := x'Px + (\rho_{1\max} - \rho_1)(x-w)'X(x-w),$$

where $\rho_1(t) := t - t_k, w := x(t_k),\ t \in [t_k, t_{k+1}),\ \rho_{1\max} := \sup_{t\geq 0}\rho_1,$ and P, X are symmetric positive definite matrices. Note that $V(t)$ is positive (for any e and w

not both equal to zero) and satisfies (7.14). Along jumps this Lyapunov function does not increase since the first term remains unchanged and the second term is non-negative before the jumps and it becomes zero right after the jumps and consequently (7.17) holds. We choose $\gamma_1(s) := ps, 0 < p < 1$; so (7.16) holds and we choose $\gamma_2(s) := g_v s^2, g_v > 0$. If the LMIs in Theorem 23 are feasible then (7.15) is satisfied and consequently Theorem 22 guarantees that system (7.11) is uniformly ISS over the class $\mathcal{S}$.

Theorem 23. *Assume that for any sampling-delay sequence belonging to set $\mathcal{S}$ defined in (7.10), there exist positive scalars $\alpha, \lambda_i, 1 \leq i \leq 4, g_v, p < 1$ and symmetric positive definite matrices P, X and (not necessarily symmetric) matrices N_1, N_2 that satisfy the following LMIs:*

$$\begin{bmatrix} M_1+\rho_{\max}M_2 & N_1A & N_1B_1 & N_1B_2 \\ * & -\tau_{\max}^{-1}\lambda_1 P & 0 & 0 \\ * & * & -\tau_{\max}^{-1}\lambda_3 P & 0 \\ * & * & * & -\tau_{\max}^{-1}\lambda_2 I \end{bmatrix} < 0, \tag{7.24a}$$

$$\begin{bmatrix} M_1+\rho_{\max}M_3 & N_1A & N_1B_1 & N_1B_2 & (N_1+N_2)A & (N_1+N_2)B_2 \\ * & -\tau_{\max}^{-1}\lambda_1 P & 0 & 0 & 0 & 0 \\ * & * & -\tau_{\max}^{-1}\lambda_3 P & 0 & 0 & 0 \\ * & * & * & -\tau_{\max}^{-1}\lambda_2 I & 0 & 0 \\ * & * & * & * & -\rho_{\max}^{-1}\lambda_1 P & 0 \\ * & * & * & * & * & -\rho_{\max}^{-1}\lambda_2 I \end{bmatrix} < 0, \tag{7.24b}$$

where

$$\bar{F} := \begin{bmatrix} A & B_1 & 0 & B_2 \end{bmatrix}, \qquad \rho_{\max} := \tau_{MATI} + \tau_{\max},$$

$$\beta_1 := (\lambda_1 + \lambda_2 g_v^{-1})\tau_{\max} p + \lambda_4 + \alpha, \qquad \beta_2 := \lambda_1 p + \lambda_2 g_v^{-1} p,$$

$$\begin{aligned} M_1 := & \begin{bmatrix} P \\ 0 \\ 0 \\ 0 \end{bmatrix} \bar{F} + \bar{F}' \begin{bmatrix} P \\ 0 \\ 0 \\ 0 \end{bmatrix}' - N_1 \begin{bmatrix} I & -I & 0 & 0 \end{bmatrix} - \begin{bmatrix} I & -I & 0 & 0 \end{bmatrix}' N_1' - N_2 \begin{bmatrix} I & 0 & -I & 0 \end{bmatrix} \\ & - \begin{bmatrix} I & 0 & -I & 0 \end{bmatrix}' N_2' - \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} X \begin{bmatrix} I & 0 & -I & 0 \end{bmatrix} - \lambda_4 g_v \begin{bmatrix} 0 \\ 0 \\ 0 \\ I \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & I \end{bmatrix} + \beta_1 \begin{bmatrix} P \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} I & 0 & 0 & 0 \end{bmatrix} \\ & + \lambda_3 p \tau_{\max} \begin{bmatrix} 0 \\ 0 \\ P \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & I & 0 \end{bmatrix}, \\ M_2 := & \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} X \bar{F} + \bar{F}' X \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix}' + (\beta_1 + \beta_2 \rho_1 \max) \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix} X \begin{bmatrix} I \\ 0 \\ -I \\ 0 \end{bmatrix}', \\ M_3 := & \beta_2 \begin{bmatrix} P \\ 0 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} I & 0 & 0 & 0 \end{bmatrix} + (N_1 + N_2) B_1 \begin{bmatrix} 0 & I & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I \\ 0 \\ 0 \end{bmatrix} B_1' (N_1 + N_2)'. \end{aligned} \tag{7.25}$$

Then, the system (7.11) *is uniformly ISS over the class* $\mathcal{S}$ *with respect to the time-varying input* $v(t)$*, i.e., inequality* (7.13) *is satisfied with the functions* β, γ *defined by*

$$\beta(r, s) = r g_1(s), \qquad \gamma(s) = g_2 s, \tag{7.26}$$

with

$$g_2 := \sqrt{\frac{g_v}{\lambda_{\min}(P)}}, \qquad g_1(s) := \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)} p^{\lfloor \frac{\alpha(s+\tau_{MATI})}{\alpha \tau_{MATI} - \log p} \rfloor}}.$$

□

See Chapter 7.5 for the proof of Theorem 23.

Theorem 23 can be applied to the tracking problem over the network with variable sampling and delays. We can convert the equation (7.7) which governs the mismatch between the desired trajectory and the system trajectory to an impulsive form given by (7.12). By applying Theorem 23 one can find the functions β, γ given by (7.26) such that

$$|e(t)| \leq \beta(|e(0)|, t) + \gamma\big(\sup_{k} \sup_{s_k \leq t < t_k} |u_d(t) - u_d(s_k)|\big). \tag{7.27}$$

We assume that $u_d(t) \in \mathcal{C}$, so the derivative of $u_d(t)$ exists. By the Mean Value Theorem there exists a $t^* \in [s_k, t]$ for any $t \leq t_{k+1}$ and for any $k \in \mathbb{N}$ such that

$$\frac{u_d(t) - u_d(s_k)}{t - s_k} \leq \frac{\mathrm{d}u_d(t^*)}{\mathrm{d}t}.$$

Note that

$$|u_d(t) - u_d(s_k)| \leq (t - s_k)\Big|\sup_{t^*>0} \frac{\mathrm{d}u_d(t^*)}{\mathrm{d}t}\Big| \leq \tau_{MATI}\Big|\sup_{t^*>0} \frac{\mathrm{d}u_d(t^*)}{\mathrm{d}t}\Big|,$$

and since the right hand side of the inequality does not depend on k then we have that

$$|e(t)| \leq \beta(|e(0)|, t) + \gamma\Big(\tau_{MATI}\Big|\sup_{t^*>0} \frac{\mathrm{d}u_d(t^*)}{\mathrm{d}t}\Big|\Big).$$

Consequently given $u_d(t)$ and the class $\mathcal{S}$ of sampling-delay sequences and given that the LMIs in Theorem 23 are feasible, one can find the bound on $|e(t)|$. It is usually the case that one would like to find a bound on the ultimate error, i.e., when the effect of $e(0)$ on $e(t)$ vanishes which happens when $t \to \infty$. To find a tighter bound for $e(t)$ the g_2 in (7.26) should be as small as possible, which can be done by line search over g_v and solving the following minimization

$$\begin{aligned} &\min \alpha \\ &\text{subject to (7.24),and} \\ &\begin{bmatrix} P & I \\ * & \alpha I \end{bmatrix} > 0. \end{aligned} \tag{7.28}$$

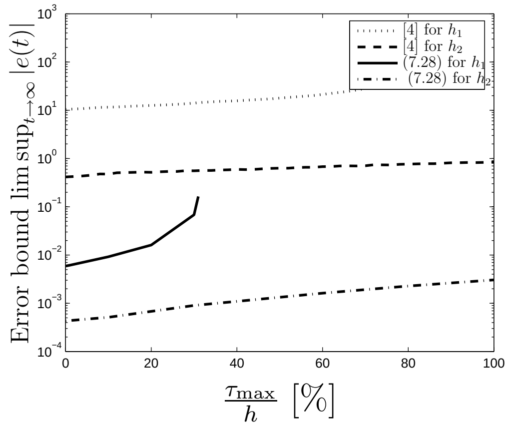


Figure 7.3. Tracking error bounds for a constant sampling interval $h_1 = 5 \times 10^{-3}$ s and $h_2 = 1 \times 10^{-3}$ s and time-varying and uncertain delays in the set $[0, \tau_{\max}]$.

Note that the above LMI is equivalent to $P > \alpha^{-1}I$ which amounts to $\lambda_{\min}(P) > \alpha^{-1}I$. Hence finding the smallest α leads to finding a matrix variable P such that minimizes g_2 in (7.26). We assumed $u(t)$ is scalar but extending the results to the vector case is very easy.

7.4 Illustrative Example

We consider an example of a motion control system from [4]. The continuous-time state-space representation can be described by (7.2), with

$$A_p = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \qquad B_p = \begin{bmatrix} 0 \\ b \end{bmatrix}, \qquad b := \frac{nr_P}{J_M + n^2 J_P},$$

where the first state represents the sheet position (of a sheet in a single motor-roller pair) and the second state is the sheet velocity. Moreover, $J_M = 1.95 \cdot 10^{-5}\text{kgm}^2$ is the inertia of the motor, $J_P = 6.5 \cdot 10^{-5}$ kgm^2 is the inertia of the pinch, $r_P = 14 \cdot 10^{-3}$ m is the radius of the pinch, $n = 0.2$ is the transmission ratio between motor

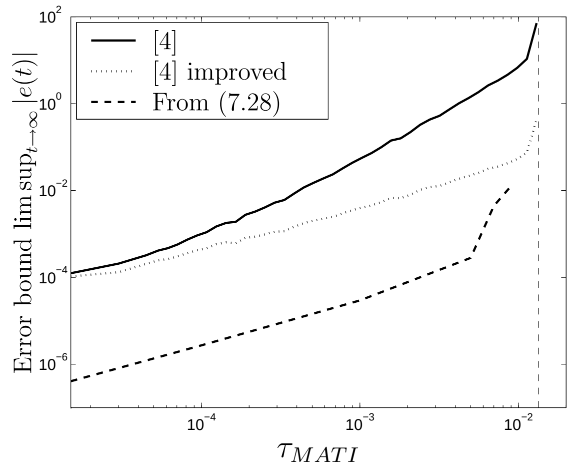


Figure 7.4. Tracking error bounds for variable sampling intervals and no delays, where $h_{\max} = 1.5h_{\min}$.

and pinch and u is the motor torque. We would like to design a control law to enforce the trajectory of the system to follow $x_d(t) = \begin{bmatrix} A_d \sin(\omega t) & A_d\omega \cos(\omega t) \end{bmatrix}'$ with $A_d = 0.01$ and $\omega = 2\pi$. We can compute $\dot{x}(t)$ and given that $A_p, B_p, x_d(t)$ are known, according to (7.4) we compute $u_d(t) = -\frac{A_d\omega^2}{b}\sin(\omega t)$. The feedback gain matrix in (7.5) is $K = \begin{bmatrix} 50 & 1.18 \end{bmatrix}$.

We first consider a constant sampling interval $h := s_{k+1} - s_k$, $\forall k \in \mathbb{N}$, but time-varying and uncertain delays such that $\tau_k \in [0, \tau_{\max}]$, $\forall k \in \mathbb{N}$. Fig. 7.3 depicts the error bounds for $\tau_{\max} \leq h$ for $h = 10^{-3}$ and $h = 5 \times 10^3$. Note that in this case $\tau_{MATI} = \tau_{\max} + h$ and by solving (7.28) we find the least conservative bound on the ultimate error. We compare our result to the discrete-time approach presented in [4]. For this particular example the delay impulsive method leads to less conservative bounds on the ultimate error, while the LMIs in discrete-time approach are feasible for larger delays.

Next, we consider the case in which the sampling intervals are variable and the delay is zero so $s_{k+1} - s_k \leq \tau_{MATI}$. Fig. 7.4 shows the ultimate error bound provided by minimization (7.28) and the results from discrete-time approach presented in [4]. The vertical line in the Fig. 7.4 shows the largest constant sampling interval for which the system is stable. Since the system is LTI for constant sampling, the system is also ISS for any constant sampling smaller than 1.34×10^{-2}. The small distance to this line shows that our result is not very conservative.

This type of plot is instrumental in determining an upper bound on the maximum sampling interval needed to guarantee a minimum level of steady-state tracking performance in the presence of delay in the loop.

7.5 Appendix

Proof of Theorem 23. Along the trajectory of the system (7.11) we have

$$\frac{\mathrm{d}V(t)}{\mathrm{d}t} = 2x'P\dot{x} - (x-w)'X(x-w) + 2(\rho_{1\max} - \rho_1)(x-w)'X\dot{x}. \tag{7.29}$$

To satisfy (7.15) we require that $\frac{\mathrm{d}V(t)}{\mathrm{d}t} \leq -\alpha V(t)$ for some $\alpha > 0$ when $V(t) \geq \gamma_1(V_m(t))$, and $V(t) \geq \gamma_2(|v(t)|)$ with

$$\gamma_1(s) := ps, \qquad\qquad \gamma_2(s) := g_v s^2. \tag{7.30}$$

where $0 < p < 1$ and $g_v > 0$. We define $\rho_2(t) := t - s_k,\ t \in [t_k, t_{k+1})$ and $\bar{\xi} := \begin{bmatrix} x' & z' & w' & v' \end{bmatrix}'$. Then for any matrix N_1, N_2 we have

$$\begin{aligned} 2\bar{\xi}'N_1(x-z) + 2\bar{\xi}'N_2(x-w) &= 2\bar{\xi}'(N_1+N_2)\int_{t-\rho_1}^{t} \dot{x}(s)ds + 2\bar{\xi}'N_1\int_{t-\rho_2}^{t-\rho_1} \dot{x}(s)ds \\ &= 2\bar{\xi}'(N_1+N_2)\int_{t-\rho_1}^{t} \big(Ax(s) + B_1 z(s) + B_2 v(s)\big)ds \\ &\quad + 2\bar{\xi}'N_1\int_{t-\rho_2}^{t-\rho_1} \big(Ax(s) + B_1 z(s) + B_2 v(s)\big)ds. \end{aligned} \tag{7.31}$$

Moreover, using the fact that $x'y \leq \lambda x'x + \lambda^{-1}y'y$, for any $\lambda > 0$, the following inequalities hold:

$$
\begin{aligned}
&2\bar{\xi}'(N_1+N_2)\int_{t-\rho_1}^{t}\big(Ax(s)+B_1z(s)+B_2v(s)\big)ds \\
&\quad \leq \lambda_1^{-1}\rho_1\bar{\xi}'(N_1+N_2)AP^{-1}A'(N_1+N_2)'\bar{\xi} \\
&\quad + \lambda_1\int_{t-\rho_1}^{t} x(s)'Px(s)ds + 2\rho_1\bar{\xi}'(N_1+N_2)B_1z \\
&\quad + \lambda_2^{-1}\rho_1\bar{\xi}'(N_1+N_2)B_2B_2'(N_1+N_2)'\bar{\xi} + \lambda_2\int_{t-\rho_1}^{t} v'(s)v(s)ds, \\
&2\bar{\xi}'N_1\int_{t-\rho_2}^{t-\rho_1}\big(Ax(s)+B_1z(s)+B_2v(s)\big)ds \leq \lambda_1^{-1}(\rho_2-\rho_1)\bar{\xi}'N_1AP^{-1}A'N_1'\bar{\xi} \\
&\quad + \lambda_1\int_{t-\rho_2}^{t-\rho_1} x'(s)Px(s)ds + \lambda_3^{-1}(\rho_2-\rho_1)\bar{\xi}'N_1B_1P^{-1}B_1N_1'\bar{\xi} \\
&\quad + \lambda_3\int_{t-\rho_2}^{t-\rho_1} z'(s)Pz(s)ds + \lambda_2^{-1}(\rho_2-\rho_1)\bar{\xi}'N_1B_2B_2'N_1'\bar{\xi} \\
&\quad + \lambda_2\int_{t-\rho_2}^{t-\rho_1} v'(s)v(s)ds,
\end{aligned}
\tag{7.32}
$$

for $\lambda_i > 0,\ i = 1,2,3$. We require that if

$$V(t) \geq pV_m(t), \qquad V(t) \geq g_v|v(t)|^2 \tag{7.33}$$

then $\frac{\mathrm{d}V(t)}{\mathrm{d}t} \leq -\alpha V(t)$, and consequently the condition (7.15) holds with γ_1, γ_2 defined in (7.30). In other words we assume $V(t) \geq pV_m(t)$, and $V(t) \geq g_v|v(t)|^2$ hold and based on these assumptions we would like to find a condition that $\frac{\mathrm{d}V(t)}{\mathrm{d}t} \leq -\alpha V(t)$. From the assumption $V(t) \geq pV_m(t)$, we conclude that $V(t) \geq pV(s)$ for $s \in [t-\rho_1, t]$ and $V(t-\rho_1) \geq pV(s)$ for $s \in [t-\tau_{\max}-\rho_2, t-\rho_1]$ for $\forall t \geq 0$.

Moreover

$$\lambda_1 \int_{t-\rho_2}^{t} x'(s)Px(s)ds \le \lambda_1 \int_{t-\rho_2}^{t} V(s)ds \le \lambda_1\rho_2 pV(t),$$
$$\lambda_3 \int_{t-\rho_2}^{t-\rho_1} x'(s)Px(s)ds \le \lambda_3 \int_{t-\rho_2}^{t-\rho_1} V(s)ds \le \lambda_3(\rho_2-\rho_1)pV(t-\rho_1) =$$
$$\lambda_3(\rho_2-\rho_1)pwPw. \tag{7.34}$$

Also we have that $V(t) \ge g_v|w(t)|^2$ for $\forall t \ge 0$, so

$$\lambda_2 \int_{t-\rho_2}^{t} v'(s)v(s)ds \le \lambda_2 \int_{t-\rho_2}^{t} g_v^{-1}V(s)ds \le \lambda_2\rho_2 g_v^{-1} pV(t), \tag{7.35}$$

We replace (7.34) and (7.35) in (7.32) and then combine with (7.31) to conclude $\triangle_1 \ge 0$ (when (7.33) holds) where

$$\begin{aligned}\triangle_1 :=& -2\bar{\xi}'N_1(x-z) - 2\bar{\xi}'N_2(x-w) + \lambda_1^{-1}\rho_1\bar{\xi}'(N_1+N_2)AP^{-1}A'(N_1+N_2)'\bar{\xi}\\ &+ 2\rho_1\bar{\xi}'(N_1+N_2)B_1 z + \lambda_2^{-1}\rho_1\bar{\xi}'(N_1+N_2)B_2B_2'(N_1+N_2)'\bar{\xi}\\ &+ \lambda_1^{-1}(\rho_2-\rho_1)\bar{\xi}'N_1AP^{-1}A'N_1'\bar{\xi} + \lambda_3^{-1}(\rho_2-\rho_1)\bar{\xi}'N_1B_1P^{-1}B_1'N_1'\bar{\xi}\\ &+ \lambda_2^{-1}(\rho_2-\rho_1)\bar{\xi}'N_1B_2B_2'N_1'\bar{\xi} + (\lambda_1\rho_2 + \lambda_2\rho_2 g_v^{-1})pV(\tilde{\xi}(t),t)\\ &+ \lambda_3(\rho_2-\rho_1)pw'Pw.\end{aligned} \tag{7.36}$$

The condition (7.15) holds if there exists a $\lambda_4, \alpha \ge 0$ such that

$$\frac{\mathrm{d}V(t)}{\mathrm{d}t} + \triangle_1 + \lambda_4(V(t) - g_v v'(t)v(t)) \le -\alpha V(t). \tag{7.37}$$

The term $\triangle_1$ reduces the conservativeness by exploiting the relationship between x, z and w. The third term is added based on S-procedure [1] and is a crucial element because otherwise the LMIs in Theorem 23 would not be feasible. We replace $\rho_2-\rho_1$ by $\tau_{\max}$ and ρ_2 by $\tau_{\max}+\rho_1$(note that $\rho_2-\rho_1 \le \tau_{max}$) and (7.37) holds if

$$\bar{M}_1 + (\rho_{1\max} - \rho_1)M_2 + \rho_1\bar{M}_3 < 0 \tag{7.38}$$

where

$$\bar{M}_1 := M_1 + \tau_{\max}\lambda_1^{-1}N_1AP^{-1}A'N_1' +$$

$$\tau_{\max}\lambda_3^{-1}N_1B_1P^{-1}B_1'N_1' + \tau_{\max}\lambda_2^{-1}N_1B_2B_2'N_1',$$

$$\bar{M}_3 := M_3 + \lambda_1^{-1}(N_1+N_2)AP^{-1}A' +$$

$$(N_1+N_2)' + \lambda_2^{-1}(N_1+N_2)B_2B_2'(N_1+N_2)',$$

and M_1, M_2, M_3 are defined in (7.25). The condition (7.38) is equivalent to (see proof of Theorem 12)

$$\bar{M}_1 + \rho_{1\max}M_2 < 0, \qquad \bar{M}_1 + \rho_{1\max}\bar{M}_3 < 0.$$

Note that $\rho_{1\max} = \sup_k(t_{k+1}-t_k) \leq \sup_k(s_{k+1}-s_k)+\sup_k(\tau_{k+1}-\tau_k) = \tau_{MATI}+\tau_{\max}$ so we replace $\rho_{1\max}$ by $\tau_{MATI} + \tau_{\max}$ then by Schur Lemma the inequalities are equivalent to (7.24a) and (7.24b). We could state the results in terms of $\rho_{1\max}$ but the constants which characterize the set $\mathcal{S}$ in (7.10) would not show up in LMIs directly. From (7.23) we can conclude

$$\lambda_{\min}(P)|x(t)|^2 \leq V(t) \leq |V_m(t)| \leq \max\{\lambda_{\max}(P)|x_m(t_0)|^2e^{-\alpha nT}, \gamma_2\|v\|_{t_0}\}$$
$$\leq \lambda_{\max}(P)|\bar{x}(t_0)|^2e^{-\alpha nT} + \gamma_2\|v\|_{t_0},$$

for $t_0 + nT + (n-1)\tau_{MATI} \leq t \leq t_0 + (n+1)T + n\tau_{MATI}$ where T is small enough such that $p \leq e^{-\alpha T}$. We pick $p = e^{-\alpha T}$ and we can show that the system is ISS over class $\mathcal{S}$ with the functions β, γ defined in (7.26) with $g_1(s), g_2$ defined in (7.26).

Chapter 8

Conclusion and future directions

In this thesis we presented a collection of results for Networked Control Systems (NCSs). We modeled NCSs as delay impulsive systems which exhibit continuous evolutions described by ODEs and state jumps or impulses that experience delay. We developed general theorems for the exponential stability of nonlinear time-varying delay impulsive systems which can be viewed as extensions of the Lyapunov-Krasovskii Theorem for time-delay systems. For linear plants and controllers, exponential stability conditions can be formulated as Linear Matrix Inequalities (LMIs), which can be solved numerically. By solving these LMIs, one can find classes of delay-sampling sequences for the different sample-hold pairs in a NCS such that exponential stability is guaranteed. We also considered the tracking problem over a network. The error dynamics, defined as the mismatch between the desired trajectory and the real trajectory of the system, can be modeled as an impulsive system driven by an external input. A sufficient condition for the ISS of the tracking error dynamics with respect to this input was given. We also provided classes of sampling-delay sequences for which the steady-state tracking error is guaranteed to be smaller than a desired level.

The timing requirements of delay-sampling sequences, obtained from the analysis described above, led to the design of communication protocols to determine which nodes gain access to the network and an algorithm to select sampling sequences. These results can be extended to design access protocols to share computation and communication resources in a NCS. Also delays produced by each shared component in a NCS can indicate the "bottlenecks" of the system and help a system designer to choose components of the system.

Some future directions and open questions are as follows:

- Although the analysis results presented in this thesis are less conservative than the existing ones in the literature, there might be ways to reduce the conservativeness of the sampling and delay bounds even further. For instance numerical simulations suggest that the Lyapunov function given in [41] may provide a necessary and sufficient condition for exponential stability of the system (3.1) (or at least the results show a great improvement). This Lyapunov function is closely related to the Lyapunov function given in (3.14). It seems that there should be more intuition behind the form of the second term of the Lyapunov function in [41].

- We found upper bounds of the sampling intervals and delays to guarantee exponential stability. However, one would expect that if "rarely" sampling intervals or delays violate these bounds, still the system remains stable. This statistical approach motives us to combine our results with the ADS results [18]. Then stability of the system also depends on the *rate* of violating bounds provided by our approach (similar to Corollary 1). This way we allow much larger delay and sampling intervals if they occur "rarely".

- The ISS results for nonlinear time-varying impulsive systems in Chapter 7 are Razumikhin-type results. There are ISS results based on Lyapunov-Krasovskii method in the literature; however, these conditions cannot be expressed in terms of LMIs for linear case and hence generally lead to conservative results [55]. However it may be possible to derive ISS results based on Lyapunov-Krasovskii theorems that can be expressed in terms of LMIs for linear case. Since Lyapunov-Krasovskii based methods usually lead to less conservative results (for stability), the bounds in illustrative example presented in Section 7.4 may be further improved.

- The effect of variable delays and sampling intervals on an adaptive control system needs to be investigated.

- There are protocols and wired networks, such as CAN and FlexRay, that are specifically designed to meet the timing requirements of control systems. However, to our best knowledge such developments have not been performed in wireless networks to make them more appropriate for control purposes.

Bibliography

[1] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishman. *Linear matrix inequalities in system and control theory.* SIAM, 1994.

[2] M. S. Branicky. Stability of hybrid systems: State of the art. In *Proc. of the 36th Conf. on Decision and Contr.*, volume 1, pages 120–125, Dec. 1997.

[3] M. S. Branicky, S. M. Phillips, and W. Zhang. Stability of networked control systems:explicit analysis of delay. In *Proc. of the 2000 Amer. Contr. Conf.*, volume 4, pages 2352–2357, June 2000.

[4] M. Cloosterman, N. van de Wouw, W. Heemels, and H. Nijmeijer. Robust stability of networked control systems with time-varying network-induced delays. In *Proc. of the 45th Conf. on Decision and Contr.*, Dec. 2006.

[5] O. V. Costa and M. D. Fragoso. Stability results for discrete-time linear systems with Markovian jumping parameters. *Journal of mathematical analysis and applications*, 179(1):154–178, Oct 1993.

[6] F. Cottet, J. Delacroix, C. Kaiser, and Z. Mammeri. *Scheduling in real-time systems.* Willy, 2002.

[7] D. B. Dacic and D. Nesic. Quadratic stabilization of linear networked control systems via simultaneous protocol and controller design. *Automatica*, 43(7): 1145–1155, July 2007.

[8] R. Davis, A. Burns, R. Bril, and J. Lukkien. Controller area network (CAN) schedulability analysis: refuted, revisited and revised. *real time systems*, 35 (3):239–272, April 2007.

[9] R. A. Decarlo, M. Branicky, S. Pettersson, and B. Lennartson. Perspectives and results on the stability and stabilizability of hybrid systems. *Proc. of IEEE*, 88(7):1069 – 1082, July 2000.

[10] D. D. Farias, J. C. Geromel, J. D. Val, and O. Costa. Output feedback control of markov jump linear systems in continuous-time. *IEEE Trans. on Automat. Contr.*, 45(5):944 – 949, May 2000.

[11] E. Fridman. A new Lyapunov technique for robust control of systems with uncertain non-small delays. *J. of Math. Contr. and Info.*, 22(3), Nov. 2005.

[12] E. Fridman, A. Seuret, and J. P. Richard. Robust sampled-data stabilization of linear systems: an input delay approach. *Automatica*, 40(8):1441–1446, Aug. 2004.

[13] E. Fridman and U. Shaked. A descriptor system approach to H_∞ control of linear time-delay systems. *IEEE Trans. on Automat. Contr.*, 47(2):253 – 270, Feb. 2002.

[14] H. Gao and C. Wang. Comments and further results on "a descriptor system approach to H_∞ control of linear time-delay system". *IEEE Trans. on Automat. Contr.*, 48, Mar. 2003.

[15] L. E. Ghaoui, F. Oustry, and M. AitRami. A cone complementarity linearization algorithm for static output-feedback and related problems. *IEEE Trans. on Automat. Contr.*, 42(8):1171–1176, Aug. 1997.

[16] K. Gu, V. L. Kharitonov, and J. Chen. *Stability of time-delay systems.* Birkhauser, 1st edition, 2003.

[17] C. N. Hadjicostis and R. Touri. Feedback control utilizing packet dropping network links. In *Proc. of the 41th Conf. on Decision and Contr.*, volume 2, pages 1205–1210, Dec. 2002.

[18] A. Hassibi, S. P. Boyd, and J. P. How. Control of asynchronous dynamical systems with rate constraints on events. In *Proc. of the 38th Conf. on Decision and Contr.*, volume 2, pages 1345–1351, Dec. 1999.

[19] J. P. Hespanha, M. L. McLaughlin, and G. Sukhatme. Haptic collaboration over the Internet. In *Proc. of the 5th Phantom Users Group Workshop*, Oct. 2000.

[20] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu. Networked control systems: analysis and design. *Proc. of IEEE*, 95(1):138–162, Jan. 2007.

[21] J. P. Hespanha and A. R. Teel. Stochastic impulsive systems driven by renewal processes. *Proc. of the 17th Int. Symp. on the Mathematical Theory of Networks and Syst.*, pages 606–618, July 2006.

[22] K. Hikichi, H. Morino, I. Arimoto, K. Sezaki, and Y. Yasuda. The evaluation of delay jitter for haptics collaboration over the Internet. In *IEEE Global Telecomm. Conf. (GLOBECOM)*, volume 2, pages 1492 – 1496, Nov. 2002.

[23] L. Hu, Y. Cao, and H. Shao. Constrained robust sampled-data control of nonlinear uncertain systems. *Int. J. of Robust and Nonlinear Contr.*, 12: 447–464, 2002.

[24] K. H. Johansson, M. Törngren, and L. Nielsen. *Handbook of networked and embedded control systems*, chapter Vehicle applications for Controller Area Networks, pages 741–766. Birkhäuser, 2005.

[25] H. Khalil. *Nonlinear systems.* Prentice-Hall, 2 edition, 1996.

[26] F.-L. Lian. *Analysis, design, modeling, and control of networked control systems.* PhD thesis, University of Michigan, 2001.

[27] F.-L. Lian, J. R. Moyne, and D. M. Tilbury. Performance evaluation of control networks. *IEEECSM*, 21(1):66–83, Feb. 2001.

[28] D. Liberzon. *Switching in systems and control.* Birkhauser, Jun 2003.

[29] H. Lin and P. J. Antsaklis. Persistent disturbance attenuation properties for networked control systems. In *Proc. of the 43th Conf. on Decision and Contr.*, volume 1, pages 953–958, Dec 2004.

[30] H. Lin, G. Zhai, and P. J. Antsaklis. Robust stability and disturbance attenuation analysis of a class of networked control systems. In *Proc. of the 42th Conf. on Decision and Contr.*, volume 2, pages 1182–1187, Dec. 2003.

[31] H. Lin, G. Zhai, L. Fang, and P. J. Antsaklis. Stability and $\mathcal{H}_\infty$ performance preserving scheduling policy for networked control systems. In *IFAC05*, July 2005.

[32] B. Lincoln and B. Bernhardsson. Optimal control over networks with long random delays. In *Proc. of the 14th Int. Symp. of Mathematical Th. of Networks and Syst.*, 2000.

[33] R. Luck and A. Ray. An observer-based compensator for distributed delays. *Automatica*, 26(5):903–908, Sept. 1990.

[34] C. Meng, T. Wang, W. Chou, S. Luan, Y. Zhang, and Z. Tian. Remote surgery case: robot-assisted teleneurosurgery. In *IEEE Int. Conf. on Robot. and Auto. (ICRA'04)*, volume 1, pages 819–823, Apr. 2004.

[35] L. A. Montestruque and P. J. Antsaklis. Model-based networked control systems: Necessary and sufficient conditions for stability. In *10th Mediterranean Contr. Conf. and Auto., Lisbon, Portugal*, 2002.

[36] L. A. Montestruque and P. J. Antsaklis. State and output feedback control in model-based networked control systems. In *Proc. of the 41th Conf. on Decision and Contr.*, volume 2, pages 1620 – 1625, Dec. 2002.

[37] L. A. Montestruque and P. J. Antsaklis. On the model-based control of networked systems. *Automatica*, 39(10):1837–1843, Oct. 2003.

[38] L. A. Montestruque and P. J. Antsaklis. Stochastic stability for model-based networked control systems. In *Proc. of the 2003 Amer. Contr. Conf.*, volume 5, pages 4119 – 4124, June 2003.

[39] L. A. Montestruque and P. J. Antsaklis. Stability of model-based networked control systems with time-varying transmission times. *IEEE Trans. on Automat. Contr.*, 49(9):1562–1572, Sept. 2004.

[40] R. M. Murray, K. J. Åström, S. P. Boyd, R. W. Brockett, and G. Stein. Control in an information rich world. *IEEE Contr. Syst. Mag.*, 23(2):20–33, Apr. 2003.

[41] P. Naghshtabrizi. A less conservaticve method to compute the upper bound of sampling, 2007. URL `http://www.ccec.ece.ucsb.edu/~payam/`.

[42] P. Naghshtabrizi and J. P. Hespanha. Designing observer-based controller for network control system. In *Proc. of the 44th Conf. on Decision and Contr.*, volume 4, pages 2876–2880, June 2005.

[43] P. Naghshtabrizi, J. P. Hespanha, and A. R. Teel. On the robust stability and stabilization of sampled-data systems: A hybrid system approach. In *Proc. of the 45th Conf. on Decision and Contr.*, pages 4873–4878, 2006.

[44] N. Navet, Y. Song, F. Simonot-Lion, and C. Wilwert. Trends in automotive communication systems. *Proc. of IEEE*, 93(6):1204 – 1223, June 2005.

[45] D. Nesic and A. R. Teel. Input-output stability properties of networked control systems. *IEEE Trans. on Automat. Contr.*, 49(10):1650–1667, Oct. 2004.

[46] D. Nesic and A. R. Teel. Input-to-state stability of networked control systems. *Automatica*, 40(12):2121–2128, Dec 2004.

[47] D. Nesic and A. R. Teel. A note on input-to-state stability of networked control systems. In *Proc. of the 43th Conf. on Decision and Contr.*, volume 5, pages 4613 – 4618, Dec 2004.

[48] J. Nilsson. *Real-Time Control Systems with Delays*. PhD thesis, Department of Automatic Control, Lund Institute of Technology, Sweden, Jan. 1998.

[49] J. Nilsson and B. Bernhardsson. Analysis of real-time control systems with time delays. In *Proc. of the 35th Conf. on Decision and Contr.*, volume 3, pages 3173–3178, Dec. 1996.

[50] J. Nilsson and B. Bernhardsson. LQG control over a Markov communication network. In *Proc. of the 36th Conf. on Decision and Contr.*, volume 5, pages 4586–4591, Dec. 1997.

[51] J. Nilsson, B. Bernhardsson, and B. Wittenmark. Stochastic analysis and

control of real-time systems with random time delays. In *Proc. of the 13th World Congress of Int. Federation of Automat. Contr.*, 1996.

[52] J. Nilsson, B. Bernhardsson, and B. Wittenmark. Some topics in real-time control. In *Proc. of the 1998 Amer. Contr. Conf.*, volume 4, pages 2386–2390, June 1998.

[53] J. Nilsson, B. Bernhardsson, and B. Wittenmark. Stochastic analysis and control of real-time systems with random time delays. *Automatica*, 34(1): 57–64, Jan. 1998.

[54] P. Ogren, E. Fiorelli, and N. E. Leonard. Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment. *IEEE Trans. on Automat. Contr.*, 49(8):1292–1302, Aug. 2004.

[55] P. Pepe and Z. P. Jiang. A Lyapunov-Krasovskii methodology for ISS and iISS of time-delay systems. *Syst. & Contr. Lett.*, 55(12):10061014, 2006.

[56] J. P. Richard. Time-delay systems: An overview of some recent advances and open problems. *Automatica*, 39(10):1667–1694, Oct. 2003.

[57] P. Seiler and R. Sengupta. Analysis of communication losses in vehicle control problems. In *Proc. of the 2001 Amer. Contr. Conf.*, volume 2, pages 1491–1496, June 2001.

[58] P. Seiler and R. Sengupta. An H_∞ approach to networked control. *IEEE Trans. on Automat. Contr.*, 50(3):356–364, Mar. 2005.

[59] S. Shirmohammadi and N. H. Woo. Evaluating decorators for haptic collaboration over the Internet. In *The 3rd IEEE Int. Workshop on Haptic, Audio and Visual Env. and their Applic.*, pages 105–109, Oct. 2004.

[60] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry. Kalman filtering with intermittent observations. *IEEE Trans. on Automat. Contr.*, 49(9):1453–1464, Sept. 2004.

[61] S. C. Smith and P. Seiler. Estimation with lossy measurements: jump estimators for jump systems. *IEEE Trans. on Automat. Contr.*, 48(12):2163–2171, Dec. 2003.

[62] W. Stallings. *Data and Computer Communications.* Prentice Hall, 6th edition, 2000.

[63] M. Tabbara, D. Nesic, and A. R. Teel. Input- output stability of wireless networked control systems. In *Proc. of the 44th Conf. on Decision and Contr.*, pages 209–214, dec 2005.

[64] S. Tatikonda. *Control Under Communication Constrains.* PhD thesis, MIT, Cambridge, MA, 2000.

[65] A. Teel. Connections between Razumikhin-type theorems and the ISS nonlinear small gain theorem. *IEEE Trans. on Automat. Contr.*, 43:960–964, 1998.

[66] G. C. Walsh, O. Beldiman, and L. G. Bushnell. Asymptotic behavior of nonlinear networked control systems. *IEEE Trans. on Automat. Contr.*, 46 (7):1093–1097, July 2001.

[67] G. C. Walsh, H. Ye, and L. Bushnell. Stability analysis of networked control systems. In *Proc. of the 1999 Amer. Contr. Conf.*, volume 4, pages 2876–2880, June 1999.

[68] G. C. Walsh, H. Ye, and L. Bushnell. Stability analysis of networked control systems. *IEEE Trans. on Contr. Syst. Tech.*, 10(3):438–446, May 2002.

[69] Y. Xu and J. P. Hespanha. Estimation under uncontrolled and controlled communications in networked control systems. In *Proc. of the 44th Conf. on Decision and Contr.*, pages 842–847, June 2005.

[70] M. Yu, L. Wang, T. Chu, and F. Hao. An LMI approach to networked control systems with data packet dropout and transmission delays. *J. of Hybrid Syst.*, 3(11), Nov. 2004.

[71] M. Yu, L. Wang, G. Xie, and T. Chu. Stabilization of networked control systems with data packet droupout via switched system approach. In *IEEE Int. symp. on computer aided control systems design, Taipei, Taiwan*, pages 362–367, Sept. 2004.

[72] D. Yue, Q. L. Han, and J. Lam. Network-based robust H_∞ control of systems with uncertainty. *Automatica*, 41(6):640–644, June 2005.

[73] D. Yue, Q.-L. Han, and C. Peng. State feedback controller design for networked control systems. *IEEE Trans. on Automat. Contr.*, 51(11):640–644, Nov. 2004.

[74] G. Zhai, B. Hu, K. Yasuda, and A. N. Michel. Qualitative analysis of discrete-time switched systems. In *Proc. of the 2002 Amer. Contr. Conf.*, volume 3, pages 1880–1885, June 2002.

[75] W. Zhang and M. S. Branicky. Stability of networked control systems with time-varying transmission period. In *Allerton Conf. of Communication, contr. and computing, Urbana, IL*, Oct. 2001.

[76] W. Zhang, M. S. Branicky, and S. M. Phillips. Stability of networked control systems. *IEEE Contr. Syst. Mag.*, 21(1):84–99, Feb. 2001.

[77] Q. Zheng and K. G. Shin. On the ability of establishing real-time channels in point-to-point packet-switched networks. *IEEE Trans. on Communications*, 42(2/3/4):1096–1105, Feb. 1994.

[78] K. Zuberi and K. Shin. Non-preemptive scheduling of messages on controller area network for real-time control applications. *Proc. Real-Time Tech. and Appl. Symp.*, pages 240 – 249, May 1995.

[79] K. Zuberi and K. Shin. Design and implementation of efficient message scheduling for controller area network. *IEEE Trans. on Computers,*, 49(2): 182–188, Feb. 2000.