TECHNISCHE UNIVERSITÄT MÜNCHEN

Lehrstuhl für Steuerungs- und Regelungstechnik

Performance-Oriented Control and Co-Design for Stochastic Networked Control Systems

Chih-Chung Chen

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:		Univ.-Prof. Dr. sc. techn. Andreas Herkersdorf
Prüfer der Dissertation:		
	1.	Univ.-Prof. Dr.-Ing. Sandra Hirche
	2.	Univ.-Prof. Dr.-Ing. habil. Boris Lohmann

Die Dissertation wurde am 23.06.2010 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 12.08.2010 angenommen.

Preface

This dissertation has emerged from four years of work at the Institute of Automatic Control Engineering, Technische Universität München where I stayed from 2005 until now.

First of all, I would like to express my gratitude to Prof. Martin Buss and Prof. Sandra Hirche for the opportunity and support of this work. I am especially grateful to Sandra for her creative suggestions which enrich the content of my thesis, for her insightful advice which enlightens my way of thinking and for her patient guidance which encourages me the exploration of unknowns. It is truly a great privilege to work with her and in such a stimulating environment at LSR.

Heartfelt thanks also to all my teammates and friends who root for me during the time, especially for Tilemachos Matiakis, Andreas Schweinberger, Haiyan Wu, Lei Lou and Neal Lii. The last and most important thanks are given to my parents and Sae. With their love and spiritual supports I could finish the PhD abroad. Thank for their understanding of my absence in many reunions for years.

All in all, a word thank will never be enough to express my gratitude to the people who have helped me.

Munich, June 2010 Chih-Chung Chen

Abstract

Due to the use of advanced communication technology, closed-loop control systems become more flexible, robust and easier to maintain. In such networked control systems (NCSs), the conventional point-to-point connection between the controllers and the physical systems are replaced by a communication network. Typical examples can be found in process control, robotics, and automotive industry.

However, the insertion of a communication network into a control loop gives rise new challenges; the use of a communication network comes at the price of non-ideal signal transmission. Random transmission delay and packet dropouts are known as a source of instability and deteriorate the control performance. One of the major challenges is to guarantee a desired control performance in the presence of communication uncertainties at efficient utilization of the limited communication resources.

This dissertation provides a comprehensive development concept for NCSs, which brings different perspectives of stability, control performance, and network resources into one joint design process. In order to guarantee the desired control performance with efficient network resource utilization, the existing approaches for sampled-data systems and stochastic switched time-delay systems are extended. As a result, two novel control and communication system co-design approaches are proposed and systematically investigated. The proposed approaches are analytically verified and experimentally validated in a networked vision-based control system and three degree-of-freedom robotic manipulator control. Both analytical and experimental results demonstrate superior performance benefits compared to the conventional system design.

Zusammenfassung

Durch den Einsatz fortschrittlicher Kommunikationstechnologie können Regelungssysteme inzwischen flexibler, robuster und wartungsgünstiger gestaltet werden. Dies gelingt durch den Einsatz vernetzter Regelungssysteme (Engl. Networked Control Systems/NCSs), in denen die Signale zwischen Prozess und Regler über ein Kommunikationsnetz übertragen werden. Wichtige Anwendungsgebiete finden sich unter anderem in der Prozessautomatisierung, der Robotik und der Fahrzeugtechnik.

Die Einbindung eines Kommunikationsnetzwerks in einen Regelkreis ist jedoch mit neuen Herausforderungen verbunden. Zum einen beeinträchtigen die durch den Datenaustausch über das Kommunikationsnetz entstehenden Zeitverzögerungen und Paketverluste die Stabilität und Regelgüte, zum anderen stehen nur beschränkte Netzwerkressourcen zur Verfügung. Die Herausforderung besteht nun darin, eine gewünschte Regelgüte bei möglichst geringer Nutzung der Kommunkationsressourcen und angesichts der Kommunikationsunsicherheiten zu erreichen.

In dieser Dissertation wird ein umfassendes Entwurfskonzept für vernetzte Regelungssysteme vorgeschlagen, welches die gemeinsame Berücksichtigung der verschiedenen Anforderungen an Stabilität, Regelgüte sowie Kommunikationsressourcen in einem Entwurfprozess ermöglicht. In diesem Sinne werden zwei neuartige Co-Design-Ansätze für Regelungssystem und Kommunikationsnetzwerk entwickelt und erforscht, welche die gewünschte Regelgüte bei kostengünstigstem Datenverkehr garantiert. Zur Analyse und Synthese derartiger Systeme werden existierende regelungstheoretische Methoden der Abtastsysteme und der stochastisch schaltenden Systeme mit Zeitverzögerung erweitert. Die Vorzüge der vorgeschlagenen Verfahren gegenüber einem konventionellen Systementwurf konnten nicht nur analytisch sondern auch in Experimenten mit einer bildbasierten vernetzten Regelung und einer Manipulatorregelung überzeugend nachgewiesen werden.

Contents

Notations

Abbreviations

A/D	Analog to Digital
BEB	Binary Exponential Backoff
BMI	Bilinear Matrix Inequality
CA	Controller-to-actuator
CPU	Central Processing Unit
CSMA	Carrier Sense Multiple Access
D/A	Digital to Analog
DiffServ	Differentiated Service
DoS	Degree of Freedom
DSA	Dynamical Service Addition
DSC	Dynamical Service Change
DSD	Dynamical Service Deletion
I/O	Input/Output
IntServ	Integrated Service
FIFO	First-in-First-out
LMI	Linear Matrix Inequality
LTI	Linear Time-Invariant
MJS	Markovian Jump System
MES	Mean Exponential Stability
NCS	Networked Control System
NVSCS	Networked Visual Servo Control System
PWM	Pulse Width Modifier
QoS	Quality of Service
RTP	Real-Rime Transport Protocol
RSVP	Resource Reservation Protocol
SC	Sensor-to-Controller
SS	Stochastic Stability
UDP	User Datagram Protocol
ZOH	Zero Order Hold

Conventions

Scalars, Vectors, and Matrices

Scalars and *Vectors* are denoted by lower case letters in italic type, whereas *Matrices* are denoted upper case letters in italic type.

x	scalar/vector
X	matrix
$\dot{x}$, $\ddot{x}$	equivalent to $\frac{d}{dt}x$, $\frac{d^2}{dt^2}x$
$\|\| \cdot \|\|$	Euclidean norm
$\lambda(\cdot)$	Eigenvalue
$\mathrm{mod}(\cdot, \cdot)$	modulo

Subscripts and Superscripts

x_c	value x associated with the controller
τ_{sc}	Sensor-to-controller delay
τ_{ca}	Controller-to-actuator delay
τ_{tx}	Transmission delay $\tau_{\mathrm{tx}} = \tau_{\mathrm{sc}} + \tau_{\mathrm{ca}}$
$\lambda_{\max}(\cdot)$	Maximal eigenvalue
$\lambda_{\min}(\cdot)$	Minimal eigenvalue
$(\cdot)^{-1}$	Inverse
$(\cdot)^{+}$	Pseudo-inverse
$(\cdot)^{T}$	Transpose

Symbols and Abbreviations

τ	Time delay
r_t	Markov process
$\mathcal{S}$	state space of a Markov process
$\mathcal{A}$	Markov probability transition rate
$\mathcal{T}$	Markov transition probability
$\mathbf{P}$	Probability
$\mathbb{E}$	Expectation
$\mathbb{R}$	The sets of real numbers
$\mathbb{N}$	The sets of natural numbers
A, B, C, D	State space representation

1 Introduction

1.1 Network control systems

Dynamical systems with spatially distributed sensors and actuators connected via a *common network* are called *networked control systems* (NCSs), see Fig. 1.1. Compared to conventional point-to-point connections, applications of control over networks have considerable advantages. To name a few, the sensors and controllers can be added or removed without wiring efforts. The re-configurability of control systems is increased. Connecting systems over a common network simplifies the diagnosis procedures at component failures as well as maintenance. This helps to reduce the general cost of systems. Furthermore, the use of common networks connects distributed physical spaces, which makes task execution from a distance easily accessible. Networks allow data to be shared efficiently. This enables an intelligent decision over a large physical space by the fusion of global information. In summary, the most important advantages of NCSs are given in the following:

- reduced complexity, wiring and cost of systems,
- easy maintenance, diagnosis and reconfiguration,
- increased flexibility and autonomy.

As a result, NCS control concept has superior benefits in the practical applications and installations over the actual control structures. Furthermore, with the recent advances in wireless networking, a large variety low-cost sensing devices can be deployed throughout a operating environment to reflect the physical phenomena with increasing fidelity. This is infeasible with conventional control structures.

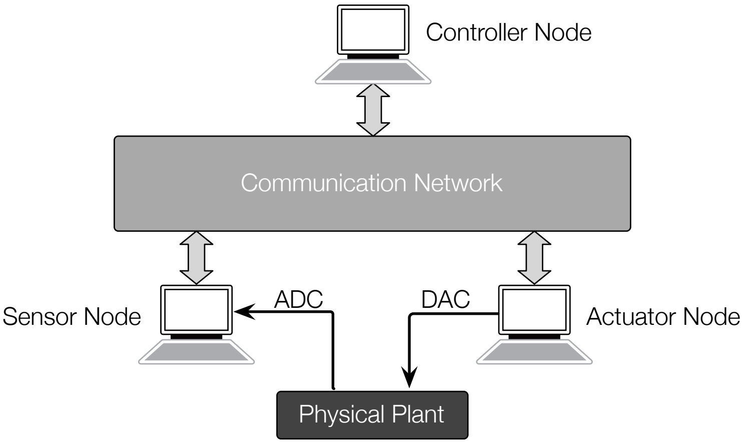


Figure 1.1: The scheme of networked control systems. Conventional point-to-point connections are replaced by a shared network.

The first NCS example is introduced by automobile industries in 1970's [100]. At that time, the motivations was the reduced cost for cabling, modularization of systems, and flexibility in car manufacturing. Nowadays, NCS technologies are widely applied in other industrial control applications. Theses applications include manufacturing automation [11, 111], automobile [30], aircraft [114], teleoperation [56], remote surgery [139], building automations [98], automated highway systems, vision-based manipulation [112], coverage control [27], environmental monitoring and surveillance [5].

Regarding the utilization of communication networks, the applications of NCSs can be roughly categorized into *shared-network control systems* and *remote control systems.* Using a shared-network for the sensor measurement and control commands transmission can reduce the complexity of connections. This class of NCS has the widest applications in the current industrial usages. On the other hand, a remote control system is referred to a complete control system remotely operated by a control unit. Examples are remote data acquisition systems and remote monitoring systems.

Challenges of control over networks

Despite of many potential advantages, the network solution for control systems introduces several issues, which differ from conventional system analysis and design, should be addressed:

i. *Delays and packet dropouts:* Exchanging data over a communication network results in non-ideal signal transmission. Delays and packet dropouts might appear during the transmission [9, 53, 123, 130]. Particularly, the network transmission delays are known as a source to jeopardize the stability and deteriorate the performance of NCSs. The value of delays strongly relates to network configurations, number of participators, routing transients, aggregate flows as well as network topologies. Hence, the transmission delays may be non-deterministic for NCSs.

ii. *Limited network resources:* Having multi-sensors and systems in a shared network, the consideration of network bandwidth becomes essential in the system design. With the limited amount of bandwidth available, the emphasis is placed on how to utilize a network more efficiently and optimally [31, 55, 94, 96, 122]. This causes the need of priority decision and scheduling issues for the data transmission in NCSs.

iii. *Synchronization:* The fusion of sensor data or coordination of actuation over a network require the synchronization of physical time for the local clocks of distributed components [32, 65, 100, 118]. Due to the inaccuracies in local clocks, the clock offset may drift away from each other in time. Hence, the observed time or durations of time intervals may differ for each component in the network. This might result in improper sensing, or even worse, unexpected actuating of systems. Therefore, the requirement of synchronization for NCSs is more essential than any other traditional point-to-point cabling system.

To cope with these problems caused by inserting the networks into control loops, the research of NCS is a multidisciplinary area affiliated with computer networking, information theory and control theory. Concerning the computer networking, for instance, the research focuses on designing network protocols so that the stability and performance of underlying NCSs are guaranteed. Typical results are [120, 128, 129], where a maximum allowable transfer interval (MATI) between two consecutive control inputs is determined to ensure closed-loop stability.

From the perspective of information theory, the emphasis is on the determination of a necessary data rate needed to stabilize a unstable plant. By using Shannon's results, the finite network-capacity stabilization problem is solved in [31, 55, 94, 122] for linear systems and [76, 95] for nonlinear systems.

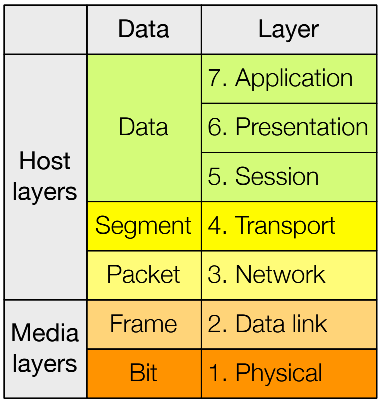
In this dissertation, the NCS is investigated from control theoretic perspective. Network-induced random delays and packet dropouts are mainly considered in the control loop. Analysis and design methods are developed to preserve the system stability and performance under network unreliability. Regarding the limited network resources, control methods involving systems and communication networks co-design are established to enable a performance trade-off from control and communication. The challenge of clock synchronization is out of the scope of this dissertation and is not further discussed. Since the network is considered as a design object, the most important network protocols used in control systems and their associated attributes are briefly introduced in the following section.

1.2 Network protocols and communication uncertainties

Generally speaking, communication networks can be categorized into two groups according to their application areas [59]. *Data networks* are specified by large data packets and high throughput. The transmission delays and packet dropouts in data networks often appear in a non-deterministic manner. Hence, their applications, in general, do not have hard real-time constraints. *Control networks*, on the contrary, shuttle small data packets frequently among system components to meet the real-time requirements. As a result, their transmission delays are deterministic or at least bounded. Furthermore, the data transmission of each component happens in sequence such that packet collisions (resulted in packet dropouts) are avoided.

In views of the transmission technology, both data and control networks can be further classified into *wired* and *wireless* networks. The wired network possesses many advantages like large bandwidth, high reliability and good security. However, due to the requirement of the wired connection between devices, the wired network has limited flexibility and mobility. Compared to the wired network, the wireless network has outstanding flexibility. However, its bandwidth is limited and the transmission is less reliable for control systems. For wired or wireless networks, the key element that distinguishes control networks from data networks is their *network protocols* being capable of supporting real-time or time-critical applications.

For the ease of development, the functionality of network protocols is conceptually arranged into different layers [145]. Each layer is a collection of similar functions which provide services to the layer above it and receives service from the layer below it. In the network protocols design, the OSI (Open System Interconnection) seven layer model developed by ISO (International Organization for Standardization) is mostly considered, see Fig. 1.2. Among the seven layers, the second layer, or more precisely the *medium access control* (MAC) sublayer of the second layer, controls the information transmission and determines the characteristics of delays and packet dropouts of networks [59, 74, 79, 89]. To



	Data	Layer
Host layers	Data	7. Application
		6. Presentation
		5. Session
	Segment	4. Transport
	Packet	3. Network
Media layers	Frame	2. Data link
	Bit	1. Physical

Figure 1.2: OSI seven layers of network protocols.

achieve the timing constraints and guarantee the performance of NCSs, the *MAC protocols* must be conjointly analyzed during the controller design of NCSs.

In the following section, the functionalities of most popular control networks used in NCSs are briefly introduced to clarify how the features of delays and packet dropouts affected by the MAC protocols. A more detail discussion and comparison of control networks can be bound in [74].

1.2.1 Wired control networks

Token-passing bus

The operation of Token-passing bus is shown in Fig. 1.3. A *token* is passed from node to node in a virtual ring, whereby the holder of the token has the access to the network. This ensures no data collisions and a maximal transmission delay. Furthermore, it provides excellent throughput and works well under heavy traffic with high degree of determinacy [68, 132]. During network operation, the token bus can dynamically add nodes to or remove nodes from the network. However, when large number of nodes are connected, a great percentage of the network time is used in passing the token. This increases the overall transmission delay and leads to performance degradation [74].

The typical commercial control networks based on token-passing protocols are PROFIBUS [132] and ControlNet [4].

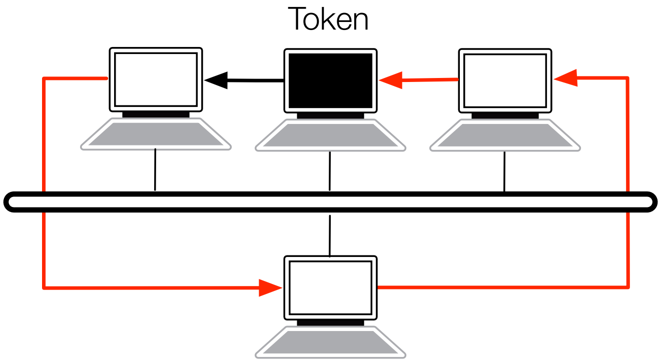


Figure 1.3: Token passing in Token Bus protocols

Controller area network

Controller area network (CAN) is a serial communication protocol developed by Bosch GmbH in 1983 for automotive industries [1]. The CAN protocol is designed for small data and uses CSMA/AMP (Carrier Sense Multiple Access, Arbitration on Message Priority) for collision avoidance. If simultaneous transmission occurs, the data collision is resolved by a priority based arbitration scheme to decide which one will be granted permission to continue transmitting. Hence, data with higher priority has guaranteed transmission delay. However, a data with high priority and large data size can block out completely the data with lower priority. The major disadvantage of CAN compared with the other networks is the slower data rate (maximum of 500 Kb/s). Therefore, CAN is only suitable for transmission of messages with small data sizes (less than 8 bytes) [3, 74].

Ethernet

Ethernet is known as the most popular communication networks and has the widest application domains [59]. In general, Ethernet can be categorized into two types: (i) hub-based Ethernet, which is commonly used for data exchange. It uses CSMA/CD (carrier sense multiple access with collision detection) mechanism for resolving contention on the communication medium, (ii) switched Ethernet with CSMA/CA (carrier sense multiple access with collision avoidance) mechanism, which is implemented in manufacturing and control environments.

Under CSMA/CD, a transmitting node first listens to the network to determine whether any other node on the network is occupying the medium. If the network is busy, the transmitting node waits until it becomes idle and continues the transmission. As soon as a collision is detected, the transmitting node stops transmitting and waits a random length of time, which is determined by the standard binary exponential backoff (BEB) algorithm, to retry its transmission.

CSMA/CA is derived from CSMA/CD. Unlike CSMA/CD, which deals with transmissions after a collision has occurred, CSMA/CA acts to prevent collisions before they happen. In CSMA/CA, before a transmitting node sends a packet, it checks the network whether the network is clear, i.e. no other node is transmitting at the time. If the network is clear, the packet is sent. Otherwise, the transmitting node waits for a randomly chosen period of time, and then checks again to see if the network is clear. This period of time is called the backoff factor, and is counted down by a backoff counter. The packet is transmitted only if the network is free and backoff counter is zero. If the network is not clear but the backoff counter expires, the backoff factor is set again, and the process is repeated.

Consequently, Ethernet is a non-deterministic protocol. Its network-introduced delay is randomly and highly depends on the traffic condition. However, Ethernet is cost-effective and has high bandwidth, popularity as well as versatility. This leads to a steady development and improvement of Ethernet technology for the application of complex control systems, see [30, 101, 121]. Furthermore, Ethernet supports the providing of different levels of communication qualities to different applications, e.g. Quality-of-Service (QoS) mechanism.

QoS mechanisms and functions are defined in MAC protocols to control data transmission. There are four types of MAC layer services characterized by QoS parameters such as delay, jitter, throughput, packet dropouts. These services can be created, changed, or deleted though the issue of Dynamic Service Addition (DSA), Dynamic Service Change (DSC), and Dynamic Service Deletion (DSD) messages. Each of these actions can be initiated by the MAC protocols and are carried out through a two or three-way-handshake mechanism in IP networks.

A number of architectures have been proposed to enable offering different levels of QoS in IP networks including the integrated services (IntServ) architecture and the differentiated service (DiffServ) architecture. In IntServ-based networks, network applications use the Resource reservation protocol (RSVP) to request and reserve resources through a network. With rapid growth of network applications, it is difficult to accept, maintain, and tear down thousands of reservations. A more suitable architecture for large-scale networks is the DiffServ architecture. In DiffServ-based networks, packets are marked according to the type of service they need. In response to these markings, routers and switches use various queuing strategies to tailor performance to their requirements. The IEEE 802.1 workgroups are typical examples supporting DiffServ-based QoS.

As mentioned later in this dissertation, by using the QoS concept, control systems and communication networks can be conjointly design such that a desired performance of NCSs can be easily achieved.

1.2.2 Wireless networks

Wireless Ethernet

Similar to CSMA/CA in wired Ethernet, the wireless Ethernet is implemented with collision avoidance mechanism. If a transmitting node wants a send a packet while the wireless network is busy, it sets its backoff counter to a randomly chosen value. As soon as the network is idle, the transmitting node waits first for an interframe space and a backoff time before sending packets. Due to the random nature of backoff time, any two nodes might have the same backoff time. The collisions cannot be entirely prevented. Thus, as soon as a packet is successfully received by its destination, the receiver confirms the sender by an acknowledgement packet (ACK). If the ACK is not received by the sender for a predefined time interval, the transmission is considered as unsuccessful and a retransmission of the packet is taken place. This results in non-deterministic network-induced transmission delays. Furthermore, due to the limited spectrum, time-varying channel and interference, setting up a wireless NCS is a challenging task.

In spite of above mentioned challenges, wireless networks present a number of unique advantages. e.g. rapid deployment, flexible installation, fully mobile operation and preventing cable wearing in the industrial automation. Therefore, a significant research effort has been devoted into the development and improvement of wireless technology for real-time applications [25, 49, 79, 105]. Typical examples are Bluetooth, IEEE 802.11 and IEEE 802.15.4 (Zigbee).

In conclusion, with increasing demands from real-time control applications, different control network protocols have been proposed with guaranteed network characteristics, e.g. deterministic transmission delay and dropouts. Based on these network specifications, analysis and design approaches are sequentially developed for NCSs such that stability and performance are preserved, see [9, 53, 96] for general overview. However, the main drawbacks with the sequential design process are that the systems are designed by robust control theories under the assumption of worst-case network-induced uncertainties. This might result in pool control performance if the worst-case network-induced uncertainties, e.g. worst-case delay, appear only rarely. Good control performance can be guaranteed by the sequential design process with the reservation of large network resources for each network application. However, this is an inefficient design for NCSs with the limited network resources and demanding performance. For NCSs, better performance can be achieved if a co-design approach is adopted where the control system is designed by taking the resource constraints into account and where the network specification is designed with the control performance in mind.

As an example, consider an NCS with a CSMA/CA based network. Within the network, the network applications are prioritized by a CSMA/CA algorithm generating waiting intervals (delays) for collision avoidance. The applications with higher priority have higher probability of shorter waiting delays and better performance, but need the provision of large network resources. It is proposed to re-specify the probability distributions of waiting delays under control performance and network capacity constraints so that the desired performance can be achieved at affordable network cost. In the practical network implementation, the probability distributions of waiting delays can be realized by choosing the backoff exponent and backoff period in the CSMA/CA algorithm. Control methods determining the probability distributions of waiting delays involves a concurrent consideration from the control and communication aspects and are the main focus in this dissertation.

1.3 Related work

The co-design problem of control systems refers to the development concept, where different perspectives of a system are brought into one design process. The first co-design example can be traced back to the 1970s [6, 48]. At that time, the limited word length, fixed-point calculations, and limited CPU speed were well-known constraints among control engineers. For the better use of computing resources, the conjoint designs of control algorithm and control software have received a considerable amount of attentions [7, 33, 45, 99, 133]. Nowadays, the shared networks are increasingly adopted in the control loops. The co-design problems towards control systems and network issues, such as limited bandwidth, delays and dropouts are widely open and deserve to be explored.

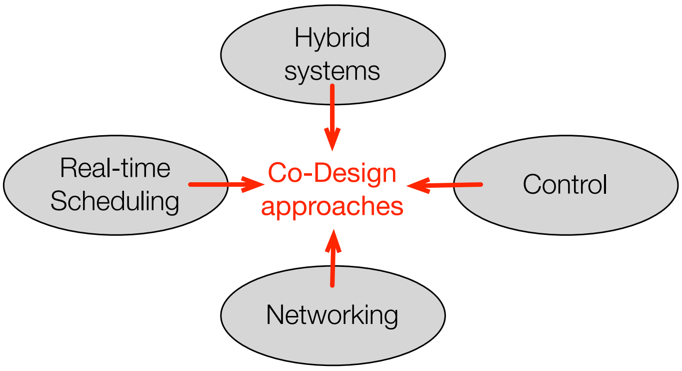


Figure 1.4: The integration of multidiscipline of control systems and communication co-design.

Aiming at performance at limited network resources, the concurrent design of control systems and communication networks for NCSs requires an integration of several disciplines, including *hybrid systems*, *control*, *networking* and *real-time scheduling* as shown in Fig. 1.4. Different analysis methodologies are applied to different network transmission characteristics. Generally, related works on the co-design of control systems and communication networks can be categorized into two categories: *deterministic* and *stochastic* design approaches, see [9, 53, 77, 123, 138] for a general overview. Some of the important results are summarized in the following.

In the framework of deterministic design approaches, the first work concerning control systems and communication networks co-design for NCSs can be found in [19], where a maximal allowable transmission interval (MATI) is determined to coordinate a set of time-discrete linear systems such that the stability of each system is guaranteed. The MATI is further extended for linear NCSs with lossy networks by using rate-monotonic scheduling algorithm in [17, 18], and for nonlinear NCSs by using perturbation theories in [127, 129], input-to-state stability in [97] and delay-impulsive systems in [93].

Similar deterministic co-design approaches can be found in [40, 75, 88, 124, 141, 144]. In [40], linear time-discrete NCSs over a limited bandwidth deterministic network are considered. The problem of optimal control and network scheduling is solved by using a mixed integer quadratic programming algorithm. The derived results are applied to control and schedule a car suspension system for validation. The design methodologies of guaranteeing acceptable control and communication performance for NCSs are studied in [75]. A dead-band controller and state estimator are proposed to dynamically adjust the the communication rate based on the control and communication performance requirement. In [88], a co-design of networked controllers and feedback scheduling policies is considered. A adaptive technique for controllers is established to enable a dynamic management of network though message scheduling. A gain scheduler middleware is proposed in [124], by which the output of existing controllers is modified by a gain scheduling algorithm with respect to the current network traffic conditions. As a result, conventional controllers designed without network consideration can be still utilized for NCSs. In [141], an algorithm to design a periodic sequence for the networking of sensors and actuators, under which the exponential stability of the NCS is preserved, is proposed. A co-design of predictive controller and network scheduling is studied in [144], where a scheduling algorithm is designed with the guarantee of system stability by ensuring the communication periods of systems within an analytical upper bound. The delayed control output is compensated by using delayed sensing data and previous control information. However, to generate the control prediction requires an exact knowledge of systems and delays. This approach is unsuitable for systems with modeling uncertainties and non-deterministic networks.

The advantage of MATI-based co-design algorithms is that it results in deterministic network scheduling protocols, which can be easily implemented into Token-ring, PROFIBUS and CAN bus. However, these algorithms require the worst-case consideration of network-induced delays and packet dropouts. This might lead to unnecessary conservative controller design when the worst-case delays or dropouts rarely happen in the practical network transmission.

To avoid the conservatism introduced by worst-case assumptions in the deterministic co-design approaches, the probability distributions of network attributes are considered in the system analysis. For this purpose, stochastic processes are introduced to accommodate the abrupt variations of network uncertainties within an analytical framework. In NCS applications, network-induced random delays and packet dropouts are modeled by Markov processes [42, 50, 66, 131]. The resulting NCS is a Markovian jump system [13, 21, 117, 142].

Its stability and stabilization conditions are derived by using stochastic Lyapunov functions (or functionals) [71] and presented in terms of linear matrix inequality (LMI) [13, 15].

Literature applying stochastic analysis methodologies to control systems and communication co-design can be found in [26, 62, 63, 79, 120, 136]. Issues concerning bandwidth limitations of NCSs with *ad hoc* wireless networks are addressed in [26]. A static sampling adaptation scheduling guaranteeing mean square stability is proposed for linear time-discrete NCSs with wireless networks. In [62], the optimal control problem of linear time-invariant (LTI) systems over lossy communication networks are explored. The control law is optimized by using stochastic dynamic programming and is derived in the form of a Riccati equation. However, this approach cannot be applied to NCSs with random delays. Stochastic optimal control and communication network co-design for NCSs with delays are considered in [63]. However, the network-induced delay is assumed to be less than one sampling interval. Therefore, this approach may be unsuitable for systems with longer time delay. In [79], a complex design index incorporating the network throughput, transmission delay and packet dropout probabilities is considered. A cross-layer framework is proposed to conjointly design the network and remote controllers such that the design index is optimized. Nonlinear NCSs employing Ethernet-like wireless and wireline networks are studied in [120]. A stochastic protocol developed by input-output stability is proposed for stabilizing a set of nonlinear NCSs with exogenous disturbances and random packet dropouts. The resource allocation problem of a communication network with bit-rate limitations is considered in [136]. An uniform quantization with white-noise errors is applied to model the effect of bit-rate limited networks. An optimal control performance is jointly achieved by allocating network resources and tuning controller parameters.

To sum up, stochastic design approaches removes the conservatism of their deterministic counterpart by considering the probability distributions of network-induced uncertainties. However, similar to deterministic design approaches, where research efforts are mainly devoted to develop scheduling algorithms such that system stability and network capacity constraints are jointly preserved. The stochastic analysis is applied either to stabilize control systems with random uncertainties or to allocate network resources under stability constraints. Until now, there are few works in the past literature, which jointly deal with the stability analysis, network resources and control performance at the same time. To the best knowledge of the author, the performance oriented conjoint design of NCS are mentioned in [75, 88] and [79, 136]. However, the network uncertainties, such as delay and packet dropouts, are excluded in [75, 136] and the network-indued delay is assumed to be less than one sampling interval in [88]. Although the random transmission delay and packet dropouts are considered in [79], no rigorous analysis is mentioned. The conjoint design of control systems and communication networks for NCS related issues is still in its infancy

This motivates an analytical exploration of co-design approaches jointly handling *random network-induced uncertainties*, *resource constraints* and *control performance*. In this dissertation, stochastic control methodologies are applied for the analysis of NCSs. A link between the stability/performance requirements of control applications and quality restrictions on the communication networks is built through statistical specification of underlying networks. Based on this, novel co-design approaches are developed for performance oriented NCS control.

1.4 Main Contribution and outlines of the dissertation

The presented dissertation focuses on the stochastic conjoint design of control system and communication network for linear time-invariant (LTI) NCSs. An important issue is the consideration of limited network capacity in the control design and the trade-off between control performance and network resources. The main contributions of this dissertation are three fold. First, stochastic control and analysis methodologies are taken into account and result in less conservative co-design approaches than conventional worst-case assumptions. Second, the proposed co-design approaches integrate the performance trade-off from control and communication. It enables the development of more efficient and affordable NCSs which can scale and adapt with limited network resources. Third, the application of the proposed co-design approaches requires no exact values of network uncertainties, but their distributions. This is simplifies the implementation in real systems and networks.

This dissertation is separated into two parts according to associated LTI systems. The first part deals with LTI systems with periodic data transmission rate and random network uncertainties, i.e. transmission delay and packet dropouts. The objective is to balance a guaranteed control performance versus required network resources. The second part concerns LTI systems with random sampling intervals and random network uncertainties. It is aimed to adapt the system data transmission rate to achieve an efficient network utilization. Results and primary work of this dissertation are published in [21–24, 57, 58, 134]. The structure of this work is give in Fig. 1.5. The associated outline of the presented results are given in the following.

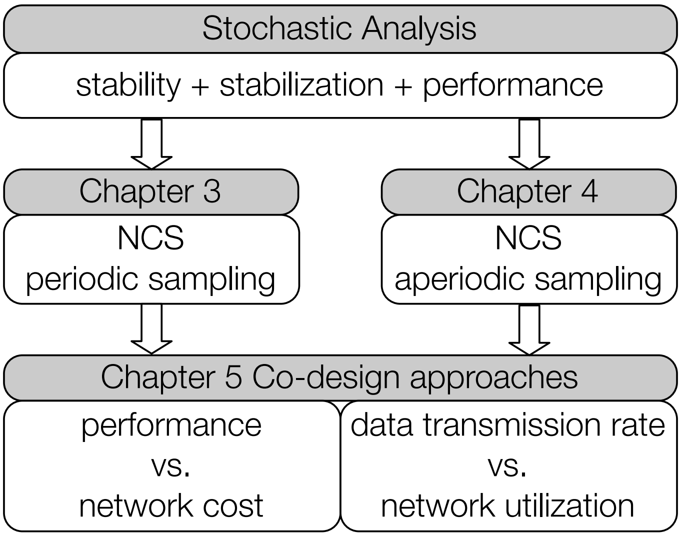


Figure 1.5: Outline of the dissertation.

In Chapter 2, the theoretical background is presented. To accommodate the stochastic variations as well as uncertainties[1] of networks within the analytical framework, a Markov process is introduced. The usefulness of Markov process in network uncertainty modeling is apparent since it enables to describe the random uncertainties by making use of the knowledge of their occurrence and statistical patterns, which truly reflects network attributes. Control systems containing Markovian parameters (or uncertainties modeled

[1] The network uncertainties are meant to network-induced delay and packet dropouts.

by Markov process) are termed Markovian jump systems (MJSs). The theoretic properties of MJSs are addressed and notions of stochastic stability are revisited. A stochastic Lyapunov function is illustrated for the stability analysis. Finally, the linear matrix inequality (LMI) and convex optimization are briefly introduced. Due to its numerical efficacy, all the stability and stabilization conditions given in this dissertation are derived in terms of LMIs.

Stochastic NCSs with periodic sampling interval and random network-induced uncertainties are considered in Chapter 3. The random network-induced delay and packet dropouts are modeled by a Markov process. In order to improve the performance, a novel delay-dependent switching controller is introduced. An MJS with mode-dependent delay is established in this chapter and the associated stability as well as stabilization conditions are derived by using stochastic analysis methodologies. Different to deterministic time-delay systems, the obtained stability as well as stabilization conditions are derived depending not only on delays, but also on their associated statistical properties. An innovative guaranteed control performance analysis is developed to maintain the stability and desired performance of NCSs under pre-defined statistical properties of delays.

Due to external traffic and limitations on network capacities, a network can be more efficiently utilized if the sampling rate of NCSs can be adapted according to network conditions. In Chapter 4, stochastic NCSs with random sampling intervals are considered. The random sampling interval together with random delays are reformulated into time-varying delays by using the input-delay approach. A set of indicator functions having independent identical distributions (i.i.d.) is introduced to describe the probabilistic occurrence of the time-varying delays. The resulting NCSs are randomly switched time-delay systems. Associated stability and stabilization conditions are obtained depending on the probabilistic distributions of sampling intervals and delays. In this chapter, a novel performance guaranteed design, which correlates the performance upper bound with probability distributions of sampling intervals, is proposed.

Two novel approaches aiming at performance oriented control system and communication network co-design are addressed in Chapter 5. In the first co-design approach, the Quality-of-Service (QoS) concept from the networking community is considered. Based on the results from Chapter 3, performance requirements of a control system and restrictions of a communication network are linked through statistical properties of the underling Markov process. QoS is then related to the ability of adjusting the probability transition rate of such Markov process. A cost-performance trade-off is achieved by appropriately parameterizing the Markov probability transition rate. According to the results from Chapter 4, the second co-design approach incorporates control performance and network usage in terms of probability distributions of associated data transmission rates. The goal is to develop a network usage efficient NCS. The performance of both considered approaches are explored in case studies. Benefits in terms of guaranteed control performance with efficient network usage are shown in the simulation results.

Chapter 6 verifies the proposed methods experimentally. Two different experiments are conducted. In the first experiment, a robotic manipulator is subjected to a QoS network. The first co-design approach addressed in Chapter 5 is used to cope with a cost-performance trade-off. The second experiment concerns a networked visual servo control system (NVSCS) with variable image transmission rate. The second co-design approach in Chapter 5 is applied to achieve a network usage-performance trade-off. In order to show the benefits of proposed approaches, benchmarks without co-design approaches are performed for comparison within the experimental validations.

The dissertation is concluded in Chapter 7 with a summary and discussion about future directions.

2 Stochastic Control Systems

As mentioned in the previous chapter, the random transmission delays and packet dropouts are the main issues in following chapters. The purpose of this chapter is to introduce the essential tools for the modeling of network uncertainties and analysis of stochastic control systems.

For the stochastic modeling, Markov processes are concerned. The Markov process, named after Andrey A. Markov (1856-1922), is an important class of stochastic processes with special feature called the *Markov property*. The Markov property allows the outcome of a Markov process at any time instant to depend only on the outcome that precedes it and none before that [103]. Markov processes are extensively used in the modeling of many communication phenomena [10, 29, 50, 66] e.g. transmission delays, packet dropouts and queuing mechanism, whenever probabilities are used to represent the unknown details of the communication networks. Within the framework of this dissertation, the effect of packet dropouts is formulated as an additional delay. The resulting random delay, which means the sum of additional delay and transmission delay, is modeled by a Markov process and termed *Markovian delay*.

A class of hybrid dynamical systems, whose discrete and continuous states are modeled by a random process and corresponding differential equations is categorized as *stochastic jump systems*. According to developed applications, two types of stochastic jump systems are considered in this chapter, e.i. *Markovian jump systems* (MJSs) and *randomly switched time-delay systems*. An MJS is a special class of stochastic jump systems whose switching between sub-systems is governed by a Markov process. Furthermore, a set of MJSs subject to a Markovian delay is classified as MJSs with mode-dependent delay. The theoretic properties of MJSs, MJSs with mode-dependent delay as well as randomly switched time-delay systems are addressed and the notions of stochastic stability are revisited. A Stochastic Lyapunov function is illustrated for stability analysis. The linear matrix inequality (LMI) and convex optimization are introduced due to their numerical benefits in system analysis and design.

2.1 Markov process

An independent stochastic process represents its outcome at any instant to depend only on the outcome that precedes it and none before that is called a Markov process, $\{r_t, t \geq 0\}$. In general, for a Markov process the time index t can be discrete or continuous. In addition, starting from some initial time $t = 0$, the process r_t changes its values in a finite (infinite) set randomly as time goes on. Due to the requirement of this dissertation, a continuous-time Markov process taking values in a finite set is considered. Within this work, a continuous-time Markov process is simply termed Markov process for abbreviation.

2.1.1 Continuous-time Markov process

Consider a continuous-time Markov process $\{r_t, t \geq 0\}$ taking values in a finite set $\mathcal{S} := \{1, \ldots, N\}$. Suppose the process r_t is in state (mode) i at time t_0, i.e. $r_{t_0} = i$, $i \in \mathcal{S}$. The transition probability that the process r_t jumps into the state j at time $t_0 + t$ is given by

$$\begin{aligned} &\mathbf{P}\{r_{t_0+t} = j | r_{t_0} = i, r_u, 0 \leq u < t_0\} \\ &\quad = \mathbf{P}\{r_{t_0+t} = j | r_{t_0} = i\}. \end{aligned} \tag{2.1}$$

Equation (2.1) means that, for a Markov process, the conditional distribution of the future mode at time $t_0 + t$, given all past states during time $0 \leq u \leq t_0$ and the present state at time t_0, depends only on the present state. Furthermore, if the transition probability $\mathbf{P}\{r_{t_0+t} = j | r_{t_0} = i\}$ is independent of initial time t_0 but only the elapsed time t, the Markov process is said to be *homogeneous*. Therefore, the transition probability in (2.1) reduces to

$$\mathbf{P}_{i,j}(t) = \mathbf{P}\{r_{t_0+t} = j | r_{t_0} = i\} \tag{2.2}$$

and satisfies

$$0 \leq \mathbf{P}_{i,j}(t) \leq 1, \quad \sum_{j \neq i} \mathbf{P}_{i,j} = 1.$$

The unconditional probability distribution of state $r_t = j$ is given by

$$\mathbf{P}_j(t) = \mathbf{P}\{r_t = j\} = \sum_i \mathbf{P}\{r_t = j | r_{t_0} = i\}\mathbf{P}\{r_{t_0} = i\} = \sum_i \mathbf{P}_i(t_0)\mathbf{P}_{i,j}(t). \tag{2.3}$$

Note that the occurrence probability of the transmission delay in a network at time t is depending on the current traffic condition, i.e. the current transmission delay. Therefore, the Markov processes concerned in this dissertation will be assumed to be homogeneous. Their joint probability distributions of different states can be specified by a linear differential equation, named *Chapman-Kolmogorov equation.*

Theorem 2.1 (**Chapman-Kolmogorov equation**)
For a homogenous Markov process $\{r_t, t \geq 0\}$ with $i, j \in \mathcal{S}$ and $t, s \geq 0$, its transition probabilities satisfy the Chapman-Kolmogorov equation given as

$$\mathbf{P}_{i,j}(t + s) = \sum_k \mathbf{P}_{i,k}(t)\mathbf{P}_{k,j}(s). \tag{2.4}$$

Proof:

$$\begin{aligned} \mathbf{P}_{i,j}(t+s) = \mathbf{P}\{r_{t+s} = j | r_{t_0} = i\} &= \sum_k \mathbf{P}\{r_{t+s} = j | r_t = k, r_{t_0} = i\}\mathbf{P}\{r_t = k | r_{t_0} = i\} \\ &= \sum_k \mathbf{P}\{r_t = k | r_{t_0} = i\}\mathbf{P}\{r_{t+s} = j | r_t = k\} \\ &= \sum_k \mathbf{P}_{i,k}(t)\mathbf{P}_{k,j}(s). \end{aligned}$$

■

Sojourn time and probability transition rate

The amount of time that a Markov process takes in one state for the change into next state is called *sojourn time.* Let δ_i denote the sojourn time at state i. The probability density function of $\{\delta_i > t\}$ can be derived as

$$F_{\delta_i}(t) = \mathbf{P}\{\delta_i > t\}, \quad t \geq 0. \tag{2.5}$$

According to (2.5), the probability of the event $\{\delta_i > t+s\}$ given $\{\delta_i > t\}$ becomes

$$\begin{aligned} F_{\delta_i}(t+s) &= \mathbf{P}\{\delta_i > t+s\} = \mathbf{P}\{\delta_i > t+s|\delta_i > s\}\mathbf{P}\{\delta_i > s\} \\ &= F_{\delta_i}(t)F_{\delta_i}(s) \end{aligned}$$

or

$$\log F_{\delta_i}(t+s) = \log F_{\delta_i}(t) + \log F_{\delta_i}(s). \tag{2.6}$$

Note that the only function satisfies (2.6) for arbitrary t and s is

$$\log F_{\delta_i}(t) = -\alpha_i t, \quad F_{\delta_i}(t) = \mathbf{P}\{\delta_i > t\} = e^{-\alpha_i t}.$$

This shows that the sojourn time of Markov processes has an exponential distribution [109]. The parameter α_i representing the transition rate out of state i. Let $\alpha_{i,i}$ denote the transition rate from i to i and $\alpha_{i,j}$ from i to j, respectively. A *probability transition rate*[1] $\mathcal{A} = (\alpha_{i,j})$, $i,j \in \mathcal{S}$ of the Markov process $\{r_t, t \geq 0\}$ is derived and satisfies

$$\alpha_{i,i} = -\alpha_i, \quad \sum_{j \neq i} \alpha_{i,j} = -\alpha_{i,i}, \quad \forall i,j \in \mathcal{S}.$$

Given the probability transition rate of a Markov process, the following lemma and theorem can be derived.

Lemma 2.1

$$\begin{aligned} &\textbf{(i)} \ \lim_{\Delta t \to 0} \frac{1 - \mathbf{P}_{i,i}(\Delta t)}{\Delta t} = \alpha_i, \\ &\textbf{(ii)} \ \lim_{\Delta t \to 0} \frac{\mathbf{P}_{i,j}(\Delta t)}{\Delta t} = \alpha_{i,j}, \quad i \neq j. \end{aligned}$$

Proof: According to (2.5), the probability that the Markov process r_t remains in i from state i in a small time interval Δt is

$$\mathbf{P}_{i,i}(\Delta t) = \mathbf{P}\{\delta_i > \Delta t\} = e^{\alpha_i \Delta t} = 1 - \alpha_i \Delta t + o(\Delta t). \tag{2.7}$$

Similarly, the probability of the Markov process r_t undergoes a jump of state from i in a small time interval Δt is given by

$$\mathbf{P}_{i,j}(\Delta t) = \mathbf{P}\{\delta_i \leq \Delta t\} = 1 - e^{\alpha_i \Delta t} = \alpha_i \Delta t + o(\Delta t). \tag{2.8}$$

where $o(\Delta t)$ is the infinitesimal of higher order terms than Δt. Divide (2.7) and (2.8) by Δt. The term $o(\Delta t)/\Delta$ tends to zero. Lemma 2.1 is derived. ■

[1] In some works, the Markov probability transition rate is named as transition generator.

Theorem 2.2 (**Kolmogorov equation**)
For all $i, j \in \mathcal{S}$ and $t \geq 0$,

$$\frac{d}{dt}\mathbf{P}_{i,j}(t) = -\mathbf{P}_{i,j}(t)\alpha_i + \sum_{k\neq j}^{N} \mathbf{P}_{i,k}\alpha_{k,j}.$$

Proof : According to Definition 2.1, it has

$$\begin{aligned}\mathbf{P}_{i,j}(t+\Delta t) - \mathbf{P}_{i,j}(t) &= \sum_{k=1}^{N} \mathbf{P}_{i,k}(t)\mathbf{P}_{k,j}(\Delta t) - \mathbf{P}_{i,j}(t)\\ &= \sum_{k\neq j}^{N} \mathbf{P}_{i,k}(t)\mathbf{P}_{k,j}(\Delta t) - \big(1 - \mathbf{P}_{i,i}(\Delta t)\big)\mathbf{P}_{i,j}(t)\end{aligned}$$

and becomes

$$\lim_{\Delta\to 0} \frac{\mathbf{P}_{i,j}(t+\Delta t) - \mathbf{P}_{i,j}(t)}{\Delta t} = \lim_{\Delta\to 0} \left\{ \sum_{k\neq j}^{N} \mathbf{P}_{i,k}(t)\frac{\mathbf{P}_{k,j}(\Delta t)}{\Delta t} - \frac{\big(1 - \mathbf{P}_{i,i}(\Delta t)\big)}{\Delta t}\mathbf{P}_{i,j}(t) \right\}. \quad (2.9)$$

Apply Lemma 2.1 to (2.9) and it completes the proof. ■

Let

$$\mathcal{A} = (\alpha_{i,j}), \quad \mathcal{T}(t) = (\mathbf{P}_{i,j}(t)), \quad i, j \in \mathcal{S}$$

represent the Markov probability transition rate and transition probability respectively. The results of Lemma 2.1 and Theorem 2.2 imply

$$\mathcal{T}(t) = e^{\mathcal{A}t}. \quad (2.10)$$

Equilibrium behavior and limiting probabilities

Given any two two state i and j, the state j is said to be *accessible* from the state i, if there exists a positive probability $\mathbf{P}_{i,j}(t)$ of reaching the state j in certain time t. If every state of a Markov process is accessible, the Markov process is said to be *irreducible*. For any irreducible Markov process, if the Markov process starting from i and returns to i at irregular times, the Markov process is said to be *aperiodic*.

For any irreducible and aperiodic Markov process r_t with finite states, its probability distribution of each state, $\mathbf{P}_j(t)$, $j \in \mathcal{S}$, will converge to a stationary distribution regardless of the initial probability distribution. Such feature of Markov process is called *equilibrium behavior*. The equilibrium behavior is governed by the *limiting probabilities*, which are defined as

$$\bar{\mathbf{P}}_j = \lim_{t\to\infty} \mathbf{P}_{i,j}(t), \quad \sum_j \bar{\mathbf{P}}_j = 1, \quad (2.11)$$

and do not depend on the initial state i. The proof is essentially based on Theorem 2.1.

Taking the limit as $t \to \infty$ in (2.4) and considering (2.11), it has

$$\bar{\mathbf{P}}_j = \sum_k \bar{\mathbf{P}}_k \mathbf{P}_{k,j}(t). \quad (2.12)$$

Differentiating both sides of (2.12), it results in

$$0 = \frac{d}{dt}\bar{\mathbf{P}}_j = \sum_k \bar{\mathbf{P}}_k \frac{d}{dt}\mathbf{P}_{k,j}(t) = \sum_k \bar{\mathbf{P}}_k \alpha_{k,j}, \quad \forall j \in \mathcal{S}.$$

This concludes the following Theorem.

Theorem 2.3 (**Limiting probabilities**)
Consider an irreducible, ergodic Markov process r_t with a transition generator $\mathcal{A} = (\alpha_{i,j})$, $i, j \in \mathcal{S}$. There exist limiting probabilities for each state satisfying

$$\sum_i \bar{\mathbf{P}}_i \alpha_{i,j} = 0, \quad \sum_i \bar{\mathbf{P}}_i = 1. \tag{2.13}$$

Theorem 2.3 shows that the probability transition rate determines the limiting probabilities of a Markov process.

2.1.2 Strong Markov process

The Markov property says that the probability of r_{t+s}, the Markov process r_t depends only on r_s (not on r_u, $0 \leq u < s$). However, this is not the case for the trajectory of time delay systems, since the trajectory of time delay systems dependent not only on the present state but also the delayed interval [46]. To cast the problem into the framework of Markov process, the definition of a *strong Markov process* is introduced.

Definition 2.1 (**Strong Markov process**)
Let $\{r_t, t > 0\}$ be a Markov process taking values in a finite set $\mathcal{S}$. Define an (random) optional time T with $\mathbf{P}(T = \infty) = 0$. Then r_t is strongly Markovian at T if the following condition holds

$$\mathbf{P}(r_{t+T} \in \mathcal{S} | r_u, 0 \leq u \leq T) = \mathbf{P}(r_{t+T} \in \mathcal{S} | r_T). \tag{2.14}$$

for all $t > 0$.

Definition 2.1 has the representation that the probability of r_{t+T}, conditioned upon the history up to T, equals the probability of r_{t+T}, conditioned on r_T only. Since (2.14) holds for any finite the optimal time T, any strong Markov process is also a Markov process.

As mentioned before, Markov processes with finite states have many applications in the networking community. It is of interest to model the communication phenomena, e.g. the transmission delays, of real networks by a Markov process. In this application, however, the Markov probability transition rate is not given but only an observation of the process is available. The coming section is devoted to determine a probability transition generator of a Markov process relies on an observation of the process.

2.1.3 Parameter identification of network-induced transmission delay

Suppose a network-induced transmission delay is observed equidistantly by a series of packets within a time interval $0 < t_1 < \ldots < t_n = T$, where $h = t_{k+1} - t_k$. The corresponding observed delay values are $\{\tau_1, \ldots, \tau_n\}$ and their empirical regularities are described by a Markov process r_t, with finite state $\mathcal{S} := \{1, \ldots, N\}$ and probability transition rate $\mathcal{A}$. Based on the observed data, the number of transitions from a state i to j within a time step h, i.e. $c_{i,j}(h)$, $i, j \in \mathcal{S}$, can be statistically determined. With the knowledge of $c_{i,j}(h)$, the number of transitions from the state i can be calculated by

$$c_i(h) = \sum_{j=1}^{N} c_{i,j}(h).$$

The transition probability from a state i to j within a time step h can be obtained by

$$\hat{\mathbf{P}}_{i,j}(h) = \frac{c_{i,j}(h)}{c_i(h)}$$

and the associated transition matrix becomes $\hat{\mathcal{T}}(h) = (\hat{\mathbf{P}}_{i,j}(h))$. According to (2.10), it implies

$$\hat{\mathcal{T}}(h) = e^{\mathcal{A}h}.$$

As a result, the approximate probability transition rate $\mathcal{A}$ can be determined by

$$\mathcal{A} = \frac{1}{h} \ln\big(\hat{\mathcal{T}}(h)\big).$$

Consider the Taylor expansion of $\ln\big(\hat{\mathcal{T}}(h)\big)$ and ignore the higher order terms, the above formula reduces to

$$\mathcal{A} \approx \frac{1}{h}\big(\hat{\mathcal{T}}(h) - I\big), \tag{2.15}$$

where I is an identical matrix.

As a summary, the algorithm for determining the probability transition rate of a Markov process based on discretely observed data is presented in Algorithm 2.1.

Algorithm 2.1 (**Determination of probability transition rate**)

```
 Input:  Observation series $\{\tau_1, \ldots, \tau_N\}$
Output:  Probability transition rate - $\mathcal{A}$
   (1)   Derive $c_{i,j}(h)$, $c_i(h)$ based on $\{\tau_1, \ldots, \tau_N\}$.
   (2)   Calculate $\hat{\mathcal{T}}(h)$ by $\hat{\mathbf{P}}_{i,j}(h) = c_{i,j}(h)/c_i(h)$.
   (3)   Determine $\mathcal{A}$ by $\mathcal{A} = \big(\hat{\mathcal{T}}(h) - I\big)/h$.
```

According to [12, 90, 91], the number of observation samples determines the accuracy of probability transition rate. For more accurate estimation of $\mathcal{A}$, the more observation samples N are required.

2.2 Stochastic jump systems

Stochastic jump systems are first introduced in the 1960's by Krasovskii and Lidskii to study the abrupt structure variations of dynamical systems, e.g. component failures, sudden environmental disturbances and changing subsystem interconnections [69, 70]. In general, stochastic jump systems are a class of hybrid systems, whose discrete and continuous states are modeled by random process and corresponding differential equations.

Within this dissertation, two types of stochastic jump systems are studied according to the applications. Stochastic NCSs with periodic sampling and Markovian transmission delays are modeled as Markovian jump systems (MJSs) with mode-dependent delay. An MJS contains a set of sub-systems and the switching (jump) between sub-systems is governed by a continuous-time Markov process with finite states. If the set of sub-systems are subjected to Markovian delays, an MJS with mode-dependent delay is established.

The second type of stochastic jump systems concerns stochastic NCSs with aperiodic sampling. By using the input-delay approach [38], the aperiodic sampling intervals are reformulated into time-varying delays. Combing the time-varying delays cased by aperiodic sampling with the network-induced delays, a randomly time-varying delay is derived. For the ease of analysis, the compound delay is reformulated into N intervals, i.e. N different time-varying delays. Associated with the delay intervals, appropriate sub-systems are determined. A set of indicator functions is introduced to conduct the random switching between sub-systems. The resulting system is a randomly switched time-delay system.

In the following section, the mathematical descriptions of MJSs, MJSs with mode-dependent delay and randomly switched time-delay systems are introduced for later analysis.

2.2.1 Markovian jump systems

The mathematical representation of an MJS is given as following

$$\dot{x}(t) = A(r_t)x(t) + B(r_t)u(t), \tag{2.16}$$

with the initial condition $x(t=0) = x_0$ and $r_{t=0} = r_0$, where $A(r_t) \in \mathbb{R}^{n\times n}$, $B(r_t) \in \mathbb{R}^{n\times m}$, $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the control input and $\{r_t, t \geq 0\}$, $r_t \in \mathcal{S}$, is a homogeneous Markov process governing the abrupt switching between different modes. According to Theorem 2.2, the switching of the MJLS is described by the probability transitions

$$\mathbf{P}\{r_{t+\Delta t} = j | r_t = i\} = \begin{cases} \alpha_{i,j}\Delta t + o(\Delta) & \text{if } i \neq j, \\ 1 + \alpha_{i,i}\Delta t + o(\Delta t) & \text{otherwises,} \end{cases}$$

where $\alpha_{i,i} = -\sum_{i\neq j} \alpha_{i,j}$, for all $i, j \in \mathcal{S}$ and $\lim_{\Delta t\to 0} \frac{o(\Delta t)}{\Delta t} = 0$.

It is noted that if the Markov process $\{r_t, t \geq 0\}$ has only one mode, the MJS in (2.16) is reduced to a deterministic linear system.

MJS with mode-dependent delay

Assume the state $x(t)$ and control input $u(t)$ of the MJS in (2.16) are subjected to random delays, which are also modeled by Markov process $\{r_t, t \geq 0\}$, $r_t \in \mathcal{S}$. This formulation results in MJSs with mode-dependent delay and has the mathematical presentation as

$$\begin{aligned} \dot{x}(t) &= A(r_t) + A_1(r_t)x(t-\tau(r_t)) + B(r_t)u(t) + B_1(r_t)u(t-\tau(r_t)), \\ x(s) &= \phi(s), \quad s \in [-\bar{\tau}, 0], \end{aligned} \tag{2.17}$$

where $A(r_t) \in \mathbb{R}^{n\times n}$, $A_1(r_t) \in \mathbb{R}^{n\times n}$, $B(r_t) \in \mathbb{R}^{n\times m}$, $B_1(r_t) \in \mathbb{R}^{n\times m}$, $\tau(r_t)$ is the delay and $\bar{\tau} = \max_{r_t\in\mathcal{S}}\{\tau(r_t)\}$. The initial condition of system (2.17) is specified as $x(s) = \phi(s)$, $s \in [-\bar{\tau}, 0]$. It means that the state evolution $\{(x(t), r_t), t \geq 0\}$ depends on its past history and contradicts the Markov property. However, the Markov property can be recovered by using the strong Markov Definition 2.1. Let $\chi(t)$ be a process taking values

$$\chi_s(t) = x(t+s), \quad t-\bar{\tau} \leq s \leq t.$$

Then $\{(\chi_s(t), r_t), t \geq 0\}$ is a strong Markov process and contains the Markov property as defined before.

2.2.2 Randomly switched time-delay systems

The second type of stochastic jump systems concerns a randomly switched time-delay system. Define a switching signal $d(t) := \{1, \ldots, N\}$. The mathematical representation of a randomly switched time-delay system is given as

$$\begin{aligned} \dot{x}(t) &= A_{d(t)}x(t) + A_{1d(t)}x(t-\tau_{s(t)}(t)) + B_{d(t)}u(t) + B_{1d(t)}u(t-\tau_{d(t)}(t)), \\ x(s) &= \phi(s), \quad s \in [-\bar{\tau}, 0], \end{aligned}$$

where $A_{d(t)} \in \mathbb{R}^{n\times n}$, $A_{1d(t)} \in \mathbb{R}^{n\times n}$, $B_{d(t)} \in \mathbb{R}^{n\times m}$, and $B_{1d(t)} \in \mathbb{R}^{n\times m}$. The variable $\tau_{d(t)}(t)$ denotes the delay and $\bar{\tau} = \max_i^N\{\tau_i(t)\}$. The initial condition of (2.18) is specified as $x(s) = \phi(s)$, $s \in [-\bar{\tau}, 0]$.

The switching signal $d(t)$ can be reformulated by a set of indicator functions

$$\beta_i = \begin{cases} 1, & d(t) = i, \quad i = 1, \ldots, N, \\ 0, & \text{otherwise.} \end{cases}$$

As a result, the above system can be written as

$$\begin{aligned} \dot{x}(t) &= \sum_{i=1}^{N} \beta_i \Big(A_i x(t) + A_{1i}x(t-\tau_i(t)) + B_i u(t) + B_{1s(t)}u(t-\tau_i(t)) \Big), \\ x(s) &= \phi(s), \quad s \in [-\bar{\tau}, 0]. \end{aligned} \tag{2.18}$$

Note that the indicator function β_i has binary values, i.e. one or zero. if it has Bernoulli distribution, its probability becomes

$$\mathbf{P}\{\beta_i = 1\} = p_i, \quad \sum_{i=1}^{N} p_i = 1.$$

Furthermore, if the indication function takes values independently of one another (memoryless) and the sojourn time is exponentially distributed, the function β_i recovers to a Markov process with two states. In such case, the randomly switched time-delay system (2.18) is identical to the MJS with mode-dependent delay in (2.17).

2.3 Stochastic stability and controllability

Stability is one of the major concerns in the design and synthesis of a closed-loop control system. Similar to the deterministic system settings, there are different definitions of stability for stochastic systems. For the system in (2.17), the following definitions of four stability notations are taken from [71].

Definition 2.2 (**Stochastic stability** [71])
Markovian jump linear system (2.17) with $u(t) = 0$ is said to be

(i) stochastically stable (SS) for any initial condition $x(t_0, r_{t_0})$, if there exists a constant $\varepsilon(\phi(\cdot), r_0)$ such that

$$\mathbb{E}\bigg\{\int_0^\infty ||x(t)||^2 dt|x(t_0, r_{t_0})\bigg\} \leq \varepsilon(\phi(\cdot), r_0);$$

(ii) mean square stability (MSS) for any initial condition $x(t_0, r_{t_0})$, if

$$\lim_{t\to\infty} \mathbb{E}\big\{||x(t)||^2|x(t_0, r_{t_0})\big\} = 0;$$

(iii) mean exponential stability (MES) for any initial condition $x(t_0, r_{t_0})$, if there exist positive constants b, and ρ such that for all $t \geq t_0$

$$\mathbb{E}\big\{||x(t)||^2|x(t_0, r_{t_0})\big\} \leq b||x(t_0, r_{t_0})||^2 e^{-\rho(t-t_0)}.$$

It is clearly that MES implies MSS and SS. Another less conservative stability notion is almost sure stability. The notation is given in the following definition.

Definition 2.3 (**Almost sure stability** [71])
Markovian jump linear system (2.17) with $u(t) = 0$ is said to be almost sure stable for any initial condition $x(t_0, r_{t_0})$, if

$$\mathbf{P}\bigg\{\lim_{t\to\infty} \big|x(t)|x(t_0, r_{t_0})\big| = 0\bigg\}.$$

The stability concepts of Definition 2.2 imply almost sure stability, but almost sure stability does not imply the stability concepts of Definition 2.2. Throughout this work, the notion of MES is mainly considered in the stability analysis.

2.3.1 Stochastic Lyapunov-Krasovskii functional

The Lyapunov-Krasovskii functional is well-known as an efficient tool for determining the stability of deterministic time-delay systems. In stochastic stability analysis, the stochastic Lyapunov-Krasovskii functional plays the same role for stochastic systems with delay. Roughly speaking, a Lyapunov-Krasovskii functional is an index measuring the values of the state trajectory $x(t)$ within the delay interval $[t - \tau, t]$, i.e. $x(t + s)$, $s \in [-\tau, 0]$. The stability is ensured if the derivative of the Lyapunov-Krasovskii functional with respect to the time is non-positive.

For stochastic Lyapunov-Krasovskii functionals, the idea of non-positive time derivative has to be refined and the notion of supermartingale is recalled. More specifically, let $V(\Xi(t))$ be a stochastic Lyapunov functional and $\mathcal{L}$ be an infinitesimal generator of $\Xi(t)$, where $\Xi(t) = x(t+s),\ s \in [-\tau, 0]$. The time derivative of $V(\Xi(t))$ is given in the following definition.

Definition 2.4 [85] Let $\mathcal{L}$ be the weak infinitesimal operator and give a function $V(\Xi(t))$, then the operator $\mathcal{L}$ acting on $V(\Xi(t))$ is defined as

$$\mathcal{L}V(\Xi(t)) = \lim_{\Delta t \to 0^+} \frac{1}{\Delta t}\Big\{\mathbb{E}\{V(\Xi(t+\Delta t)|\Xi(t))\} - V(\Xi(t))\Big\}.$$

Non-positive $\mathcal{L}V(\Xi(t))$ indicates that $\Xi(t)$ is supermartingale and means that the stochastic system is stochastically stable. The more precise statement is the following theorem.

Theorem 2.4 (**Stochastic Lyapunov-Krasovskii functional**)
For a stochastic jump system with $\Xi(t) = x(t+s)$, $s \in [-\tau, 0]$ and infinitesimal generator $\mathcal{L}$, if there exists a stochastic Lyapunov-Krasovskii functional $V(\Xi(t))$ and continuous non-decreasing functions u, v and w, where $u(s)$ and $v(s)$ are positive for $s > 0$, and $u(0) = v(0) = 0$ such that

(i) $V(0) = 0$,

(ii) $u(||\Xi(0)||) \leq V(\Xi(t)) \leq v(||\Xi(0)||_c)$, where $||\Xi(0)||_c = \max_{s\in[-\tau,0]} ||x(s)||$,

(iii) $\mathcal{L}V(\Xi(t)) \leq -w(||\Xi(0)||)$,

then the system is MES.

Note that Theorem 2.4 is limited to MES. The reason for focusing on this aspect of stochastic stability is that MES implies SS, MSS and almost sure stability. The operator $\mathcal{L}$ goes into the usual Lyapunov operator $dV(\Xi(t))/dt$, when the process $\Xi(t)$ is deterministic and can be described by a system of differential equations. By using the stochastic Lyapunov functional, it is possible to verify the qualitative properties of the trajectories $\Xi(t)$. Furthermore, the stochastic Lyapunov functional can also be used for designing control laws with stabilizing conditions.

As in the deterministic case, however, a general difficulty is to find suitable stochastic Lyapunov functionals. The most widely used Lyapunov candidate is the quadratic stochastic Lyapunov-Krasovskii functional. Due to numerical efficiency, the stability and controller design conditions obtained based on quadratic Lyapunov-Krasovskii functionals are cast into the form of linear matrix inequalities (LMIs). In Section 2.4, features of convex optimization and LMI are briefly presented.

2.3.2 Controllability

The controllability and observability introduced by Kalman in 1963, are known as a key element in the development of system analysis. For deterministic systems, the notions of controllability and observability concerning deterministic systems are given by differential or difference equations. For stochastic jump systems, on the contrary, there are no similar notations. Depending on the randomness of underlying stochastic processes, different notions of controllability are proposed [64, 87, 115, 140] in the literature. The definition of *relative controllability* is introduced in this dissertation.

Definition 2.5 (ε**-controllability**)
An initial condition $x(t_0) = x_0$ of the system (2.17) and (2.18) is ε-controllable with probability σ over the time interval $[t_0, t_f]$ if there exists a control law $u(t, x(t))$ such that

$$\mathbf{P}\{||x(t_f)||^2 \geq \varepsilon | x(t_0) = x_0\} \leq 1 - \sigma.$$

The name *relative* controllability is chosen to stress the difference with dichotomic controllability classification in deterministic settings. The relative controllability measures a relative degree of controllability, and can be used to rank candidate actuator configuration. This is specially interesting for systems with flexible structures, such NCSs, where there are many degrees of freedom in the choice of components location. The notion of relative controllability is extended to *global* controllability in [64, 87].

The algebraic test of ε-controllability for LTI MJLs is given in [64, 87], whereas the algebraic test of global controllability for MJLs with constant delay is determined in [140]. The observability and simple rank-test conditions of ε-controllability for stochastic jump system with time-varying or random delay are still open. Throughout this dissertation, the stochastic jump systems are considered to be global controllable and observable.

2.4 Convex optimization and linear matrix inequality

Generally, a convex optimization is a subfield of optimization problems concerning the minimization of convex functions. Given a real-valued function $f(x)$, which is said to be *convex* if the function lies below or on the straight line between any two points within a interval, see Fig. 2.1. The most important feature of a convex function refers to the uniqueness of extremum. That means, any minimum (or maximum) found in a convex optimization problem is the best achievable solution. Convex optimization has applications in a wide range of disciplines. One of typical examples in the control theory is the linear matrix inequality (LMI).

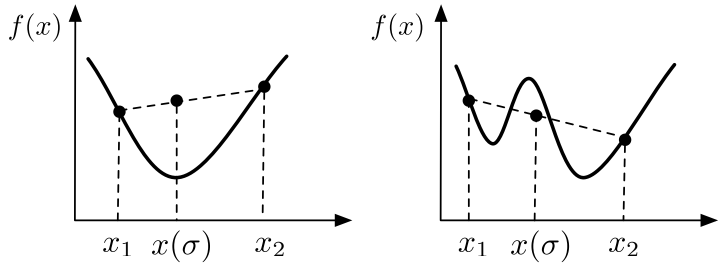


Figure 2.1: Illustration of convex function (links) and non-convex function (right).

An LMI approach refers to a kind of convex optimization problems in which its constraints appear as LMIs. The LMI constraints on a vector $y = [y_1, \ldots, y_m] \in \mathbb{R}^m$ have the form

$$F(y) = F_0 + y_1 F_1 + \ldots + y_n F_m < 0, \tag{2.19}$$

where $F_i = F_i^T \in \mathbb{R}^{n \times n}$, $i = 1, \ldots, m$ are known constant real matrices. The optimization problem is

$$\begin{aligned} &\text{minimize } c^T y \\ &\text{s.t. } F(y) < 0, \end{aligned}$$

where $c \in \mathbb{R}^n$ and $F(z) < 0$ means $F(z)$ is negative definite. The simplest LMI arising in the control theory is the so-called *Lyapunov inequality* [15]

$$A^T Y + YA < 0,$$

where $A \in \mathbb{R}^{n\times n}$ is known and $Y = Y^T \in \mathbb{R}^{n\times n}$ is the variable to be determined. Let $Y_1, \ldots, Y_m$ be a basis for the symmetric matrix Y, where $m = n(n+1)/2$ due to $Y = Y^T$. Consider $F_0 = 0$ and $F_i = A^T Y_i + Y_i A$. The Lyapunov inequality can be reformulated into (2.19) and yields

$$F(Y) = \sum_{i=1}^{m} F_i < 0.$$

The formulation of an LMI is particularly attractive in the control community due to the following reasons:

- **Numerical efficiency:** The LMI problem can be solved very efficiently by using interior-point method. As a results, a numerical solution can be easily found even if no analytical or closed-form solution is known.

- **Optimality:** Any feasible solution in LMI satisfied the inequality constraints and minimizes the corresponding convex cost function.

- **Multi-criteria:** The LMI formulation cast many different specifications in the analysis and design processes into a single criterion. This enables the exploration of trade-offs, e.g. performance and stability.

As shown in the later chapters, the stability and controller design conditions are derived in form of LMIs. The delays and sampling intervals are expressed in a single LMI, which allows a trade-off between delay length, sampling rate and stability.

2.5 Summary and discussion

In order to be able to follow the control approaches developed in this dissertation, the necessary mathematical backgrounds concerning Markov processes, stochastic jump systems, stochastic stability and linear matrix inequalities (LMIs) are illustrated in this chapter.

An independent stochastic process whose future state depending only on the present state and ignoring its past is called Markov process. Markov processes are applied to a wide variety of problems involving random uncertainties, such as telephone traffic, inventory control, machine breakdown and repair, air-traffic control, and communication networks. In the modeling of communication phenomena, e.g. transmission delays, packet dropouts and queuing mechanism, the Markov processes and Markovian properties are extensively used in the past literature [10, 29, 50, 66]. Particularly for the transmission delays, the main focus of this dissertation, their probabilistic appearance is characterized by a Markov probability probability transition rate, which can be determined by Algorithm 2.1 and demonstrates a good agreement with real networks [131]. According to Markovian properties, the limiting probabilities of delays are determined by probability transition rates. This enables a stochastic conjoint design of systems as well as networks as illustrated in Chapter 5.

Hybrid dynamical systems subjected to random switching signal are categorized as stochastic jump systems. Stochastic jump systems have the major merits due to their capability of maintaining an acceptable behavior and meeting some performance requirements even in the presence of abrupt changes in the systems [28]. These changes, within the framework of this dissertation, refer to the network uncertainties, i.e. transmission delays. Based on the later applications, two types of stochastic jump systems are considered in this chapter. NCSs with aperiodic sampling rate and Markovian delays is modeled as Markovian jump systems (MJSs) with mode-dependent delay, whereas NCSs with aperiodic sampling is formulated as a randomly switched time-delay systems. Towards the analysis of stochastic jump systems, the definitions of stochastic stability are revisited. The notion of mean exponential stability (MES) is mainly considered in following chapters since MES implies stochastic stability (SS) and mean square stability (MES). Moreover, a quadratic stochastic Lyapunov-Krasovskii method is considered for examining MES of stochastic jump systems with delay.

MES of a stochastic jump system can be guaranteed by a negative definite Lyapunov inequality. Due to the numerical efficiency, the related Lyapunov inequality is reformulated into an LMI. As shown later, the stability and controller design conditions are derived by means of different specifications, e.g. delays and sampling intervals, in a single criterion, which allows a trade-off between performance and stability.

This chapter is mainly based on [14, 103] and [71]. For more details on Markov process theories and stochastic jump systems, the reader is encouraged to look into [67, 108] and [20, 87]. A very good introduction of LMI and convex optimization is further presented in [15, 16].

3 Stochastic NCS with Periodic Sampling and Random Delay

The use of communication networks in the automation technologies has many advantages such as low cost, high flexibility, easy installation and maintenance. However, the use of a communication network comes at the price of non-ideal signal transmission: the sampled data sent through the network, e.g. Ethernet, experience *random transmission delays* and *random packet dropouts* as discussed in [9, 53]. The random delays, particularly, are well known as a source of instability and deteriorates the control performance [44, 84, 107]. It would be desirable to have a analysis and design approach, which can ensure the stability and guarantee the control performance in the presence of random delays.

Traditionally, random delays are treated as constant by considering their worst case. Based on this simplification, various approaches from the literature [38, 72, 78, 82, 89, 104, 106, 143], refer to Section 1.3 for a detailed discussion, have been proposed to cope with delays. However, these analysis and design results are derived for the worst-case delay, and discard the probabilistic distribution of delays. This might result in conservative controller design for systems with random delays. Studies with random delays are available in [92, 100, 117, 135, 137, 142][1], where Markovian processes are used for the modeling of delays. However, the associated stability and design conditions are determined by ignoring the impact of data sampling rate. Furthermore, these results require the exact knowledge of Markov probability transition rates. Any perturbation on it could lead to instability or affect the control performance.

Aiming at these shortcomings in the existing literature, the major innovation in this chapter is to develop an analysis and design approach, which involves random delays, packet dropouts and sampling rate in a single criterion. According to this criterion, guaranteed bounds on stability regions are determined and expected control performance are guaranteed in the presence of uncertain Markovian delay models. As a result, the stability restrictions and performance requirements of NCSs with periodic sampling are linked through network specifications (or statistical features of delays). This correlation motivates a novel co-design approach of control systems and communication networks as discussed in Chapter 5.

The reminder of this chapter is organized as follows. First, the network-induced transmission delay and packet dropouts are modeled by a Markov process and a MJLS with mode-dependent delay is introduced in Section 3.1. The stability analysis and controller design algorithms are presented in Section3.2 for NCSs with state-feedback controller, and in Section 3.3 for NCSs with output-feedback controller. Towards the uncertainties in the delay modeling, the guaranteed control performance is addressed in Section 3.4. Finally, the chapter is closed with summary and discussion in Section 3.5. The introductions of software tool for the controller design algorithms derived in chapter are given in the Appendix A.1.1.

[1] The detailed discussion is given in Section 1.3.

3.1 MJLS with random delay

Consider an LTI system as the physical plant

$$\begin{cases} \dot{x}(t) &= Ax(t) + B\bar{u}(t), \\ y(t) &= Cx(t), \end{cases} \tag{3.1}$$

where $x \in \mathbb{R}^n$ is the state, $\bar{u} \in \mathbb{R}^m$ is the control input and $y \in \mathbb{R}^q$ is the measured output. A, B and C are constant matrices with appropriate dimensions, (A, B) is controllable and (A, C) is observable. The plant is interconnected by a controller over a communication network, see Fig. 3.1. The sensor is periodically sampled with the sampling interval h_1, whereas the controller is an event-based digital controller. The actuator is fed by the control input held constant by a Zero-Order-Hold (ZOH).

The sensor-to-controller (SC) and controller-to-actuator (CA) transmission delays are modeled by Markovian delays $\tau_{\text{sc}}(r_t)$ and $\tau_{\text{ca}}(r_t)$. The switching of Markovian delays is governed by a Markov process $r_t \in \mathcal{S}$ taking values from the finite set $\mathcal{S} := \{1, \ldots, N\}$. The switching rate from mode i to mode j is defined by $\alpha_{i,j}$, where $i, j \in \mathcal{S}$.

According to (3.1) and Fig. 3.1, the piecewise constant measurement from SC at the sampled time t_k is given by

$$\begin{aligned} \bar{y}(t) &= y(t_k - \tau_{\text{sc}}(r_t)) = y(t - \tau_{h_1}(t) - \tau_{\text{sc}}(r_t)) \\ &= Cx(t - \tau_{h_1}(t) - \tau(r_t^{\text{sc}})), \\ \tau_{h_1}(t) &= t - t_k, \quad t_k \leq t < t_{k+1}. \end{aligned} \tag{3.2}$$

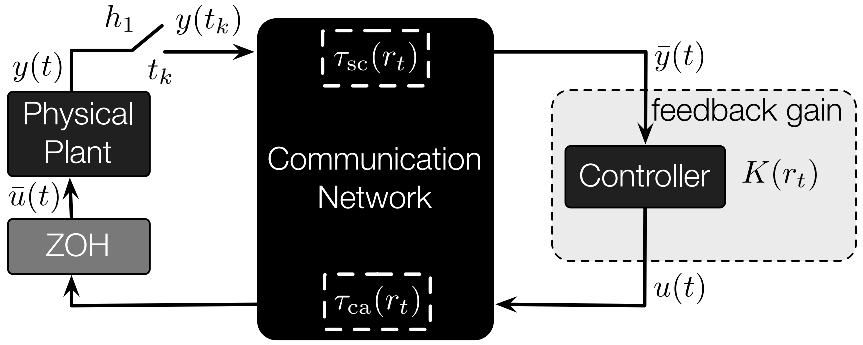
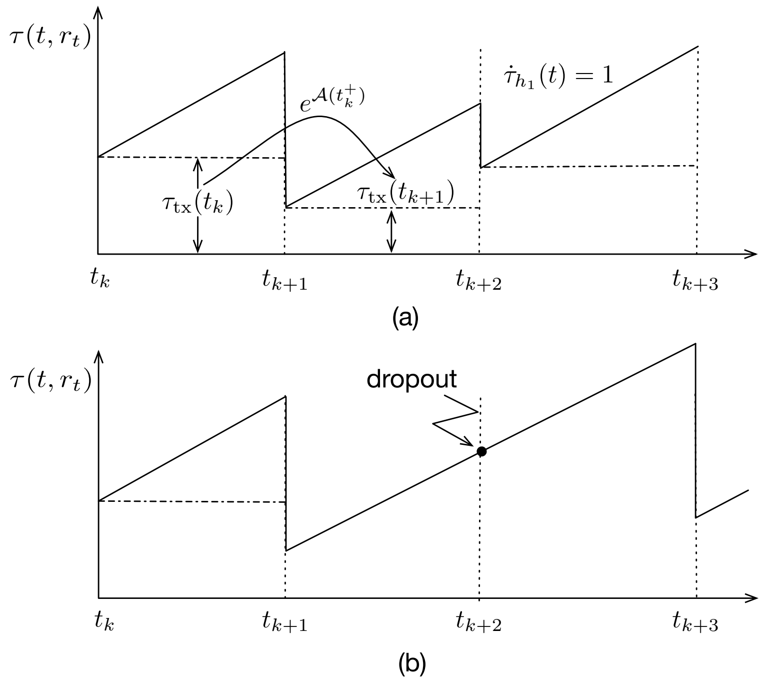


Figure 3.1: Illustration of a sampled-data NCS with sensor-to-controller (SC) delay $\tau_{\text{sc}}(r_t)$ and controller-to-actuator (CA) $\tau_{\text{ca}}(r_t)$. The sensor output, $\bar{y}(t) = y(t_k - \tau_{\text{sc}}(r_t))$, is periodically sampled a sampling interval by h_1 and the control input, $\bar{u}(t) = K(r_t)\bar{y}(t - \tau_{\text{ca}}(r_t))$, is derived by an event-based digital controller and held by a Zero-Order-Hold (ZOH).

Delay-dependent switching controller

Assume a remote controller being able to monitor the SC delay $\tau_{\text{sc}}(r_t)$, e.g. using time-stamping, and synchronously switches the control laws with it. The control commands are fed back to the plant with CA delay $\tau_{\text{ca}}(r_t)$. Consider a state-feedback controller, i.e. $\bar{y}(t) = x(t - \tau_{h_1}(t) - \tau_{\text{sc}}(r_t))$, and the control law yields

$$\bar{u}(t) = K(r_t)x(t - \tau_{h_1}(t) - \tau_{\text{sc}}(r_t) - \tau_{\text{ca}}(r_t)) \tag{3.3}$$



(a)

(b)

Figure 3.2: The evolution of time delay $\tau(t,r_t)$ for certain sample path of $\tau_{\text{tx}}(r_t)$ (a); and the evolution of time delay $\tau(t,r_t)$ with packet dropouts (b).

Substitute (3.3) into system (3.1), the closed-loop system becomes

$$\begin{aligned}\dot{x}(t) &= Ax(t) + BK(r_t)x(t-\tau(t,r_t)),\\ \tau(t,r_t) &= \tau_{h_1}(t) + \tau_{\text{sc}}(r_t) + \tau_{\text{ca}}(r_t).\end{aligned} \tag{3.4}$$

System (3.4) is an MJLS with piecewise random delay $\tau(t,r_t)$.

Piecewise Random delay

The resulting delay $\tau(t,r_t)$ in system (3.4) contains a periodically time-varying component $\tau_{h_1}(t) = t - t_k$ generated by the inter-sampling effect, and a piecewise random component

$$\tau_{\text{tx}}(r_t) = \tau_{\text{sc}}(r_t) + \tau_{\text{ca}}(r_t)$$

generated by the transmission delay as shown in Fig. 3.2.

The periodically time-varying component is bounded by a sampling interval, i.e. $\tau_{h_1}(t) \le h_1$, and has the derivative $\dot{\tau}_{h_1}(t) = 1$. As shown in Fig. 3.2 (b), the packet dropout can be viewed as a delay $\tau_{h_1}(t)$ which grows by accumulating sampling interval h_1. For an NCS with maximal d_1 consecutive dropouts, the time-varying delay is bounded by $\tau_{h_1}(t) \le (1+d_1)h_1$. For stability analysis, the upper bound of the time-varying component is taken into account. As a result, the delay $\tau(t,r_t)$ recovers to

$$\tau(r_t) = (1+d_1)h_1 + \tau_{\text{sc}}(r_t) + \tau_{\text{ca}}(r_t) \tag{3.5}$$

with associated upper and lower bounds defined as

$$\begin{aligned}\bar{\tau} &= (1+d_1)h_1 + \max_{i\in\mathcal{S}}\left\{\tau_{\text{sc}}(i) + \tau_{\text{ca}}(i)\right\},\\ \underline{\tau} &= (1+d_1)h_1 + \min_{i\in\mathcal{S}}\left\{\tau_{\text{sc}}(i) + \tau_{\text{ca}}(i)\right\}\end{aligned}$$

in the later analysis.

Although the stability result derived in Section 3.2 is developed by the upper bound of the time-varying component, it robustly ensures the stability for the time-varying delay satisfying $\tau_{h_1}(t) \leq (1+d_1)h_1$. As the time-varying component has $\dot{\tau}_{h_1}(t) = 1$ within sampling intervals, considering its the upper bound does not impose conservatism into the stability analysis as discussed in [38, 51].

Remark 3.1 The computation delay required by the controller is short and ignorable. Due to the utilization of a common network, the SC and CA transmission suffers the same network traffic. As a result, the SC and CA delays can be approximately modeled by a Markov process r_t.

Remark 3.2 The switching of transmission delays may result in sampled sequence disorder. In this dissertation, the disordering in the sampled sequence is excluded, i.e. with the following assumption

A1: $\mathbf{P}(|\tau(r_{t_{k+1+d_1}}) - \tau(r_{t_k})| \geq h_1) = 0$,

where $d_1 \geq 0$ is maximal number of consecutive dropouts. The assumption A1 restricts that the switching difference of consecutive delays is less than one sampling interval. This assumption can be made as the current transmission delay in the real communication networks is usually correlated to the previous delay. In single-path networks the assumption is fulfilled.

3.2 Stability and stabilization with delay-dependent state-feedback controller

It is useful to design a controller with a given performance. For this purpose, before the stability and stabilization conditions for delay-dependent state-feedback controller are introduced, a new variable $z(t)$ satisfying

$$z(t) = e^{\gamma t}x(t) \tag{3.6}$$

with $\gamma \geq 0$ is introduced. Substitute (3.5) and (3.6) into (3.4), the closed-loop system in (3.4) becomes

$$\dot{z}(t) = (A+\gamma I)z(t) + e^{\gamma\tau(r_t)}BK(r_t)z(t-\tau(r_t)). \tag{3.7}$$

Note that

$$z(t) - z(t-\tau(r_t)) = \int_{t-\tau(r_t)}^{t} \dot{z}(s)ds.$$

The associated system NCS (3.7) has the equivalent form

$$\dot{z}(t) = \big(A+\gamma I + e^{\gamma\tau(r_t)}BK(r_t)\big)z(t) - e^{\gamma\tau(r_t)}BK(r_t)\int_{t-\tau(r_t)}^{t} \dot{z}(s)ds, \tag{3.8}$$

Let $\xi^T(t) = [z^T(t) \;\; \dot{z}^T(t)]^T$, the closed-loop system (3.8) has an equivalent form

$$E\dot{\xi}(t) = \hat{A}(r_t)\xi(t) - \hat{A}_1(r_t)\int_{t-\tau(r_t)}^{t} \xi(s)ds, \tag{3.9}$$

where

$$E = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \ \hat{A}(r_t) = \begin{bmatrix} 0 & I \\ A + \gamma I + e^{\gamma\tau(r_t)} BK(r_t) & -I \end{bmatrix}, \ \hat{A}_1(r_t) = \begin{bmatrix} 0 & 0 \\ 0 & e^{\gamma\tau(r_t)} BK(r_t) \end{bmatrix}.$$

The system described by (3.8) is reformulated into an equivalent system (3.9) by means of the *descriptor transformation* [37]. The descriptor transformation reduces the conservatism in the delay-dependent stability analysis as the transformation introduces no additional dynamics to the original system [36, 39]. The stability of the system represented by (3.9) implies the stability of the system in (3.8). The transformed system in (3.9) is considered in the following stability analysis.

Remark 3.3 The state $\{\xi(t), r_t, t \geq 0\}$ depends on the history $\xi(t+\theta)$, $\theta \in [-2\tau(r_t), 0]$, which implies $\{\xi(t), r_t, t \geq 0\}$ is not a Markov process. According to Definition 2.1, a strong Markov process $\{\Xi(t), r_t, t \geq 0\}$ can be formulated by the following transformation

$$\Xi(t) = \xi(s+t), \ s \in \big[t - 2\tau(r_t), t\big].$$

3.2.1 Stability analysis

In this subsection, delay-dependent stability for state-feedback NCSs with piecewise random delay is derived by using the Lyapunov-Krasovskii functional approach. The random delay $\tau(r_t)$ contains the transmission delay and accumulating sampling intervals caused by packet dropouts, see (3.5). Accordingly, the transmission delay, consecutive dropouts as well as the sampling interval are conjointly treated in the derived stability condition. The solution of stability condition indicates the trade-off between transmission delays $\tau_{\text{sc}}(r_t) + \tau_{\text{ca}}(r_t)$, maximal consecutive packet dropouts d_1 and the sampling interval h_1 for which the stochastic exponential mean square stability can be guaranteed. The details of stability condition are given in Theorem 3.1.

Theorem 3.1 For the closed-loop system (3.9) with Markovian delay $\tau(r_t)$, $r_t \in \mathcal{S}$ and a given $\gamma \geq 0$, if there exist matrices $Q > 0$, $W > 0$ and $X(i) > 0$, $i \in \mathcal{S}$ such that the following LMI's hold

$$\begin{bmatrix} Q & \hat{A}_1^T(i) \\ * & W \end{bmatrix} \geq 0, \tag{3.10}$$

$$\begin{bmatrix} \Psi_1(i) & \Psi_2(i) & \Psi_3(i) \\ * & -(\tau(i) + \hat{\tau}\alpha_i)Q^{-1} & 0 \\ * & * & -\Gamma(i) \end{bmatrix} < 0, \tag{3.11}$$

where

$$\hat{\tau} = \frac{1}{2}(\bar{\tau}^2 - \underline{\tau}^2), \qquad \alpha_i = -\alpha_{i,i},$$

$$\Psi_1(i) = \hat{A}(i)X(i) + X^T(i)\hat{A}^T(i) + \tau(i)W + \alpha_{i,i}EX^T(i), \quad \Psi_2(i) = (\tau(i) + \hat{\tau})X^T(i),$$
$$\Psi_3(i) = \big[\sqrt{\alpha_{i,1}}EX^T(i) \cdots \sqrt{\alpha_{i,N}}EX^T(i)\big], \quad \Gamma(i) = \operatorname{diag}\big\{X(1), \ldots, X(N)\big\},$$

then the system is MES.

Proof: Define a set of positive definite matrices $P(r_t) = X^{-1}(r_t)$ and consider a Lyapunov candidate as follows

$$V(\Xi(t), r_t) = V_1(\Xi(t), r_t) + V_2(\Xi(t), r_t) + V_3(\Xi(t), r_t), \tag{3.12}$$

where

$$V_1(\Xi(t),r_t)=\xi^T(t)EP(r_t)\xi(t),\quad V_2(\Xi(t),r_t)=\int_{-\tau(r_t)}^{0}\int_{t+\theta}^{t}\xi^T(s)Q\xi(s)dsd\theta,$$
$$V_3(\Xi(t),r_t)=\alpha_i\int_{-\bar{\tau}}^{-\underline{\tau}}\int_{t+\theta}^{t}\xi^T(s)Q\xi(s)(s-t-\theta)dsd\theta.$$

Suppose $r_t=i\in\mathcal{S}$, then

$$\begin{aligned}\mathcal{L}V_1(\Xi(t),r_t)&=\dot{\xi}^T(t)EP(r_t)\xi(t)+\xi^T(t)P^T(r_t)E\dot{\xi}(t)\\&=\xi^T(t)\Big[\hat{A}^T(r_t)P(r_t)+P^T(r_t)\hat{A}(r_t)+\sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\&\quad-2\xi^T(t)P^T(r_t)\hat{A}_1(r_t)\int_{t-\tau(r_t)}^{t}\xi(s)ds.\end{aligned}$$

According to Lemma A.1, $\mathcal{L}V_1(\Xi(t),r_t)$ becomes

$$\begin{aligned}\mathcal{L}V_1(\Xi(t),r_t)\le{}&\xi^T(t)\Big[\hat{A}^T(r_t)P(r_t)+P^T(r_t)\hat{A}(r_t)+\sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\&+\tau(r_t)\xi^T(t)P^T(r_t)WP(r_t)\xi(t)\\&+\int_{t-\tau(r_t)}^{t}\xi^T(s)\hat{A}_1(r_t)W^{-1}\hat{A}_1^T(r_t)\xi(s)ds.\end{aligned}$$

Set

$$Q\ge\hat{A}_1^T(r_t)W^{-1}\hat{A}_1(r_t),\tag{3.13}$$

then $\mathcal{L}V_1(\Xi(t),r_t)$ yields

$$\begin{aligned}\mathcal{L}V_1(\Xi(t),r_t)\le{}&\xi^T(t)\Big[\hat{A}^T(r_t)P(r_t)+P^T(r_t)\hat{A}(r_t)+\sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\&+\tau(r_t)\xi^T(t)P^T(r_t)WP(r_t)\xi(t)\\&+\int_{t-\tau(r_t)}^{t}\xi^T(s)Q\xi(s)ds.\end{aligned}\tag{3.14}$$

According to Lemma A.4,

$$\begin{aligned}\mathcal{L}V_2(\Xi(t),r_t)\le{}&\tau(r_t)\xi^T(t)Q\xi(t)-\int_{t-\tau(r_t)}^{t}\xi^T(s)Q\xi(s)ds\\&+\alpha_i\int_{-\bar{\tau}}^{-\underline{\tau}}\int_{t+\theta}^{t}\xi^T(s)Q\xi(s)dsd\theta.\end{aligned}\tag{3.15}$$

$$\mathcal{L}V_3(\Xi(t),r_t)=\frac{1}{2}\alpha_i(\bar{\tau}^2-\underline{\tau}^2)\xi^T(t)Q\xi(t)-\alpha_i\int_{-\bar{\tau}}^{-\underline{\tau}}\int_{t+\theta}^{t}\xi^T(s)Q\xi(s)dsd\theta,\tag{3.16}$$

Combining (3.14)-(3.16) results in

$$\begin{aligned}\mathcal{L}V(\Xi(t),r_t) \leq{}& \xi^T(t)\Big[\hat{A}^T(r_t)P(r_t)+P^T(r_t)\hat{A}(r_t)+\sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\ &+\tau(r_t)\xi^T(t)P^T(r_t)WP(r_t)\xi(t)\\ &+\Big(\tau(r_t)+\frac{1}{2}\alpha_i(\bar{\tau}^2-\underline{\tau}^2)\Big)\xi^T(t)Q\xi(t)\\ ={}& \xi^T(t)\Theta(r_t)\xi(t).\end{aligned} \tag{3.17}$$

Pre- and post-multiple $\Theta(r_t)$ by $X^T(r_t)$ and $X(r_t)$, it gives

$$\begin{aligned}&\hat{A}(r_t)X(r_t)+X^T(r_t)\hat{A}^T(r_t)+\tau(r_t)W+\alpha_{i,i}X^T(i)E\\ &+\sum_{j\neq i}^{N}\alpha_{i,j}X^T(r_t)EX^{-1}(j)X(r_t)\\ &+\Big(\tau(r_t)+\frac{1}{2}\alpha_i(\bar{\tau}^2-\underline{\tau}^2)\Big)X^T(r_t)QX(r_t)<0.\end{aligned} \tag{3.18}$$

Applying Schur complement to (3.13) and (3.18) results in (3.10) and (3.11).

Since $\max_{\theta\in[-2\bar{\tau},0]}\{||\xi(t+\theta)||\} \leq \varphi||\xi(t)||$ for some $\varphi>0$ [83], the following can be established

$$\begin{aligned}V(\Xi(t),r_t) \leq{}& \Big[\lambda_{\max}(EP(r_t))+\varsigma\lambda_{\max}(Q)\Big]||\xi(t)||^2\\ \leq{}& \Lambda_{\max}(r_t)||\xi(t)||^2,\end{aligned} \tag{3.19}$$

where

$$\varsigma=\frac{1}{2}\bar{\tau}^2\varphi+\frac{1}{6}(\bar{\tau}^3-\underline{\tau}^3)\bar{\alpha}\varphi, \quad \bar{\alpha}=\max_{i\in\mathcal{S}}\{\alpha_i\}.$$

$$\Lambda_{\max}(r_t)=\lambda_{\max}(EP(r_t))+\varsigma\lambda_{\max}(Q).$$

Combining (3.17) and (3.19) yields

$$\frac{\mathcal{L}V(\Xi(t),r_t)}{V(\Xi(t),r_t)} \leq -\min_{r_t\in\mathcal{S}}\left\{\frac{\lambda_{\min}(-\Theta(r_t))}{\Lambda_{\max}(r_t)}\right\} \triangleq -\rho_0$$

and

$$\mathbb{E}\{\mathcal{L}V(\Xi(t),r_t)\} \leq -\rho_0\mathbb{E}\{V(\Xi(t),r_t)\}. \tag{3.20}$$

By applying Dynkin's formula into (4.15) it becomes

$$\begin{aligned}\mathbb{E}\{V(\Xi(t),r_t)\}-\mathbb{E}\{V(\Xi(0),r_0)\} ={}& \mathbb{E}\Big[\int_0^t \mathcal{L}V(\Xi(s),r_s)ds\big|\Xi(0),r_0\Big]\\ \leq{}& -\rho_0\int_0^t \mathbb{E}\{\mathcal{L}V(\Xi(s),r_s)\}ds.\end{aligned} \tag{3.21}$$

Using the Gronwall-Bellman lemma, (4.16) results in

$$\mathbb{E}\{V(\Xi(t),r_t)\} \leq e^{-\rho_0 t}\mathbb{E}\{V(\Xi(0),r_0)\}.$$

Since

$$V(\Xi(t), r_t) \geq \Big[\lambda_{\min}(EP(r_t)) + \varsigma\lambda_{\min}(Q)\Big]||\xi(t)||^2 = \Lambda_{\min}(r_t)||\xi(t)||^2,$$

it is established that

$$\mathbb{E}\{||\xi(t)||^2\} \leq e^{-\rho_0 t}\frac{\mathbb{E}\{V(\Xi(0), r_0)\}}{\min_{r_t \in \mathcal{S}}\big\{\Lambda_{\min}(r_t)\big\}}. \tag{3.22}$$

Equation (4.17) provides the proof for stochastic exponential mean square stability. ■

Remark 3.4 It is noted that $\mathbb{E}\{||\xi(t)||^2\} \geq \mathbb{E}\{||z(t)||^2\}$, and $z(t) = e^{\gamma t}x(t)$. Therefore, the inequality (4.17) can be rewritten as

$$\mathbb{E}\{||x(t)||^2\} \leq e^{-(\rho_0+2\gamma)t}\frac{\mathbb{E}\{V(\Xi(0), r_0)\}}{\min_{r_t \in \mathcal{S}}\big\{\Lambda_{\min}(r_t)\big\}}. \tag{3.23}$$

As shown in (3.23) , the given γ in Theorem 3.1 ensures the decay rate of trajectory $\mathbb{E}\{||x(t)||^2\}$ and determines the control performance of the closed-loop system (3.4).

The Lyapunov candidates $V_2(\Xi(t), r_t)$ and $V_3(\Xi(t), r_t)$ are chosen to compensate the integral terms caused by the derivative of $V_1(\Xi(t), r_t)$. The stability condition in Theorem 3.1 contains the transmission delays as well as their statistical properties, consecutive dropouts and sampling intervals in a single LMI condition. This allows the exploration of trade-offs between different parameters for the controller design discussed in the following section. The stability analysis method proposed in Theorem 3.1 can be applied to nonlinear NCSs by linearizing the system at equilibrium states or by the input-output linearization [126].

To illustrate the influence of random delays and transition generator on system stability, the following example is given.

Example 3.1 Consider an NCS governed by the two-state Markovian jump delay differential equation as

$$\begin{aligned} \dot{x}(t) &= -x(t) - 0.164x(t - \tau(1)) \\ \dot{x}(t) &= -x(t) - 0.082x(t - \tau(2)). \end{aligned} \tag{3.24}$$

According to Theorem 3.1 and setting $\gamma = 0$, the maximal Markovian delay[2] of system (3.24) has the values

$$\tau(1) = 20\,\text{ms}, \quad \tau(2) = 40\,\text{ms} \tag{3.25}$$

and transition generator

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{3.26}$$

As mentioned before, the Markovian delay combines the sampling intervals, transmission delay and maximal consecutive dropouts of NCSs. With known sampling interval and maximal number of consecutive dropouts of system (3.24), the maximal allowable transmission delay can be easily derived.

[2] The maximal Markovian delay is determined by the feasible solutions of Theorem 3.1.

The transition generator of a Markovian delay determines the equilibrium behavior of its states. The transition generator in (3.26) indicates the Markovian delay $\tau(1) = 20\,\text{ms}$ in (3.25) has stationary probability 50%. However, the stability condition can be still guaranteed by the stationary probability distribution of the short delay, i.e. $\tau(1) = 20\,\text{ms}$, at 39% or

$$\mathcal{A} = \begin{bmatrix} -1.3 & 1.3 \\ 0.9 & -0.9 \end{bmatrix}.$$

The relationship of the stability condition and Markovian transition generators will be analytically explored in Section 3.4.

The parameter γ determines how fast the trajectory of an NCS converge to the origin. Given the Markovian delay and transition generator as above, the variation of γ and the corresponding feedback gains of system (3.24) are given in Table 3.1. The trend shows the bigger γ is, the more aggressive the feedback gains are.

Table 3.1: The values of γ and corresponding state-feedback gains.

γ	0	0.2	0.4	0.6	0.8
$K(1)$	-0.164	-0.164	-0.171	-0.188	-0.191
$K(2)$	-0.082	-0.084	-0.092	-0.112	-0.120

3.2.2 State-feedback stabilization

The difficulty in solving switching feedback gain $K(i)$ in the matrix inequality (3.11) involves nonlinear terms, i.e. $\hat{A}(i)X(i)$ in $\Psi_1(i)$ and cannot be considered as an LMI problem. However, by introducing special settings of $X(i)$ the nonlinear terms can be eliminated and the LMI problem is recovered.

Theorem 3.2 For given scalars $n_1(i) \geq 0$, $n_2(i) \geq 0$, $\varepsilon(i) \geq 1$ and $\gamma \geq 0$, if there exist matrices $Y(i)$, $W > 0$ and $X_1(i) = X_1^T(i) > 0$, $i \in \mathcal{S}$ satisfying

$$X(i) = \begin{bmatrix} X_1(i) & 0 \\ -n_1(i)X_1(i) & n_2(i)X_1(i) \end{bmatrix} \tag{3.27}$$

such that

$$\begin{bmatrix} \hat{\Psi}_1(i) & \hat{\Psi}_2(i) & \hat{\Psi}_3(i) \\ * & -\varepsilon(i)(\tau(i) + \hat{\tau}\alpha_i)W & 0 \\ * & * & -\Gamma(i) \end{bmatrix} < 0, \tag{3.28}$$

where

$$\hat{\tau} = \frac{1}{2}(\bar{\tau}^2 - \underline{\tau}^2), \qquad \alpha_i = -\alpha_{i,i},$$

$$
\begin{aligned}
\hat{\Psi}_1(i) &= \begin{bmatrix} -n_1(i)X_1(i) & n_2(i)X_1(i) \\ AX_1(i) + \gamma X_1(i) + e^{\gamma\tau(i)}BY(i) + n_1(i)X_1(i) & -n_2(i)X_1(i) \end{bmatrix} \\
&\quad + \begin{bmatrix} -n_1(i)X_1(i) & n_2(i)X_1(i) \\ AX_1(i) + \gamma X_1(i) + e^{\gamma\tau(i)}BY(i) + n_1(i)X_1(i) & -n_2(i)X_1(i) \end{bmatrix}^T \\
&\quad + \tau(i)W + \alpha_{i,i}EX^T(i), \\
\hat{\Psi}_2(i) &= \varepsilon(i)\big(\tau(i) + \hat{\tau}\alpha_i\big) \begin{bmatrix} 0 & -n_1(i)e^{\gamma\tau(i)}Y^T(i)B^T \\ 0 & n_2(i)e^{\gamma\tau(i)}Y^T(i)B^T \end{bmatrix}, \\
\hat{\Psi}_3(i) &= \big[\sqrt{\alpha_{i,1}}EX^T(i) \cdots \sqrt{\alpha_{i,N}}EX^T(i)\big], \\
\Gamma(i) &= \operatorname{diag}\big\{X(1), \ldots, X(N)\big\}
\end{aligned}
$$

holds, the closed-loop system (3.9) is MES with the feedback gain

$$K(i) = Y(i)X_1^{-1}(i). \tag{3.29}$$

Proof: According to Theorem 3.1, the switching controller (3.3) stabilizes the closed-loop system (3.4) if the inequalities (3.13) and (3.18) are satisfied. Choose a set of $\varepsilon(i) \geq 1$ and let $Q = \varepsilon(i)\hat{A}_1^T(i)W^{-1}\hat{A}_1(i)$, (3.13) becomes

$$\varepsilon(i)\hat{A}_1^T(i)W^{-1}\hat{A}_1(i) \geq \hat{A}_1^T(i)W^{-1}\hat{A}_1(i). \tag{3.30}$$

Substitute (3.30) and (3.27) into (3.18) and let $Y(i) = K(i)X_1(i)$. The nonlinear terms in (3.11) are eliminated and the LMI (3.28) is derived. ■

Remark 3.5 The structure of $X(i)$ in Theorem 3.2 is made due to the requirement $P(i) = X^{-1}(i)$, in the Lyapunov candidate satisfying $EP(i) = P^T(i)E > 0$. Therefore, $X(i)$ is determined as follows

$$X(i) = \begin{bmatrix} X_1(i) & 0 \\ X_2(i) & X_3(i) \end{bmatrix}, \quad X_1(i) = X_1(i)^T > 0. \tag{3.31}$$

However, by expanding $\hat{\Psi}_2(i)$ and $\hat{\Psi}_3(i)$ it results in terms, e.g. $e^{\gamma\tau(i)}BK(i)X_2(i)$ and $e^{\gamma\tau(i)}BK(i)X_3(i)$, which make deriving an LMI formulation not possible. In order to obtain an LMI formulation, one possibility is to set $X_2(i)$ and $X_3(i)$ in (3.31) as $-n_1(i)X_1(i)$ and $n_2(i)X_1(i)$, where $n_1(i)$ and $n_2(i)$ are positive real numbers.

Although the LMI algorithm can be efficiently solved by the LMI toolbox, the restriction on matrix $X(i)$ introduces conservatism in the controller design. The design algorithms (3.27)-(3.29) in Theorem 3.2 might not provide a feasible solution, even if there exists one.

A less conservative approach is to set $X(i)$ back to (3.31) and solve the BMI (bilinear matrix inequality) directly. However, solving an BMI has the drawback that the feasible feedback gains can only be found strongly depending on the initial conditions. A brute-force numerical search regarding any possible initial conditions is unavoidable, e.g. using the V-K iteration [135] or cone complementary linearization [142]. The solution of the LMI algorithm in Theorem 3.2 can be used as an initial condition for solving the BMI. In this case, less conservative feedback gains can be derived. To illustrate the results of Theorem 3.2, the following numerical example is considered.

Example 3.2 Consider an NCS with dynamics described by (3.1) and assume the system has Markovian delays $\tau(r_t) = [20\ 40]\,\text{ms}$. The switching of Markovian delays is governed by the generator $\mathcal{A}$ given by

$$\mathcal{A} = \begin{bmatrix} -3 & 3 \\ 1 & -1 \end{bmatrix}.$$

The system parameters are

$$A = \begin{bmatrix} 0 & 1 \\ 1 & -50 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix}.$$

Set $\gamma = 1.2$, $n_1(1) = 8.100 \times 10^4$, $n_2(1) = 1.327 \times 10^5$, $n_1(2) = 7.290 \times 10^5$, $n_2(2) = 9.677 \times 10^5$, $\varepsilon(1) = 3.051$ and $\varepsilon(2) = 1.332$. Solving Theorem 3.2, the feasible stabilizing state-feedback gains are derived as

$$K(1) = [-4.567\ -1.983], \quad K(2) = [-2.000\ -1.357].$$

With the initial condition $x^T(\theta) = [1\ 2]$, $\theta \in [-\bar{\tau}, 0]$, the simulation is performed 500 times with different sample paths of transmission delays for a time horizon of $T = 3\,\text{s}$. One sample path of the Markovian delay and associated probability distributions are shown in Fig. 3.3 (a). Note that the stationary probability distribution of $\tau(2) = 40\,\text{ms}$ is 75% and $\tau(1) = 20\,\text{ms}$ is 25%. For comparison, two controller design approaches are investigated. In the proposed delay-dependent switching controller design, the delay is monitored using the time-stamping technique and the feedback gain is synchronously switched with the Markovian delays. The second approach holds the random delay constant by using the buffering technique at the controller side, i.e. the controller is designed with the worst-case delay $\tau(2) = 40\,\text{ms}$. The evolution of mean trajectory

$$||\bar{x}(t)|| = \sqrt{\bar{x}_1^2(t) + \bar{x}_2^2(t)}$$

is shown in Fig. 3.3. For the NCS with delay-depend switching controller, the mean trajectory converges exponentially towards a ball around the origin of radius $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 1.575\,\text{s}$. The performance is 78.7% improved than the NCS with worst-case design controller $t_{0.05} = 2.814\,\text{s}$.

Consider the positive definite matrix X as in (3.31) and take the feasible solution of Theorem 3.2 as an initial condition for BMI. The less conservative stabilizing state-feedback gains are

$$K(1) = [-30.368\ -27.041], \quad K(2) = [-20.650\ -5.032],$$

with

$$X_1(1) = \begin{bmatrix} 0.318 & -0.143 \\ -0.143 & 0.125 \end{bmatrix}, \quad X_1(2) = \begin{bmatrix} 0.424 & -0.170 \\ -0.170 & 0.183 \end{bmatrix},$$

$$W = 10^2 \times \begin{bmatrix} 0.353 & -0.068 & 0.018 & 0.006 \\ -0.068 & 0.387 & -0.059 & 0.001 \\ -0.018 & -0.059 & 0.522 & -0.021 \\ 0.006 & 0.001 & -0.021 & 0.575 \end{bmatrix}.$$

The same comparison is also executed for the NCS with the less conservative delay-dependent switching controller. The NCS converges towards $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 0.068\,\text{s}$, up to 23 times faster than the delay-dependent controller derived by the LMI algorithm. Furthermore, the delay-dependent switching controller derived by BMI algorithm is 79.4% faster than the worst-case deign controller derived by BMI algorithm.

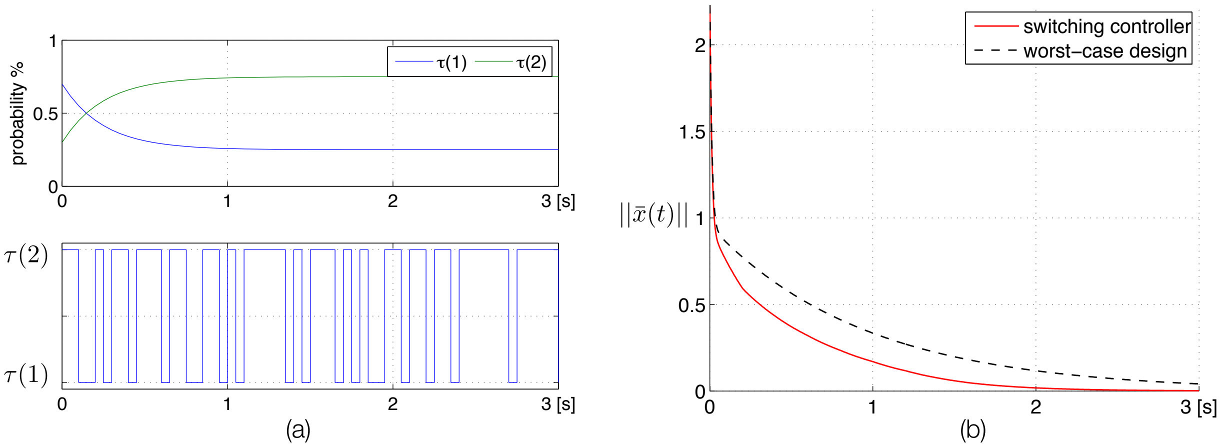


Figure 3.3: One sample path of the Markovian delay and associated probability distributions (a) and the mean state trajectory of NCS with delay-dependent switching controller (solid line) and NCS with worst-case design controller (dashed line) (b).

By and large, the delay-dependent switching controller design algorithm in Theorem 3.2 has superior control performance than the conventional worst-case design. It is very promising for NCSs with non-constant delay.

3.3 Stability and stabilization with delay-dependent output-feedback controller

In this section, the stability of NCSs with delay-dependent dynamical output-feedback controller is studied. As shown in Fig. 3.4, the actuator is event driven, as while the sensor and the the dynamical output-feedback controller are periodically sampled by h_1 and h_2, respectively. The switching output-feedback controller has the form

$$\begin{aligned} \dot{x}_c(t) &= A_c(r_t)x_c(t) + B_c(r_t)\bar{y}(t), \\ \bar{u}(t) &= C_c(r_t)x_c(t), \end{aligned} \tag{3.32}$$

where $x_c \in \mathbb{R}^n$ is the controller state with $x_c = 0$ for $t \le 0$, $A_c(r_t)$, $B_c(r_t)$ and $C_c(r_t)$ are designed parameters with appropriate dimensions. According to (3.2), the piecewise constant sensor measurement at sampled time t_k is

$$\begin{aligned} \bar{y}(t) &= Cx(t - \tau_{h_1}(t) - \tau_{\mathrm{sc}}(r_t)), \\ \tau_{h_1}(t) &= t - t_k, \quad t_k \le t < t_{k+d_1+1}, \end{aligned}$$

where d_1 is the maximal number of consecutive dropouts in SC channel. Similarly, the piecewise constant control output at sampled time t_l becomes

$$\begin{aligned} \bar{u}(t) &= C_c(r_t)x_c(t - \tau_{h_2}(t) - \tau_{\mathrm{ca}}(r_t)) \\ \tau_{h_2}(t) &= t - t_l, \quad t_l \le t < t_{l+d_2+1}, \end{aligned} \tag{3.33}$$

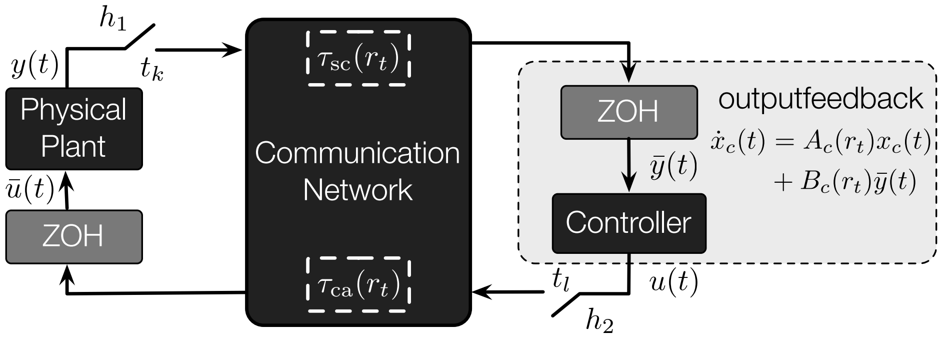


Figure 3.4: Illustration of a sampled-data NCS with SC delay $\tau_{\mathsf{sc}}(r_t)$ and CA delay $\tau_{\mathsf{ca}}(r_t)$. The sensor output, $\bar{y}(t) = Cx(t - \tau_{h_1}(t) - \tau_{\mathsf{sc}}(r_t))$, is periodically sampled by h_1 and held by a ZOH. The output-feedback controller output, $\bar{u}(t) = C_c(r_t)x_c(t - \tau_{h_2}(t) - \tau_{\mathsf{ca}}(r_t))$, is periodically sampled by h_2 and held by a ZOH.

where d_2 is the maximal number of consecutive dropouts in CA channel. For any $\gamma \geq 0$, consider $z(t) = e^{\gamma t}x(t)$ and $z_c(t) = e^{\gamma t}x_c(t)$. Define $\chi^T(t) = [z^T(t)\ z_c^T(t)]$. Combining (3.1), (3.32) and (3.33), the closed-loop system in Fig. 3.4 becomes

$$\dot{\chi}(t) = \bar{A}_0(r_t)\chi(t) + \bar{A}_1(r_t)\chi(t - \tau_1(t, r_t)) + \bar{A}_2(r_t)\chi(t - \tau_2(t, r_t)), \tag{3.34}$$

where

$$\bar{A}_0(r_t) = \begin{bmatrix} A + \gamma I & 0 \\ 0 & A_c(r_t) + \gamma I \end{bmatrix},\ \bar{A}_1(r_t) = \begin{bmatrix} 0 & 0 \\ e^{\gamma\tau_1(t,r_t)}B_c(r_t)C & 0 \end{bmatrix},$$
$$\bar{A}_2(r_t) = \begin{bmatrix} 0 & e^{\gamma\tau_2(t,r_t)}BC_c(r_t) \\ 0 & 0 \end{bmatrix}.$$

The resulting closed-loop system (3.34) is an MJLS with multiple piecewise random delays, $\tau_1(t, r_t) = t - t_k + \tau_{\text{sc}}(r_t)$ and $\tau_2(t, r_t) = t - t_l + \tau_{\text{ca}}(r_t)$. Consider the upper bound of time-varying delays in SC and CA channels in the stability analysis. Hence, the delays in the closed-loop system (3.34) become

$$\tau_1(r_t) = (1 + d_1)h_1 + \tau_{\text{sc}}(r_t), \quad \tau_2(r_t) = (1 + d_2)h_2 + \tau_{\text{ca}}(r_t) \tag{3.35}$$

Apply the descriptor transformation to the system (3.34) and let $\xi^T(t) = [\chi^T(t)\ \dot{\chi}^T(t)]$, it becomes

$$E\dot{\xi}(t) = \hat{A}(r_t)\xi(t) - \hat{A}_1(r_t)\int_{t-\tau_1(r_t)}^{t} \xi(s)ds - \hat{A}_2(r_t)\int_{t-\tau_2(r_t)}^{t} \xi(s)ds, \tag{3.36}$$

where

$$E = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \qquad \hat{A}(r_t) = \begin{bmatrix} 0 & I \\ \bar{A}_0(r_t) + \bar{A}_1(r_t) + \bar{A}_2(r_t) & -I \end{bmatrix},$$
$$\hat{A}_1(r_t) = \begin{bmatrix} 0 & 0 \\ 0 & \bar{A}_1(r_t) \end{bmatrix},\ \hat{A}_2(r_t) = \begin{bmatrix} 0 & 0 \\ 0 & \bar{A}_2(r_t) \end{bmatrix}.$$

As mentioned before, the system described by (3.34) is equivalent to the transformed system in (3.36). Therefore, the system represented by (3.36) is considered for the stability analysis in the subsequent section.

Remark 3.6 The state $\{\xi(t), r_t, t \geq 0\}$ depends on the history $\xi(t+\theta),\ \theta \in [-2\tilde{\tau}(r_t), 0]$, where $\tilde{\tau}(r_t) = \max\{\tau_1(r_t), \tau_2(r_t)\}$. It implies $\{\xi(t), r_t, t \geq 0\}$ is not a Markov process. According to Definition 2.1, a strong Markov process $\{\Xi(t), r_t, t \geq 0\}$ can be formulated by the following transformation

$$\Xi(t) = \xi(s+t),\ s \in \big[t - 2\tilde{\tau}(r_t), t\big],$$

where $\tilde{\tau}(r_t) = \max\{\tau_1(r_t), \tau_2(r_t)\}$.

3.3.1 Stability analysis

Similar to Section 3.2.1, a delay-dependent stability for output-feedback NCSs is derived by using the Lyapunov-Krasovskii functional approach. The resulting NCSs in (3.34) have multiple random delays $\tau_1(r_t)$ and $\tau_2(r_t)$. The derived stability condition deals with both random delays. Any feasible solution of the stability condition indicates the trade-off between transmission delay: $\tau_{\mathrm{sc}}(r_t)$ and $\tau_{\mathrm{ca}}(r_t)$, consecutive dropouts: d_1 and d_2, and the sampling intervals: h_1 and h_2, such that the stochastic exponential mean square stability can be guaranteed. The details are given in the following Theorem 3.3.

Theorem 3.3 For the closed-loop system (3.36) with a given $\gamma \geq 0$, if there exist matrices $Q_1 > 0$, $Q_2 > 0$, $P(i) > 0$, $i \in \mathcal{S}$ such that the following LMI's hold

$$\begin{bmatrix} \Psi_1(i) & \tau_1(i)P^T(i)\hat{A}_1(i) & \tau_2(i)P^T(i)\hat{A}_2(i) \\ * & -\tau_1(i)Q_1 & 0 \\ * & * & -\tau_2(i)Q_2 \end{bmatrix} < 0, \tag{3.37}$$

where $\alpha_i = -\alpha_{i,i}$,

$$\bar{\tau}_1 = \max_{i\in\mathcal{S}}\big\{\tau_1(i)\big\},\ \underline{\tau}_1 = \min_{i\in\mathcal{S}}\big\{\tau_1(i)\big\},\ \hat{\tau}_1 = \frac{1}{2}(\bar{\tau}_1^2 - \underline{\tau}_1^2),$$

$$\bar{\tau}_2 = \max_{i\in\mathcal{S}}\big\{\tau_2(i)\big\},\ \underline{\tau}_2 = \min_{i\in\mathcal{S}}\big\{\tau_2(i)\big\},\ \hat{\tau}_2 = \frac{1}{2}(\bar{\tau}_2^2 - \underline{\tau}_2^2),$$

$$\begin{aligned} \Psi_1(i) = {} & \hat{A}^T(r_t)P(i) + P^T(i)\hat{A}(r_t) + \sum_{j=1}^{N}\alpha_{i,j}EP(j) \\ & + \big(\tau_1(i) + \hat{\tau}_1\alpha_i\big)Q_1 + \big(\tau_2(i) + \hat{\tau}_2\alpha_i\big)Q_2, \end{aligned}$$

then the system is MES.

Proof: Consider a Lyapunov candidate as follows

$$V(\Xi(t), r_t) = V_1(\Xi(t), r_t) + V_2(\Xi(t), r_t) + V_3(\Xi(t), r_t), \tag{3.38}$$

where

$$\begin{aligned} V_1(\Xi(t), r_t) &= \xi^T(t)EP(r_t)\xi(t), \\ V_2(\Xi(t), r_t) &= \int_{-\tau_1(r_t)}^{0}\int_{t+\theta}^{t}\xi^T(s)Q_1\xi(s)dsd\theta + \int_{-\tau_2(r_t)}^{0}\int_{t+\theta}^{t}\xi^T(s)Q_2\xi(s)dsd\theta, \\ V_3(\Xi(t), r_t) &= \alpha_i\int_{-\bar{\tau}_1}^{-\underline{\tau}_1}\int_{t+\theta}^{t}\xi^T(s)Q_1\xi(s)(s-t-\theta)dsd\theta \\ &\quad + \alpha_i\int_{-\bar{\tau}_2}^{-\underline{\tau}_2}\int_{t+\theta}^{t}\xi^T(s)Q_2\xi(s)(s-t-\theta)dsd\theta. \end{aligned}$$

Suppose $r_t = i \in \mathcal{S}$. According to Lemma A.1, it has

$$\begin{aligned}\mathcal{L}V_1(\Xi(t), r_t) \le{}& \xi^T(t)\Big[\hat{A}^T(r_t)P(r_t) + P^T(r_t)\hat{A}(r_t) + \sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\ &+ \xi^T(t)\Big[\tau_1(r_t)P^T(r_t)\hat{A}_1(r_t)Q_1^{-1}\hat{A}_1^T(r_t)P(r_t)\\ &+ \tau_2(r_t)P^T(r_t)\hat{A}_2(r_t)Q_2^{-1}\hat{A}_2^T(r_t)P(r_t)\Big]\xi(t)\\ &+ \int_{t-\tau_1(r_t)}^{t}\xi^T(s)Q_1\xi(s)ds + \int_{t-\tau_2}^{t}\xi^T(s)Q_2\xi(s)ds.\end{aligned} \tag{3.39}$$

$$\begin{aligned}\mathcal{L}V_2(\Xi(t), r_t) \le{}& \tau_1(r_t)\xi^T(t)Q_1\xi(t) + \tau_2(r_t)\xi^T(t)Q_2\xi(t)\\ &- \int_{t-\tau_1(r_t)}^{t}\xi^T(s)Q_1\xi(s)ds - \int_{t-\tau_2(r_t)}^{t}\xi^T(s)Q_2\xi(s)ds\\ &+ \alpha_i\int_{-\bar{\tau}_1}^{-\underline{\tau}_1}\int_{t+\theta}^{t}\xi^T(s)Q_1\xi(s)dsd\theta + \alpha_i\int_{-\bar{\tau}_2}^{-\underline{\tau}_2}\int_{t+\theta}^{t}\xi^T(s)Q_2\xi(s)dsd\theta.\end{aligned} \tag{3.40}$$

$$\begin{aligned}\mathcal{L}V_3(\Xi(t), r_t) ={}& \frac{1}{2}\alpha_i(\bar{\tau}_1^2 - \underline{\tau}_1^2)\xi^T(t)Q_1\xi(t) - \alpha_i\int_{-\bar{\tau}_1}^{-\underline{\tau}_1}\int_{t+\theta}^{t}\xi^T(s)Q_1\xi(s)dsd\theta\\ &+ \frac{1}{2}\alpha_i(\bar{\tau}_2^2 - \underline{\tau}_2^2)\xi^T(t)Q_2\xi(t) - \alpha_i\int_{-\bar{\tau}_2}^{-\underline{\tau}_2}\int_{t+\theta}^{t}\xi^T(s)Q_2\xi(s)dsd\theta.\end{aligned} \tag{3.41}$$

Combining (3.39)-(3.41) results in

$$\begin{aligned}\mathcal{L}V(\Xi(t), r_t) \le{}& \xi^T(t)\Big[\hat{A}^T(r_t)P(r_t) + P^T(r_t)\hat{A}(r_t) + \sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\ &+ \xi^T(t)\Big[\tau_1(r_t)P^T(r_t)\hat{A}_1(r_t)Q_1^{-1}\hat{A}_1^T(r_t)P(r_t)\\ &+ \tau_2(r_t)P^T(r_t)\hat{A}_1(r_t)Q_2^{-1}\hat{A}_1^T(r_t)P(r_t)\Big]\xi(t)\\ &+ \Big(\tau_1(r_t) + \frac{1}{2}\alpha_i(\bar{\tau}_1^2 - \underline{\tau}_1^2)\Big)\xi^T(t)Q_1\xi(t)\\ &+ \Big(\tau_2(r_t) + \frac{1}{2}\alpha_i(\bar{\tau}_2^2 - \underline{\tau}_2^2)\Big)\xi^T(t)Q_2\xi(t)\\ ={}& \xi^T(t)\Theta(r_t)\xi(t).\end{aligned} \tag{3.42}$$

Applying the Schur complement to $\Theta(r_t)$, it results in (3.37).

Due to the fact $\max_{\theta\in[-2\tau,0]}\{||\xi(t+\theta)||\} \le \varphi||\xi(t)||$ for some $\varphi > 0$ [83], the Lyapunov candidate in (3.38) satisfies

$$\begin{aligned}V(\Xi(t), r_t) \le{}& \Big[\lambda_{\max}(EP(r_t)) + \varsigma_1\lambda_{\max}(Q_1) + \varsigma_2\lambda_{\max}(Q_2)\Big]||\xi(t)||^2\\ \le{}& \Lambda_{\max}(r_t)||\xi(t)||^2,\end{aligned} \tag{3.43}$$

where $\bar{\alpha} = \max_{i\in\mathcal{S}}\{\alpha_i\}$,

$$\varsigma_1 = \frac{1}{2}\bar{\tau}_1^2\varphi + \frac{1}{6}(\bar{\tau}_1^3 - \underline{\tau}_1^3)\bar{\alpha}\varphi, \quad \varsigma_2 = \frac{1}{2}\bar{\tau}_2^2\varphi + \frac{1}{6}(\bar{\tau}_2^3 - \underline{\tau}_2^3)\bar{\alpha}\varphi,$$

$$\Lambda_{\max}(r_t) = \lambda_{\max}(EP(r_t)) + \varsigma_1 \lambda_{\max}(Q_1) + \varsigma_2 \lambda_{\max}(Q_2).$$

Apply the Dynkin's formula and use the Gronwall-Bellman lemma, it has

$$\mathbb{E}\{||\xi(t)||^2\} \leq e^{-\rho_0 t} \frac{\mathbb{E}\{V(\Xi(0), r_0)\}}{\min_{r_t \in \mathcal{S}} \{\Lambda_{\min}(r_t)\}}, \tag{3.44}$$

where

$$\rho_0 = \min_{r_t \in \mathcal{S}} \left\{ \frac{\Lambda_{\min}(-\Theta(r_t))}{\Lambda_{\max}(r_t)} \right\}.$$

Equation (3.44) provides the proof for stochastic exponential mean square stability. ■

Remark 3.7 In Theorem 3.3, the control performance of system (3.4) with output-feedback controller (3.32) can be pre-specified by $\gamma \geq 0$. The decay rate of trajectory $\mathbb{E}\{||x(t)||^2\}$ is determined in (3.23).

The Lyapunov candidates are chosen to compensate the integrals caused by the delays $\tau_1(r_t)$ and $\tau_2(r_t)$. Theorem 3.3 can be also applied to NCSs with nonlinear plants by using linearization techniques. The relationship between Markovian delays and stability is discussed in the following example.

Example 3.3 Consider an NCS

$$\dot{x}(t) = -x(t) + \bar{u}(t)$$

with a set of dynamical output-feedback controllers

$$\begin{cases} \dot{x}_c(t) & = -10x_c(t) - 3x(t - \tau_1(1)), \\ \bar{u}(t) & = -0.5x_c(t - \tau_2(1)), \end{cases} \tag{3.45}$$

and

$$\begin{cases} \dot{x}_c(t) & = -6x_c(t) - 2x(t - \tau_1(2)), \\ \bar{u}(t) & = -2.5x_c(t - \tau_2(2)). \end{cases} \tag{3.46}$$

According the stability condition in Theorem 3.3 and setting $\gamma = 0$, the maximal feasible Markovian delays of the NCS are

$$\begin{aligned} \tau_1(1) &= 20\,\text{ms}, \quad \tau_1(2) = 40\,\text{ms}, \\ \tau_2(1) &= 10\,\text{ms}, \quad \tau_2(2) = 20\,\text{ms}, \end{aligned}$$

with transition generator

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 2 & -2 \end{bmatrix}.$$

As shown in (3.32)-(3.33), the Markovian delays $\tau_1(r_t)$ and $\tau_2(r_t)$ contain the sampling intervals and networked-induced delays[3] in SC and CA channels. With the knowledge of network-induced delays, the sampling intervals in both channels can be easily determined and vice versa.

[3] The network-induced delay is defined as the composition of the delays caused by transmission and packet dropouts.

The Markov transition generator indicates that the Markovian delays $\tau_1(1)$ and $\tau_2(1)$ have the stationary probability distribution 66.7%. The stability condition is still valid by the stationary probability distribution of $\tau_1(1)$ and $\tau_2(1)$ at 33.3% or

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 0.5 & -0.5 \end{bmatrix},$$

refer to Section 3.4 for details.

The parameter γ ensures the decay rate of system trajectories. The bigger γ is chosen, the more aggressive the output-feedback controllers will be, see Table 3.2.

Table 3.2: The values of γ and corresponding output-feedback controllers.

γ	0	0.2	0.4
$A_c(1),\ B_c(1),\ C_c(1)$	-10, -3, -0.5	-12, -3, -0.4	-16, -3, -0.1
$A_c(2),\ B_c(2),\ C_c(2)$	-6, -2, -2.5	-7, -2, -1.8	-5, -2.1, -1.3

3.3.2 Output-feedback stabilization

Solving the switching output-feedback controller parameters $A_c(r_t)$, $B_c(r_t)$ and $C_c(r_t)$ concerns the nonlinear terms in matrix inequality (3.37), i.e. $P^T(r_t)\hat{A}(r_t)$, $P^T(r_t)\hat{A}_1(r_t)$ and $P^T(r_t)\hat{A}_2(r_t)$. However, the LMI condition can be recovered by the diagonal requirement of $P(r_t)$. The details are shown in the following Theorem 3.4.

Theorem 3.4 For given scalars $n_1(i) \geq 0$, $n_2(i) \geq 0$ and $\gamma \geq 0$, if there exist matrices $F(i)$, $G(i)$, $H(i)$, $Q_1 > 0$, $Q_2 > 0$, $P_1(i) = P_1^T(i) > 0$, $i \in \mathcal{S}$ satisfying

$$P_1(i) = \begin{bmatrix} P_{11}(i) & 0 \\ 0 & P_{12}(i) \end{bmatrix},$$
$$P(i) = \begin{bmatrix} P_1(i) & 0 \\ -n_1(i)P_1(i) & n_2(i)P_1(i) \end{bmatrix}, \tag{3.47}$$

such that

$$\begin{bmatrix} \Psi_1(i) & \Psi_2(i) & \Psi_3(i) \\ * & -\tau_1(i)Q_1 & 0 \\ * & * & -\tau_2(i)Q_2 \end{bmatrix} < 0, \tag{3.48}$$

where $\alpha_i = -\alpha_{i,i}$

$$\bar{\tau}_1 = \max_{i\in\mathcal{S}}\{\tau_1(i)\},\ \underline{\tau}_1 = \min_{i\in\mathcal{S}}\{\tau_1(i)\},\ \hat{\tau}_1 = \frac{1}{2}(\bar{\tau}_1^2 - \underline{\tau}_1^2),$$
$$\bar{\tau}_2 = \max_{i\in\mathcal{S}}\{\tau_2(i)\},\ \underline{\tau}_2 = \min_{i\in\mathcal{S}}\{\tau_2(i)\},\ \hat{\tau}_2 = \frac{1}{2}(\bar{\tau}_2^2 - \underline{\tau}_2^2),$$

$$\Psi_1(i) = \begin{bmatrix} \Pi_1(i) & \Pi_2(i) \\ (1-n_1(i))P_1(i) & n_2(i)P_1(i) \end{bmatrix}^T + \begin{bmatrix} \Pi_1(i) & \Pi_2(i) \\ (1-n_1(i))P_1(i) & n_2(i)P_1(i) \end{bmatrix}$$
$$+ \tau_1(i)Q_1 + \tau_2(i)Q_2 + \sum_{j=1}^{N} \alpha_{i,j} EP(j),$$

$$\Pi_1(i) = n_1(i)\begin{bmatrix} P_{11}(i)A + \gamma P_{11}(i) & e^{\gamma\tau_2(i)}H(i) \\ e^{\gamma\tau_1(i)}G(i) & F(i) + \gamma P_{12}(i) \end{bmatrix},$$
$$\Pi_2(i) = n_2(i)\begin{bmatrix} P_{11}(i)A + \gamma P_{11}(i) & e^{\gamma\tau_2(i)}H(i) \\ e^{\gamma\tau_1(i)}G(i) & F(i) + \gamma P_{12}(i) \end{bmatrix},$$
$$\Psi_2(i) = \tau_1(i)\begin{bmatrix} 0 & -n_1(i)\begin{bmatrix} 0 & 0 \\ e^{\gamma\tau_1(i)}G(i) & 0 \end{bmatrix} \\ 0 & n_2(i)\begin{bmatrix} 0 & 0 \\ e^{\gamma\tau_1(i)}G(i) & 0 \end{bmatrix} \end{bmatrix}, \quad \Psi_3(i) = \tau_2(i)\begin{bmatrix} 0 & -n_1(i)\begin{bmatrix} 0 & e^{\gamma\tau_2(i)}H(i) \\ 0 & 0 \end{bmatrix} \\ 0 & n_2(i)\begin{bmatrix} 0 & e^{\gamma\tau_2(i)}H(i) \\ 0 & 0 \end{bmatrix} \end{bmatrix},$$

holds, the closed-loop system (3.36) is MES under the output-feedback controller of the form

$$A_c(i) = P_{11}^{-1}(i)F(i), \quad B_c(i) = P_{12}^{-1}(i)G(i)(i)C^+, \quad C_c(i) = B^+P_{11}^{-1}(i)H(i). \tag{3.49}$$

Proof: Substitute (3.47) into (3.42) and let

$$F(i) = P_{11}(i)A_c(i), \quad G(i) = P_{12}(i)B_c(i)C, \quad H(i) = P_{11}(i)BC_c(i). \tag{3.50}$$

The nonlinear terms are eliminated and the LMI in (3.48) is derived. ■

Remark 3.8 The structure of $P(i)$ in Theorem 3.4 is made due to the condition $EP(i) = P^T(i)E > 0$ in the Lyapunov candidate. Generally, the positive definite matrix $P(i)$ has the form as

$$P(i) = \begin{bmatrix} P_1(i) & 0 \\ P_2(i) & P_3(i) \end{bmatrix}, \quad P_1(i) = P_1^T(i) > 0. \tag{3.51}$$

However, in order to avoid the nonlinear terms $\bar{A}_1^T(i)P_2(i)$, $\bar{A}_2^T(i)P_2(i)$, $\bar{A}_1^T(i)P_3(i)$ and $\bar{A}_2^T(i)X_3(i)$, the matrices $P_2(i)$ and $P_3(i)$ are replaced by $-n_1(i)P_1(i)$ and $n_2(i)P_1(i)$. In addition, $P_1(i)$ is determined as a diagonal matrix, i.e.

$$P_1(i) = \begin{bmatrix} P_{11}(i) & 0 \\ 0 & P_{12}(i) \end{bmatrix},$$

so as to make the products of $\bar{A}_2^T(i)P_1(i)$ and $\bar{A}_3^T(i)P_1(i)$ resulting in $F(i) = P_{11}(i)A_c(i)$, $G(i) = P_{12}(i)B_c(i)C$ and $H(i) = P_{11}(i)BC_c(i)$.

The LMI algorithm is derived by the structure requirement of matrices $P_1(i)$. These requirements, however, introduce certain conservatism in the output-feedback controller design. In order to reduce the conservatism, matrices $X(i)$ and $X_1(i)$ are set back to (3.51) and solve the resulting BMI directly. The numerical efficiency of solving BMI can be increased by taking the solution of the LMI in Theorem 3.4 as an initial condition. Therefore, a less conservative output-feedback controller can be derived as shown in the numerical example.

Example 3.4 Consider an NCS with an output-feedback controller as described by (3.34). Assume the Markovian delays $\tau_1(r_t) = [20\ 25]$ ms and $\tau_2(r_t) = [10\ 40]$ ms. The switching of Markovian delays is governed by the generator $\mathcal{A}$ given by

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 3 & -3 \end{bmatrix}.$$

The system parameters are as follows

$$A = -0.7, \quad B = 1, \quad C = 1.$$

Consider the positive definite matrices $P(i)$ and $P_1(i)$ as defined in (3.51). Set $\gamma = 0.4$ and solve the BMI in (3.48), the output-feedback controllers are derived as

$$\begin{aligned} A_c(1) &= -16.000, \quad & B_c(1) &= -3.000, \quad & C_c(1) &= -0.100, \\ A_c(2) &= -4.992, \quad & B_c(2) &= -2.099, \quad & C_c(2) &= -1.373, \end{aligned}$$

with

$$P_1(1) = \begin{bmatrix} 0.691 & 0.136 \\ 0.136 & 0.759 \end{bmatrix}, \quad P_1(2) = \begin{bmatrix} 0.901 & 0.006 \\ 0.006 & 0.635 \end{bmatrix},$$

$$Q_1 = \begin{bmatrix} 0.181 & 0.004 & -0.001 & 0.008 \\ 0.004 & 0.191 & -0.005 & -0.001 \\ -0.001 & -0.005 & 0.197 & 0.020 \\ 0.008 & -0.001 & 0.020 & 0.171 \end{bmatrix}, \quad Q_2 = \begin{bmatrix} 0.372 & 0.009 & -0.002 & 0.014 \\ 0.009 & 0.395 & -0.007 & -0.003 \\ -0.002 & -0.007 & 0.406 & 0.038 \\ 0.014 & -0.003 & 0.038 & 0.354 \end{bmatrix}.$$

With the initial condition $x^T(\theta) = 1$, $\theta \in [-\bar{\theta}, 0]$, the simulation is ran 500 times with different sample paths of transmission delays for a time horizon of $T = 5\,\text{s}$. One sample path of Markovian delays and their probability distributions are shown in Fig. 3.5 (a). The stationary probability distribution of $\tau_1(2)$ and $\tau_2(1)$ is 75%, $\tau_1(2)$ and $\tau_2(2)$ is 25%. The mean trajectory of the NCS with delay-dependent switching controller exponentially converges towards a ball around the origin of radius $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 2.222\,\text{s}$. The performance is 30.2% improved than the counterpart NCS, i.e. NCS with worst-case design controller, $t_{0.05} = 3.182\,\text{s}$.

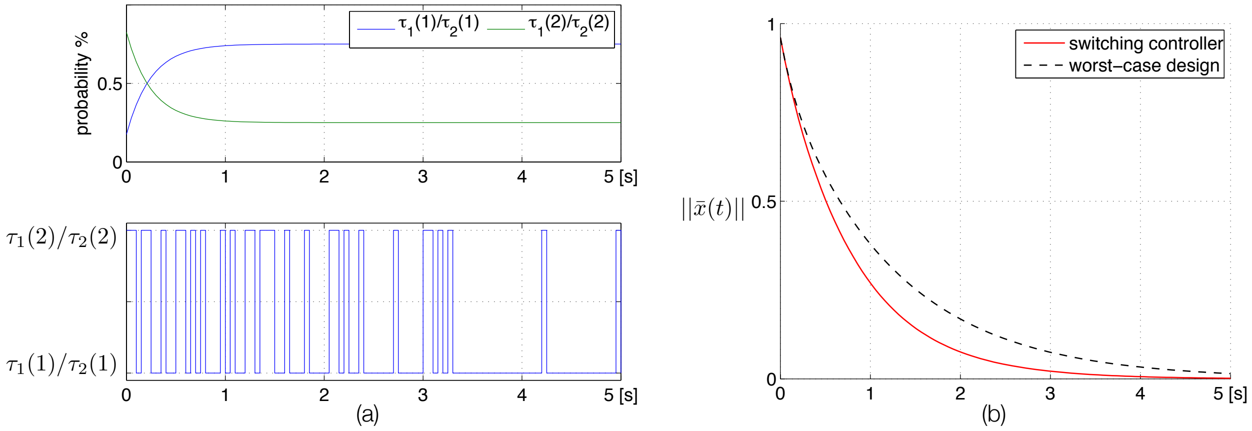


Figure 3.5: One sample path of the Markovian delays and associated probability distributions (a) and the mean state trajectory of NCS with delay-dependent switching controller (solid line) and NCS with worst-case design controller (dashed line) (b).

In summary, the proposed delay-dependent output-feedback controller design algorithm has superior control performance and is promising for NCS with random delays.

3.4 Guaranteed control performance for NCS with random delay

Section 3.2-3.2.2 concern the controller design ensuring the mean exponential stability for NCSs if the probability transition rates of delays, i.e. $\mathcal{A} = (\alpha_{i,j})$, $i, j \in \mathcal{S}$, are exactly known. However, for the real communication networks, the probability transition rates of delays are usually subjected to exogenous disturbance. The probability transition rate of delays might affect system stability and the control performance. It is useful to design an NCS, whose desired control performance is guaranteed despite the uncertainties in probability transition rate. The guaranteed control problem is therefore developed to maintain the stability and desired control performance of NCSs.

Let $R(r_t)$ be a set of symmetric, positive definite matrices. A cost function can be defined as

$$J_{per}(r_t) = \mathbb{E}\left\{ \int_0^\infty z^T(t)R(r_t)z(t)dt \big| z_0, r_0 \right\}. \tag{3.52}$$

Associated to the cost function (3.52), the guaranteed control performance is defined as follows.

Definition 3.1 (**Guaranteed control performance**)
Consider a set of MJLSs. If there exists a positive scalar $\bar{J}_{per}$ such that the cost function (3.52) satisfies $J_{per}(r_t) \leq \bar{J}_{per}$, then $\bar{J}_{per}$ is said to be a guaranteed cost on the control performance of NCSs.

3.4.1 State-feedback guaranteed control performance analysis

In this subsection, the guaranteed performance for a class of NCSs with state-feedback controller is studied. Consider the probability transition rate is perturbed by $\Delta\alpha_{i,j}$, $i, j \in \mathcal{S}$. According to properties of Markov probability transition rates, the uncertainty $\Delta\alpha_{i,j}$ has the properties:

(i) $\sum_{j=1}^{N} \Delta\alpha_{i,j} = 0$,

(ii) $\Delta\alpha_{i,j} > -\alpha_{i,j}$, $i \neq j$,

(iii) $\Delta\alpha_{i,i} < -\alpha_{i,i}$, otherwise.

As a result,the perturbed probability transition rate becomes $\mathcal{A} = (\alpha_{i,j} + \Delta\alpha_{i,j})$.

Consider a state-feedback controller, $K(i)$, of an NCS is determined by Theorem 3.2 with a probability transition rate $\mathcal{A} = (\alpha_{i,j})$, $i, j \in \mathcal{S}$. The following theorem determines an upper bound on the uncertainties $\Delta\alpha_{i,j}$ under which the mean exponential stability is ensured and the control performance is guaranteed.

Theorem 3.5 Consider an NCS in (3.4) with state-feedback controller (3.3) satisfying the matrix inequality (3.28) in Theorem 3.2. Let the Markov process transition generator be perturbed by $\Delta\alpha_{i,j}$, $i, j \in \mathcal{S}$. For given scalars $\gamma \geq 0$ and matrices $R(i) > 0$, if there exist matrices $Q > 0$ and $P_1(i) = P_1^T(i) > 0$, $i \in \mathcal{S}$ satisfying

$$P(i) = \begin{bmatrix} P_1(i) & 0 \\ P_2(i) & P_3(i) \end{bmatrix},$$

such that

$$\bar{\Theta}(i) = \begin{bmatrix} \Psi_1(i) & \tau(i)P^T(i)\hat{A}_1(i) \\ * & -\tau(i)Q \end{bmatrix} < 0, \tag{3.53}$$

where

$$\hat{\tau} = \frac{1}{2}(\bar{\tau}^2 - \underline{\tau}^2), \qquad \alpha_i = -\alpha_{i,i},$$

$$\Psi_1(i) = \hat{A}^T(i)P(i) + P^T(i)\hat{A}(i) + \sum_{j=1}^{N} \alpha_{i,j} EP(j) + \big(\tau(i) + \alpha_i\hat{\tau}\big)Q + \begin{bmatrix} I \\ 0 \end{bmatrix} R(i)[I \ 0]$$

holds and the perturbations of probability transition rate are bounded by

$$\Delta\alpha_i \leq \Delta\bar{\alpha}_i = \frac{\lambda_{\min}\big(-\bar{\Theta}(i)\big)}{\lambda_{\max}\big(\hat{\tau}Q\big) + \lambda_{\max}\big(\bar{P}(i)\big)}, \tag{3.54}$$

where $\bar{P}(i) = \sum_{j\neq i}^{N} EP(j) - EP(i)$, then the NCS is MES and the cost function defined in (3.52) is bounded by

$$\begin{aligned} J_{per}(r_t) \leq \bar{J}_{per}(r_0, \bar{\alpha}_{r_0} + \Delta\alpha_{r_0}) = \xi^T(0)EP(r_0)\xi(0) + \int_{-\tau(r_0)}^{0}\int_{\theta}^{0} \xi^T(s)Q\xi(s)dsd\theta \\ + (\alpha_{r_0} + \Delta\alpha_{r_0})\int_{-\bar{\tau}}^{-\underline{\tau}}\int_{\theta}^{0} \xi^T(s)Q\xi(s)(s-\theta)dsd\theta. \end{aligned} \tag{3.55}$$

Proof: Consider the same Lyapunov candidate as in Theorem 3.2 with non-perturbed probability transition rate $\mathcal{A} = (\alpha_{i,j})$. Based on Lemma A.1, the infinitesimal generator of the Lyapunov candidate is known as

$$\begin{aligned} \mathcal{L}V(\Xi(t), r_t) \leq\ & \xi^T(t)\Big[\hat{A}^T(r_t)P(r_t) + P^T(r_t)\hat{A}(r_t) + \sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t) \\ & + \tau(r_t)\xi^T(t)P^T(r_t)\hat{A}_1(r_t)Q^{-1}\hat{A}_1^T(r_t)P(r_t)\xi(t) \\ & + \Big(\tau(r_t) + \frac{1}{2}\alpha_i(\bar{\tau}^2 - \underline{\tau}^2)\Big)\xi^T(t)Q\xi(t) \\ =\ & \xi^T(t)\Theta(r_t)\xi(t), \end{aligned}$$

The mean exponential stability is ensured by $\Theta(i) < 0$. According to Dynkin's formula, it has

$$\mathbb{E}\Big\{\int_0^T \mathcal{L}V(\Xi(t), r_t)dt\big|\Xi(0), r_0\Big\} = \mathbb{E}\{V(\Xi(T)), r_T\} - \mathbb{E}\{V(\Xi(0), r_0)\}.$$

Note that $z(t) = [I \ 0]\xi(t)$. The cost function in (3.52) becomes

$$\begin{aligned} J_{per}(r_t) =\ & \mathbb{E}\Big\{\int_0^T \big[\xi^T(t)\begin{bmatrix} I \\ 0 \end{bmatrix} R(r_t)[I \ 0]\xi(t) + \mathcal{L}V(\Xi(t), r_t)\big]dt\big|\Xi(0), r_0\Big\} \\ & - \mathbb{E}\{V(\Xi(T), r_T)\} + \mathbb{E}\{V(\Xi(0), r_0)\} \\ \leq\ & \mathbb{E}\Big\{\int_0^T \xi^T(t)\bar{\Theta}(r_t)\xi(t)dt + V(\Xi(0), r_0)\Big\}, \end{aligned}$$

where $\bar{\Theta}(r_t) = \Theta(r_t) + \begin{bmatrix} I \\ 0 \end{bmatrix} R(r_t)[I\ 0]$. By the requirement of $\bar{\Theta}(r_t) < 0$, it yields

$$J_{per}(r_t) = \mathbb{E}\left\{ \int_0^\infty z^T(t)R(r_t)z(t)dt\big|z_0, r_0 \right\} \leq \mathbb{E}\{V(\Xi(0), r_0)\} = \bar{J}_{per}(r_0, \bar{\alpha}_{r_0} + \Delta\alpha_{r_0}).$$

Apply Schur complement to $\bar{\Theta}(r_t)$, it results in (3.53).

Assume the transition generator of Markov process r_t is perturbed by $\Delta\alpha_{i,j}$, i.e. $\mathcal{A} = (\alpha_{i,j} + \Delta\alpha_{i,j})$, $i, j \in \mathcal{S}$. Take $\alpha_{i,j} + \Delta\alpha_{i,j}$ into $\bar{\Theta}(i)$, it requires

$$\hat{\tau}\Delta\alpha_i Q + \sum_{N}^{j=1} \Delta\alpha_{i,j} EP(j) + \bar{\Theta}(i) < 0 \tag{3.56}$$

such that the mean exponential stability is ensured and the guaranteed control performance is guaranteed. Note that $\Delta\alpha_i = -\Delta\alpha_{i,i} = \sum_{i \neq j}^{N} \Delta\alpha_{i,j}$, it has

$$\sum_{N}^{j=1} \Delta\alpha_{i,j} EP(j) \leq \Delta\alpha_i \bigg(\sum_{j \neq i}^{N} EP(j) - EP(i) \bigg).$$

Choose a $\Delta\alpha_i$ in (3.56) such that the following inequality is satisfied

$$\Delta\alpha_i \lambda_{\max}(\hat{\tau}Q) + \Delta\alpha_i \lambda_{\max}\bigg(\sum_{j \neq i}^{N} EP(j) - EP(i) \bigg) \leq \lambda_{\max}(-\bar{\Theta}(i)).$$

As a result, the perturbation upper bound $\Delta\bar{\alpha}_i$ can be determined by

$$\Delta\bar{\alpha}_i = \frac{\lambda_{\min}\big(- \bar{\Theta}(i)\big)}{\lambda_{\max}\big(\hat{\tau}Q\big) + \lambda_{\max}\big(\bar{P}(i)\big)}, \tag{3.57}$$

where $\bar{P}(i) = \sum_{j \neq i}^{N} EP(j) - EP(i)$.

Based on (3.56), the perturbations on the probability transition rate is bounded by (3.57) and the cost function in (3.52) is bounded by (3.55). ■

Remark 3.9 The guaranteed control performance is an upper bound on the control performance obtained by a stochastic Lyapunov functional. It is an useful evaluation method to measure the expected value of performance index for systems with probabilistic uncertainties.

For a given NCS with Markovian random delay and delay-dependent state-feedback controller, the perturbation upper bound on the probability transition rate and guaranteed cost of the control performance can be determined by above theorem. The following numerical example shows the usefulness of previous results.

Example 3.5 Consider an NCS with an state-feedback controller as described by (3.3). Assume the Markovian delays having values $\tau(r_t) = [20\ 50]$ ms with probability transition rate

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}.$$

the system parameters and feedback gains are

$$A = \begin{bmatrix} -1 & 1 \\ 0 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix},$$

and

$$K(1) = [-6.3 \ -5.5], \quad K(2) = [-2.6 \ -2.2].$$

Set $\gamma = 0$ and

$$R(1) = R(2) = 10^{-4} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Solving Theorem 3.5, the positive definite matrices are

$$P_1(1) = 10^{-3} \times \begin{bmatrix} 0.154 & 0.017 \\ 0.017 & 0.064 \end{bmatrix}, \quad P_1(2) = 10^{-3} \times \begin{bmatrix} 0.152 & -0.002 \\ -0.002 & 0.073 \end{bmatrix},$$

$$Q = 10^{-3} \times \begin{bmatrix} 0.782 & 0.087 & 0.002 & 0.154 \\ 0.087 & 0.661 & -0.173 & 0.016 \\ 0.002 & -0.173 & 0.794 & -0.022 \\ 0.154 & 0.016 & -0.022 & 0.336 \end{bmatrix}.$$

The upper bounds on the perturbation of Markov process probability transition rate, $\Delta\bar{\alpha}_1$ and $\Delta\bar{\alpha}_2$, are determined by (3.54) and have the values

$$\Delta\alpha_1 = 0.207, \quad \Delta\alpha_2 = 0.559.$$

The variations of the probability transition rate become

$$\begin{bmatrix} -1.207 & 1.207 \\ 0.441 & -0.441 \end{bmatrix} \leq \mathcal{A} \leq \begin{bmatrix} -0.739 & 0.739 \\ 1.559 & -1.559 \end{bmatrix}.$$

This implies the stationary probability distributions of $\tau(1) = 20\,\text{ms}$ ranging from 26.8% to 66.3% (or from 33.7% to 73.2% for $\tau(2) = 50\,\text{ms}$). In the multi-hop wireless LAN, e.g. IEEE 802.15.4, the Markov probability transition rate well defines the probability distribution of waiting delays generated for collision avoidance. For NCS applications, the probability of waiting delays are meant to be adapted. The perturbation bounds on the Markov probability rate determines the feasible adaptable ranges of waiting delays where the stability and performance of underlying control systems are ensured.

3.4.2 Output-feedback guaranteed control performance analysis

The guaranteed cost on the performance of the output-feedback controller (3.32) can be established in the similar way as shown in Theorem 3.5. Define the cost function in (3.52) as

$$J_{per}(r_t) = \mathbb{E}\left\{ \int_0^\infty \chi^T(t) R(r_t) \chi(t) dt \big| z_0, r_0 \right\}, \tag{3.58}$$

where $R(t) >$. Assume an output-feedback controller is determined by Theorem 3.3 with a probability transition rate, $\mathcal{A} = \alpha_{i,j}$, $i, j \in \mathcal{S}$. Assume the probability transition rate is perturbed by $\Delta\alpha_{i,j}$ satisfying the conditions (i), (ii) and (iii) in the previous section. An upper bound of $\Delta\alpha_{i,j}$ will be determined in the following theorem such that the mean exponential stability is still ensured and the control performance (3.58) is guaranteed.

Theorem 3.6 Consider an NCS in (3.4) with output-feedback controller (3.32) satisfying the matrix inequality (3.48) in Theorem 3.4. Let the Markov process transition rate be perturbed by $\Delta\alpha_{i,j}$, for $i, j \in \mathcal{S}$. For given scalars $\gamma \geq 0$ and matrices $R(i) > 0$, if there exist matrices $Q_1 > 0$, $Q_2 > 0$ and $P_1(i) = P_1^T(i) > 0$, $i \in \mathcal{S}$ satisfying

$$P(i) = \begin{bmatrix} P_1(i) & 0 \\ P_2(i) & P_3(i) \end{bmatrix},$$

such that

$$\bar{\Theta}(i) = \begin{bmatrix} \Psi_1(i) & \tau_1(i)P^T(i)\hat{A}_1(i) & \tau_2(i)P^T(i)\hat{A}_2(i) \\ * & -\tau_1(i)Q_1 & 0 \\ * & * & -\tau_2(i)Q_2 \end{bmatrix} < 0, \tag{3.59}$$

where $\alpha_i = -\alpha_{i,i}$

$$\bar{\tau}_1 = \max_{i\in\mathcal{S}}\{\tau_1(i)\},\ \underline{\tau}_1 = \min_{i\in\mathcal{S}}\{\tau_1(i)\},\ \hat{\tau}_1 = \frac{1}{2}(\bar{\tau}_1^2 - \underline{\tau}_1^2),$$
$$\bar{\tau}_2 = \max_{i\in\mathcal{S}}\{\tau_2(i)\},\ \underline{\tau}_2 = \min_{i\in\mathcal{S}}\{\tau_2(i)\},\ \hat{\tau}_2 = \frac{1}{2}(\bar{\tau}_2^2 - \underline{\tau}_2^2),$$

$$\Psi_1(i) = \hat{A}^T(i)P(i) + P^T(i)\hat{A}(i) + \sum_{j=1}^{N}\alpha_{i,j}EP(j) + \big(\tau_1(i) + \alpha_i\hat{\tau}_1\big)Q_1 + \big(\tau_2(i) + \alpha_i\hat{\tau}_2\big)Q_2 + \begin{bmatrix} I \\ 0 \end{bmatrix} R(i)[I\ 0]$$

holds and the perturbations on probability transition generator are bounded by

$$\Delta\alpha_i \leq \Delta\bar{\alpha}_i = \frac{\lambda_{\min}(-\bar{\Theta}(i))}{\lambda_{\max}(\hat{\tau}_1 Q_1) + \lambda_{\max}(\hat{\tau}_2 Q_2) + \lambda_{\max}(\bar{P}(i))}, \tag{3.60}$$

where $\bar{P}(i) = \sum_{j\neq i}^{N} EP(j) - EP(i)$, then the system is also MES and the cost function in (3.58) is still bounded by

$$\begin{aligned} J_{per}(r_t) &\leq \bar{J}_{per}(r_0, \bar{\alpha}_{r_0} + \Delta\alpha_{r_0}) \\ &= \xi^T(0)EP(r_0)\xi(0) + \int_{-\tau_1(r_0)}^{0}\int_{\theta}^{0}\xi^T(s)Q_1\xi(s)dsd\theta \\ &\quad + \int_{-\tau_2(r_0)}^{0}\int_{\theta}^{0}\xi^T(s)Q_2\xi(s)dsd\theta \\ &\quad + (\alpha_{r_0} + \Delta\alpha_{r_0})\int_{-\bar{\tau}_1}^{-\underline{\tau}_1}\int_{\theta}^{0}\xi^T(s)Q_1\xi(s)(s-\theta)dsd\theta \\ &\quad + (\alpha_{r_0} + \Delta\alpha_{r_0})\int_{-\bar{\tau}_2}^{-\underline{\tau}_2}\int_{\theta}^{0}\xi^T(s)Q_2\xi(s)(s-\theta)dsd\theta. \end{aligned} \tag{3.61}$$

Proof: Consider the Lyapunov candidate in Theorem 3.4 with non-perturbed probability transition rate $\mathcal{A} = (\alpha_{i,j})$. The infinitesimal generator of the Lyapunov candidate, $\mathcal{L}V(\Xi(t), r_t)$, is given as

$$\begin{aligned}\mathcal{L}V(\Xi(t), r_t) \leq{}& \xi^T(t)\Big[\hat{A}^T(r_t)P(r_t) + P^T(r_t)\hat{A}(r_t) + \sum_{j=1}^{N}\alpha_{i,j}EP(j)\Big]\xi(t)\\ &+ \xi^T(t)\Big[\tau_1(r_t)P^T(r_t)\hat{A}_1(r_t)Q_1^{-1}\hat{A}_1^T(r_t)P(r_t)\\ &+ \tau_2(r_t)P^T(r_t)\hat{A}_1(r_t)Q_2^{-1}\hat{A}_1^T(r_t)P(r_t)\Big]\xi(t)\\ &+ \Big(\tau_1(r_t) + \frac{1}{2}\alpha_i(\bar{\tau}_1^2 - \underline{\tau}_1^2)\Big)\xi^T(t)Q_1\xi(t)\\ &+ \Big(\tau_2(r_t) + \frac{1}{2}\alpha_i(\bar{\tau}_2^2 - \underline{\tau}_2^2)\Big)\xi^T(t)Q_2\xi(t)\\ ={}& \xi^T(t)\Theta(r_t)\xi(t).\end{aligned}$$

The mean exponential stability is ensured by $\Theta(r_t) < 0$. Similar to the proof of Theorem 3.5, the cost function in (3.52) is bounded if the following inequality

$$\bar{\Theta}(i) = \Theta(i) + \begin{bmatrix} I \\ 0 \end{bmatrix} R(r_t) \begin{bmatrix} I & 0 \end{bmatrix}$$

is satisfied. Applying Schur complement to $\bar{\Theta}(r_t)$, the LMI in (3.59) is derived.

Consider the perturbed probability transition rate as $\mathcal{A} = (\alpha_{i,j} + \Delta\alpha_{i,j})$, $i, j \in \mathcal{S}$. Take $\alpha_{i,j} + \Delta\alpha_{i,j}$ into $\bar{\Theta}(i)$, it needs

$$\hat{\tau}_1\Delta\alpha_i Q_1 + \hat{\tau}_2\Delta\alpha_i Q_2 + \sum_{N}^{j=1}\Delta\alpha_{i,j}EP(j) + \bar{\Theta}(i) < 0 \tag{3.62}$$

such that the mean exponential stability is ensured and the guaranteed control performance is guaranteed. Let $\bar{P}(i) = \sum_{j\neq i}^{N} EP(j) - EP(i)$. The same as in Theorem 3.5, the perturbation upper bound $\bar{\Delta}\alpha_i$ is determined as

$$\Delta\bar{\alpha}_i = \frac{\lambda_{\min}\big(-\bar{\Theta}(i)\big)}{\lambda_{\max}\big(\hat{\tau}_1 Q_1\big) + \lambda_{\max}\big(\hat{\tau}_2 Q_2\big) + \lambda_{\max}\big(\bar{P}(i)\big)}. \tag{3.63}$$

According to (3.62), the perturbations on the probability transition rate is bounded by (3.63) and the cost function in (3.52) is bounded by (3.61). ■

The results for NCSs with output-feedback controller are demonstrated by the following numerical example.

Example 3.6 Consider an NCS with an output-feedback controller as described by (3.34). Assume the Markovian delays having values $\tau_1(r_t) = [15\ 32]$ ms and $\tau_2(r_t) = [15\ 28]$ ms. Set the Markov process probability transition rate as

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The system parameters are defined

$$A = -1.3, \quad B = 1, \quad C = 1.$$

with output-feedback controllers

$$\begin{aligned} A_c(1) &= -16.578, \quad B_c(1) = -1.348, \quad C_c(1) = -0.186; \\ A_c(2) &= -2.690, \quad B_c(2) = -2.099, \quad C_c(2) = -1.267. \end{aligned}$$

Set $\gamma = 0.4$ and

$$R(1) = R(2) = 10^{-4} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Solving Theorem 3.6, the positive definite matrices are

$$P_1(1) = 10^{-3} \times \begin{bmatrix} 0.434 & 0.009 \\ 0.009 & 0.314 \end{bmatrix}, \quad P_1(2) = 10^{-3} \times \begin{bmatrix} 0.902 & -0.261 \\ -0.261 & 0.389 \end{bmatrix},$$

$$Q_1 = 10^{-3} \times \begin{bmatrix} 1.363 & 0.622 & -0.112 & 0.131 \\ 0.622 & 1.640 & 0.049 & -0.056 \\ -0.112 & 0.049 & 2.235 & 0.082 \\ 0.131 & -0.056 & 0.082 & 0.397 \end{bmatrix},$$
$$Q_2 = 10^{-3} \times \begin{bmatrix} 1.967 & 0.066 & -0.007 & 0.010 \\ 0.066 & 1.999 & 0.020 & -0.007 \\ -0.007 & 0.020 & 2.037 & 0.001 \\ 0.010 & -0.007 & 0.001 & 1.791 \end{bmatrix}.$$

The upper bounds on the perturbation of Markov process probability transition rate, $\Delta\bar{\alpha}_1$ and $\Delta\bar{\alpha}_2$, are determined by (3.60) and have the values

$$\Delta\alpha_1 = 0.006, \quad \Delta\alpha_2 = 0.447.$$

According to (2.13) in Chapter 2, the perturbation bounds allow the stationary probability distribution of delays $\tau_1(1) = \tau_2(1) = 15\,\text{ms}$ to be varied between 35.5% and 59.3%.

Consider $\tau_1(r_t)$ and $\tau_2(r_t)$ as waiting delays of CSMA/CA network in SC and CA channels, respectively. Varying of probability distributions of waiting delays means adjusting the transmission priority within a CSMA/CA (Carrier Sensing Multiple Access/Collision Avoidance) network. For applications with higher transmission priority, the probability of short waiting delay is increased. Hence, the application performance is guaranteed. Increasing transmission priority of network applications requires the provision of large network resources, e.g. bandwidth. Due to the limited network capacities, it is desirable to design an NCS which consumes only as much network resources as required to guaranteed a certain performance level. Theorem 3.5 and Theorem 3.6 proposed in this section enables a trade-off design of control systems and their networks. Based on the two theorems, a novel Quality-of-Service (QoS) co-design approach will be explored in Chapter 5.

3.5 Summary and discussion

It is well-known that data sampling rates and communication qualities, e.g. random delays and packet dropouts, of NCSs have strong impact on their stability and control performance. Stochastic design approaches handling different specifications in a single criterion, and balancing control performance and system stability are, however, not available in the known literature. The goal here is to develop analysis and design tools for NCSs, in which the system and network specifications are conjointly considered. Targeted at the uncertainties caused by the network, guaranteed bounds on stability regions and control performance are required.

In this chapter, continuous-time LTI NCSs with periodic sampling are mainly considered. By applying the sampled-data system approach, the effects of periodic sampling and random packet dropouts are reformulated into delays. In order to accommodate the randomness of transmission delays within the analytical framework, a Markov process is introduced. LTI systems with Markovian delay are classified as MJSs with mode-dependent delay. Based on stochastic analysis methodologies, stability and stabilization conditions are determined for NCSs with delay-dependent state-feedback as well as output-feedback controllers. The switching controller design algorithms are derived in terms of multi-criteria LMIs depending on transmission delays, packet dropouts, sampling rate, and associated statistical properties of delays. This correlation allows an arbitration between transmission delays as well as sampling rate, and enables the exploration of stability (feasibility) versus statistical properties of delays in the analysis (or design) process. Furthermore, the robustness of stabilization conditions is analyzed towards uncertain Markovian delay models. The bounds on the stability region are determined for uncertain Markov probability transition rates of delays. The expected performance is shown to be limited by a guaranteed cost, which is derived by Markov probability transition rates of delays.

With known delay lengths and their probability transition rates, the stabilizing state-feedback (or output-feedback) controllers can be easily determined for continuous LTI systems by solving the proposed LMIs. The design algorithms can be also applied to nonlinear systems by using the standard linearization techniques. Although the LMI algorithms can be efficiently solved by existing LMI tools, the restrictions on LMIs introduce conservatism in the design. The LMIs might not find a feasible solution, even if there exists one. A less conservative algorithm involving BMI is also discussed in this chapter. An iterative method, which concerns the solutions of LMI as initial conditions, to tackle the numerical complicities of BMIs.

Numerical examples are considered for validations. The numerical results demonstrate the superior performance benefit of the proposed stochastic controller design algorithms over the worst-case design counterpart. This confirms the proposed control design algorithms are promising for NCS applications.

By referring the probability transition rates of delays to network specifications, it defines the probability distribution of waiting delays generated for collision avoidance in CSMA/CA networks. In the consideration of network capacity constraints, it is desirable to design an NCS and its underlying network conjointly such that the network resources are efficiently used and the control performance is guaranteed. The approaches presented in this chapter form an important basis for the control system and communication network co-design discussed in Chapter 5.

4 NCS with Aperiodic Sampling

In general, systems with higher sampling rate result in better control performance [35]. In views of NCSs with limited network capacities, however, higher sampling rate incurs higher traffic load, which might lead to data traffic congestion on a shared network. In order to alleviate data congestion, it could be advantageous if the sampling rate of an NCS can be adapted to network traffic conditions [75, 102, 110, 119]. This results in *NCSs with aperiodic data sampling.*

In order to keep the non-deterministic network attributes within the analytical framework, in Chapter 3, Markov processes are introduced to describe the random network-induced delays[1] by making use of the knowledge of their probability transition rates. However, the Markovian modeling requires the assumption of Markov properties. Namely, the outcome of the next sample depends *only* upon the current sample and independent of the past; the sojourn times (time between jumps) are random variables with exponential distribution. According to Chapter 3, the Markovian delay model contains the transmission delays, packet dropouts and sampling intervals. For NCSs with random sampling intervals, the sojourn time is equal to sampling intervals and has the same stochastic properties. If the sampling interval is not exponential distributed, the Markovian modeling is no more applicable.

The innovation of this chapter is to remove the Markovian restrictions posed on the aperiodic sampling modeling. To facilitate this, a set of indication functions is introduced for the modeling of aperiodic sampling intervals. Unlike approaches proposed in [60, 102, 110, 116, 119], where classical robust control methodologies are modified for systems with uncertain sampling intervals, the focus is to develop stochastic analysis and synthesis methods for NCSs, which are relevant to network usage. As a result, the proposed stability and stabilization conditions are determined in terms of probabilistic distributions of aperiodic sampling intervals. Different from the existing stochastic control approaches addressed in [41, 47, 54], innovative performance design algorithms are developed. These design algorithms relate the performance bound to the probability distributions of sampling intervals. This correlation enables a trade-off design between control performance and sampled-data flow. The analysis and design methods proposed in this chapter provides an important basis for the second co-design approach in Chapter 5, where a *optimal probabilistic sampling* is targeted.

The remainder of this chapter is organized as follows. In Section 4.1, the problem statement of an NCS with aperiodic sampling is formulated and a *randomly switched time-varying system* is introduced. The stability analysis and state-feedback controller design algorithm for NCSs with aperiodic sampling are presented in Section 4.2, whereas the output-feedback stability conditions and associated controller design algorithm are derived in Section 4.3. Concerning the network constraints, the guaranteed control performance design is studied in Section 4.4. The chapter is closed with discussion in Section 4.5. The

[1] The network-induced delays are composite of transmission delay and fictitious delays caused by dropouts.

introductions of software tool for the controller design algorithms derived in chapter are given in the Appendix A.1.2.

4.1 Random delays and aperiodic sampling intervals

Reconsider the LTI plant in Chapter 3

$$\begin{cases} \dot{x}(t) & = Ax(t) + B\bar{u}(t), \\ y(t) & = Cx(t). \end{cases} \tag{4.1}$$

In order to alleviate data congestion of the underlying communication network, the plant is aperiodically sampled. At the sampling instant t_k, $k \in \mathbb{N}$, the SC/CA delays are $\tau_{\text{sc}}(k)/\tau_{\text{ca}}(k)$ and the aperiodic sampling interval is $h(k)$, see Figure 4.1.

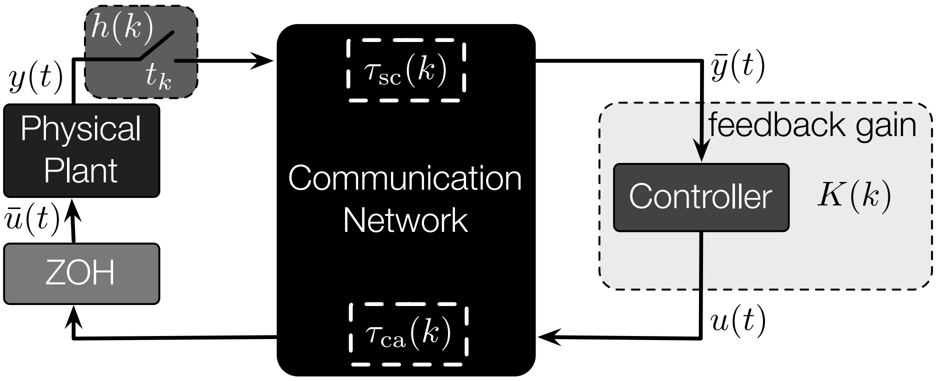


Figure 4.1: Illustration of a sampled-data NCS over communication network with random sampling interval $h(k)$ and random SC/CA delays $\tau_{\text{sc}}(k)/\tau_{\text{ca}}(k)$.

Assume the system (4.1) has a state-feedback controller, i.e. $y(t) = x(t)$. The control command $\bar{u}(t)$ becomes

$$\bar{u}(t) = Kx(t_k), \quad t \in [t_k + \tau_{\text{tx}}(k), t_{k+1} + \tau_{\text{tx}}(k+1)), \tag{4.2}$$

where $\tau_{\text{tx}}(k) = \tau_{\text{sc}}(k) + \tau_{\text{ca}}(k)$. Rewrite the piecewise constant $x(t_k)$ as

$$x(t_k) = x(t - (t - t_k)) = x(t - \tau(t)), \quad t \in [t_k + \tau_{\text{tx}}(k), t_{k+1} + \tau_{\text{tx}}(k+1)).$$

Substitute $x(t - \tau(t))$ into (4.2), the closed-loop system becomes

$$\dot{x}(t) = Ax(t) + BKx(t - \tau(t)), \quad t \in [t_k + \tau_{\text{tx}}(k), t_{k+1} + \tau_{\text{tx}}(k+1)). \tag{4.3}$$

System (4.3) is derived by embedding the transmission delay into a sampling interval. The non-equidistant sampling interval is reformulated as a time-varying delay by applying the input-delay approach [8, 38]. The resulting system (4.3) becomes a continuous-time system with time-varying delay. The stochastic time-varying delay $\tau(t)$ has the sojourn time correlated to the sampling intervals. Therefore, the Markovian modeling is no more applicable. In order to analyze system (4.3) within the framework of stochastic control, a set of independent identical distributed (i.i.d.) processes is used for delay modeling. As a consequence, a *randomly switched time-delay system* will be introduced in the following section.

Remark 4.1 If the packet containing $x(t_{k+1})$ is dropped during the transmission, the previous data $x(t_k)$ is utilized by the closed-loop system (4.3) due to the zero-order hold (ZOH). As a result, the effect of dropouts can be viewed as additional delay which grows by accumulating sampling periods. Assume the maximal number of consecutive dropouts as m. The closed-loop system in (4.3) has maximal and minimal delays are given in the following

$$\bar{\tau} = \max_{k\in\mathbb{N}}\{t_{k+m+1} - t_k + \tau_{\text{tx}}(k+m+1)\}, \quad \underline{\tau} = \min_{k\in\mathbb{N}}\{\tau_{\text{tx}}(k)\}.$$

4.1.1 Randomly switched time-delay system

The delay $\tau(t)$ in system (4.3) is classified into $N \geq 2$ intervals, which are assigned by the most significant maximal values of $\tau(t)$ between two consecutive sampling intervals as shown in Fig. 4.2. The n intervals are defined by $s_i > 0$, $i = 1, \ldots, N-1$, satisfying $s_i < s_{i+1}$, $s_0 = \underline{\tau}$ and $s_N = \bar{\tau}$. The n interval delays are defined as

$$\begin{aligned} \tau_1 &= \{\tau(t) | s_0 \leq \tau(t) < s_1\}, \\ \tau_2 &= \{\tau(t) | s_1 \leq \tau(t) < s_2\}, \\ &\vdots \\ \tau_N &= \{\tau(t) | s_{N-1} \leq \tau(t) < s_N\}. \end{aligned} \tag{4.4}$$

The occurrence of τ_i, $i = 1, \ldots, n$, is described by a set of indicator functions

$$\beta_i = \begin{cases} 1, & s_{i-1} \leq \tau(t) < s_i, \quad i = 1, \ldots, N, \\ 0, & \text{otherwise}. \end{cases} \tag{4.5}$$

The indicator functions have the occurrence probabilities

$$\mathbf{P}\{\beta_i = 1\} = p_i, \quad \sum_{i=1}^{N} p_i = 1.$$

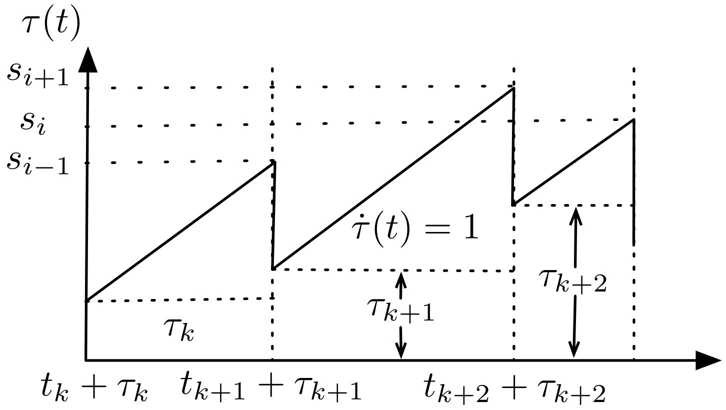


Figure 4.2: The evolution of time-varying delay $\tau(t)$.

Due to the i.i.d. assumption on the delays, the occurrence of τ_i is also i.i.d. [43]. This implies the indicator function β_i has Bernoulli distribution (i.i.d. process with binary random variables). As a result, the occurrence probability of $\tau_i(t)$ becomes

$$\mathbb{E}\{\beta_i\} = p_i, \quad \mathbb{E}\{(\beta_i - p_i)^2\} = p_i(1 - p_i).$$

Assume the state-feedback controller being able to switch its feedback gain according to delay intervals. For the ease of stability analysis, the upper bound s_i of each delay interval τ_i is considered in the controller design. As a result, the control input of the system in (4.1) becomes

$$\bar{u}(t) = \sum_{i=1}^{N} \beta_i K_i x(t - s_i), \quad i = 1, \ldots, N. \tag{4.6}$$

According to (4.6)-(4.5), the closed-loop system (4.1) can be rewritten as

$$\dot{x}(t) = Ax(t) + \sum_{i=1}^{N} \beta_i BK_i x(t - s_i). \tag{4.7}$$

Remark 4.2 Note that the feedback gain K_i of system (4.7) is switched according to delays s_i and results in a *randomly switched time-delay system.* It is assumed that each switching of (4.7) is separated by a finite time interval. Therefore, the Zeno solutions are excluded in this dissertation.

4.2 Stability and stabilization with delay-dependent state-feedback controller

In order to derive a delay-dependent condition, the descriptor transformation used in Chapter 3 is considered. Set a new variable

$$z(t) = e^{\gamma t} x(t),$$

where $\gamma \geq 0$. The closed-loop system in (4.7) becomes

$$\dot{z}(t) = (A + \gamma)z(t) + \sum_{i=1}^{N} e^{\gamma s_i} \beta_i K_i z(t - s_i). \tag{4.8}$$

Let $\xi^T(t) = [z^T(t)\ \dot{z}^T(t)]$, the closed-loop system in (4.8) becomes

$$E\dot{\xi}(t) = \hat{A}\xi(t) - \sum_{i=1}^{N} \hat{A}_i \int_{t-s_i}^{t} \xi(s)ds, \tag{4.9}$$

where

$$E = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \ \hat{A} = \begin{bmatrix} 0 & I \\ A + \gamma I + \sum_{i=1}^{N} e^{\gamma s_i} \beta_i BK_i & -I \end{bmatrix}, \ \hat{A}_i = \begin{bmatrix} 0 & 0 \\ 0 & e^{\gamma s_i} \beta_i BK_i \end{bmatrix}.$$

The stability of the system represented by (4.9) implies the stability of the origin system in (4.8). Therefore, in the following section, the transformed system (4.9) is considered for the analysis.

4.2.1 Stability analysis

The objective of this section is to derive a mean exponential stability condition for the system in (4.9). As system (4.9) contains stochastic variables and time-delays, the Lyapunov-Krasovskii approach proposed in Chapter 3 is applied for the stability analysis. The stability results derived in Chapter 3 are conditioned by the transition generator of Markovian delays, while the stability results in this section are determined by the occurrence probabilities of random delays. The details are given in Theorem 4.1.

Theorem 4.1 For the closed-loop system in (4.9) with a given $\gamma \geq 0$, if there exist symmetric matrices, $Q_i > 0$, $i = 1, \dots, N$, $P_1 > 0$ and real matrices P_2 and P_3 with

$$P = \begin{bmatrix} P_1 & 0 \\ P_2 & P_3 \end{bmatrix},$$

such that the following LMI satisfies

$$\begin{bmatrix} \Psi & s_1 P^T & \cdots & s_N P^T \\ * & -s_1 Q_1 & 0 & \vdots \\ \vdots & 0 & \ddots & * \\ * & \cdots & * & -s_N Q_N \end{bmatrix} < 0, \tag{4.10}$$

where

$$\Psi = \begin{bmatrix} \Xi_1 & \Xi_2 \\ P_1 - P_2 & -P_3 \end{bmatrix} + \begin{bmatrix} \Xi_1 & \Xi_2 \\ P_1 - P_2 & -P_3 \end{bmatrix}^T + \sum_{i=1}^{N} s_i \begin{bmatrix} 0 & 0 \\ 0 & e^{\gamma s_i} p_i B K_i \end{bmatrix}^T Q_i \begin{bmatrix} 0 & 0 \\ 0 & e^{\gamma s_i} p_i B K_i \end{bmatrix},$$

$$\Xi_1 = A^T P_2 + \sum_{i=1}^{N} e^{\gamma s_i} p_i (BK_i)^T P_2, \quad \Xi_2 = A^T P_3 + \sum_{i=1}^{N} e^{\gamma s_i} p_i (BK_i)^T P_3,$$

then the system MES.

Proof: Consider a Lyapunov candidate

$$V(\xi(t)) = V_0(\xi(t)) + \sum_{i=1}^{N} V_i(\xi(t)),$$

where

$$V_0(\xi(t)) = \xi^T(t) E P \xi(t), \quad V_i(\xi(t)) = \int_{-s_i}^{0} \int_{t+\theta}^{t} \xi^T(s) \hat{A}_i^T Q_i \hat{A}_i \xi(s)\, ds\, d\theta.$$

It has

$$\begin{aligned} \mathcal{L}V_0(\xi(t)) &= \dot{\xi}^T(t) E P \xi(t) + \xi^T(t) P^T E \dot{\xi}(t) \\ &= \xi^T(t) \big[\hat{A}^T P + P^T \hat{A}\big] \xi(t) - 2 \sum_{i=1}^{N} \xi^T(t) P^T \hat{A}_i \int_{t-s_i}^{t} \xi(s)\, ds. \end{aligned}$$

According to Lemma A.1, $\mathcal{L}V_0(\xi(t))$ becomes

$$\begin{aligned}\mathcal{L}V_0(\xi(t)) &\leq \xi^T(t)\big[\hat{A}^TP + P^T\hat{A}\big]\xi(t) + \sum_{i=1}^{N} s_i\xi^T(t)P^TQ_i^{-1}P\xi(t) \\ &\quad + \sum_{i=1}^{N}\int_{t-s_i}^{t}\xi^T(s)\hat{A}_i^TQ_i\hat{A}_i\xi(s)\,ds.\end{aligned} \tag{4.11}$$

Likewise, it has

$$\sum_{i=1}^{N}\mathcal{L}V_i(\xi(t)) = \sum_{i=1}^{N} s_i\xi^T(t)\hat{A}_i^TQ_i\hat{A}_i\xi(t) - \sum_{i=1}^{N}\int_{t-s_i}^{t}\xi^T(s)\hat{A}_i^TQ_i\hat{A}_i\xi(s)\,ds. \tag{4.12}$$

Combine (4.11) and (4.12), it yields

$$\begin{aligned}\mathcal{L}V(\xi(t)) &\leq \xi^T(t)[\hat{A}^TP + P^T\hat{A} + \sum_{i=1}^{N} s_i\hat{A}_i^TQ_i\hat{A}_i + \sum_{i=1}^{N} s_iP^TQ_i^{-1}P]\xi(t) \\ &= \xi^T(t)\Theta\xi(t).\end{aligned} \tag{4.13}$$

Apply Schur complement to (4.13), it results in (4.10).

Note that $\max_{\theta\in[-\bar{\tau},0]}\{||\xi(t+\theta)||\} \leq \phi||\xi(t)||$ for some $\varphi > 0$ [83], the following inequality can be established

$$\begin{aligned}V(\xi(t)) &\leq \Big[\lambda_{\max}(EP) + \sum_{i=1}^{N}\frac{s_i^2}{2}\lambda_{\max}(\hat{A}_i^TQ_i\hat{A}_i)\Big]||\xi(t)||^2 \\ &\leq \Lambda_{\max}||\xi(t)||^2.\end{aligned} \tag{4.14}$$

Combining (4.13) and (4.14) yields

$$\frac{\mathcal{L}V(\xi(t))}{V(\xi(t))} \leq -\frac{\lambda_{\min}(-\Theta)}{\Lambda_{\max}} \triangleq -\rho_0$$

and

$$\mathbb{E}\{\mathcal{L}V(\xi(t))\} \leq -\rho_0\mathbb{E}\{V(\xi(t))\}. \tag{4.15}$$

By applying Dynkin's formula into (4.15) it becomes

$$\mathbb{E}\{V(\xi(t))\} - \mathbb{E}\{V(\xi(0))\} = \mathbb{E}\bigg\{\int_0^t \mathcal{L}V(\xi(s))ds\bigg\} \leq -\rho_0\int_0^t \mathbb{E}\{V(\xi(s))\}\,ds. \tag{4.16}$$

Using the Gronwall-Bellman lemma, (4.16) results in

$$\mathbb{E}\{V(\xi(t))\} \leq e^{-\rho_0 t}\mathbb{E}\{V(\xi(0))\}.$$

Since

$$V(\xi(t)) \geq \Big[\lambda_{\min}(EP) + \sum_{i=1}^{N}\frac{s_i^2}{2}\lambda_{\min}(Q_i)\Big]||\xi(t)||^2 = \Lambda_{\min}||\xi(t)||^2,$$

it is established that

$$\mathbb{E}\{||\xi(t)||^2\} \leq e^{-\rho_0 t}\frac{\mathbb{E}\{V(\xi(0))\}}{\Lambda_{\min}}. \tag{4.17}$$

Equation (4.17) provides the proof for exponential mean square stability. ■

Remark 4.3 Note that $\mathbb{E}\{||\xi(t)||^2\} \geq \mathbb{E}\{||z(t)||^2\}$, and $z(t) = e^{\gamma t}x(t)$. Therefore, it has

$$\mathbb{E}\{||x(t)||^2\} \leq e^{-(\rho_0+2\gamma)t}\frac{\mathbb{E}\{V(\xi(0))\}}{\Lambda_{\min}}. \tag{4.18}$$

The given $\gamma \geq 0$ in Theorem 4.1 ensures the decay rate of trajectory $\mathbb{E}\{||x(t)||^2\}$ and determines the control performance of the closed-loop system (4.7).

The stochastic Lyapunov functional $V(\xi(t))$ is derived based on its quadratic form $V_1(\xi(t))$. The Lyapunov candidate $V_2(\xi(t))$ is chosen to compensate the integral terms caused by the derivative of $V_1(\xi(t))$. The main difference of stability results derived in Theorem 4.1 and Theorem 3.1 in Chapter 3 is that Theorem 4.1 is conditioned by occurrence probabilities of random delays, while Theorem 3.1 is determined by probability transition rates of Markovian delays. Furthermore, Theorem 4.1 allows the probabilistic sampling intervals of LTI systems. It can be used to determine a proper probability distribution of sampling intervals for LTI systems under the stability constraint; meanwhile, the data flow on the shared network can be reduced. This results in a data congestion control as discussed in Section 5.3.

For nonlinear NCSs, Theorem 4.1 can be applied by using standard linearizing techniques.

4.2.2 State-feedback stabilization

Solving feedback gains K_i, $i = 1, \ldots, N$ in Theorem 4.1 involves nonlinear terms, e.g. $P_2^T BK_i$ and $P_3^T BK_i$ in (4.10). These nonlinear terms render the inequality in (4.10) into a bilinear matrix inequality (BMI) problem, whose solutions are difficult to find as it is non-convex and NP-hard. However, the nonlinear terms can be eliminated by choosing a special matrix $X = P^{-1}$ and an LMI formulation is recovered. The controller design algorithm is given in the following theorem.

Theorem 4.2 For given positive scalars $r_1 > 0$, $r_2 > 0$ and $\gamma \geq 0$, if there exist symmetric matrices $U_i > 0$, $i = 1, \ldots, N$, and $X_1 = X_1^T > 0$ satisfying

$$X = \begin{bmatrix} X_1 & 0 \\ -r_1X_1 & r_2X_1 \end{bmatrix},$$

such that

$$\begin{bmatrix} \hat{\Psi} & \hat{\Psi}_1^T & \cdots & \hat{\Psi}_N^T \\ * & -s_1U_1 & 0 & \vdots \\ \vdots & 0 & \ddots & * \\ * & \cdots & * & -s_NU_N \end{bmatrix} < 0, \tag{4.19}$$

where

$$\hat{\Psi} = \begin{bmatrix} -r_1X_1 & r_2X_1 \\ \Xi_3 & -r_2X_1 \end{bmatrix} + \begin{bmatrix} -r_1X_1 & r_2X_1 \\ \Xi_3 & -r_2X_1 \end{bmatrix}^T + \sum_{i=1}^{N} s_iU_i,$$

$$\Xi_3 = AX_1 + \gamma X_1 + \sum_{i=1}^{N} e^{\gamma s_i}p_iBY_i + r_1X_1,$$

$$\hat{\Psi}_1 = s_1\hat{A}_1X = s_1 \begin{bmatrix} 0 & 0 \\ -r_1e^{\gamma s_1}p_1BY_1 & r_2e^{\gamma s_1}p_1BY_1 \end{bmatrix},$$
$$\vdots$$
$$\hat{\Psi}_N = s_N\hat{A}_NX = s_N \begin{bmatrix} 0 & 0 \\ -r_1e^{\gamma s_N}p_NBY_N & r_2e^{\gamma s_N}p_NBY_N \end{bmatrix},$$

holds, then the closed-loop system (4.8) is MES with the feedback gain

$$K_i = Y_iX_1^{-1}, \quad i = 1, \ldots, N. \tag{4.20}$$

Proof: Define

$$X = P^{-1} = \begin{bmatrix} X_1 & 0 \\ -r_1X_1 & r_2X_1 \end{bmatrix}.$$

Pre- and post-multiply Θ in (4.13) by X^T and X, it becomes

$$\hat{A}X + X^T\hat{A}^T + \sum_{i=1}^{N} s_iQ_i^{-1} + \sum_{i=1}^{N} s_iX^T\hat{A}_i^TQ_i\hat{A}_iX < 0. \tag{4.21}$$

Let $U_i = Q_i^{-1}$ and $Y_i = K_iX_1$, $i = 1, \ldots, N$. Applying Schur complement to (4.21) results in (4.19). ■

If no feasible solutions can be found by LMI (4.19), a less conservative approach is to set X as

$$X = \begin{bmatrix} X_1 & 0 \\ X_2 & X_3 \end{bmatrix}, \quad X_1 = X_1^T > 0 \tag{4.22}$$

and solve the BMI (bilinear matrix inequality). This is shown in the following numerical example.

Example 4.1 Consider an NCS with dynamics described by (4.1), which is i.i.d. sampled between 5 ms to 45 ms. The used network has i.i.d. transmission delay with the value of 30 ms to 40 ms depending on the network traffic. Set the parameters $\gamma = 1.3$ and $N = 2$. Choose $s_1 = 45\,\text{ms}$, $p_1 = 60\%$, $s_2 = 65\,\text{ms}$, $p_2 = 30\%$ and $s_3 = 85\,\text{ms}$, $p_3 = 10\%$. The system parameters are

$$A = \begin{bmatrix} 0 & 1 \\ 1 & -50 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}.$$

Set the positive definite matrix X as in (4.22). The stabilizing state-feedback gains are derived as

$$K_1 = [-11.956 \ -23.876], \quad K_2 = [-6.193 \ -12.171], \quad K_3 = [-2.832 \ -5.542]$$

with

$$X_1 = \begin{bmatrix} 169.956 & -10.193 \\ -10.193 & 6.588 \end{bmatrix}$$

$$U_1 = 10^4 \times \begin{bmatrix} 0.250 & -0.001 & -0.101 & -0.063 \\ -0.001 & 0.200 & -0.219 & -0.136 \\ -0.101 & -0.219 & 1.007 & 0.514 \\ -0.063 & -0.136 & 0.514 & 0.336 \end{bmatrix},$$

$$U_2 = 10^3 \times \begin{bmatrix} 2.124 & -0.254 & -0.325 & -0.192 \\ -0.254 & 1.308 & -0.503 & -0.369 \\ -0.325 & -0.503 & 3.915 & 0.803 \\ -0.192 & -0.369 & 0.803 & 1.134 \end{bmatrix},$$

$$U_3 = 10^3 \times \begin{bmatrix} 1.877 & -0.328 & -0.091 & -0.011 \\ -0.328 & 0.974 & 0.027 & -0.111 \\ -0.091 & 0.027 & 2.055 & -0.611 \\ -0.011 & -0.111 & -0.611 & 0.556 \end{bmatrix}.$$

With the initial condition $x^T(\theta) = [2 \ -2]$, $\theta \in [-\bar{\tau}, 0]$, the simulation is performed 500 times with different sample paths of delays for a time horizon of $T = 1.5\,\text{s}$. The proposed switching controller design and worst-case design approaches are investigated. The mean trajectory of the switching controller design converges towards a ball around $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 0.472\,\text{s}$, 48.7% faster than the worst-case design $t_{0.05} = 0.914\,\text{s}$, see Fig. 4.3 (a).

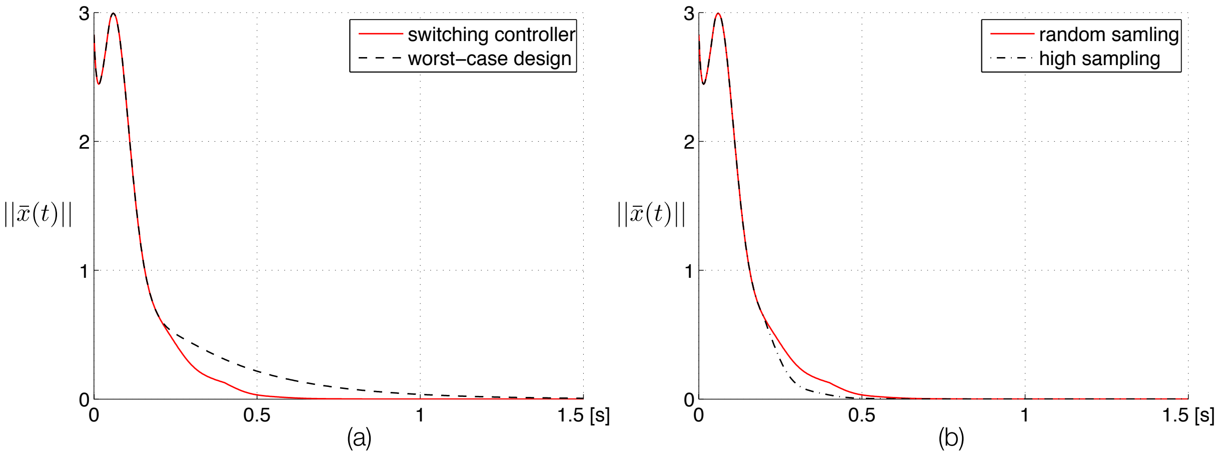


Figure 4.3: The mean state trajectory of NCS with delay-dependent switching controller (solid line) and NCS with worst-case design controller (dashed line) (a) and the mean state trajectory of NCS with random sampling (solid line) and NCS with high sampling (dashed line) (b).

In order to illustrate the benefits of random sampling for NCSs, consider the same NCS sampled by three i.i.d. sampling intervals 5 ms, 25 ms and 45 ms with probability distributions 60%, 30% and 10%, respectively. Assume the used network has constant transmission delay 40 ms. Hence, the NCS is subjected to three delay intervals $s_1 = 45\,\text{ms}$, $p_1 = 60\%$, $s_2 = 65\,\text{ms}$, $p_2 = 30\%$ and $s_3 = 85\,\text{ms}$, $p_3 = 10\%$. Set $\gamma = 1.3$ and $N = 2$, the NCS has the same state-feedback controllers as described above. With the initial condition $x^T(\theta) = [2 \ -2]$, $\theta \in [-\bar{\tau}, 0]$, another simulation is performed 500 times for NCS with random sampling intervals for a time horizon of $T = 1.5\,\text{s}$. The evolution of mean trajectory is shown in Fig. 4.3 (b). The mean trajectory of the NCS with random sampling converges towards a ball around $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 0.472\,\text{s}$, close to the NCS with high sampling rate $t_{0.05} = 0.368\,\text{s}$ (+28.3%). However, the data flow is only 33.3% of the NCS with high sampling.

As a summary, the switching controller design algorithm proposed in this Chapter demonstrates a superior performance over the traditional worst-case design. In views of data flow reduction, it is shown that the NCS with random sampling achieves acceptable performance at low data flow.

4.3 Stability and stabilization with delay-dependent output-feedback controller

The stability of NCSs with aperiodic sampling and delay-dependent output-feedback controller is studied in this section. Consider the LTI plant in (4.1) with an output-feedback controller

$$\begin{aligned}\dot{x}_c(t) &= A_c x(t) + B_c \bar{y}(t),\\ \bar{u}(t) &= C_c x_c(t),\end{aligned} \tag{4.23}$$

where $x_c(t) \in \mathbb{R}$ is the controller state with $x_c(t) = 0$ for $t \leq 0$, A_c, B_c and C_c are designed parameters with appropriate dimensions.

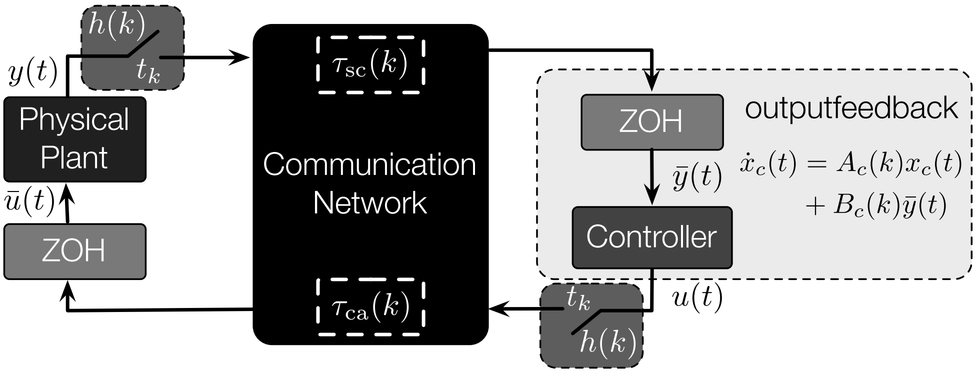


Figure 4.4: Illustration of a sampled-data NCS over communication network with random sampling interval $h(k)$ and random and random SC/CA delays $\tau_{\text{sc}}(k)/\tau_{\text{ca}}(k)$.

As shown in Figure 4.4, the sensor and output-feedback controller are aperiodically sampled by $h(k)$ at the sampling instant t_k, $k \in \mathbb{K}$. Assume the SC and CA delays having the same value at the sampling instant t_k. It implies $\tau_{\text{sc}}(k) = \tau_{\text{ca}}(k) = \tau(k)$. Therefore, the sampled measurement at $t \in [t_k + \tau(k), t_{k+1} + \tau(k+1))$ becomes

$$\begin{aligned}\bar{y}(t) &= Cx(t_k) = Cx(t - \tau(t)),\\ \bar{u}(t) &= C_c x_c(t_k) = C_c x_c(t - \tau(t)).\end{aligned}$$

Assume the maximal consecutive dropouts as m. The time-varying delay $\tau(t)$ is upper and lower bounded by

$$\bar{\tau} = \max_{k\in\mathbb{N}}\{t_{k+m+1} - t_k + \tau(k+m+1)\}, \quad \underline{\tau} = \min_{k\in\mathbb{N}}\{\tau(k)\}.$$

Categorize $\tau(t)$ into $N \geq 2$ intervals. The delay-dependent output-feedback controller in (4.23) becomes

$$\begin{aligned}\dot{x}_c(t) &= \sum_{i=1}^{N} \beta_i \Big(A_{ci} x_c(t) + B_{ci} C x(t - s_i) \Big), \\ \bar{u}(t) &= \sum_{i=1}^{N} \beta_i C_{ci} x_c(t - s_i).\end{aligned} \tag{4.24}$$

Combining (4.1) and (4.24), the closed-loop system becomes

$$\begin{bmatrix} \dot{x}(t) \\ \dot{x}_c(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ 0 & \sum_{i=1}^{N} \beta_i A_{ci} \end{bmatrix} \begin{bmatrix} x(t) \\ x_c(t) \end{bmatrix} + \sum_{i=1}^{N} \begin{bmatrix} 0 & \beta_i B C_{ci} \\ \beta_i B_{ci} C & 0 \end{bmatrix} \begin{bmatrix} x(t - s_i) \\ x_c(t - s_i) \end{bmatrix} \tag{4.25}$$

Remark 4.4 Note that the parameters of the output-feedback controller A_{ci}, B_{ci} and C_{ci} are switched according to delays s_i. The same as in section 4.1.1, system (4.25) is *randomly switched time-delay system.* The Zeno solutions of system (4.25) are excluded by the assumption that each consecutive switching is separated by a finite time interval.

For any $\gamma \geq 0$, consider $z(t) = e^{\gamma t} x(t)$ and $z_c(t) = e^{\gamma t} x_c(t)$. Define $\chi^T(t) = [z^T(t) \; z_c^T(t)]$. The closed-loop system in (4.25) yields

$$\dot{\chi}(t) = \bar{A}_0 \chi(t) + \sum_{i=1}^{N} \bar{A}_i \chi(t - s_i), \tag{4.26}$$

where

$$\bar{A}_0 = \begin{bmatrix} A + \gamma I & 0 \\ 0 & \sum_{i=1}^{N} \beta_i A_{ci} + \gamma I \end{bmatrix}, \quad \bar{A}_i = \begin{bmatrix} 0 & \beta_i e^{\gamma s_i} B C_{ci} \\ \beta_i e^{\gamma s_i} B_{ci} C & 0 \end{bmatrix}.$$

Apply the system transformation and let $\xi^T(t) = [\chi^T(t) \; \dot{\chi}^T(t)]$, it becomes

$$E \dot{\xi}(t) = \hat{A} \xi(t) - \sum_{i=1}^{N} \hat{A}_i \int_{t - s_i}^{t} \xi(s) ds. \tag{4.27}$$

$$E = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \quad \hat{A} = \begin{bmatrix} 0 & I \\ \bar{A}_0 + \sum_{i=1}^{N} \bar{A}_i & -I \end{bmatrix}, \quad \hat{A}_i = \begin{bmatrix} 0 & 0 \\ 0 & \bar{A}_i \end{bmatrix}.$$

As mentioned in the previous section, the transformed system in (4.27) is equivalent to the original system in (4.26). The system in (4.27) is considered in the following section for the stability analysis.

4.3.1 Stability analysis

The delay-dependent stability for output-feedback controller is derived by using the Lyapunov-Krasovskii functional approach. Since the transformed system in (4.27) has similar form as the system with state-feedback controller in (4.9), the same Lyapunov candidate as used in Theorem 4.1 is considered. The stability results are conditioned by the occurrence probabilities of random delays. Details of the stability results are summarized in Theorem 4.3.

Theorem 4.3 For the closed-loop system in (4.26) with a given $\gamma \geq 0$, if there exist symmetric matrices, $Q_i > 0$, $i = 1, \ldots, N$, $P_1 > 0$ and real matrices P_2 and P_3 with

$$P = \begin{bmatrix} P_1 & 0 \\ P_2 & P_3 \end{bmatrix},$$

such that the following LMI satisfies

$$\begin{bmatrix} \Psi & s_1 P^T & \cdots & s_N P^T \\ * & -s_1 Q_1 & 0 & \vdots \\ \vdots & 0 & \ddots & * \\ * & \cdots & * & -s_N Q_N \end{bmatrix} < 0, \tag{4.28}$$

where

$$\Psi = \begin{bmatrix} \Xi_1 & \Xi_2 \\ P_1 - P_2 & -P_3 \end{bmatrix} + \begin{bmatrix} \Xi_1 & \Xi_2 \\ P_1 - P_2 & -P_3 \end{bmatrix}^T$$
$$+ \sum_{i=1}^{N} s_i \begin{bmatrix} 0 & 0 \\ 0 & \begin{bmatrix} 0 & e^{\gamma s_i} p_i B C_{ci} \\ e^{\gamma s_i} p_i B_{ci} C & 0 \end{bmatrix} \end{bmatrix}^T Q_i \begin{bmatrix} 0 & 0 \\ 0 & \begin{bmatrix} 0 & e^{\gamma s_i} p_i B C_{ci} \\ e^{\gamma s_i} p_i B_{ci} C & 0 \end{bmatrix} \end{bmatrix},$$

$$\Xi_1 = \begin{bmatrix} A + \gamma I & 0 \\ 0 & \sum_{i=1}^N p_i A_{ci} + \gamma I \end{bmatrix}^T P_2 + \sum_{i=1}^{N} \begin{bmatrix} 0 & e^{\gamma s_i} p_i B C_{ci} \\ e^{\gamma s_i} p_i B_{ci} C & 0 \end{bmatrix}^T P_2,$$

$$\Xi_2 = \begin{bmatrix} A + \gamma I & 0 \\ 0 & \sum_{i=1}^N p_i A_{ci} + \gamma I \end{bmatrix}^T P_3 + \sum_{i=1}^{N} \begin{bmatrix} 0 & e^{\gamma s_i} p_i B C_{ci} \\ e^{\gamma s_i} p_i B_{ci} C & 0 \end{bmatrix}^T P_3,$$

then the system is MES.

Proof: see the proof of Theorem 4.1. ■

Remark 4.5 In Theorem 4.3, the control performance of system (4.25) can be pre-defined by $\gamma \geq 0$. The decay rate of trajectory $\mathbb{E}\{||x(t)||^2\}$ is determined in (4.18).

4.3.2 Output-feedback stabilization

Solving the output-feedback controller parameters A_{ci}, B_{ci} and C_{ci}, $i = 1, \ldots, N$ in Theorem 4.3 involves nonlinear terms, e.g. Ψ in (4.28). These nonlinear terms render the inequality in (4.28) into a bilinear matrix inequality (BMI) problem. However, there nonlinear terms can be eliminated by a special structure requirement of $X = P^{-1}$. The details about output-feedback parameters designed are summarized in Theorem 4.6.

Theorem 4.4 For given positive scalars $r_1 > 0$, $r_2 > 0$ and $\gamma \geq 0$, if there exist symmetric matrices $U_i > 0$, $i = 1, \ldots, N$, and $X_1 = X_1^T > 0$ satisfying

$$X_1 = \begin{bmatrix} X_{11} & 0 \\ 0 & X_{12} \end{bmatrix},$$

$$X = \begin{bmatrix} X_1 & 0 \\ -r_1 X_1 & r_2 X_1 \end{bmatrix},$$

such that

$$\begin{bmatrix} \hat{\Psi} & \hat{\Psi}_1^T & \cdots & \hat{\Psi}_N^T \\ * & -s_1 U_1 & 0 & \vdots \\ \vdots & 0 & \ddots & * \\ * & \cdots & * & -s_N U_N \end{bmatrix} < 0, \tag{4.29}$$

where

$$\hat{\Psi} = \begin{bmatrix} -r_1 X_1 & r_2 X_1 \\ \Xi_3 & -r_2 X_1 \end{bmatrix} + \begin{bmatrix} -r_1 X_1 & r_2 X_1 \\ \Xi_3 & -r_2 X_1 \end{bmatrix}^T + \sum_{i=1}^{N} s_i U_i,$$

$$\Xi_3 = \begin{bmatrix} AX_{11} + \gamma X_{11} & 0 \\ 0 & \sum_{i=1}^N p_i F_i + \gamma X_{12} \end{bmatrix} + r_1 X_1,$$

$$\hat{\Psi}_1 = s_1 \hat{A}_1 X = s_1 \begin{bmatrix} 0 & 0 \\ -r_1 e^{\gamma s_1} p_1 \begin{bmatrix} 0 & H_1 \\ G_1 & 0 \end{bmatrix} & r_2 e^{\gamma s_1} \begin{bmatrix} 0 & H_1 \\ G_1 & 0 \end{bmatrix} \end{bmatrix},$$

$$\vdots$$

$$\hat{\Psi}_N = s_N \hat{A}_N X = s_N \begin{bmatrix} 0 & 0 \\ -r_1 e^{\gamma s_N} p_N \begin{bmatrix} 0 & H_N \\ G_N & 0 \end{bmatrix} & r_2 e^{\gamma s_N} \begin{bmatrix} 0 & H_N \\ G_N & 0 \end{bmatrix} \end{bmatrix},$$

holds, then the closed-loop system (4.26) is MES under the output-feedback controller of the form

$$A_{ci} = F_i X_{12}^{-1}, \quad B_{ci} = G_i X_{11}^{-1} C^+, \quad C_{ci} = B^+ H_i X_{12}^{-1}. \tag{4.30}$$

Proof: Define

$$X = \begin{bmatrix} X_1 & 0 \\ -r_1 X_1 & r_2 X_2 \end{bmatrix}, \quad X_1 = \begin{bmatrix} X_{11} & 0 \\ 0 & X_{12} \end{bmatrix}.$$

Pre- and post-multiply Θ in (4.13) by X^T and X, it results in (4.21). Let $U_i = Q_i^{-1}$, $F_i = A_{ci} X_{12}$, $G_i = B_{ci} C X_{11}$ and $H_i = BC_{ci} X_{12}$, where $i = 1, \ldots, N$. Applying Schur complement to (4.21) results in (4.29). ■

Remark 4.6 The structure of X in Theorem 4.6 is made due to the conditions $X^{-1} = P$ and $EP = P^T E$ in the Lyapunov candidate. Generally, the positive definite matrix X has the form given in (4.22). However, in order to avoid the nonlinear terms $\bar{A}_i X_2$ and $\bar{A}_i X_3$, the matrices X_2 and X_3 are replaced by $-r_1 X_1$ and $r_2 X_1$. Furthermore, X_1 is determined as a diagonal matrix, i.e.

$$X_1 = \begin{bmatrix} X_{11} & 0 \\ 0 & X_{12} \end{bmatrix},$$

so as to make the products of $\bar{A}_i X_1$ resulting in $F_i = A_{ci} X_{12}$, $G_i = B_{ci} C X_{11}$ and $H_i = BC_{ci} X_{12}$.

The LMI algorithm is recovered by structure restrictions of matrices X and X_1. This restriction, however, introduce certain conservatism in the output-feedback controller design. A less conservative design approach is to set matrices X and X_1 as described in (4.22). The resulting BMI can be easily solved by using the solution of Theorem 4.6 as an initial condition. This approach is demonstrated and discussed in the following numerical example.

Example 4.2 Consider an NCS with an output-feedback controller as described in (4.23). It is assumed the plant and controller are i.i.d. sampled by two sampling intervals 5 ms and 30 ms with corresponding probability distributions 60% and 40%, respectively. The NCS is connected by a network with constant delay 30 ms. Hence, the considered NCS is subjected to delay intervals $s_1 = 35\,\text{ms}$, $p_1 = 60\%$ and $s_2 = 60\,\text{ms}$, $p_2 = 40\%$. Set the parameters $\gamma = 0.4$ and the system parameters as

$$A = \begin{bmatrix} -2.1 & -0.1 \\ -0.2 & 0.3 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ -0.5 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 1 \end{bmatrix}.$$

Consider the positive definite matrix X as in (4.22) and consider the solutions of Theorem 4.6 as an initial condition for solving BMI. The stabilizing output-feedback controller are derived as

$$A_{c1} = \begin{bmatrix} -70.453 & -67.677 \\ -67.677 & -70.453 \end{bmatrix}, \quad B_{c1} = \begin{bmatrix} -15.914 \\ -15.914 \end{bmatrix}, \quad C_{c1} = \begin{bmatrix} -4.947 & -4.947 \end{bmatrix},$$
$$A_{c2} = \begin{bmatrix} -40.550 & -39.590 \\ -39.590 & -40.550 \end{bmatrix}, \quad B_{c2} = \begin{bmatrix} -6.025 \\ -6.025 \end{bmatrix}, \quad C_{c2} = \begin{bmatrix} -4.281 & -4.281 \end{bmatrix},$$

with

$$X_1 = \begin{bmatrix} 0.108 & 0.055 & -0.017 & -0.017 \\ 0.055 & 0.124 & -0.019 & -0.019 \\ -0.017 & -0.019 & 0.135 & -0.126 \\ -0.017 & -0.019 & -0.126 & 0.135 \end{bmatrix},$$

$$U_1 = \begin{bmatrix} 0.483 & 0.034 & -0.022 & -0.022 & -0.292 & -0.136 & 0.078 & 0.078 \\ 0.034 & 0.329 & -0.001 & -0.001 & -0.044 & -0.296 & -0.042 & -0.042 \\ -0.022 & -0.001 & 0.589 & 0.034 & 0.015 & 0.008 & -0.240 & -0.242 \\ -0.022 & -0.001 & 0.034 & 0.589 & 0.015 & 0.008 & -0.242 & -0.240 \\ -0.292 & -0.044 & 0.015 & 0.015 & 0.334 & 0.049 & 0.026 & 0.026 \\ -0.136 & -0.296 & 0.008 & 0.008 & 0.049 & 0.488 & -0.027 & -0.029 \\ 0.078 & -0.042 & -0.240 & -0.242 & 0.026 & -0.029 & 1.144 & 0.589 \\ 0.078 & -0.042 & -0.242 & -0.240 & 0.026 & -0.029 & 0.589 & 1.144 \end{bmatrix},$$

$$U_2 = \begin{bmatrix} 0.310 & -0.013 & -0.009 & -0.009 & -0.252 & 0.031 & 0.050 & 0.050 \\ -0.013 & 0.303 & -0.004 & -0.004 & -0.030 & -0.257 & -0.025 & -0.025 \\ -0.009 & -0.004 & 0.529 & -0.016 & 0.016 & -0.001 & -0.149 & -0.152 \\ -0.009 & -0.004 & -0.016 & 0.529 & 0.016 & -0.001 & -0.152 & -0.150 \\ -0.252 & -0.030 & 0.016 & 0.016 & 0.304 & -0.006 & 0.017 & 0.017 \\ 0.031 & -0.257 & -0.001 & -0.001 & -0.006 & 0.305 & -0.020 & -0.020 \\ 0.050 & -0.025 & -0.150 & -0.152 & 0.017 & -0.020 & 0.828 & 0.283 \\ 0.050 & -0.025 & -0.152 & -0.150 & 0.017 & -0.020 & 0.283 & 0.828 \end{bmatrix}.$$

With the initial condition $x^T(\theta) = [2 \; -2]$, $\theta \in [-\bar{\tau}, 0]$, the simulation is performed 500 times with different sample paths of delays for a time horizon of $T = 20\,\text{s}$. For NCS with delay-dependent switching output-feedback controller, its mean trajectory converges towards a ball around $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 8.081\,\text{s}$, superior to the mean trajectory of the NCS with worst-case design $t_{0.05} = 15.354\,\text{s}$ (-47.4%), as shown in Fig. 4.5 (a).

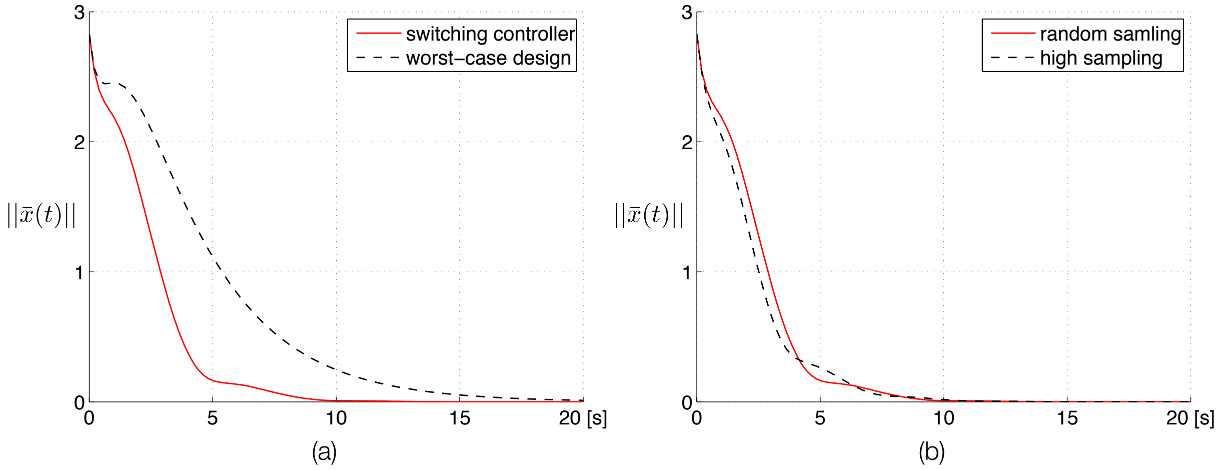


Figure 4.5: The mean state trajectory of NCS with delay-dependent switching controller (solid line) and NCS with worst-case design controller (dashed line) (a) and the mean state trajectory of NCS with random sampling (solid line) and NCS with high sampling (dashed line) (b).

In addition, a benchmark numerical experiment of NCS with constant sampling interval, i.e. 5 ms, is executed for comparison as shown in Fig. 4.5 (b). It is observed that the mean trajectory of NCS with high sampling converges within $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 7.677\,\mathrm{s}$, close to NCS with random sampling $t_{0.05} = 8.081\,\mathrm{s}$ (+5.26%). However, the data flow of NCS with random sampling rate is 66.7% less than the data flow of the NCS with high sampling.

The numerical results show that the delay-dependent switching controller design algorithm enables a good control performance compared to worst-case design. With the consideration of network capacity constraints, the NCS with random sampling efficiently reduces the data flow and meanwhile preserves the control performance. This gives rise to an interesting question like *how much data flow can be reduced without affecting performance?* In the flowing section, a performance design approach will be studied, which provides a performance upper bound for admissible random sampling intervals.

4.4 Guaranteed control performance for NCS with random sampling and delay

Due to the external traffic and limitations on network resources [22,128], a network can be more efficiently utilized if the sampling rate of control systems can be varied according to network conditions. However, improper sampling rate (or sampling intervals) of systems might result in performance degradation. In this section, a sampling distribution related performance index will be proposed. Based on the index, a admissible distribution of sampling intervals for an NCS can be determined, so that the control performance is guaranteed by certain sampling distributions.

For this purpose, define a cost function as

$$J_{per} = \mathbb{E}\Big\{ \int_0^T z^T(t) R z(t) dt \Big\}, \tag{4.31}$$

where R is a symmetric, positive definite matrix. Associated to the control function (4.31), the guaranteed control performance defined in Definition 3.1 is recalled.

4.4.1 Guaranteed cost state-feedback controller

Consider the delay-dependent state-feedback controller

$$\bar{u}(t) = \sum_{i=1}^{N} \beta_i B K_i x(t - s_i), \quad i = 1, \dots, N,$$

where s_i satisfies (4.4). The objective in this section is to design a set of K_i so that the resulting closed-loop system in (4.8) is MES and the cost function in (4.31) is bounded by some specified scalar.

Theorem 4.5 For given positive scalars $r_1 > 0$, $r_2 > 0$, $\gamma \geq 0$ and matrix $R > 0$, if there exist symmetric matrices $U_i > 0$, $i = 1, \dots, N+1$, and $X_1 = X_1^T > 0$ satisfying

$$X = \begin{bmatrix} X_1 & 0 \\ -r_1 X_1 & r_2 X_1 \end{bmatrix},$$

such that

$$\begin{bmatrix} \hat{\Psi} & \hat{\Psi}_1^T & \cdots & \hat{\Psi}_N^T & \hat{\Psi}_{N+1}^T \\ * & -s_1 U_1 & 0 & \cdots & 0 \\ \vdots & * & \ddots & 0 & \vdots \\ * & \cdots & * & -s_N U_N & 0 \\ * & * & \cdots & * & -U_{N+1} \end{bmatrix} < 0, \tag{4.32}$$

where

$$\hat{\Psi} = \begin{bmatrix} -r_1 X_1 & r_2 X_1 \\ \Xi_3 & -r_2 X_1 \end{bmatrix} + \begin{bmatrix} -r_1 X_1 & r_2 X_1 \\ \Xi_3 & -r_2 X_1 \end{bmatrix}^T + \sum_{i=1}^{N} s_i U_i,$$

$$\Xi_3 = A X_1 + \gamma X_1 + \sum_{i=1}^{N} e^{\gamma s_i} p_i B Y_i + r_1 X_1,$$

$$\hat{\Psi}_1 = s_1 \hat{A}_1 X = s_1 \begin{bmatrix} 0 & 0 \\ -r_1 e^{\gamma s_1} p_1 B Y_1 & r_2 e^{\gamma s_1} p_1 B Y_1 \end{bmatrix},$$

$$\vdots$$

$$\hat{\Psi}_N = s_N \hat{A}_N X = s_N \begin{bmatrix} 0 & 0 \\ -r_1 e^{\gamma s_N} p_N B Y_N & r_2 e^{\gamma s_N} p_N B Y_N \end{bmatrix},$$

$$\hat{\Psi}_{N+1} = \begin{bmatrix} X_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad U_{N+1} = \begin{bmatrix} R^{-1} & 0 \\ 0 & R^{-1} \end{bmatrix}$$

holds, then the closed-loop system (4.8) is MES with the feedback gain

$$K_i = Y_i X_1^{-1}, \quad i = 1, \ldots, N \tag{4.33}$$

and the cost function in (4.31) is bounded by

$$J_{per} \leq \bar{J}_{per}(p_1, \ldots, p_N) = \xi^T(0) E P \xi(0) + \sum_{i=1}^{N} \int_{-s_i}^{0} \int_{\theta}^{0} \xi^T(s) \hat{A}_i^T Q_i \hat{A}_i \xi(s)\, ds\, d\theta. \tag{4.34}$$

Proof: Consider the same Lyapunov candidate in Theorem 4.1, the closed-loop system is MES if the following inequality

$$\begin{aligned}\mathcal{L}V(\xi(t)) &\leq \xi^T(t)[\hat{A}^T P + P^T \hat{A} + \sum_{i=1}^{N} s_i \hat{A}_i^T Q_i \hat{A}_i + \sum_{i=1}^{N} s_i P^T Q_i^{-1} P]\xi(t) \\ &= \xi^T(t) \Theta \xi(t) < 0\end{aligned}$$

is satisfied. Define $z(t) = [I\ 0]\xi(t)$. According to Dynkin's formula, the cost function in (4.31) becomes

$$\begin{aligned}J_{per}(r_t) &= \mathbb{E}\Big\{ \int_0^T \xi^T(t) \begin{bmatrix} I \\ 0 \end{bmatrix} R[I\ 0]\xi(t) + \mathcal{L}V(\xi(t))\big] dt \Big\} - \mathbb{E}\{V(\xi(T))\} + \mathbb{E}\{V(\xi(0))\} \\ &\leq \mathbb{E}\Big\{ \int_0^T \xi^T(t) \bar{\Theta} \xi(t) dt + V(\xi(0)) \Big\},\end{aligned}$$

where $\bar{\Theta} = \Theta + \begin{bmatrix} I \\ 0 \end{bmatrix} R[I\ 0]$. It is clear that if $\bar{\Theta} < 0$, the cost function (4.31) is bounded by

$$J_{per} = \mathbb{E}\Big\{ \int_0^\infty z^T(t) R z(t) dt \Big\} \leq \mathbb{E}\{V(\xi(0))\} = \bar{J}_{per}(s_1, \ldots, s_N).$$

Pre- and post-multiply $\bar{\Theta}$ by X^T and X and let $U_i = Q_i^{-1}$, $Y_i = K_i X$, $i = 1, \ldots, N$ and $U_{N+1} = \text{diag}\{R^{-1}, R^{-1}\}$. Applying Schur complement, it results in (4.32). ■

Remark 4.7 Based on Theorem 4.5, the upper bound of the cost index obtained in above theorem depends on the initial condition $\xi(0)$. Consider $P = X^{-1}$ and $Q_i = U_i^{-1}$, $i = 1, \ldots, N$. The guaranteed cost bound can be optimized by solving the linear optimization problem

$$\begin{aligned}&\min_{X_1 > 0, U_i > 0} \ \xi^T(0) E P \xi(0) + \sum_{i=1}^{N} \int_{-s_i}^{0} \int_{\theta}^{0} \xi^T(s) \hat{A}_i^T Q_i \hat{A}_i \xi(s)\, ds\, d\theta. \\ &\text{s.t. } (4.32)\end{aligned} \tag{4.35}$$

As mentioned in Theorem 4.2, the conservatism in controller design is introduced by the special structure requirement of X. The conservatism can be reduced by considering X in original form, see (4.22), and using the BMI solver for controller design. The BMI design approach can be more efficiently solved by using the solutions of Theorem 4.5 as initial conditions. The illustration and discussion of the derived theorem are given in the following example.

Example 4.3 Consider the NCS in Example 4.1 with the same parameter settings and initial conditions. Set the weighting matrix R in (4.31) as

$$R = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}.$$

Solving Theorem4.5, the stabilizing feedback gains are

$$K_1 = [-12.572 \ \ -21.933], \quad K_2 = [-6.368 \ \ -11.218], \quad K_3 = [-2.903 \ \ -5.108],$$

with

$$X_1 = \begin{bmatrix} 0.738 & -0.010 \\ -0.010 & 0.050 \end{bmatrix},$$

$$U_1 = 10^3 \times \begin{bmatrix} 1.191 & -0.602 & -1.195 & 0.583 \\ -0.602 & 0.413 & 0.608 & -0.386 \\ -1.195 & 0.608 & 1.210 & -0.586 \\ 0.583 & -0.386 & -0.586 & 0.383 \end{bmatrix},$$

$$U_2 = \begin{bmatrix} 171.554 & -54.703 & -164.390 & 69.083 \\ -54.703 & 50.094 & 55.655 & -42.568 \\ -164.390 & 55.655 & 164.471 & -67.208 \\ 69.083 & -42.568 & -67.208 & 52.106 \end{bmatrix},$$

$$U_3 = \begin{bmatrix} 375.054 & -187.750 & -367.166 & 203.450 \\ -187.750 & 111.353 & 187.803 & -107.545 \\ -367.166 & 187.803 & 364.810 & -201.510 \\ 203.450 & -107.545 & -201.510 & 116.336 \end{bmatrix}$$

at $J_{per} = 7.067 \times 10^4$.

As shown in (4.34), the performance bound is a function of probability distribution of sampling intervals (or delay intervals). The feasible probability distributions of sampling intervals and associated cost index are shown in Table 4.1.

Table 4.1: The feasible probability distributions of sampling intervals and associated cost indices.

p_1, p_2, p_3	70%, 20%, 10%	60%, 30%, 10%	40%, 30%, 30%	20%, 30%, 50%
J_{per}	6.552×10^4	7.067×10^4	9.378×10^4	2.157×10^5

According to Table 4.1, it is obviously that higher probability of fast sampling has less bounds on its performance. However, it requires the provision of larger network bandwidth. Considering the performance bounds and data flow in one cost index, a novel design approach can be developed for NCSs by minimizing the cost index. In Chapter 5, the data flow will be represented in forms of normalized network cost. By applying Theorem 4.5, an NCS can be designed so that its performance is guaranteed by most economic network resource consumption.

4.4.2 Guaranteed cost output-feedback controller

Consider the delay-dependent output-feedback controller

$$\dot{x}_c(t) = \sum_{i=1}^{N} \beta_i \Big(A_{ci}x_c(t) + B_{ci}Cx(t-s_i) \Big),$$
$$\bar{u}(t) = \sum_{i=1}^{N} \beta_i C_{ci} x_c(t-s_i),$$

where s_i satisfies (4.4) and $i = 1, \ldots, N$. Consider the cost function as

$$J_{per} = \mathbb{E}\bigg\{ \int_0^T \chi^T(t)R\chi(t)dt \bigg\}, \tag{4.36}$$

where $R > 0$ and Definition 3.1. It is aimed to find a set of A_{ci}, B_{ci}, C_{ci} such that the closed-loop system in (4.26) is MES and the cost function (4.36) is bounded by a certain value.

Theorem 4.6 For given positive scalars $r_1 > 0$, $r_2 > 0$, $\gamma \geq 0$ and matrix $R > 0$, if there exist symmetric matrices $U_i > 0$, $i = 1, \ldots, N+1$, and $X_1 = X_1^T > 0$ satisfying

$$X_1 = \begin{bmatrix} X_{11} & 0 \\ 0 & X_{12} \end{bmatrix},$$

$$X = \begin{bmatrix} X_1 & 0 \\ -r_1X_1 & r_2X_1 \end{bmatrix},$$

such that

$$\begin{bmatrix} \hat{\Psi} & \hat{\Psi}_1^T & \cdots & \hat{\Psi}_N^T & \hat{\Psi}_{N+1}^T \\ * & -s_1U_1 & 0 & \cdots & 0 \\ \vdots & * & \ddots & * & \vdots \\ * & \cdots & * & -s_NU_N & 0 \\ * & * & \cdots & * & -U_{N+1} \end{bmatrix} < 0, \tag{4.37}$$

where

$$\hat{\Psi} = \begin{bmatrix} -r_1X_1 & r_2X_1 \\ \Xi_3 & -r_2X_1 \end{bmatrix} + \begin{bmatrix} -r_1X_1 & r_2X_1 \\ \Xi_3 & -r_2X_1 \end{bmatrix}^T + \sum_{i=1}^{N} s_iU_i,$$

$$\Xi_3 = \begin{bmatrix} AX_{11} + \gamma X_{11} & 0 \\ 0 & \sum_{i=1}^{N} p_iF_i + \gamma X_{12} \end{bmatrix} + r_1X_1,$$

$$\hat{\Psi}_1 = s_1\hat{A}_1X = s_1 \begin{bmatrix} 0 & 0 \\ -r_1e^{\gamma s_1}p_1 \begin{bmatrix} 0 & H_1 \\ G_1 & 0 \end{bmatrix} & r_2e^{\gamma s_1} \begin{bmatrix} 0 & H_1 \\ G_1 & 0 \end{bmatrix} \end{bmatrix},$$

$$\vdots$$

$$\hat{\Psi}_N = s_N\hat{A}_NX = s_N \begin{bmatrix} 0 & 0 \\ -r_1e^{\gamma s_N}p_N \begin{bmatrix} 0 & H_N \\ G_N & 0 \end{bmatrix} & r_2e^{\gamma s_N} \begin{bmatrix} 0 & H_N \\ G_N & 0 \end{bmatrix} \end{bmatrix},$$

$$\hat{\Psi}_{N+1} = \begin{bmatrix} X_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad U_{N+1} = \begin{bmatrix} R^{-1} & 0 \\ 0 & R^{-1} \end{bmatrix}$$

holds, then the closed-loop system (4.26) is MES under the output-feedback controller of the form

$$A_{ci} = F_i X_{12}^{-1}, \quad B_{ci} = G_i X_{11}^{-1} C^{+}, \quad C_{ci} = B^{+} H_i X_{12}^{-1}. \tag{4.38}$$

and the cost function in (4.36) is bounded by

$$J_{per} \leq \bar{J}_{per}(p_1, \dots, p_N) = \xi^T(0) E P \xi(0) + \sum_{i=1}^{N} \int_{-s_i}^{0} \int_{\theta}^{0} \xi^T(s) \hat{A}_i^T Q_i \hat{A}_i \xi(s)\, ds\, d\theta. \tag{4.39}$$

Proof: According to the proof the Theorem 4.5, it is shown known that the closed-loop system is MES and the cost function in (4.36) is bonded if

$$\bar{\Theta} = \Theta + \begin{bmatrix} I \\ 0 \end{bmatrix} R \begin{bmatrix} I & 0 \end{bmatrix} < 0.$$

Define

$$X = \begin{bmatrix} X_1 & 0 \\ -r_1 X_1 & r_2 X_2 \end{bmatrix}, \quad X_1 = \begin{bmatrix} X_{11} & 0 \\ 0 & X_{12} \end{bmatrix}.$$

Pre- and post-multiply $\bar{\Theta}$ by X^T and X and let $U_i = Q_i^{-1}$,

$$F_i = A_{ci} X_{12}, \quad G_i = B_{ci} C X_{11}, \quad H_i = B C_{ci} X_{12},$$

where $i = 1, \dots, N$ and $U_{N+1} = \text{diag}\{R^{-1}, R^{-1}\}$. Applying Schur complement, it results in (4.37). ■

Remark 4.8 The upper bound of the cost function (4.36) depends on the initial condition $\xi(0)$. Note the fact $P = X^{-1}$ and $Q_i = U_i^{-1}$, $i = 1, \dots, N$. The optimal guaranteed cost bound can be derived by solving the linear optimization problem

$$\begin{aligned} &\min_{X_1 > 0, U_i > 0} \quad \xi^T(0) E P \xi(0) + \sum_{i=1}^{N} \int_{-s_i}^{0} \int_{\theta}^{0} \xi^T(s) \hat{A}_i^T Q_i \hat{A}_i \xi(s)\, ds\, d\theta. \\ &\text{s.t. } (4.37) \end{aligned} \tag{4.40}$$

The structure restrictions of matrices X and X1 results in LMI algorithm (4.37) and introduces conservatism in controller design. A less conservative design approach is to set matrices X and X1 as described in (4.22). The resulting BMI can be easily solved by using the solution of Theorem 4.6 as an initial condition. This approach is demonstrated and discussed in the following numerical example.

Example 4.4 Consider the NCS in Example 4.2 with the same sampling, delay and initial conditions. Set the parameter $\gamma = 0$ and $N = 2$. Choose $s_1 = 35\,\text{ms}$, $p_1 = 60\%$, $s_2 = 60\,\text{ms}$, $p_2 = 40\%$ and

$$R = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}.$$

Solving Theorem 4.6, the stabilizing output-feedback controller are

$$A_{c1} = \begin{bmatrix} -31.741 & -30.628 \\ -30.628 & -45.777 \end{bmatrix}, \quad B_{c1} = \begin{bmatrix} -7.112 \\ -6.308 \end{bmatrix}, \quad C_{c1} = \begin{bmatrix} -4.356 & -8.750 \end{bmatrix},$$
$$A_{c2} = \begin{bmatrix} -13.294 & -12.286 \\ -12.286 & -18.010 \end{bmatrix}, \quad B_{c2} = \begin{bmatrix} -6.042 \\ -7.956 \end{bmatrix}, \quad C_{c2} = \begin{bmatrix} -4.065 & -5.456 \end{bmatrix},$$

with

$$X_1 = \begin{bmatrix} 0.015 & 0.002 & -0.006 & 0.001 \\ 0.002 & 0.015 & -0.004 & -0.001 \\ -0.006 & -0.004 & 0.008 & -0.004 \\ 0.001 & -0.001 & -0.004 & 0.004 \end{bmatrix},$$

$$U_1 = \begin{bmatrix} 0.248 & -0.077 & -0.018 & -0.004 & -0.167 & 0.092 & -0.047 & 0.065 \\ -0.077 & 0.276 & 0.021 & -0.011 & -0.001 & -0.236 & 0.006 & -0.018 \\ -0.018 & 0.021 & 0.589 & -0.205 & -0.047 & -0.030 & -0.204 & 0.081 \\ -0.004 & -0.011 & -0.205 & 0.535 & 0.094 & -0.021 & 0.047 & -0.289 \\ -0.167 & -0.001 & -0.047 & 0.094 & 0.244 & -0.059 & -0.019 & 0.006 \\ 0.092 & -0.236 & -0.030 & -0.021 & -0.059 & 0.288 & 0.004 & -0.001 \\ -0.047 & 0.006 & -0.204 & 0.047 & -0.019 & 0.004 & 0.515 & -0.206 \\ 0.065 & -0.018 & 0.081 & -0.289 & 0.006 & -0.001 & -0.206 & 0.499 \end{bmatrix},$$

$$U_2 = \begin{bmatrix} 0.148 & -0.042 & -0.011 & 0.008 & -0.097 & 0.047 & -0.026 & 0.028 \\ -0.042 & 0.161 & 0.017 & -0.015 & -0.001 & -0.139 & 0.001 & 0 \\ -0.011 & 0.017 & 0.422 & -0.094 & -0.026 & -0.024 & -0.195 & 0.029 \\ 0.008 & -0.015 & -0.094 & 0.224 & 0.047 & -0.003 & 0.006 & -0.076 \\ -0.097 & -0.001 & -0.026 & 0.047 & 0.141 & -0.033 & -0.013 & 0.010 \\ 0.047 & -0.139 & -0.024 & -0.003 & -0.033 & 0.173 & 0.006 & -0.012 \\ -0.026 & 0.001 & -0.195 & 0.006 & -0.013 & 0.006 & 0.374 & -0.106 \\ 0.028 & 0 & 0.029 & -0.076 & 0.010 & -0.012 & -0.106 & 0.193 \end{bmatrix}$$

at $J_{per} = 2.307 \times 10^3$.

According to the setting in Example 4.2, the ratio of high to low sampling has the factor of six. Heuristically, it can be assumed that the low sampling rate has data flow of $C_2 = 0.6 \times 10^3$, whereas the high sampling rate has $C_1 = 3.6 \times 10^3$. In oder to highlight the trade-off between performance and data flow, the required data flow

$$J_{net} = p_1 C1 + p_2 C2$$

and performance bounds are established in Table 4.2

Table 4.2: The feasible probability distributions of sampling intervals, associated performance bounds and data flow indices.

p_1, p_2	80%, 20%	60%, 40%	40%, 60%	20%, 80%
J_{per}	1.747×10^4	2.307×10^4	3.598×10^4	9.367×10^5
J_{net}	3×10^3	2.4×10^3	1.8×10^3	1.2×10^3
$J_{per} + J_{net}$	4.747×10^4	4.707×10^4	5.398×10^4	9.379×10^5

From Table 4.2, it is obvious that higher probability of sampling is not suitable if the network capacity is taken into account. As shown in above examples, the NCS design algorithms proposed in this chapter delicately formulate guaranteed performance bounds in terms of probability distribution of sampling intervals. This enables a conjoint conjoint consideration of control performance and network usage in the NCS design. In Chapter 5, a novel cost function will be developed, which facilitates a performance and network usage trade-off and results in a *control system and communication network co-design approach*.

4.5 Summary and discussion

Nowadays, control systems are getting more complex and intelligent due to a large number deployment of sensors and actuators. To manage these sensors and actuators in an efficient way, communication networks, such as ControlNet, DeriveNet, Ethernet, Profibus etc., are increasingly being used in control systems. Facing the increasing probability of sensor data congestion in a control network, it is desirable to develop an NCS, whose data sampling rate can be varied according to network traffic.

In this chapter, NCSs with random sampling are systematically investigated. Main focus is to derive network usage relevant methods for NCS analysis and synthesis. The analysis approach is different from Chapter 3, where the periodic sampling intervals are embedded into random delays. In this chapter, on the contrary, the delays are combined into random sampling intervals and reformulated into a time-varying delay by using the input-delay approach. The time-varying delay is classified into N number of intervals, whose probabilistic occurrence is described by associated indicator functions. In order to provide a framework for NCS analysis and synthesis relevant to the network usage, the obtained stability and stabilization conditions are presented in terms of probabilistic distributions of aperiodic sampling intervals. Based on this framework, an innovative performance guaranteed algorithm is developed to correlate the performance bound with probability distributions of sampling intervals. For network applications, this allows an arbitration between control performance and sensor data flow. For the underlying network, the algorithm enables controlling traffic entry into the network, so as to avoid data congestion and guarantees communication qualities.

The proposed approaches are numerically validated by different NCSs. It is shown that the prosed approaches can efficiently stabilized unstable plants with probabilistic sampling. The simulation results demonstrate superior performance over traditional worst-case design. Furthermore, the proposed design algorithm achieves acceptable performance at considerably less data loads.

Compared to the results derived in Chapter 3, the Markovian requirements in the delay modeling are successfully removed in Chapter 4. The stability and stabilization results in chapter contains only the static probability distributions of probabilistic samplings, whereas the results in Chapter 3 enclose the information of statistical network dynamics, i.e. Markov probability transition rates. In views of design functionality, the results of Chapter 4 provide the basis for optimal network scheduling and Chapter 3 is aimed at QoS network control. The details will be given in the following chapter.

5 Control Systems and Communication Networks Co-Design

It is well-know that the stability and performance of an NCS strongly depend on the communication quality, e.g. the transmission delay. Guaranteed short transmission delay leads to good control performance but needs the provision of large network resources, such as bandwidth. Considering the finite resources of a network, it is desirable that every system consumes only as much network resources as required to guarantee a certain level of performance, i.e. a *cost-performance trade-off* is required. Furthermore, inefficient utilization of network resources might result in higher possibility of longer delay and cause control performance degradation. For example, the data transmission rate (sampling rate) of a system gets higher, the network traffic becomes heavier. The possibility of longer transmission delay or more packet dropouts increases. Hence, it is beneficial to design a data transmission scheduling for a system so that the network resources is more efficiently used; meanwhile, the control performance of connected systems is ensured, i.e. a *network usage-performance trade-off* is required. Motivated by these requirements, two novel approaches dealing with networks and systems co-design are proposed in this chapter.

In the first co-design approach, the Quality-of-Service (QoS) concept from the networking community is considered. QoS refers to the capability of a network to provide different levels of communication quality to different applications. Within the first approach, performance requirements of systems and restrictions of networks are linked through statistical properties of the Markovian transmission delays. QoS is then related to the ability of adjusting the probability transition rate of such Markov process. A cost-performance trade-off is achieved by appropriately parameterizing the Markov probability transition rate. This approach can be implemented for multihop wireless LAN IEEE 802.15.4 with the MAC protocol CSMA/CA (Carrier Sensing Multiple Access/Collision Avoidance), where QoS is considered to re-specify the probability distribution of the waiting delay upon the priority of packets. Re-specifying the probability distribution for the waiting delay in real network implementation can be realized by choosing the backoff exponent and backoff period in CSMA/CA algorithm. As a result, a communication network and control system co-design problem reduces to the parameterizing of Markov process transition generator.

In the second co-design approach, the sampling interval is considered as random with certain probability distributions. A cost function incorporating the control performance and network usage is formulated by means of the probability distributions of associated sampling intervals. In oder to balance the performance and network usage, the probability distribution of sampling intervals is optimized by minimizing the cost function under performance constraint, so as good performance is achieved at economic network consumption. Different from the existing MATI-based data scheduling, which precisely regulate sensor data by an maximal allowable interval, the second approach manages the traffic entry in percentage. This eases the implementation and increases the performance as shown in this chapter.

Compared to the existing literature, the major innovation of the proposed two co-design approaches are three fold. First, the proposed approaches take stochastic control and random delays into account. This stochastic consideration results in less conservative controller design than related co-design approaches in [58, 79, 136, 141], where the results are derived by worst-case assumptions. Second, the proposed approaches integrate the performance trade-off from control and communication. They enable the development of more efficient and affordable NCSs which can scale and adapt with available network resources. Third, unlike [58], where deterministic piecewise constant transmission delays are needed. The application of the proposed approaches requires no exact values of delays or sampling intervals, but their distributions. This simplifies the implementation in real networks.

The remainder of this chapter is organized as follows. First, the quality-of-service (QoS) network is briefly presented in the next section. Based on the QoS concept, a networks and systems co-design approach incorporating the cost-performance trade-off is introduced in Section 5.2. In views of efficient network utilization, a networks and systems co-design approach concerning optimal sampling interval distributions is given in Section 5.3. Discussions and conclusions are summarized in Section 5.4.

5.1 Quality-of-Service network

"Quality-of-Service" (QoS) is a popular and overlapped term which is taken from different perspectives by the networking and application-development communities. In the networking, QoS refers to the capability of a network to provide different treatment to different classes of network traffic. The primary goal of QoS mechanism is to increase the overall utility of the network by granting priorities to higher-value or more performance-sensitive data flows. The priority indicates for example, the desired bit rate, delay and packet dropping probability of data flows. In the networking, it should be noted that QoS does not prevent network congestion. However, QoS adds an intelligent transmission interface allowing the network to make informed decisions about how to transmit the data flows.

In the application-development community, the view of QoS focuses on the relationship between application performance and network utility. It is known that real-time network applications are sensitive to the network traffic. Higher application performance requires more network resources (e.g. larger bandwidth and shorter transmission delay) and results in higher network cost. Due to the limited network resources, it is desirable to allocate the network resources to each application to achieve certain level of performance. QoS is considered as a key to meet the application performance and network utility in a cost-effective manner.

There are two popular networking architectures developed to engineer preferential treatment of applications within QoS networks. In the small-scale networks, the IntServ (or integrated services) architecture is used to reserve network resources. In this architecture, network applications use the Resource reservation protocol (RSVP) to request and reserve resources through a network. For large-scale IP networks, the DiffServ (or Differentiated Services) architecture marks the network applications according to the type of service they need. In response to these markings, routers and switches use various queueing strategies to tailor performance to requirements.

Generally, QoS of a communication network is affected by four factors; *delay*, *packet dropout*, *packet error* and *jitter*. The jitter concerns the variation of delays. The packet error happens when packets are misdirected or corrupted during *en route* and can be treated as packet dropout. As mentioned in Chapter 3, the effect of packet dropouts is considered as a fictitious delay. In this dissertation, QoS of a communication network is characterized by the features of random transmission delays. The typical exampled of IEEE 802.15.4 wireless LAN will be depicted in the following section.

5.2 Networks and control systems co-design: a cost-performance trade-off

Consider a control system connected by a multi-hop wireless LAN IEEE 802.15.4 with the MAC protocol CSMA/CA (Carrier Sensing Multiple Access/Collision Avoidance). The CSMA/CA algorithm for IEEE 802.15.4 uses the randomly generated waiting delay[1] for the collision avoidance. For applications with higher priority, the probability of shorter waiting delay is increased. This results in better control performance but leads to higher network cost. According to [50,66], the random waiting delay in CSMA/CA network can be modeled by a Markov process. Based on Markov properties, re-specifying the probability distribution of a Markovian (waiting) delay means adjusting its associated probability transition rate. Consider the probability distribution of CSMA/CA waiting delay as a QoS parameter. An *optimal cost-performance trade-off* can be achieved by optimizing the Markov probability transition rate. In real network implementation, this is realized by choosing the back-off exponent and back-off period in CSMA/CA algorithm.

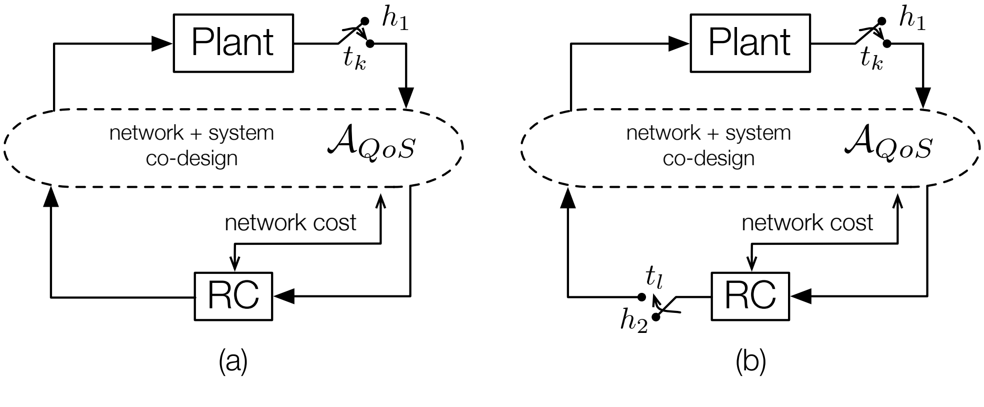


Figure 5.1: Scheme of NCS with QoS network and state-feedback controller (a) and output-feedback controller (b).

The control structure is illustrated in Fig. 5.1. The sensor is periodically sampled by h_1 (output-feedback controller is periodically sampled by h_2). The random SC and CA delays are modeled by Markovian delays. The resulting system is an MJLS with mode-dependent delay. The results from Chapter 3, i.e. guaranteed control performance for NCS with random delay, are considered. The performance requirements of systems and restrictions of networks are linked through statistical properties of the Markovian transmission delays.

[1] The random waiting delay is represented by the sensor-to-controller delay $\tau_{\mathrm{sc}}(r_t)$ in Chapter 3.

As a result, the network and system co-design problem reduces to the parameterizing of Markov probability transition rate. The remote controller (RC) is designed with predefined the communication quality, i.e. the probability distribution of transmission delays, such that an optimal the cost-performance trade-off is achieved as shown in the following.

5.2.1 Optimal cost-performance trade-off

The conjoint design of QoS network and control system is to optimize the resources allocation such that the control performance versus network cost is balanced. For this purpose, the network cost $C(r_t)$, which represents the cost of the reservation of network resources and is associated with the delay $\tau(r_t)$, can be introduced into the cost function in (3.52). As a result, a new cost function incorporating the trade-off between network resources and control performance can be formulated in the following

$$J(r_t) = \lim_{T\to\infty} \mathbb{E}\left\{ \int_0^T z^T(t)R(r_t)z(t)dt + \frac{1}{T}\int_0^T C(r_t)dt \right\}. \tag{5.1}$$

The first term in (5.1) refers to the control performance and the second term refers to the network cost associated with the communication quality, i.e. transmission delay.

The control performance is shown to be bounded by (3.53) in Theorem 3.5. With known initial distribution of Markov process r_0, the expected values of control performance is guaranteed by

$$\lim_{T\to\infty} \mathbb{E}\left\{ \int_0^T x^T(t)R(r_t)x(t)dt \right\} \leq \bar{J}_{per}(r_0, \bar{\alpha}_{r_0} + \Delta\alpha_{r_0}) = \sum_{i=1} \mathbf{P}_j(0)V(\Xi(0), r_0 = i), \tag{5.2}$$

where $\mathbf{P}_i(0) = \mathbf{P}\{r_0 = i\}$.

The expected values of the normalized network cost over the runtime T are considered in the second term of (5.1). According to the Markov properties, each ergodic and irreducible Markov process has a stationary probability distribution $\bar{\mathbb{P}} = [\bar{\mathbf{P}}_1 \dots \bar{\mathbf{P}}_N]$, see [103]. With the known transition generator $\mathcal{A} = (\alpha_{i,j})$, the stationary probability distribution is determined as

$$\sum_i^N \bar{\mathbf{P}}_i \alpha_{i,j} = 0, \quad \sum_i^N \bar{\mathbf{P}}_i = 1.$$

Therefore, the expected values of the normalized network cost in (5.1) can be determined by

$$\lim_{T\to\infty} \mathbb{E}\left\{ \frac{1}{T}\int_0^T C(r_t)dt \right\} = \sum_i \bar{\mathbf{P}}_i C_i \tag{5.3}$$

Combining (5.2) and (5.3) into (5.1), the stochastic cost function (5.1) recovers to a deterministic function

$$J(\mathcal{A}) = \bar{J}_{per}(r_0, \bar{\alpha}_{r_0} + \Delta\alpha_{r_0}) + \sum_i \bar{\mathbf{P}}_i C_i \tag{5.4}$$

depending on the Markov process transition generator. Hence, the performance-cost trade-off results in minimizing (5.4) by appropriate choice of the transition generator within the stability constraints, i.e. the perturbation upper bound on the transition generator determined in Theorem 2. The details of the optimization problem is formulated as follows.

Proposition 5.1 Let $\Delta\bar{\alpha}_i$ be the perturbation upper bound of Markov process transition generator determined by (3.53) Theorem 3.5. An optimal cost-performance is given by

$$\min_{\mathcal{A}_{\text{QoS}} \in \mathbb{A}} J(\mathcal{A}_{\text{QoS}}), \tag{5.5}$$

where $\mathbb{A} = \{\nu_{i,j}, i, j \in \mathcal{S}\}$ is the set of admissible transition generators satisfying $\nu_i = -\nu_{i,i} = \sum_{j \neq i}^{N} \nu_{i,j}$ and

$$\alpha_i - \Delta\bar{\alpha}_i \leq \nu_i \leq \alpha_i + \Delta\bar{\alpha}_i.$$

For NCSs with output-feedback controllers, a similar optimization problem can be determined. Assume the NCS in Fig. 5.1 having an output-feedback controller. The cost function in (5.1) becomes

$$J(r_t) = \lim_{T \to \infty} \mathbb{E}\left\{ \int_0^T \chi^T(t) R(r_t) \chi(t) dt + \frac{1}{T} \int_0^T C(r_t) dt \right\}. \tag{5.6}$$

The first term in (5.6) is bounded by (3.61) in Theorem 4.6 and the second term is determined by (5.3). As a result, the optimization problem is summarized in Proposition 5.2.

Proposition 5.2 Let $\Delta\bar{\alpha}_i$ be the perturbation upper bound of Markov process transition generator determined by (3.59) in Theorem 3.6. An optimal cost-performance is given by

$$\min_{\mathcal{A}_{\text{QoS}} \in \mathbb{A}} J(\mathcal{A}_{\text{QoS}}), \tag{5.7}$$

where $\mathbb{A} = \{\nu_{i,j}, i, j \in \mathcal{S}\}$ is the set of admissible transition generators satisfying $\nu_i = -\nu_{i,i} = \sum_{j \neq i}^{N} \nu_{i,j}$ and

$$\alpha_i - \Delta\bar{\alpha}_i \leq \nu_i \leq \alpha_i + \Delta\bar{\alpha}_i.$$

Proposition 5.1 and Proposition 5.2 are static optimization problem with linear inequality and equality constraints. A local minimum can be easily found by any commercial optimization algorithm, e.g. `fmincon` in Matlab. The QoS co-design approach is based on the analysis and synthesis methods proposed in Chapter 3, and aims at linear[2] time-invariant NCSs with constant sampling rate. The transmission delays are mainly caused by the waiting delays used for collision avoidance in CSMA/CA networks. Furthermore, the design approach can be easily extended for a CSMA/CA network with multiple control systems by considering these control systems as subsystems of an NCS.

The benefit of QoS control is studied in the following example. A comparison between the NCS with QoS network co-design and NCS without QoS network co-design is performed with respect to control performance and network cost.

[2] Nonlinear NCSs can be linearized by standard linearization methods.

5.2.2 Case study: NCS with QoS network

Case A: NCS with State-feedback controller

Consider an QoS communication network having Markovian transmission delays, i.e. $\tau(1) = 20\,\text{ms}$ and $\tau(2) = 50\,\text{ms}$. The associated network cost is given by $C(1) = 4 \times 10^{-3}$ and $C(2) = 2 \times 10^{-3}$, meaning higher cost for shorter transmission delay. The probability transition rate, i.e. $\mathcal{A} = (\alpha_{i,j})$, $i, j \in \mathcal{S} = \{1, 2\}$, is considered as a QoS parameter of network and conjointly designed with the controller under cost-performance constraint.

Reconsider the LTI plant in Example 3.5 as

$$\dot{x}(t) = \begin{bmatrix} -1 & 1 \\ 0 & 0.5 \end{bmatrix} x(t) + \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} u(t) \tag{5.8}$$

with feedback gains

$$K(1) = [-6.3 \ \ -5.5], \quad K(2) = [-2.6 \ \ -2.2].$$

Set $\gamma = 0$ and the probability transition rate as

$$\mathcal{A} = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}.$$

The associated stationary probability for $\tau(1)$ is $\bar{\mathbf{P}}_1 = 50\%$ and $\bar{\mathbf{P}}_2 = 50\%$ for $\tau(2)$. Solving the (3.54) in Theorem 3.5, the upper bounds on the perturbation of Markov process transition generator, $\Delta\bar{\alpha}_1$ and $\Delta\bar{\alpha}_2$, are determined as (other associated positive definite matrices are given in Example 3.5)

$$\Delta\bar{\alpha}_1 = 0.207, \quad \Delta\bar{\alpha}_2 = 0.559.$$

The optimization problem in Proposition 5.1 is numerically solved by the `fmincon` algorithm from the Matlab optimization toolbox. With the initial condition $x^T(t_0 + \theta) = [1 \ 2]$, $\theta \in [-\bar{\tau}, 0]$ and the initial probability distribution of Markov process $\mathbb{P}(t_0) = [80\% \ 20\%]$, the cost-performance trade-off in (5.7) is minimized by the probability transition rate

$$\mathcal{A}^*_{\text{QoS}} = \begin{bmatrix} -1.207 & 1.207 \\ 0.441 & -0.441 \end{bmatrix}$$

with $\bar{J}(r_0) = 4.016 \times 10^{-3}$ and associated limiting probabilities are $\bar{\mathbb{P}} = [26.8\% \ 73.2\%]$.

The simulation is performance 500 times with different sample paths of transmission delay for a time horizon of $T = 3\,\text{ms}$. A sample path of transmission delay is shown in Fig 5.3. For comparison, two communication networks are investigated. In the proposed QoS communication network, the transition generator, i.e. the probability distribution of transmission delays, is designated such that the control performance and network cost are optimized. Furthermore, the controller is synchronously switched with the transmission delays. The benchmark communication network considers the worst-case delay $\tau(2)$ and the system is designed with associated feedback gain $K(2)$. The evolution of mean trajectory $\bar{x}^T(t) = [\bar{x}_1^T(t) \ \bar{x}_2^T(t)]$ is shown in Fig. 5.4 (a) and (b) for comparison. The mean trajectories converge exponentially towards a ball around the origin of radius $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 0.60\,\text{s}$, faster than the worst-case design with transmission delay (-42.3%), see Table 5.1. However, the network cost is only 27% more than the worst-case design. Clearly, the proposed approach has superior performance the trade-off between control performance and network cost is achieved.

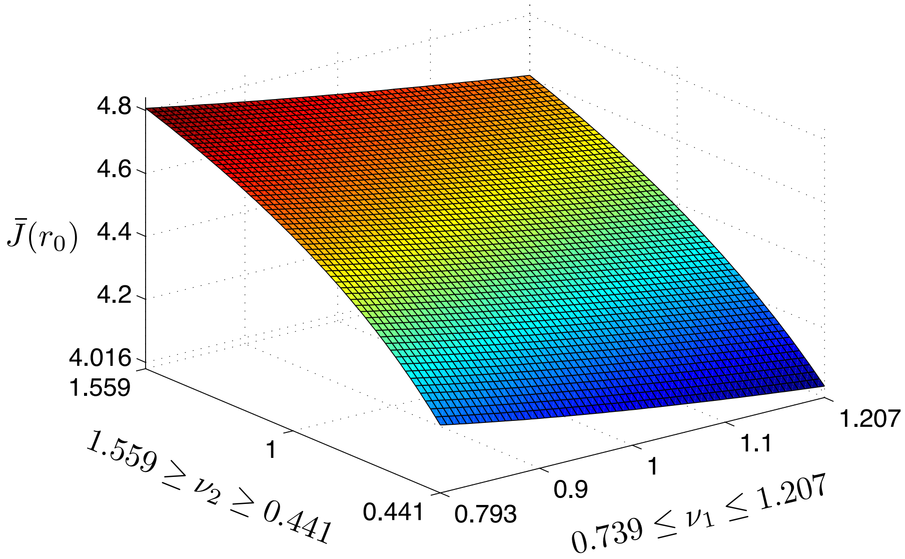


Figure 5.2: The cost function $\bar{J}(r_0)$ and transition generator $\alpha_i - \Delta\bar{\alpha}_i \leq \nu_i \leq \alpha_i + \Delta\bar{\alpha}_i$.

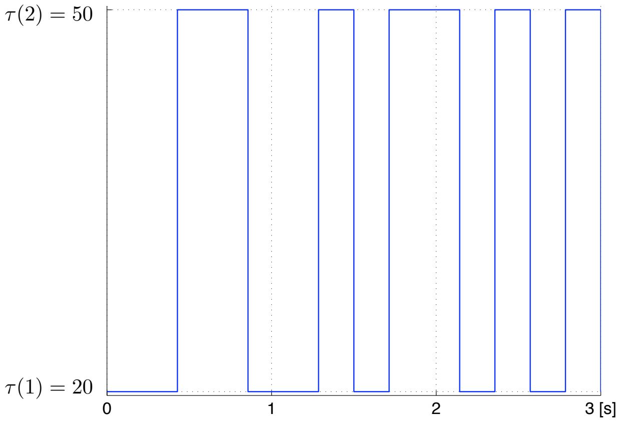


Figure 5.3: The sample path of transmission delay.

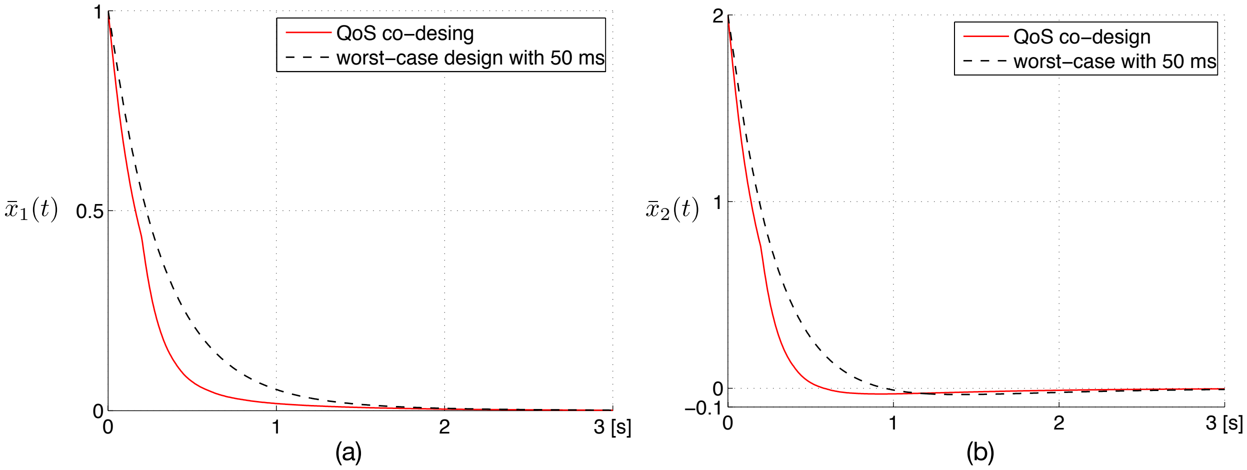


Figure 5.4: The mean trajectories of system with QoS co-design (solid line) and worst-case design with delay $\tau(2)$ (dashed line).

Table 5.1: Control performance and network cost.

	$t_{0.05}$ [s]	Network cost [unit]
QoS network	0.60	2.54×10^{-3}
$K(2)$ with delay $\tau(2)$	1.04	2×10^{-3}

Case B: NCS with Output-feedback controller

To illustrate the network and control system co-design approach for NCSs with output-feedback controller, the LTI system in Example 3.6 is reconsidered. Assume the Markovian transmission delays having the values $\tau_1(r_t) = [15\ 32]$ ms and $\tau_2(r_t) = [15\ 28]$ ms. Set $\gamma = 0.4$, the Markov process probability transition rate as

$$\mathcal{A} = \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix},$$

and

$$R(1) = R(2) = 10^{-4} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The perturbation upper bounds of the probability transition rate are (other associated positive definite matrices are given in Example 3.6)

$$\Delta\bar{\alpha}_1 = 0.006, \quad \Delta\bar{\alpha}_2 = 0.447.$$

With the initial condition $x(t_0 + \theta) = 1$, $\theta \in [-\bar{\tau}, 0]$, the initial probability distribution $\mathbb{P}(t_0) = [65\%\ 35\%]$ and corresponding network cost $C(1) = 4.5 \times 10^{-4}$ and $C(2) = 3.7 \times 10^{-4}$, the optimal performance-cost trade-off is achieved by the optimal transition generator

$$\mathcal{A}^*_{\text{QoS}} = \begin{bmatrix} -0.994 & 0.994 \\ 1.447 & -1.447 \end{bmatrix}$$

with $\bar{J}(r_0) = 1.135 \times 10^{-3}$ and associated limiting probabilities $\bar{\mathbb{P}} = [59.3\%\ 40.7\%]$.

The simulation is performed 500 times with different sample paths of delays for a time horizon $T = 5$ s. The mean trajectory is shown in Fig. 5.6. The trajectory converges exponentially towards a ball around the origin of radius $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 1.67$ s, faster than worst-case design with longer transmission delay (−37.2%). However, the network cost is only increased by +12.7%, see Table 5.7. As a result, the trade-off between control performance and network cost is achieved for output-feedback NCS.

The communication network and control system co-design are targeted by parameterizing the transition generator within the perturbation upper bound. The benefits of the co-design approach include i) the guaranteed control performance is achieved by the efficient usage of network resources; ii) the cost-performance optimization results in re-specifying the probability distribution of random delays. This can be easily implemented e.g. for the CSMA/CA algorithm in IEEE 802.15.4 wireless networks, where the random waiting delay is generated for collision avoidance.

According to the case studies, it is shown that the QoS co-design approach results in superior performance and requires relatively low network resources compared to the worst-case design. For groups of dynamical systems closed via a shared network (or multiple NCSs), each dynamical systems can be viewed as a sub-system of an aggregate NCS. Therefore, the same analysis and design approach can be applied.

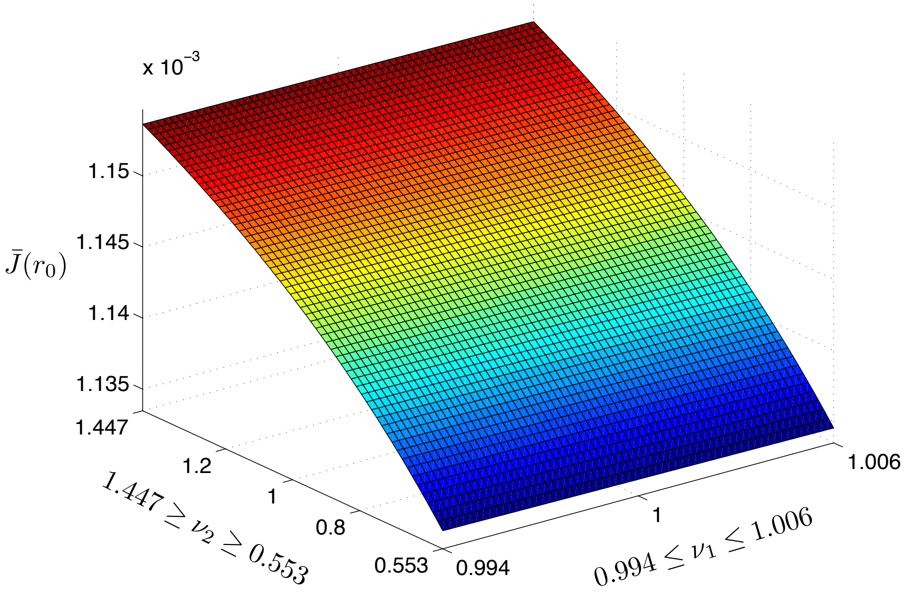


Figure 5.5: The cost function $\bar{J}(r_0)$ and transition generator $\alpha_i - \Delta\bar{\alpha}_i \leq \nu_i \leq \alpha_i + \Delta\bar{\alpha}_i$

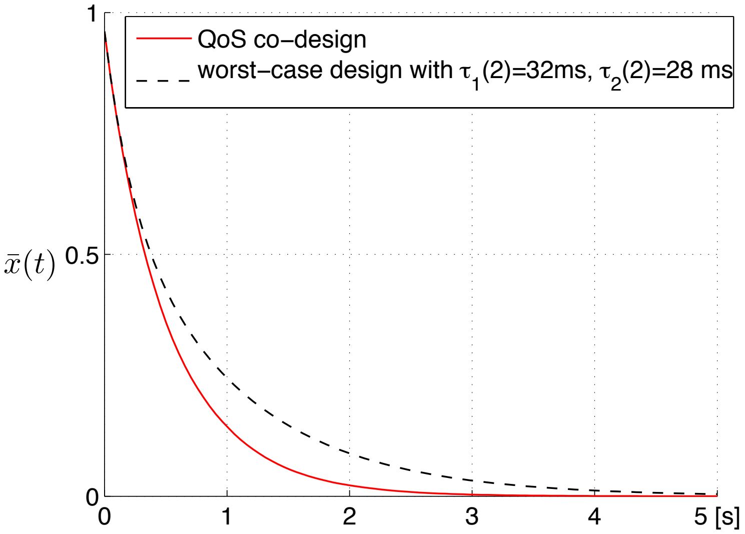


Figure 5.6: The mean trajectories of system with QoS co-design (solid line) and worst-case design with longer delay $\tau_1(2) = 32\,\text{ms}$, $\tau_2(2) = 28\,\text{ms}$.

Figure 5.7: Control performance and network cost.

	$t_{0.05}$ [s]	Network cost [unit]
QoS network	1.62	4.17×10^{-3}
$K(2)$ with delay $\tau_1(2)$ and $\tau_2(2)$	2.58	3.7×10^{-4}

5.3 Networks and systems co-design: optimal random sampling

For traditional sampled-data systems, the sensor measurement and control commands are point-to-point connected. In this case, the transmission uncertainties are neglected. A smaller sampling interval implies better control performance, see Fig. 5.8. For NCSs, however, commercial control networks, such as ControlNet, DeviceNet, Ethernet, Profibus, Flexrate, etc., are implemented into the feedback loop. Due to external traffic and bandwidth limitations, a smaller sampling interval can lead to unnecessary traffic load on these networks [73,128]. Heavy traffic of networks might result in higher possibility of longer delay and cause control performance degradation (see point C in Fig. 5.8). In this section, a random sampling approach is proposed for NCSs. In oder to balance the performance and network usage, a cost function incorporating the control performance and network usage is formulated in terms of the probability distributions of associated sampling intervals. By minimizing the cost function under performance constraint, an *optimal random sampling* can be targeted by optimizing the probability distributions of sampling intervals, so as good control performance is achieved at economic network consumption. This approach is different from the existing MATI-based data scheduling, which precisely regulate sensor data by an maximal allowable intervals. It manages the traffic entry in percentage. This eases the implementation and increases the performance as shown in later case studies.

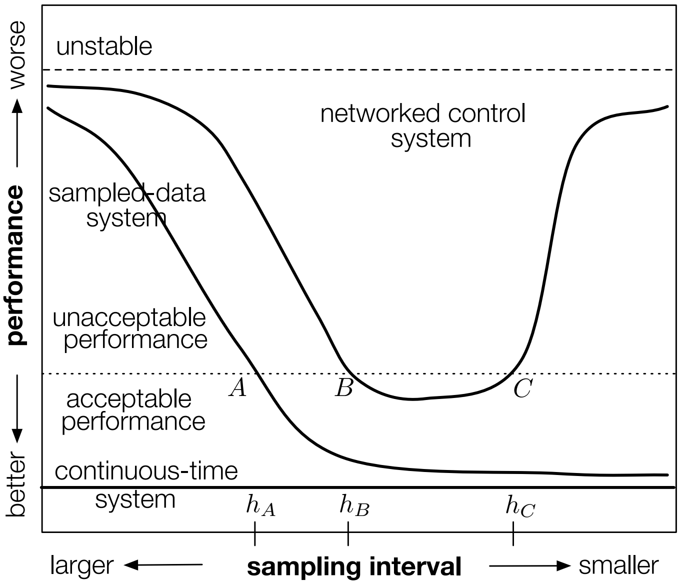


Figure 5.8: Performance comparison of continuous-time system, sampled-data system, and networked control system [73].

The control structure is illustrated in Fig. 5.9. The sensor is aperiodically sampled by $h(k)$ (The output-feedback controller is aperiodically by $h(k)$ as well). The aperiodic sampling intervals as well as transmission delays are described by a set of indication functions with certain probability distributions. The resulting system is a randomly switched time-varying delay system. The guaranteed control performance derived for NCS with random sampling in Chapter 4 is re-considered. A static network scheduling and remote controller

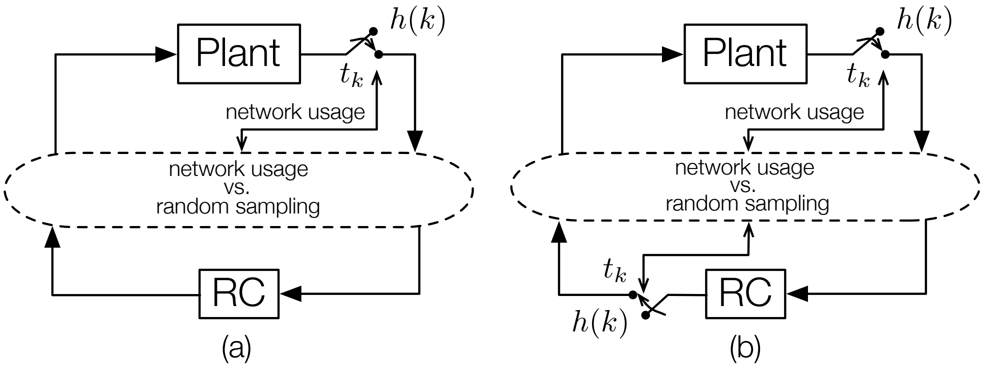


Figure 5.9: Scheme of NCS with optimal probabilistic sampling and state-feedback controller (a) and output-feedback controller (b).

are conjointly derived by arbitrating the control performance and network usage in terms of probability distributions of sampling intervals in a cost function.

5.3.1 Optimal sampling distribution

Assume an NCS with state-feedback controller in Fig. 5.9 is aperiodically sampled by a sampling interval $h(k)$, $k \in \mathbb{K}$, where

$$\underline{h} = \min_{k\in\mathbb{K}}\{h(k)\}, \quad \bar{h} = \max_{k\in\mathbb{K}}\{h(k)\}.$$

For further technical development, the sampling interval $h(k)$ is categorized into $q \geq 2$ intervals by $s_j^h > 0$, $j = 1, \dots, q-1$, satisfying $s_j^h < s_{j+1}^h$, $s_0^h = \underline{h}$ and $s_q^h = \bar{h}$. Each sampling interval s_j^h has occurrence probability

$$\mathbf{P}\{s_j^h\} = p_j^h, \quad \sum_{j=1}^{q} p_j^h = 1.$$

Recall the cost function in (4.31) and let $C_j(t)$ represent the network usage factor of associated sampling interval s_j^h, where

$$C_j(t) = \begin{cases} C_j, & s_{j-1}^h \leq h(k) < s_j^h, \quad j = 1, \dots, q, \quad k \in \mathbb{K}, \\ 0, & \text{otherwise.} \end{cases}$$

The cost function toward the trade-off between network usage and control performance can be formulated as

$$J = \lim_{T\to\infty} \mathbb{E}\bigg\{ \int_0^T z^T(t)Rz(t)dt + \frac{1}{T}\sum_{j=1}^{q}\int_0^T C_j(t)dt \bigg\}. \tag{5.9}$$

The first term in above cost function concerns the control performance and the second term concerns the network usage associated with sampling intervals.

Remark 5.1 Consider the transmission delay in Fig. 5.9 as $\tau_{\text{tx}}(k)$, where

$$\underline{\tau}_{\text{tx}} = \min_{k\in\mathbb{K}}\{\tau_{\text{tx}}(k)\}, \quad \bar{\tau}_{\text{tx}} = \max_{k\in\mathbb{K}}\{\tau_{\text{tx}}(k)\}.$$

Categorize the transmission delay $\tau_{\mathrm{tx}}(k)$ into $l \geq 2$ intervals by $s_l^{\mathrm{tx}} > 0$, $l = 1, \ldots, r-1$, satisfying $s_l^{\mathrm{tx}} < s_{l+1}^{\mathrm{tx}}$, $s_0^{\mathrm{tx}} = \underline{\tau}_{\mathrm{tx}}$ and $s_l^{\mathrm{tx}} = \bar{\tau}_{\mathrm{tx}}$. The occurrence probability of each delay interval s_l^{tx} becomes

$$\mathbf{P}\{s_l^{\mathrm{tx}}\} = p_l^{\mathrm{tx}}, \quad \sum_{l=1}^{r} p_l^{\mathrm{tx}} = 1.$$

The delay interval n delay intervals defined by (4.4) in Chapter 4 can be reformulated as

$$s_i = \sum_{l=1}^{r}\sum_{j=1}^{q} s_l^{\mathrm{tx}} + s_j^h, \quad p_i = \sum_{l=1}^{r}\sum_{j=1}^{q} p_l^{\mathrm{tx}} p_j^h. \tag{5.10}$$

Therefore, the results derived in Chapter 4 can be applied to the NCSs discussed in this section.

The control performance is bounded by (4.34) in Theorem 4.5. Given the probability distribution p_i of each delay intervals s_i, $i = 1, \ldots, n$, the expected values of control performance becomes

$$\lim_{T\to\infty} \mathbb{E}\left\{\int_0^T z^T(t)Rz(t)dt\right\} \leq \bar{J}_{per}(s_1, \ldots, s_n) = V_0(\xi(0)) + \sum_{i=1}^{n} p_i V_i(\xi(0)). \tag{5.11}$$

With known probability distribution of C_j, the expected values of the normalized network usage index over the runtime T can be estimated as

$$\lim_{T\to\infty} \mathbb{E}\left\{\frac{1}{T}\sum_{j=1}^{q}\int_0^T C_j(t)dt\right\} = \sum_{j=1}^{q} p_j^h C_j. \tag{5.12}$$

Combine (5.11) and (5.12) into (5.9), the stochastic cost function (5.9) becomes a deterministic function

$$J(p_1^h, \ldots, p_q^h) = \bar{J}_{per}(s_1, \ldots, s_n) + \sum_{j=1}^{q} p_j^h C_j \tag{5.13}$$

depending on probability distributions of delays and sampling intervals. Therefore, an optimal network utilization results in minimizing (5.13) by an appropriate choose of probability distributions of sampling intervals p_j^h, $j = 1, \ldots, q$. Changing the probability distribution of sampling intervals p_j^h results in variations of p_i. However, any variation of p_i might change the validity of the stability condition derived by Theorem 4.5. In oder to ensure the stability, the LMI (4.32) in Theorem 4.5 has to be considered during the optimization. The details of the optimal network utilization problem is formulated as follows.

Proposition 5.3 An optimal random sampling is given by

$$\begin{aligned} &\min_{p_1^h,\ldots,p_n^h} J(p_1^h, \ldots, p_q^h), \\ &\text{s.t.}(4.32) \end{aligned} \tag{5.14}$$

where $p_1^h, \ldots, p_q^h$ satisfying $\sum_{j=1}^{q} p_j^h = 1$ is the set of admissible probability distribution of sampling intervals.

Note that the above optimization problem is derived for NCSs with state-feedback controllers. For NCSs with output-feedback controllers, a similar result can be also obtained. Consider the NCS in Fig. 5.9 having an output-feedback controller. The cost function in (5.9) becomes

$$J = \lim_{T\to\infty} \mathbb{E}\left\{ \int_0^T \chi^T(t) R \chi(t) dt + \frac{1}{T}\sum_{j=1}^{q} \int_0^T C_j(t) dt \right\}. \tag{5.15}$$

Assume the sensor and output-feedback controller are aperiodically sampled by $h(k)$, which is categorized into $q \geq 2$ intervals. According to Theorem 4.6 and (5.12), the stochastic cost function in (5.15) reduces to deterministic function having the same form as in (5.13). As a result, the optimal network utilization problem for NCSs with output-feedback controller is given as follows.

Proposition 5.4 An optimal random sampling is given by

$$\begin{aligned} &\min_{p_1^h,\ldots,p_n^h} J(p_1^h, \ldots, p_q^h), \\ &\text{s.t.}(4.37) \end{aligned} \tag{5.16}$$

where $p_1^h, \ldots, p_q^h$ satisfying $\sum_{j=1}^{q} p_j^h = 1$ is the set of admissible probability distribution of sampling intervals.

Proposition 5.3 and Proposition 5.4 are static optimization problems with linear matrix inequality constraints. A local minimum can be found easily by the optimization toolbox `fmincon` in Matlab. The proposed optimal random sampling approach can be envisioned as a static scheduling network protocols. Different from the existing static scheduling protocols, such as token ring or token bus, which can provide constant delay as in [129,143], the proposed approach considers random delay and can be applied to modern control networks like wireless LAN, switched Ethernet or Ethernet (see Chapter 6). The benefit of optimal network utilization is studied in the following examples.

5.3.2 Case study: NCS with efficient network utilization

Case A: Optimal random sampling for NCS with state-feedback controller

Assume an NCS

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 1 & -50 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u(t) \tag{5.17}$$

subjected to a network with probabilistic delay distributions: $s_1^{\text{tx}} = 25\,\text{ms}$, $p_1^{\text{tx}} = 60\%$ and $s_2^{\text{tx}} = 40\,\text{ms}$, $p_2^{\text{tx}} = 40\%$. Consider the NCS is sampled by two sampling intervals: $s_1^h = 5\,\text{ms}$ and $s_2^h = 20\,\text{ms}$. Associated to different sampling interval, the network usage factor are heuristically set as $C_1 = 4 \times 10^4$ and $C_2 = 1 \times 10^4$, meaning higher network usage for shorter sampling interval. According to (5.10), the delay intervals become

$$\begin{aligned} s_1 &= s_1^h + s_1^{\text{tx}} = 30, \quad & p_1 &= 0.6p_1^h, \\ s_2 &= s_2^h + s_2^{\text{tx}} = 45, \quad & p_1 &= 0.4p_1^h + 0.6p_2^h, \\ s_3 &= s_3^h + s_3^{\text{tx}} = 60, \quad & p_1 &= 0.4p_2^h. \end{aligned}$$

The probability distributions of sampling intervals p_1^h and p_2^h are designed such that optimal network utilization is achieved.

Set the parameters $\gamma = 1.3$ and

$$R = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}.$$

The optimization problem in Proposition 5.3 is numerically solved by the optimization `fmincon` as well as `Yalmip toolbox` [80] in Matlab. With the initial condition $x^T(t_0 + \theta) = [-2\ 2]$, $\theta \in [-s_3, 0]$, the cost function (5.15) in Proposition 5.3 is optimize by $[p_1^h\ p_2^h] = [36.25\%\ 63.75\%]$ for $J = 3.099 \times 10^4$. The associated stabilizing state-feedback gains are

$$K_1 = [-35.317\ \ -61.632], \quad K_2 = [-9.051\ \ -15.781], \quad K_3 = [-3.5751\ \ -6.2144],$$

where

$$X_1 = \begin{bmatrix} 0.993 & -0.012 \\ -0.012 & 0.087 \end{bmatrix},$$

$$U_1 = 10^3 \times \begin{bmatrix} 4.259 & -2.362 & -4.277 & 2.299 \\ -2.362 & 1.549 & 2.386 & -1.456 \\ -4.277 & 2.386 & 4.324 & -2.306 \\ 2.299 & -1.456 & -2.306 & 1.429 \end{bmatrix},$$

$$U_2 = 10^3 \times \begin{bmatrix} 1.629 & -0.778 & -1.609 & 0.821 \\ -0.778 & 0.508 & 0.774 & -0.498 \\ -1.609 & 0.774 & 1.609 & -0.807 \\ 0.820 & -0.498 & -0.807 & 0.529 \end{bmatrix},$$

$$U_3 = \begin{bmatrix} 834.718 & -397.468 & -808.276 & 442.465 \\ -397.468 & 242.407 & 394.827 & -236.252 \\ -808.276 & 394.827 & 796.051 & -432.471 \\ 442.465 & -236.252 & -432.471 & 260.926, \end{bmatrix}.$$

The simulation is performed 500 times with different sample paths of delays for a time horizon $T = 1\,\mathrm{s}$. The mean trajectory of the considered NCS is shown in Fig. 5.10. The trajectory of the NCS with optimal random sampling converges exponentially towards a ball around the origin of radius $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 0.355\,\mathrm{s}$, faster than the counterpart NCS with low sampling rate (-27.2%), see Table 5.2. However, the network usage is 47.8% less than NCS with high sampling rate, see Table 5.2. The results demonstrate that good control performance at less (economic) network utilization is achieved by the optimal random sampling approach.

Table 5.2: Control performance and network usage.

	$t_{0.05}$ [s]	Network usage [unit]
optimal sampling rate	0.355	2.088×10^4
high sampling rate	0.253	4×10^4
low sampling rate	0.483	1×10^4

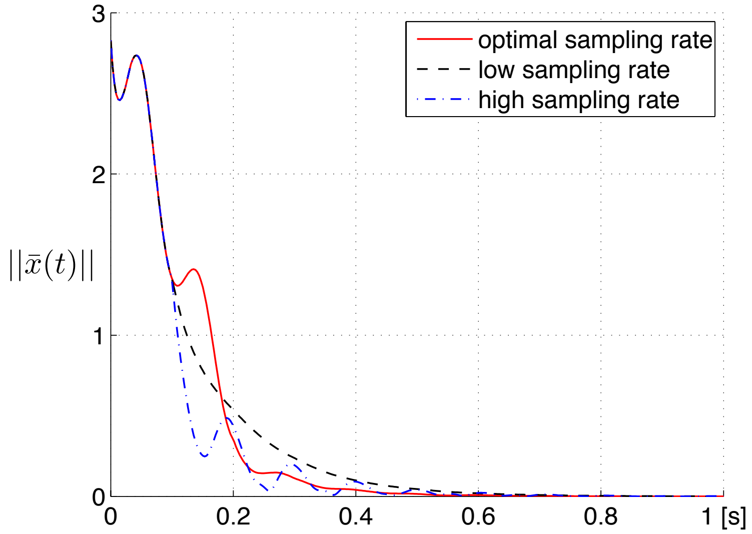


Figure 5.10: The mean norm of trajectories of NCS with optimal sampling rate (solid line), lower sampling rate (dashed line) and high sampling rate (dash-dot line).

It should be pointed out that the network usage factors are chosen to represent the data flow of associated sampling rate. In order to have a fair arbitration between different sampling rates, the ratio of network usage factors is parameterized equal to the ratio of sampling rates, i.e. $C_1/C_2 = h_1/h_2$.

Case B: Optimal network utilization for NCS with output-feedback controller

To present the optimal network utilization for NCSs with output-feedback controller, the system in Example 4.4 is recalled. Assume the sensor and output-feedback controller are sampled by two sampling intervals, i.e. $s_1^h = 5\,\text{ms}$ and $s_2^h = 20\,\text{ms}$ accompanied with network usage factor $C_1 = 4 \times 10^3$ and $C_2 = 1 \times 10^3$, respectively. Note that $C_1/C_2 = h_1/h_2$. The sensor measurement as well as control commands are transferred via a communication network with constant delay $s_1^{\text{tx}} = 25\,\text{ms}$. Based on (5.10), the delay intervals become

$$\begin{aligned} s_1 &= s_1^h + s_1^{\text{tx}} = 30, \quad p1 = p_1^h, \\ s_2 &= s_2^h + s_2^{\text{tx}} = 45, \quad p2 = p_2^h. \end{aligned}$$

A set of distributions of sampling intervals p_1^h and p_1^h is determined such that the underlying network can be optimally utilized.

Set the parameters $\gamma = 0$ and

$$R = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix}.$$

The optimization problem in Proposition 5.4 is numerically solved by the optimization `fmincon` and `Yalmip toolbox` in Matlab. With initial condition $x^T(t_0 + \theta) = [2\ -2\ 0\ 0]$,

$\theta \in [-s_2, 0]$, the cost function (5.16) in Proposition 5.4 is minimized at $J = 3.269 \times 10^3$ by $[p_1^h \; p_2^h] = [24.46\% \; 77.54\%]$. The associated stabilizing output-feedback controllers are

$$A_{c1} = \begin{bmatrix} -29.675 & -33.862 \\ -33.862 & -43.738 \end{bmatrix}, \quad B_{c1} = \begin{bmatrix} -6.100 \\ -6.490 \end{bmatrix}, \quad C_{c1} = \begin{bmatrix} -5.000 & -7.750 \end{bmatrix},$$

$$A_{c2} = \begin{bmatrix} -10.225 & -11.668 \\ -11.668 & -15.070 \end{bmatrix}, \quad B_{c2} = \begin{bmatrix} -5.900 \\ -6.490 \end{bmatrix}, \quad C_{c2} = \begin{bmatrix} -3.333 & -5.000 \end{bmatrix},$$

with

$$X_1 = \begin{bmatrix} 0.022 & 0.005 & -0.015 & 0.005 \\ 0.005 & 0.019 & -0.009 & 0.001 \\ -0.015 & -0.009 & 0.023 & -0.012 \\ 0.005 & 0.001 & -0.012 & 0.009 \end{bmatrix},$$

$$U_1 = \begin{bmatrix} 0.387 & -0.094 & 0.098 & -0.065 & -0.314 & 0.146 & -0.063 & 0.068 \\ -0.095 & 0.526 & 0.007 & -0.020 & -0.062 & -0.460 & 0.061 & -0.026 \\ 0.098 & 0.007 & 0.634 & -0.143 & -0.128 & -0.049 & -0.443 & 0.204 \\ -0.065 & -0.020 & -0.143 & 0.550 & 0.177 & -0.016 & 0.182 & -0.321 \\ -0.314 & -0.062 & -0.126 & 0.177 & 0.469 & -0.024 & 0.026 & -0.021 \\ 0.146 & -0.460 & -0.049 & -0.016 & -0.024 & 0.535 & -0.060 & 0.011 \\ -0.063 & 0.061 & -0.443 & 0.182 & 0.026 & -0.060 & 0.638 & -0.254 \\ 0.068 & -0.026 & 0.204 & -0.321 & -0.021 & 0.011 & -0.254 & 0.488 \end{bmatrix},$$

$$U_2 = \begin{bmatrix} 0.280 & -0.045 & 0.058 & -0.050 & -0.219 & 0.065 & -0.038 & 0.055 \\ -0.045 & 0.365 & 0.005 & -0.012 & -0.049 & -0.335 & 0.036 & -0.018 \\ 0.058 & 0.005 & 0.453 & -0.062 & -0.084 & -0.029 & -0.334 & 0.089 \\ -0.050 & -0.012 & -0.062 & 0.401 & 0.116 & -0.009 & 0.076 & -0.267 \\ -0.219 & -0.049 & -0.084 & 0.116 & 0.317 & -0.001 & 0.020 & -0.013 \\ 0.065 & -0.335 & -0.029 & -0.009 & -0.001 & 0.406 & -0.037 & 0.003 \\ -0.038 & 0.036 & -0.334 & 0.076 & 0.020 & -0.037 & 0.475 & -0.106 \\ 0.055 & -0.018 & 0.089 & -0.267 & -0.013 & 0.003 & -0.106 & 0.404 \end{bmatrix}.$$

Table 5.3: Control performance and network usage.

	$t_{0.05}$ [s]	Network usage [unit]
optimal sampling rate	9.495	1.674×10^3
high sampling rate	8.485	4×10^3
low sampling rate	17.172	1×10^3

The simulation is performed 500 times with different sample paths of delays for a time horizon $T = 20\,\text{s}$. The mean trajectory is shown in Fig. 5.11. The trajectory of NCS with optimal random sampling converges exponentially towards a ball around the origin of radius $||\bar{x}(t)|| = 0.05$ after $t_{0.05} = 9.495\,\text{s}$, closed to the counterpart NCS with high sampling rate (+17.21%). However, the network usage is 58.15% less than NCS with high sampling rate, see Table 5.3. Clearly, the NCS with random sampling efficiently reduces the data flow and preserves the control performance by the proposed approach.

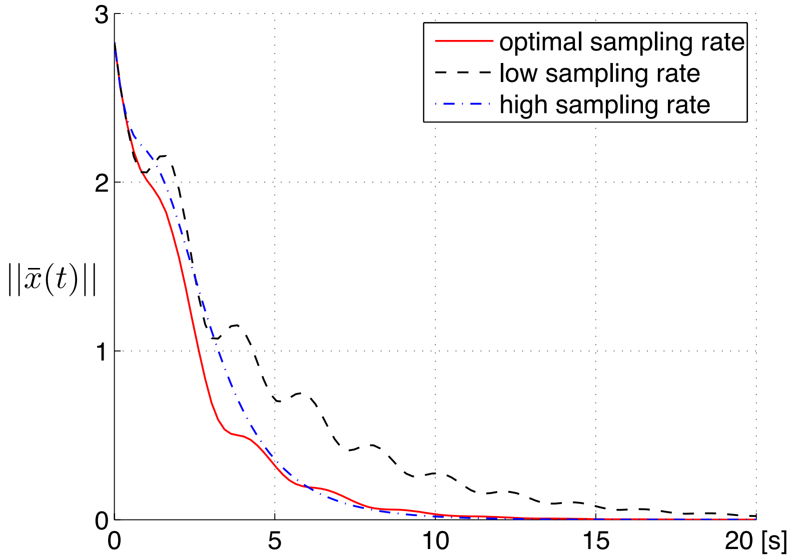


Figure 5.11: The mean norm of trajectories of NCS with optimal sampling rate (solid line), lower sampling rate (dashed line) and high sampling rate (dash-dot line).

The optimal random sampling approach concerns the static sampling interval scheduling of NCSs with performance and network capacity constraints. A stochastic cost function is employed to describe the guaranteed performance versus the probability distribution of random sampling intervals. The probability distribution of random sampling intervals are scheduled to obtain the optimal guaranteed performance based on the stabilization theorems derived Chapter 4. In this way, the control systems and shared networks can be conjointly designed to meet the requirements of control performance and network resources. Different from the existing static scheduling algorithms [129, 143], the optimal random sampling co-design approach allows random delay in the network and can be applied to broadband networks such as switched Ethernet, Ethernet and wireless LAN. Furthermore, the optimal random sampling co-design approach requires no accurate regulation of sensor data by an maximal allowable interval, which eases the implementation.

5.4 Summary and discussion

Practical NCSs are usually subjected to communication networks with limited resources. In views of network bandwidth constraint, efficient use of networks becomes an essential issue in the NCS design. Aiming at this requirement, in this chapter, two novel approaches are developed to deal with a conjoint design of control systems and communication networks.

In the first approach, the Quality-of-Service (QoS) concept from networking community is considered. Based on the results derived in Chapter 3, performance requirements of systems and restrictions of networks are related in terms of Markov probability transition rates of delays. A cost-performance trade-off is then achieved by appropriately parameterizing the Markov probability transition rate. This approach can be implemented for the

MAC protocol, where QoS is considered to re-specify the probability distribution of the waiting delay upon the priority of packets

In the second approach, a probabilistic network scheduling is addressed. According to the results derived in Chapter 4, the control performance and network usage are embodied in a cost index containing the probability distributions of different data transmission rates. An optimal network scheduling is targeted by minimizing the cost function under performance and network capacity constraints. This approach allows random transmission delay and are suitable for broadband networks like switched Ethernet, Ethernet and wireless LAN.

The performance of both considered approaches are studied in numerical examples. All results demonstrate an impressive control performance at reduced network consumption. The benefits of the proposed two co-design approaches include

- guaranteed control performance is achieved by the efficient usage of network resources.
- design parameters of networks, e.g. delays or data sampling data, require no exact values but their distributions. This simplifies the implementation in real networks.

This chapter concludes the theoretical stability and performance analysis of Chapter 3 and Chapter 4 in two control systems and communication networks co-design approaches. Experimental validation of the proposed approaches follows in the next chapter.

6 Experimental Validation

In this chapter, the two control system and communication network co-design approaches proposed in Chapter 5 are validated by two different experiments. In the first experiment, a three degrees-of-freedom (DoF) robotic manipulator with periodic sampling and QoS network are considered. The desired QoS communication characteristics is provided by a network emulator. The stochastic analysis and controller design approaches described in Chapter 3 are applied and a set of delay-dependent switching controllers in joint-space is accordingly determined. Within the first co-design approach, the system performance requirements and network restrictions are linked through the probability transition rates of waiting delays in CSMA/CA network by a stochastic cost function. QoS is related to the ability of adjusting the probability transition rates of waiting delays. By solving the cost function, a cost-performance trade-off is achieved by appropriately choosing the QoS parameters.

In the second experiment, the image stream is considered in the feedback loop and a so-called networked visual servo control system (NVSCS) is developed for experimental validation. In order to relieve the burden on the UDP-based Ethernet, the probability distributions of image transmission rate are considered as variables in the experiment. The delay-dependent switching controllers for the NVSCS are determined by the stochastic design approaches from addressed in Chapter 4. Within the second experiment, a cost function incorporating the control performance and network usage is formulated by means of the probability distributions of associated image transmission rate. A network usage-performance trade-off is targeted by minimizing the cost function under performance constraint, so as good performance is achieved at economic network consumption.

The innovation of the two experiments concerns designing NCSs by cost-performance trade-off approaches, which delicately incorporate the stability requirement, guaranteed performance constraint and random delay properties in one condition. In both experiments, benchmarks without co-design approaches are implemented for comparison. Both experimental results demonstrate significant performance improvements by the proposed co-design approaches.

6.1 Networked robotic manipulator control

6.1.1 Experimental setup

The experimental setup is composed of a 3 DoF robotic manipulator system ViSHaRD3 [125], two PCs running real-time (RT) Linux, and a network emulator. As shown in Fig. 6.1, the ViSHaRD3 device is connected to a remote controller through a network emulated by Netem [2]. The communication between the remote controller and ViSHaRD3 is done based on a QoS network with UDP/IP protocol. In order to increase the performance, the control law of the remote controller is switched according to the sensor-to-controller transmission delay by using the time-stamp technique.

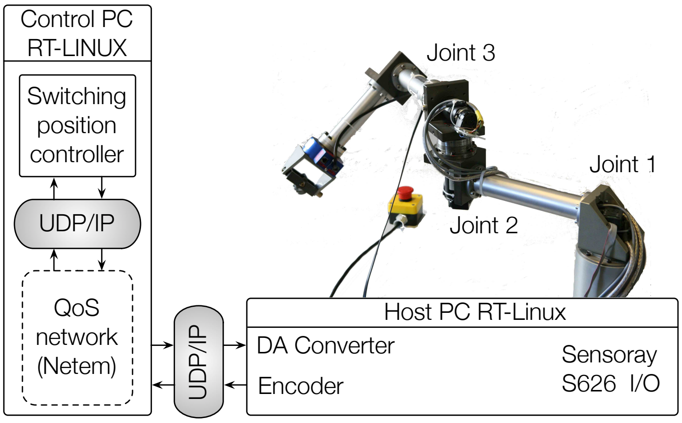


Figure 6.1: Experimental 3 DoF ViSHaRD3 system.

The control for ViSHaRD3 relies on the measurement and amendment of joint displacement. The control commands as well as sensor measurement are provided and read though a Sensoray S626 I/O card. The I/O card is connected to the host PC, and serves as an A/D and D/A converters for the incoming and outgoing signals. Each joint is actuated by a DC motor coupled with a harmonic drive gear. The DC-motor torque is modulated by the PWM amplifier operated under current control. The reference signal is given in voltage from the I/O card and is considered approximately proportional to the DC-motor torque.

The control-loop is implemented in Matlab/Simulink blocksets. Standalone real-time code is generated directly from the Simulink models. The sampling interval is $h = 5\,\text{ms}$. Before demonstrating the experimental results, a detailed description of ViSHaRD3 device, network-induced delay and control structure is presented.

6.1.2 ViSHaRD3

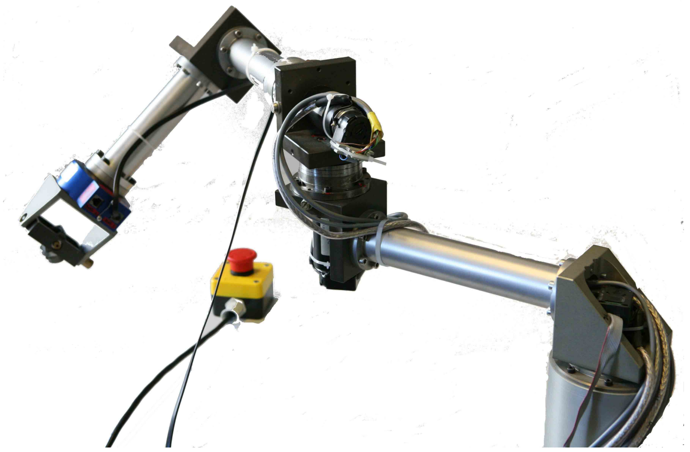


Figure 6.2: The robotic manipulator system ViSHaRD3.

The hardware and the kinematic design of the ViSHaRD3 can be seen in Fig. 6.2 and Fig. 6.3 respectively. ViSHaRD3 is designed as a general purpose haptic interface in order

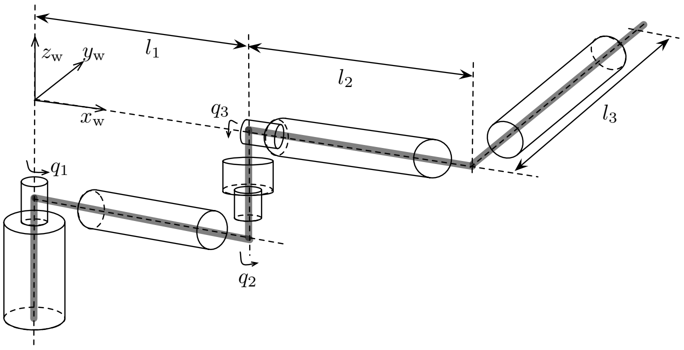


Figure 6.3: The kinematic design of ViSHaRD3 system.

to be used in a variety of application domains. The ViSHaRD3 robotic manipulator has three revolute joints. The first two joints are arranged in SCARA-configuration with vertical axes, avoiding the need for gravitational compensation. All the links have equal lengths of 0.3 m. The end effector is a gimbal mounted thimble with three passive, freely rotating DOF. The rectangular workspace is $0.6 \times 0.25 \times 0.4$ m in width, depth and height. The mass of the moving parts is around 5.5 kg, resulting in an apparent inertia of the end-effector between 1.9 to 18 kg. The torque capability is provided by 150 W Maxon DC motors coupled with harmonic drive gears. The angles of the joints are measured by optical incremental encoders with a resolution of 2000 counts per revolution. Multiplied by the gear ratio 1:100 it results in a resolution of $\pi 10^{-5}$ rad. For a more detailed description of ViSHaRD3 as well as issues concerning control aspects, friction compensation etc. see [125].

6.1.3 QoS scheduler: Netem

The QoS network used in the experiment is provided by a Linux kernel based network emulator *Netem* [2, 52]. Netem is an enhancement of the traffic control function in Linux and built by using differentiated services (diffserv) facilities in Linux kernel. It incorporates a variety of network attributes, including round-trip delay, packet dropouts, jitter and other scenarios. Netem works as a classful queuing discipline (qdisc) implemented between Transport and Network layer of the OSI reference model. Its basic architecture is shown in Fig. 6.4. The qdisc decides which packets from which task will be putted into the sending buffer according to current netem settings with priority First-In-First-Out (pFIFO) mechanism. The highest priority traffic is placed into priority 0 and is always serviced first. Similarly, priority 1 is always emptied of pending packets before priority 2 is dequeued. Based on the qdisc, the random delays of the considered QoS network is configured.

As default, the qdisc settings are configured by command line via the Netlink socket interface. In the ViSHaRD3 experimental testbed, a Matlab/Simulink interface is developed by the `S-function` and Netlink socket such that parameters could be easily changed within Simulink. As pointed out in [52], the accuracy of emulated transmission delay is limited by the system tick rate. In the ViSHaRD3 setup, Linux kernel version 2.6.31 with system tick rate 1000 Hz is used, the precision of the transmission delay is 1 ms. Within the experimental validation, the random delay with exponential distributions is measured up to 2 ms. Exact settings of delays and associated probability transition rate are given in the next section.

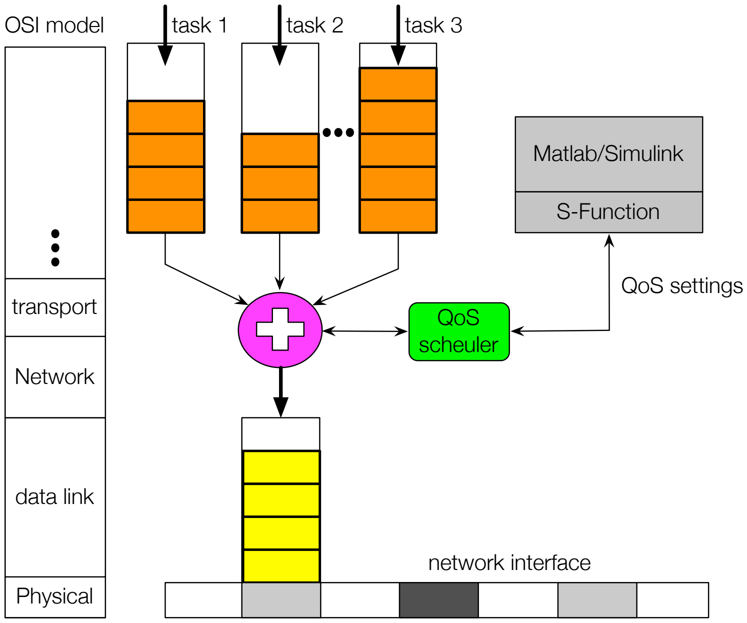


Figure 6.4: Netem basic architecture.

6.1.4 Controller design

Due to the requirement of the proposed approach, the ViSHaRD3 device is linearized by computed torque feedforward approach [113]. Combined with friction compensation, the linearized ViSHaRD3 system is decoupled into three systems

$$\frac{d}{dt}\begin{bmatrix} q_i(t) \\ \dot{q}_i(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -150 \end{bmatrix}\begin{bmatrix} q_i(t) \\ \dot{q}_i(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i(t), \tag{6.1}$$

where $i = 1, 2, 3$ for joint 1, 2 and 3. The ViSHaRD3 system is equipped with a set of PD controllers, which are synchronously switched with the SC delay.

The joint vector q of ViSHaRD3 is fed to the remote controller through a communication network having the SC delay $\tau_{\text{sc}}(r_t) \in \{5, 7, 8\}$ ms, CA delay $\tau_{\text{ca}}(r_t) \in \{2, 4, 5\}$ ms. With the sampling interval $h = 5$ ms, the resulting delay[1] has the values $\tau(1) = 12$ ms, $\tau(2) = 16$ ms, $\tau(3) = 18$ ms. The network cost with associated delays are given in Table 6.1. The Markov probability transition rate is set to

Table 6.1: The network cost and associated delays.

SC delay [ms]	CA delay [ms]	Round-trip delay [ms]	Network cost [unit]
$\tau_{\text{sc}}(1) = 5$	$\tau_{\text{ca}}(1) = 2$	$\tau(1) = 12$	6×10^{-3}
$\tau_{\text{sc}}(2) = 7$	$\tau_{\text{ca}}(2) = 4$	$\tau(2) = 16$	4×10^{-3}
$\tau_{\text{sc}}(3) = 8$	$\tau_{\text{ca}}(3) = 5$	$\tau(3) = 18$	3×10^{-3}

$$\mathcal{A} = \begin{bmatrix} -4 & 1 & 3 \\ 1 & -3 & 2 \\ 1 & 1 & -2 \end{bmatrix}.$$

[1] The resulting delay is determined by (3.5) in Chapter 3 as $\tau(r_t) = h + \tau_{\text{sc}}(r_t) + \tau_{\text{ca}}(r_t)$.

Meaning the stationary probability for $[\tau(1)\ \tau(2)\ \tau(3)]$ are $[\bar{\mathbf{P}}_1\ \bar{\mathbf{P}}_2\ \bar{\mathbf{P}}_3] = [20\%\ 25\%\ 55\%]$.

The ViSHaRD3 system is equipped with a set of PD controllers, which are synchronously switched with the SC delay. Combine the switching PD controller into (6.1), it yields

$$\frac{d}{dt}\begin{bmatrix} q_i(t) \\ \dot{q}_i(t) \end{bmatrix} = A_i \begin{bmatrix} q_i(t) \\ \dot{q}_i(t) \end{bmatrix} + \bar{K}(r_t) \begin{bmatrix} q_i(t-\tau(r_t)) \\ \dot{q}_i(t-\tau(r_t)) \end{bmatrix}, \tag{6.2}$$

where $i = 1, 2, 3$ and

$$A_1 = A_2 = A_3 = \begin{bmatrix} 0 & 1 \\ -1 & -150 \end{bmatrix}, \quad \bar{K}(r_t) = \begin{bmatrix} 0 & 0 \\ -K_P(r_t) & -K_D(r_t) \end{bmatrix}.$$

The PD gains in (6.2) are computed by (3.28)-(3.29) in Theorem 3.5 in Chapter 3 using the `Yalmip toolbox` in MATLAB. The LMI (3.53) in Theorem 3.5 is solved for the decay rate of $\gamma = 0$ and

$$R(1) = R(2) = R(3) = 10^{-4} \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The feasible PD gains and perturbation upper bonds are summarized in Table 6.2. Consider the initial joint vector of ViSHaRD3 as $q^T(t_0) = [q_1(t_0)\ q_1(t_0)\ q_3(t_3)] = [0\ 0\ -0.5\pi]$ rad, $\dot{q}^T(t_0) = [0\ 0\ 0]$ rad/s and the initial probability distribution of Markovian delay $\mathbf{P}(t_0) = [20\%\ 40\%\ 40\%]$. The optimization problem in Proposition 5.1 is optimized by the probability transition rate

$$\mathcal{A}^*_{\text{QoS}} = \begin{bmatrix} -4.000 & 1.000 & 3.000 \\ 0.998 & -2.997 & 1.999 \\ 0.997 & 0.997 & -1.994 \end{bmatrix}$$

with $\bar{J}(r_0) = 1.712 \times 10^{-3}$.

Table 6.2: The feasible switching PD controller for ViSHaRD3 device.

Markovian delay [ms]	Joint 1, 2 and 3	Perturbation bounds
$\tau(1) = 12$	$K_P(1) = 70.65$, $K_D(1) = 5.50$	$\Delta\bar{\alpha}_1 = 8.14 \times 10^{-5}$
$\tau(2) = 14$	$K_P(2) = 26.54$, $K_D(2) = 2.20$	$\Delta\bar{\alpha}_2 = 2.90 \times 10^{-3}$
$\tau(3) = 18$	$K_P(3) = 15.21$, $K_D(3) = 1.55$	$\Delta\bar{\alpha}_3 = 5.10 \times 10^{-3}$

6.1.5 Experimental results

A sinusoidal function, which has the amplitude 0.2 and frequency 0.5 rad/s, serves as position reference q_r to the system. The experiments are run 10 times with different sample paths of the delay. A sample path of the transmission delay is shown in Fig 6.5. For comparison, the random delays are rendered as constant by buffering and the NCS with worst-case design is investigated. The evolutions of normalized mean control error

$$\bar{e}(t) = \frac{q(t) - q_r(t-\tau(r_t)}{\max\{||q_r(t)||\}}$$

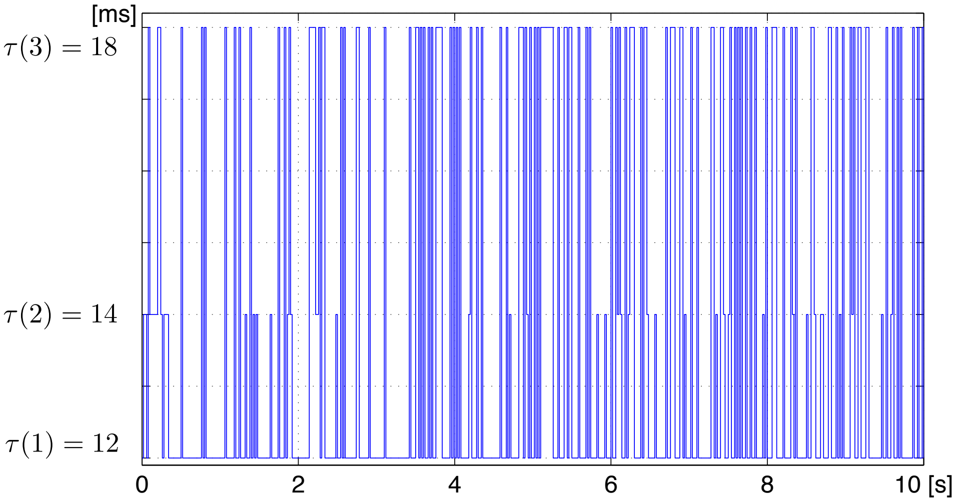


Figure 6.5: The sample path of Markovian delay for the experiment.

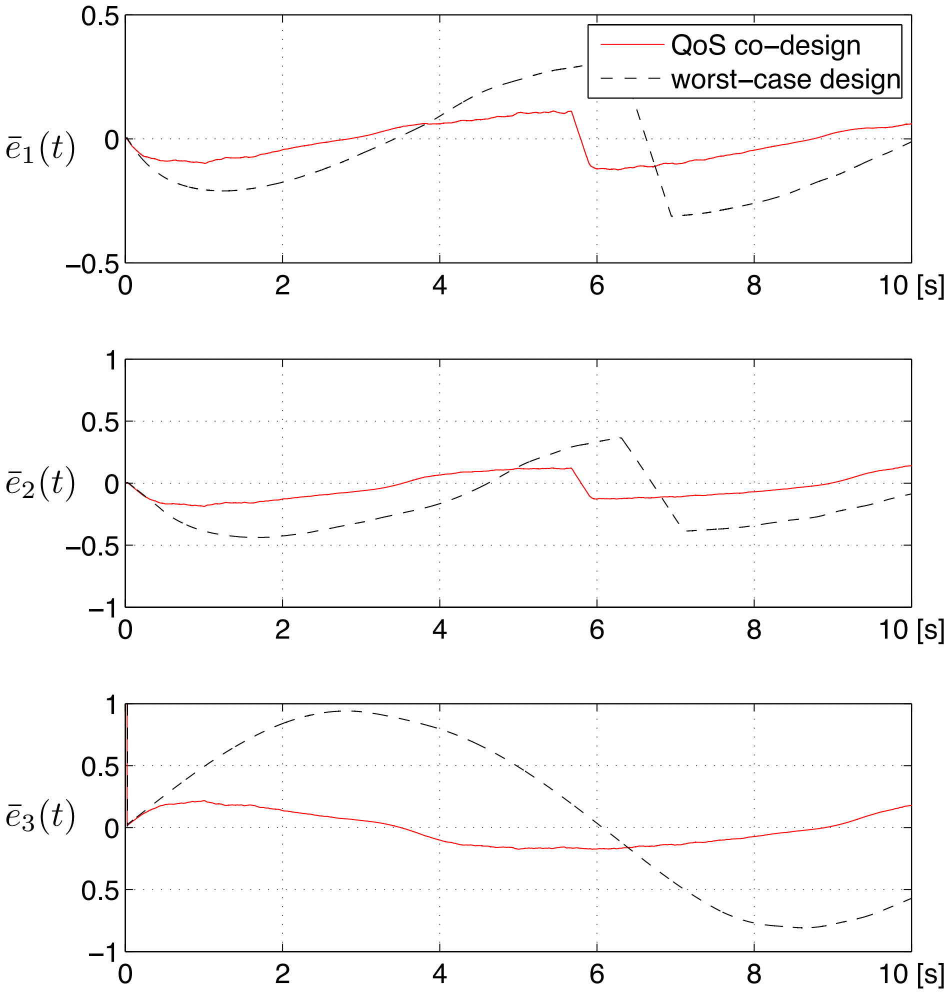


Figure 6.6: The normalized mean control error evolutions with QoS co-design approach (solid line) and worst-case design (dashed line).

are shown in Fig. 6.6. It is observed that the normalized mean control errors of the proposed QoS co-design approach (solid line) are smaller than the worst-case design approach (dashed line). The L_2 norm of normalized mean control error over the experimental time horizon $[0\ t_f]$, $t_f = 10\,\text{s}$, i.e.

$$||\bar{e}(t_f)||_2 = \sqrt{\int_0^{t_0} \bar{e}(t)\bar{e}^T(t)\,dt},$$

is measured to be $||\bar{e}(t_f)||_2 = 1.42$ for the proposed approach and $||\bar{e}(t_f)||_2 = 3.92$ for worst-case design. The QoS co-design approach has 63.8% less control error than the worst-case design. However, the network cost is only 26.7% more.

Table 6.3: The feasible switching PD controller for ViSHaRD3 device.

	$\|\|\bar{e}(t_f)\|\|_2$	Network cost [unit]
NCS with QoS co-design	1.42	0.38
NCS with worst-case design	3.92	0.3

As a result, the QoS co-design approach has superior performance benefits over the counterpart even when the total switching difference of delay is small, as in this case of only 6 ms. In case of larger switching delay differences, the performance benefit is likely to be more obvious.

6.2 Networked visual servo control

6.2.1 Experimental setup

The second experiment concerns a networked visual servo control system (NVSCS). Visual servo control refers to the use of visual-data in the feedback control loop, see [61] for an overview for its advantages and challenges. With recent advances in communication and computing technologies, video grabbing, image processing and control can be implemented on different platforms across a common communication network. This kind of setup results in NVSCSs. The benefits of an NVSCS include: an NVSCS employs different cameras over a network; it provides wide-range visual feedback and increases system autonomy. An NVSCS has distributed computation for image processing; it enables high-speed vision feedback and is more robust to occlusions, see [112] for details.

The considered NVSCS is composed of two commercial linear motors from Copley Control Corp, two PCs running RT-Linux and a camera (Mikrotron EoSens MC1363). The experiment refers to the synchronization of the two linear motor modules by using the camera as a position sensor. As shown in Fig. 6.7, an object is mounted on a reference linear motor module; and a controlled linear motor module is equipped with a camera. The two linear motor modules are connected to host PCs running RT Linux via a Sensory S626 I/O card. To enable a high-speed vision feedback in the control loop, the image frames captured by the camera with resolution of 648 × 480 pixels are processed by distributed standalone PCs (X86-64 AMD Phenom II ×4 810 processor) implemented with pose estimation algorithms over the network. In order to relieve the burden on the network, the camera framerate is considered as variable and the main objective to be determined in the following section.

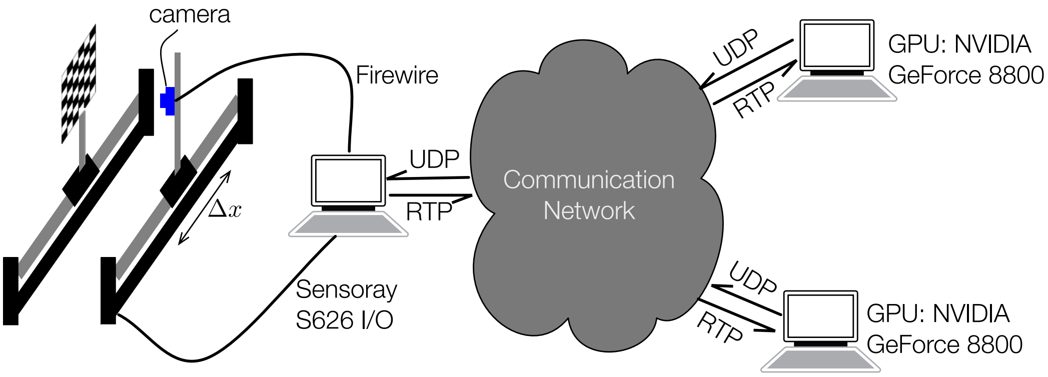


Figure 6.7: Experimental setup of networked visual servo control.

The control functions are implemented in Matlab/Simulink blocksets. Standalone real-time code is generated directly from Simulink models. The sampling period of the control function is $h = 1\,\mathrm{ms}$.

6.2.2 Pose estimation and distributed computation

Computation delay

The position of the controlled module, $x(t)$, is estimated by using pose estimation. By given matched feature pairs, the problem of pose estimation could be considered as dual problem of 2D visual servoing proposed in [86]. A virtual camera is applied and is moved by using a visual servoing law to minimize the position errors between current observed image features and previous ones.

In order to increase the accuracy of position measurement, a scale invariant feature transform (SIFT) algorithm [81], which is known for its robust character, is applied for feature extraction. To improve the performance, the SIFT algorithm is implemented on a GPU (Graphics Processing Units) by exploiting its massive parallel processing capability. Matched feature pairs contain outliers, which lead to errors of the pose estimation. Therefore, a RANSAC (RANdom Sample Consensus) algorithm [34] is used for the rejection of outliers.

The number of matched feature pairs has impact on the time required for pose estimation. Moreover, image features vary from frame to frame due to different view angles, illumination conditions and noise. As a result, the image processing delay is random as shown in Fig. 6.8. The image processing delay has mean value $\bar{\tau}_{cp} = 20.33\,\mathrm{ms}$ and standard deviation 2.98 ms. The whole image processing time ranges from $\tau_{\mathrm{cp}} = [14\ 40]\,\mathrm{ms}$ (NVIDIA GeForce 8800) in the experiment.

Transmission delay

For computing efficiency, a distributed computation over the network is considered in the experiment. As shown in Fig. 6.7, image frames captured by the camera are transmitted

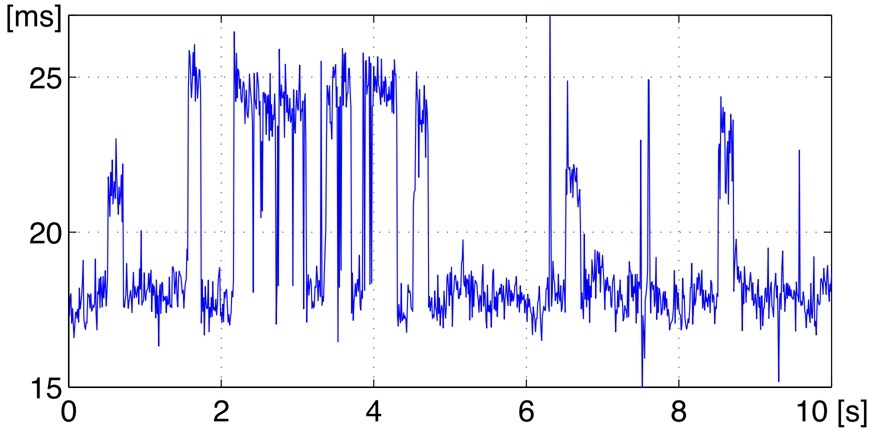


Figure 6.8: Random image processing delay with mean value 20.33 ms.

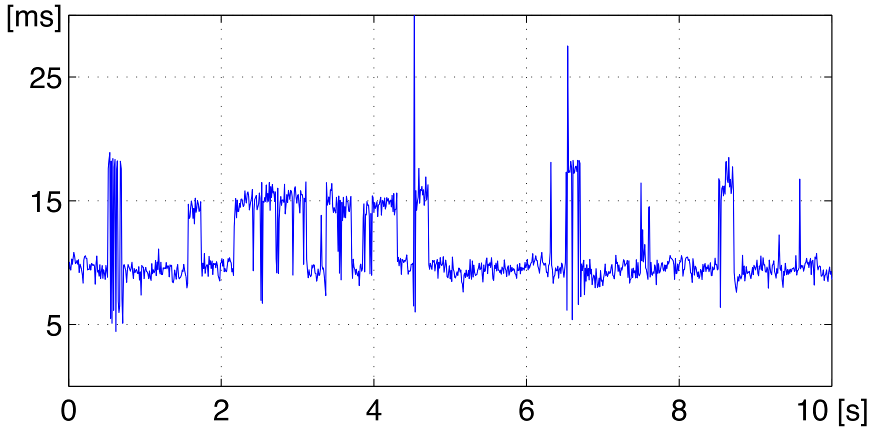


Figure 6.9: Random image transmission delay with mean value 13.83 ms.

to differently standalone PCs for pose estimation. The results of pose estimation, i.e. difference between current and desired camera pose, are fed back through a communication network to the host PC. The communication is done by Ethernet with an enhancement UDP protocol specialized in audio and video streaming, namely, real-time transport protocol (RTP). The First-in-First-out (FIFO) mechanism is considered during the transmission. The packet dropouts are considered as additional delays. The round-trip transmission delay in the experiment ranges from $\tau_{\text{tx}} = [3\ 39]$ ms and has mean value $\bar{\tau}_{\text{cp}} = 13.83$ ms with standard deviation 6.95 ms.

Sampling intervals

Each standlone PC (X86-64 AMD Phenom II ×4 810 processor 8G RAM) can execute the process estimation at maximal camera framerate 40 Hz. Within the experiment, two standalone PCs are assumed to be available over the network. The policy for distributed computation follows a sequential assignment. As a result, the camera frame rate can be improved to 80 Hz.

It should be pointed out that each image frame has the resolution 640 × 480 pixels and the data size 2.4 Mb. For the camera running at framerate 80 Hz, the network data flow is 192Mb/s. In order to reduce the data flow, the co-design approach studied in Section 5.3.1 is considered. Two sampling intervals, i.e. $s_1^h = 12.5\,\text{ms}$ (80 Hz) and $s_2^h = 25\,\text{ms}$ (40 Hz), are considered in the system analysis. An optimal distribution of the data transmission intervals, i.e. s_1^h and s_2^h, will be determined so as good control performance can be achieved at economic network consumption.

For later system analysis and controller design, the computation delay, transmission delay and sampling intervals and the total delay are summarized in Table 6.4.

Table 6.4: The total delay of the considered networked visual servo control system.

	τ_{cp} [ms]	τ_{tx} [ms]	s_l^h [ms]	total delay $= \tau_{\text{cp}} + \tau_{\text{tx}} + s_l^h$ [ms]
40 Hz	[14 40]	[3 39]	25	[42 104]
80 Hz	[14 40]	[3 39]	12.5	[29.5 91.5]

6.2.3 Controller design and optimal data transmission scheduling

The linear motor with input torque and output position is a second order system. The system parameters are obtained by standard least square identification of the response to square pulse input and yields

$$\frac{d}{dt}\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0.959 & -1169.9 \end{bmatrix}\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t). \tag{6.3}$$

Consider the maximal values of computation and transmission delays. According to (5.10) in Section 5.3.1 of Chapter 5, the delay intervals become

$$\begin{aligned} s_1 &= s_1^h + \max\{\tau_{\text{cp}} + \tau_{\text{tx}}\} = 91.5, \quad & p_1 &= p_1^h, \\ s_2 &= s_2^h + \max\{\tau_{\text{cp}} + \tau_{\text{tx}}\} = 104, \quad & p_2 &= p_2^h. \end{aligned}$$

The probability distributions of sampling intervals p_1^h and p_2^h are designed such that optimal network utilization is achieved. The control module is equipped with a set of delay-dependent PD controllers. Combine the switching PD controller into (6.3)

$$\frac{d}{dt}\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0.959 & -1169.9 \end{bmatrix}\begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} + \sum_{i=1}^{2} \beta_i(t)\bar{K}_i \begin{bmatrix} x(t-s_i) \\ \dot{x}(t-s_i) \end{bmatrix}, \tag{6.4}$$

where

$$\bar{K}_i = \begin{bmatrix} 0 & 0 \\ -K_{Pi} & -K_{Di} \end{bmatrix}.$$

Set the parameters $\gamma = 0$, $C_1 = 2$, $C_2 = 1$ and

$$R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

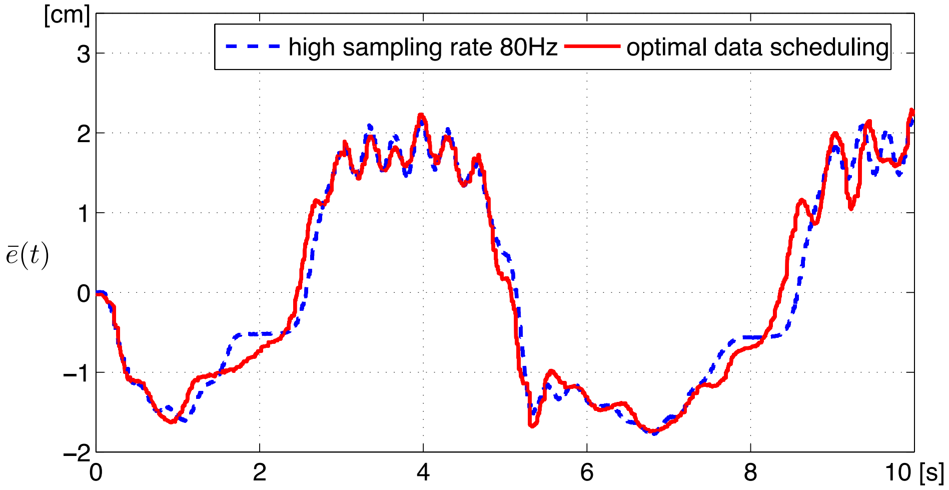


Figure 6.10: The mean control error evolution of NCS with optimal sampling distribution (solid line) and with high sampling rate 80 Hz (dash line).

The optimization problem in Proposition 5.3 is numerically solved by the optimization `fmincon` as well as `Yalmip toolbox` in Matlab. With the initial condition $[x(0)\ \dot{x}(0)]^T = [0\ 0]$, $\theta \in [-s_2, 0]$, the cost function (5.15) in Proposition 5.3 is optimize by $[p_1^h\ p_2^h] = [50\%\ 50\%]$ for $J = 1.67$. The associated stabilizing state-feedback gains are

$$\bar{K}_1 = \begin{bmatrix} 0 & 0 \\ -900 & -15 \end{bmatrix}, \quad \bar{K}_2 = \begin{bmatrix} 0 & 0 \\ -600 & -5 \end{bmatrix}.$$

6.2.4 Experimental results

The reference module moves along sinusoidal trajectory with the amplitude of 20 cm and frequency of 0.17 Hz. The experiment run 10 times from the same initial condition of both modules. The control error, defined by

$$e(t) = x_r(t) - x_c(t),$$

where $x_r(t)$ denotes the position of the reference module and $x_c(t)$ denotes the position of the controlled module, are compared. The evolution of mean control error is shown in Fig. 6.10. The optimal data rate design approach has maximal tracking error $\bar{e}_{\max} = 2.32\,\text{cm}$ and variance of tracking error $\bar{e}_{\text{var}} = 1.77 \times 10^{-4}\,\text{cm}^2$, similar to the maximal tracking error of high data rate design approach (+5.93%), see Table 6.5. However, the network usage (data flow) is 25% less than high data rate design approach. As a result, an optimal network utilization is achieved by NCS with optimal sampling rate. The experimental results show that controller design algorithm proposed in Chapter 5 enables a good control performance at low network resource consumption compared to the conventional design.

It should be pointed out that the optimal sampling distribution can be envisioned as a static network scheduling. However, unlike the MATI-based network scheduling, which can only be applied to network with constant delay, such as token ring or token bus. The

Table 6.5: Control performance and network usage.

	$\bar{e}_{\text{max}}$ [cm]	$\bar{e}_{\text{var}}$ [cm^2]	Network usage [unit]
optimal data rate	2.32	1.77×10^{-4}	1.5
high data rate	2.19	1.71×10^{-4}	2

optimal sampling distribution approach allows random delay and can be applied broadband networks like Ethernet.

6.3 Summary and discussion

Two different experiments are conducted for the validation of the proposed co-design approaches developed in the previous chapter. The focus of the first experiment is to conjoint design the control system and underlying QoS communication network. The QoS network is and diffserv-based (differentiated services) and emulated by the emulator Netem. The waiting delays of sensor data are considered as QoS parameter and adjusted by their probability transition rates. Based on the results obtained in Chapter 3, the performance requirements of control systems and restrictions of QoS networks are related in terms of the probability transition rates of waiting delays. A conjoint design of control systems and communication networks is achieved by parameterizing the probability transition rates of delays. This co-design approach can be applied to control systems over CSMA/CA networks.

In the second experiment, a networked visual servo control system is addressed. In order to relieve the network burden, a static data transmission scheduling is conjointly considered within controller design. An optimal sampling distribution of control systems is derived by a cost function, so as the performance is preserved by an acceptable level and the data flow is reduced. Superior to existing MATI-based network scheduling algorithms, the optimal sampling distribution supports the networks with random delays, e.g. Ethernet. In addition, this approach can be further extend to dynamic data scheduling by feeding the network traffic into the control loop.

According to the experimental results, both approaches demonstrate superior performance benefits over conventional worst-case design and are very promising for future NCSs.

7 Conclusion and Future Work

7.1 Conclusion

In most NCS related works, the control systems are either designed robustly against the network-induced uncertainties or are developed according to the known communication quality [53]. The restriction on the former approaches concerns the worst-case assumptions of network-induced delays or dropouts, which often result in conservation controller design. The later approaches result in good control performance with guaranteed network quality. However, their applications might be restrictive in cases of limited network resources. In order to improve the restrictions in previous works, a novel stochastic conjoint design scheme of control systems and communication networks for a set of LTI NCSs is presented in this dissertation. The major contributions are three fold: i) stochastic control and analysis methodologies are taken into account during the design phase. The resulting co-design approaches consider the probability distributions of network uncertainties rather than their worst-case assumptions. ii) the proposed co-design approaches integrate the performance trade-off from control and communication. It enables the development of more efficient and affordable NCSs with limited network resources. iii) the application of the proposed co-design approaches requires no exact values of external uncertainties, but their distributions. This simplifies the implementation in real systems and networks.

Two kinds of LTI systems are investigated in this dissertation. For LTI systems with periodic data transmission rate, a link between guaranteed control performance and required network quality is established. The main objective is to balance the performance versus required network resources. For the LTI systems with random data transmission rate, an optimal data transmission scheduling is developed. It is aimed to adapt the system data transmission rate to achieve an efficient network utilization.

For this purpose, the theoretic properties of MJLSs are addressed and notions of stochastic stability are revisited in Chapter 2. Stochastic NCSs with periodic and aperiodic sampling intervals are studied in Chapter 3 and Chapter 4, respectively. In Chapter 3, the random network-induced delay and packet dropouts are modeled by a Markov process. Regarding the delays, a novel delay-dependent switching controller is proposed to improve the performance. The resulting closed-loop system is an MJS with mode-dependent delay. The associated stability as well as stabilization conditions are derived by using stochastic Lyapunov functionals. Unconventional to deterministic time-delay systems, the obtained stability as well as stabilization conditions are derived depending not only on delays, but also on their associated statistical properties. An innovative guaranteed control performance analysis is developed to link the statistical properties of delays and desired performance of NCSs.

In Chapter 4, the aperiodic sampling intervals and transmission delays are reformulated into time-varying delays by the input-delay approach. A set of indicator functions having independent identical distributions (i.i.d.) is introduced to described the occurrence of the time-varying delays. This closed-loop system becomes a randomly switched time-delay system. Associated stability and stabilization conditions are obtained depending on the probabilistic distributions of sampling intervals and delays. A novel performance guaranteed design, which correlates the performance upper bound with probability distributions of sampling intervals, is developed.

Based on the control methodologies derived in Chapter 3 and Chapter 4, two novel approaches aiming at performance oriented control system and communication network co-design are addressed in Chapter 5. The first co-design approach concerns a trade-off between network cost and control performance. It is well-know that performance of an NCS strongly depends on the underlying communication qualities, e.g. transmission delay. Guaranteed short transmission delay results in good control performance. However, this needs the provision of large network resources. Inspired by the Quality-of-Service (QoS) concept from the networking community, the resources of a network can be assigned to different applications for different performance requirements. According to the results from Chapter 3, the performance requirements of a control system and restrictions of a communication network are linked through statistical properties of an underling Markov process. QoS is then referred to the ability of adjusting the probability transition rate of such Markov process. A cost-performance trade-off can be achieved by appropriately parameterizing the Markov probability transition rate.

The second co-design approach refers to the design of an NCS with acceptable performance at affordable network usage. The focus there is to determine a data transmission scheduling for NCSs such that the network resources is more efficiently used; meanwhile, the control performance of connected systems is also preserved. In accordance to the results derived in Chapter 4, a stochastic cost function incorporates control performance and network usage in terms of probability distributions of associated data rate is developed. Consider probability distributions of data rate as design variables, an probabilistic data scheduling can be determined by minimizing the cost function. The performance of both considered approaches are explored in case studies. Benefits in terms of guaranteed control performance with efficient network usage are shown in the simulation results.

The experimental validation of the developed con-design approaches in this dissertation is presented in Chapter 6. Two different experiments are conducted. In the first experiment, a 3-DoF robotic manipulator is subjected to a QoS network. The first co-design approach addressed in Chapter 5 is used to cope with a cost-performance trade-off. Compared to the conventional worst-case design approach, the QoS co-design approach has 63.5% less control error. The second experiment concerns a networked visual servo control system (NVSCS) with variable image transmission rate. The second co-design approach in Chapter 5 is applied to achieve a network usage-performance trade-off. Within the experiment, an optimal data scheduling is determined by the proposed approach, which reduce the total data flow up to 25% at similar control performance.

According to numerical or experimental results, the proposed co-design approaches demonstrate superior performance benefits and are promising for the future NCSs.

7.2 Outlook

Due to the modularity, re-configurability, and versatility, NCSs are increasingly considered as a replacement of traditional control systems in the automation industry. Furthermore, the ongoing development of communication technologies, e.g. wireless communication, has speeded up this trends. An important issue in the further research of NCSs concerns closing the gap between theories and the praxis, as well as extend the existing analysis methodologies to more complex systems with non-ideal communication networks.

Research on future directions connected to the presented co-design scheme includes the extension of the analysis methods derived in Chapter 3 and Chapter 4 to stochastic nonlinear feedback systems, e.g. small-gain theorem. The controller design algorithms are obtained in terms of linear matrix inequality (LMI) for numerical efficiency. However, it introduces conservatism. A less conservative but more efficient controller design algorithm is desirable for NCSs. Furthermore, there is a number of research directions emerging from this dissertation which may further have significant impact in the applications. Some of them are

- **Visual servo control systems**: The integration of vision into control is recognized as a key element to increase the accuracy, autonomy and application domains of robots in manufacturing. However, using visual-data in the feedback loop causes random delays from image acquisition, image processing and data transmission, which deteriorates the control performance. The transmission of high-speed video stream over network needs large network bandwidth, which is limited. The knowledge and intuition gained from this dissertation can be applied to visual servo control systems. Compared to conventional visual servo control design, a compatible control performance at less network resource requirement can be achieved.

- **Complex systems**: Many practical complex dynamical systems suffer lots of abrupt and unknown uncertainties, e.g. sensor or actuator failures. From operation point of view, it is desirable to know how much uncertainties a complex system can tolerate. The proposed approaches of this dissertation concerns statistical properties of network-induced uncertainties into the control design. This can be extended to complex dynamical systems in the face of component failures. By using the proposed approaches, the stability and performance of dynamical systems can be guaranteed to certain percentage of uncertainties or component failures.

- **Smart power grid**: The trend in the current power generation is moving from a centralized supply architecture to a distributed module, where many small energy sources (e.g. solar cells, wind turbines) are interconnected to each other to supply the entire energy consumption. The aggregate system can be viewed as a complex NCS comprising two parts with different characteristics, i.e. the energy network, and the communication network which is used to exchange information among the energy sources and energy consumers. An important issue of smart grid technology concerns how to use different energy sources more efficiently to lower the energy cost for consumers. Similar to the QoS concept, different energy sources can be viewed as different power supply quality. By modeling the energy consuming as a dynamical system, a energy source allocation strategy can be determined by applying the proposed QoS design approach. Furthermore, by feeding the current energy consumption back to different power plants via the communication network, a dynamic energy control can be developed to increase the energy independence.

A Design Tools and Preliminary Lemmas

A.1 Design Tools

As mentioned earlier, the controller design algorithms derived in Chapter 3 and Chapter 4 are represented in terms of LMIs. In order to solve the LMI problems, in the rest of this section, a numerical tool will be introduced based on `Yalmip Toolbox` [80] for Matlab. For the controller design, the *ready-to-run* Matlab scripts can be found in the accompanying CD. The system requirements for using these scripts are

- Matlab 2007 (or higher),
- Yalmip Toolbox, *http://users.isy.liu.se/johanl/yalmip*,
- SeDuMi, *http://sedumi.ie.lehigh.edu/*,
- PENBMI, *http://www.penopt.com*.

The instructions of these scripts are given in the following subsections.

A.1.1 NCS with periodic sampling and random delay

The design goal is to compute a controller $K(r_t)$ for a linear control system

$$\dot{x}(t) = Ax(t) + BK(r_t)x(t - \tau(r_t))$$

with Markovian delay $\tau(r_t) = [\tau(1), \dots, \tau(N)]$ and probability transition rate $\mathcal{A} = (\alpha_{i,j})$, $i, j \in \mathcal{S} := [1, \dots, N]$.

The ready-to-run script for state-feedback controller `ps_statfeedbacklmi.m` can be found in the accompanying CD. According to Theorem 3.2 in Chapter 3, the known parameters of `ps_statfeedbacklmi.m` are defined as:

```
    A   = [0 1; 1 -50];
    B   = [0.5 ; 1];
   Tr   = [-3 3; 1 -1];   %probability transition rate
    t   = [0.02 0.04];    %Markovian delay
    g   = 1.2;            %decay rate: gamma
 n1_1   = 8.1e+04;
 n2_1   = 1.327e+05;
 n1_2   = 7.29e+05;
 n2_2   = 9.676e+05;
  ep1   = 3.051;
  ep2   = 1.332;
```

The unknown variables of `ps_statfeedbacklmi.m` are defined as:

```
     W    = sdpvar(4,4);       % W>0
 x11_1    = sdpvar(2,2);       % x11_1>0
    Y1    = sdpvar(1,2);
 x11_2    = sdpvar(2,2);       % x11_2>0
    Y2    = sdpvar(1,2);
    Z2    = zeros(2);
    X1    = [x11_1 Z2; -n1_1*x11_1 n2_1*x11_1];
    X2    = [x11_2 Z2; -n1_2*x11_2 n2_2*x11_2];
```

Running `ps_statfeedbacklmi.m` in Matlab environment, the delay-dependent switching feedback controller can be determined as:

```
 K1   =
        -4.5670    -1.9830
 K2   =
        -2.0000    -1.3570
```

If `PENBMI` is installed, the feedback controller can be also solved by a ready-to-run BMI script `ps_statfeedbackbmi.m`. The parameter settings of `ps_statfeedbackbmi.m` are similar to `ps_statfeedbackbmi.m`. The only different is in the unknown variable settings:

```
     W    = sdpvar(4,4);      % W>0
 x11_1    = sdpvar(2,2);      % x11_1>0
 x21_1    = sdpvar(2,2);      % x21_1>0
 x22_1    = sdpvar(2,2);
 x11_2    = sdpvar(2,2);
 x21_2    = sdpvar(2,2);
 x22_2    = sdpvar(2,2);
    K1    = sdpvar(1,2);       %K1 can be defined as unknown variable or given value
    K2    = sdpvar(1,2);       %K2 can be defined as unknown variable or given value
    X1    = [x11_1 Z2; x21_1 x22_1];
    X2    = [x11_2 Z2; x21_2 x22_2];
```

It should be pointed out that the state-feedback gains are defined as known variables in `ps_statfeedbackbmi.m`. However, in order to increase the numerical efficiency, the state-feedback gains K1 and K2 are often defined as known values. Likewise, running `ps_statfeedbackbmi.m` in Matlab environment, the delay-dependent switching feedback controller can be determined as:

```
 K1   =
        -30.3680    -20.0410
 K2   =
        -10.6500    -5.0320
```

Remark A.1 Similar Matlab scripts for designing delay-dependent output-feedback controllers based on Theorem 3.4 can be found in the accompanying CD, i.e. `ps_outputfeedbacklmi.m` (LMI) and `ps_outputfeedbackbmi.m` (BMI). The parameter settings of output-feedback controller are similar to state-feedback controller design and therefore omitted here.

A.1.2 NCS with aperiodic sampling

For NCSs with aperiodic sampling, the design purpose is to compute a set of K_i, $i = 1, \dots, N$, which stabilizes a randomly switched time-delay system

$$\dot{x}(t) = Ax(t) + \sum_{i=1}^{N} \beta_i B K_i x(t - s_i),$$

with the occurrence probability of s_i

$$\mathbb{E}\{\beta_i\} = p_i, \quad \sum_{i=1}^{N} p_i = 1.$$

A Matlab script for state-feedback controller `as_statfeedbacklmi.m` can be found in the accompanying CD. According to Theorem 4.2 in Chapter 4, the known parameters of `as_statfeedbacklmi.m` are defined as:

```
  A   = [0 1; 1 -50];
  B   = [0.5 ; 1];
 s1   = 0.045;     %45ms delay
 s2   = 0.065;     %65ms delay
 s3   = 0.085;     %85 ms delay
 p1   = 0.6;       %p1 = 60%
 p2   = 0.3;       %p2 = 30%
 p3   = 0.1;       %p3 = 10%
  g   = 0;         %decay rate: gamma
 r1   = 3.051;
 r2   = 1.332;
```

The unknown variables of `as_statfeedbacklmi.m` are defined as:

```
U1   = sdpvar(4,4);     % U1>0
U2   = sdpvar(4,4);     % U2>0
U3   = sdpvar(4,4);     % U3>0
Y1   = sdpvar(1,2);
Y2   = sdpvar(1,2);
Y3   = sdpvar(1,2);
X1   = sdpvar(2,2);     % X1>0
Z2   = zeros(2);
 X   = [X1 Z2; -r1*X1 r2*X1];
```

Then, the state-feedback gains can determined by LMI in the Matlab environment. In order to derive a less conservative state-feedback controllers, a BMI Matlab script `as_statfeedbackbmi.m` can be found in the CD. The difference in the variable setting of `as_statfeedbackbmi.m` to the above mentioned LMI script concerns:

```
U1  = sdpvar(4,4);    % U1>0
U2  = sdpvar(4,4);    % U2>0
U3  = sdpvar(4,4);    % U3>0
K1  = sdpvar(1,2);    %K1 can be defined as unknown variable or given value
K2  = sdpvar(1,2);    %K2 can be defined as unknown variable or given value
K3  = sdpvar(1,2);    %K3 can be defined as unknown variable or given value
X1  = sdpvar(2,2);    % X1>0
X2  = sdpvar(2,2);
X3  = sdpvar(2,2);
 X  = [X1 Z2; X2 X3];
```

Note that the feedback gains K1, K2 an K3 can be either defined as unknown variables or, in order to reduce the numerical complexity, given values. Executing `as_statfeedbackbmi.m` in the Matlab environment, the feedback gains are determined as:

```
K1  =
      -11.9560    -23.8775
K2  =
      -6.1933    -12.1706
K3  =
      -2.8319    -5.5417
```

Remark A.2 The Matlab scripts for designing delay-dependent output-feedback controllers can be also found in the accompanying CD, i.e. `as_outputfeedbacklmi.m` (LMI) and `as_outputfeedbackbmi.m` (BMI).

A.2 Lemmas

This section introduces a number of Lemmas, which are extensively used in this dissertation in various proofs of the proposed theorems.

Lemma A.1 [14] Let X and Y be real constant matrices with appropriate dimensions. Then

$$X^TY + Y^TX \le \varepsilon X^TX + \frac{1}{\varepsilon}Y^TY$$

holds for any $\varepsilon > 0$.

Lemma A.2 Let Y be a symmetric matrix, H and E be any known matrices of appropriate dimensions and F satisfy $F^TF \le I$. For any scalar $\zeta > 0$, it has

(i) $HFE + E^TF^TH^T \le \zeta HH^T + \zeta^{-1}E^TE$

(ii) $Y + HFE + E^TF^TH^T < 0$ holds if and only if there exists a scalar $\zeta > 0$ such that $Y + \zeta HH^T + \zeta^{-1}E^TE < 0$.

Lemma A.3 Let X, Y be positive definite matrices and a, b be scalars satisfying $a > 0$ and $a > b$. Then

$$\lambda_{\max}(aX + bY) \le \lambda_{\max}(aX + aY).$$

Proof: It is noted that

$$(aX+bY)^T(aX+bY) \le (aX+aY)^T(aX+aY).$$

Pre- and post-multiply the above inequality by the normalized eigenvector v^T and v, which corresponds to the maximal eigenvalue, i.e. $\lambda_{\max}(aX+bY)$. It becomes

$$\begin{aligned}\lambda^2_{\max}(aX+bY) &= v^T(aX+bY)^T(aX+bY)v \\ &\le v^T(aX+aY)^T(aX+aY)v.\end{aligned} \tag{A.1}$$

According to the definition of second order induced norm (Euclidean norm) of matrix, it has

$$\begin{aligned}\lambda^2_{\max}(aX+aY) &= ||aX+aY||^2 \\ &= \max_{\|v\|_2=1} v^T(aX+aY)^T(aX+aY)v\end{aligned} \tag{A.2}$$

Combine (A.1) and (A.2), it yields

$$\lambda_{\max}(aX+bY) \le \lambda_{\max}(aX+aY)$$

■

Lemma A.4 Consider a function

$$V(z(t),r_t) = \int_{-\tau(r_t)}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta.$$

For $r_t = i$, $\mathcal{L}V(z(t),r_t)$ has the inequality

$$\begin{aligned}\mathcal{L}V(z(t),r_t) \le \tau(i)z^T(t)Qz(t) &- \int_{t-\tau(i)}^{t} z^T(s)Qz(s)ds \\ &+ \bar{\alpha}\int_{-\bar{\tau}}^{-\underline{\tau}}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta.\end{aligned} \tag{A.3}$$

Proof: Since

$$\begin{aligned}&\mathbb{E}\{V(z(t+\Delta t),r_{t+\Delta t}|z(t),r_t=i)\} \\ &\quad = \mathbb{E}\Big\{\int_{-\tau(r_{t+\Delta t})}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\theta\Big| z(t),r_t=i\Big\}\end{aligned}$$

and

$$\begin{aligned}
&\mathbb{E}\Big\{\int_{-\tau(r_{t+\Delta t})}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\theta\Big| z(t), r_t = i\Big\}\\
&\quad= \sum_{j\neq i}^{N}\mathbb{E}\Big\{I_{\{r_{t+\Delta t}=j\}}\int_{-\tau(j)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\theta\Big| z(t), r_t = i\Big\}\\
&\qquad+\mathbb{E}\Big\{I_{\{r_{t+\Delta t}=i\}}\int_{-\tau(i)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\theta\Big| z(t), r_t = i\Big\}\\
&\quad= \sum_{j\neq i}^{N}\mathbf{P}\{r_{t+\Delta t} = j|r_t = i\}\int_{-\tau(j)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\\
&\qquad+\mathbf{P}\{r_{t+\Delta t} = i|r_t = i\}\int_{-\tau(i)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\\
&\quad= \sum_{j\neq i}^{N}\big(\alpha_{i,j}\Delta t + o(\Delta t)\big)\int_{-\tau(j)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\\
&\qquad+\big(1+\alpha_{i,i}\Delta t + o(\Delta t)\big)\int_{-\tau(i)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd,
\end{aligned}$$

where $I_{\{\cdot\}}$ is an indicator function. Applying above two equations, it becomes

$$\begin{aligned}
&\mathbb{E}\{V(z(t+\Delta t), r_{t+\Delta t}|z(t), r_t = i)\} - V(z(t), t)\\
&\quad= \int_{-\tau(i)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd - \int_{-\tau(i)}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\\
&\qquad+\big(\alpha_{i,i}\Delta t + o(\Delta t)\big)\int_{-\tau(i)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd\\
&\qquad+\sum_{j\neq i}^{N}\big(\alpha_{i,j}\Delta t + o(\Delta t)\big)\int_{-\tau(j)}^{0}\int_{t+\Delta t+\theta}^{t+\Delta t} z^T(s)Qz(s)dsd.
\end{aligned}$$

According to Definition 2.4 and due to the fact $\lim_{\Delta t\to 0} o(\Delta t)/\Delta t = 0$, it yields

$$\begin{aligned}
\mathcal{L}V(z(t), r_t) &= \lim_{\Delta t\to 0^+}\frac{1}{\Delta t}\Big\{\mathbb{E}\{V(z(t+\Delta t), r_{t+\Delta t}|z(t), r_t = i)\} - V(z(t), r_t)\Big\}\\
&= \tau(i)z^T(t)Qz(t) - \int_{t-\tau(i)}^{t} z^T(s)Qz(s)ds\\
&\quad+\sum_{j=1}^{N}\alpha_{i,j}\int_{-\tau(j)}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta.
\end{aligned} \tag{A.4}$$

Since $\alpha_{i,j} > 0$, $i \neq j$ and $-\alpha_{i,i} = \sum_{i\neq j}^{N} \alpha_{i,j}$ for $i, j \in \mathcal{S}$, equation (A.4) can be written as

$$\begin{aligned}
\mathcal{L}V(z(t), r_t) &= \tau(i)z^T(t)Qz(t) - \int_{t-\tau(i)}^{t} z^T(s)Qz(s)ds + \alpha_{i,i}\int_{-\tau(i)}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta \\
&\quad + \sum_{i\neq j}^{N} \alpha_{i,j}\int_{-\tau(j)}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta \\
&\leq \tau(i)z^T(t)Qz(t) - \int_{t-\tau(i)}^{t} z^T(s)Qz(s)ds + \alpha_{i,i}\int_{-\underline{\tau}}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta \\
&\quad + \sum_{i\neq j}^{N} \alpha_{i,j}\int_{-\bar{\tau}}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta \\
&= \tau(i)z^T(t)Qz(t) - \int_{t-\tau(i)}^{t} z^T(s)Qz(s)ds + \alpha_{i,i}\int_{-\underline{\tau}}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta \\
&\quad + \alpha_i\int_{-\bar{\tau}}^{0}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta.
\end{aligned}$$

Define $\bar{\alpha} = \max_{i\in\mathcal{S}}\{\alpha_i\}$, it becomes

$$\mathcal{L}V(z(t), r_t) \leq \tau(i)z^T(t)Qz(t) - \int_{t-\tau(i)}^{t} z^T(s)Qz(s)ds + \bar{\alpha}\int_{-\bar{\tau}}^{-\underline{\tau}}\int_{t+\theta}^{t} z^T(s)Qz(s)dsd\theta.$$

and completes the proof. ■

Bibliography

[1] Controller Area Network (CAN), http://www.can-cia.de/.

[2] Net:Netem, http://www.linuxfoundation.org/en/Net:Netem.

[3] *DeviceNet Specifications.* Open DeviceNet Vendors Association, Boca Raton, FL, 2nd edition, 1997.

[4] *ControlNet specifications.* ControlNet International, Boca Raton, FL, 2nd edition, 1998.

[5] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393 – 422, 2002.

[6] K.-E. Årzén and A. Cervin. Control and embedded computing: survey of research directions. In *16th IFAC World Congress*, Prague, Czech Republic, 2005.

[7] K.-E. Årzén, A. Cervin, J. Eker, and L. Sha. An introduction to control and scheduling co-design. In *Proceedings of the 39th Conference on Decision and Control*, Sydney, 2000.

[8] K. Åström and B. Wittenmark. *Adaptive Control.* Addison-Wesley, 1955.

[9] J. Baillieul and P. Antsaklis. Control and communication challenges in networked real-time systems. *Proceedings of the IEEE*, 95(1):9–28, January 2007.

[10] S. L. Beuerman and E. J. Coyle. The delay characteristics of CSMA/CD networks. *IEEE Transactions on Communications*, 36(5):553–563, 1988.

[11] S. Biegacki and D. VanGompel. The application of DeviceNet in process control. *ISA Transactions*, 35:169–176, 1996.

[12] M. Bladt and M. Sørensen. Statistical inference for discretely observed Markov jump processes. *Journal of the Royal Statistical Society*, 67(3):395–419, 2005.

[13] E.-K. Boukas. Control of stochastic systems with time-varying multiple time delays: LMI approach. *Journal of Optimization Theory and Applications*, 119(1):19–36, Oct. 2003.

[14] E.-K. Boukas and Z.-K. Liu. *Deterministic and Stochastic Time Delay Systems.* Birkhäuser, Boston, 2002.

[15] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM, Philadelphia, 1994.

[16] S. Boyd and L. Vandenberghe. *Convex optimization.* Cambridge University Press, UK, 2004.

[17] M. S. Branicky, V. Liberatore, and S. M. Phillips. Networked control system co-simulation for co-design. In *Proceedings of the American Control Conference*, Denver, USA, 2003.

[18] M. S. Branicky, S. M. Phillips, and W. Zhang. Scheduling and feedback co-design for networked control systems. In *41st IEEE Conference on Decision and Control*, 2002.

[19] R. Brockett. Stabilization of motor networks. In *Proceedings of the 34th Conference on Decision and Control*, New Orleans, LA, 1995.

[20] C. G. Cassandras and J. Lygeros. *Stochastic hybrid systems.* Taylor & Francis Group, Florida, 2007.

[21] C.-C. Chen, S. Hirche, and M. Buss. Controller design and experimental validation for networked control systems with time-varying random delay. *Journal of the Society of Instrument and Control Engineers*, 47(8):676–685, 2008.

[22] C.-C. Chen, A. Molin, and S. Hirche. Guaranteed cost control over quality-of-service networks. In *European Control Conference*, Budapest, Hungary, 2009.

[23] C.-C. Chen and M. Buss S. Hirche. Sampled-data networked control systems with random time delay. In *17th IFAC World Congress*, Seoul, 2008.

[24] C.-C. Chen, H. Wu, K. Kühnlenz, and S. Hirche. A switching controller for a networked vision-based control system. *to appear in Automatisierungstechnik.*

[25] J. Colandairaj, G. W. Irwin, and W. G. Scanlon. Analysis and co-simulation of an IEEE 802.11b wireless networked control system. In *16th IFAC World Congress*, Prague, Czech Republic, 2005.

[26] J. Colandairaj, G.W. Irwin, and W.G. Scanlon. Wireless networked control systems with QoS-based sampling. *IET Control Theory & Applications*, 1(1):430–438, 2007.

[27] J. Cortes, S. Martinez, T. Karatas, and F. Bullo. Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation*, 20(2):243–255, April 2004.

[28] O. L. V. Costa, M. D. Fragoso, and R. P. Marques. *Discrete-time Markov jump linear systems.* Springer-Verlag, London, 2005.

[29] E. Coyle and B. Liu. A matrix representation of CSMA/CD networks. *IEEE Transactions on Communications*, 33(1):53–64, Jan 1985.

[30] R.M. Daoud, H.H. Amer, H.M. Elsayed, and Y. Sallez. Ethernet-based car control network. In *IEEE CCECE/CCGEI*, Ottawa, 2006.

[31] N. Elia and S. K. Mitter. Stabilization of linear systems with limited information. *IEEE Transactions and Automatic Control*, 46(9):1384–1400, 2001.

[32] J. E. Elson. *Time Synchronization in Wireless Sensor Networks.* PhD thesis, University of California Los Angeles, 2003.

[33] R. Ernst. Codesign of embedded systems: Status and trends. *IEEE Design & Test*, 15(2):45–54, 1998.

[34] M. A. Fischer and R. C. Bolles. Random sampled consensus: A paradigm for modeling fitting with applications to image analysis and automated cartography. *Communication of the ACM*, 2(381-395), 1981.

[35] G. F. Franklin, J. D. Powell, and M. Workman. *Digital control of dynamic systems*. Addison Wesley Longman, Menlo Park, 3rd edition, 1998.

[36] E. Fridman. New Lyapunov-Krasovskii functionals for stability of linear retarded and neutral type systems. *Systems & Control Letters*, 43:309–319, 2001.

[37] E. Fridman. Stability of linear descriptor systems with delay: A Lyapunov-based approach. *Journal of Mathematical Analysis and Applications*, 273:22–24, 2002.

[38] E. Fridman, A. Seuret, and J.-P. Richard. Robust sampled-data stabilization of linear systems: an input delay approach. *Automatica*, 40:1441–1446, 2004.

[39] E. Fridman and U. Shaked. A descriptor system approach to H_∞ control of linear time-delay systems. *IEEE Transactions on Automatic Control*, 47(2):253–270, Feb 2002.

[40] M.E.M.B. Gaid, A. Çela, and Y. Hamam. Optimal integrated control and scheduling of networked control systems with communication constraints: application to a car suspension system. *IEEE Transactions on Control Systems Technology*, 14(4):776–787, 2006.

[41] H. Gao, J. Wu, and P. Shi. Robust sampled-data H_∞ control with stochastic sampling. *Automatica*, 45:1729–1736, 2009.

[42] P. Giacomazzi. Closed-form analysis of end-to-end netowrk delay with Marko-modulated Poisson and fluid traffic. *Computer Communications*, 32:640–648, 2009.

[43] Robert M. Gray and Lee D. Davisson. *An Introduction to Statistical Signal Processing*. Cambrige University Press, Cambrige, UK, 2004.

[44] K. Gu, V. L. Kharitonov, and J. Chen. *Stability of Time-Delay Systems*. Birkhäuser, Boston, 2003.

[45] R. K. Gupta. *Co-Synthesis of Hardware and Software for Digital Embedded Systems*. PhD thesis, Stanford Universit, 1995.

[46] J. K. Hale and S. M. V. Lunel. *Introduction to Functional Differential Equations*. Springer-Verlag, 1993.

[47] K. Han. *Randomly sampled-data control systems*. PhD thesis, University of California Los Angeles, 1990.

[48] H. Hanselmann. Implementation of digital controllers—a survey. *Automatica*, 23(1):7–32, 1987.

[49] M. S. Hasan, H. Yu, and A. Carrington. Overview of wireless networked control systems over mobile ad-hoc network. In *Proceedings of the 14th International Conference on Automation & Computing*, Brunel University, London, 2008.

[50] J. He, Z. Tang, H. H. Chen, and S. Wang. An accurate Markov model for slotted CSMA/CA algorithm in IEEE 802.15.4 networks. *IEEE Communications Letters*, 12(6):420–422, June 2008.

[51] Y. He, Q.-G. Wang, C. Lin, and M. Wu. Delay-range-dependent stability for systems with time-varying delay. *Automatica*, 43:371–376, 2007.

[52] S. Hemminger. Network emulation with netem. In *Linux conf Au*, 2005.

[53] J. Hespanha, P. Naghshtabrizi, and Y. Xu. A survey of recent results in networked control systems. *Proc. of IEEE Special Issue on Technology of Networked Control Systems*, 95(1):137–162, 2007.

[54] J. Hespanha and A. Teel. Stochastic impulsive systems driven by renewal processes. In *International Symposium on the Mathematical Theory of Networks and Systems*, 2006.

[55] J. P. Hespanha, A. Ortega, and L. Vasudevan. Towards the control of linear systems with minimum bit-rate. In *International Symposium on the Mathematical Theory of Networks and Systems*, 2002.

[56] S. Hirche and M. Buss. Transparent data reduction in networked telepresence and teleaction systems part ii: Time-delayed communication. *PRESENCE: Teleoperators and Virtual Environments*, 16(5):532–542, 2007.

[57] S. Hirche, C.-C. Chen, and M. Buss. Performance oriented control over networks: switching controllers and switched time delay. In *45th IEEE Conference on Decision and Control*, San Diego, US, 2006.

[58] S. Hirche, C.-C. Chen, and M. Buss. Performance oriented control over networks: switching controllers and switched time delay. *Asian Journal of Control*, (1):24–33, Jan. 2008.

[59] D. Hristu-Varsakelis and W. S. Levine. *Handbook of networked and embedded control systems*. Birkhäuser, Boston, 2005.

[60] B. Hu and A. N. Michel. Robustness analysis of digital feedback control systems with time-varying sampling periods. *Journal of the Franklin Institute*, (337):117–130, 2000.

[61] S. Hutchison, G.D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, 12(5):651–670, 1996.

[62] O. C. Imer, S. Yüksel, and T. Başar. Optimal control of LTI systems over unreliable communication links. *Automatica*, 42(9):1429 – 1439, 2006.

[63] K. Ji and W.-J. Kim. Stochastic optimal control and network co-design for networked control systems. *International Journal of Control, Automation and Systems*, 5(5):515–525, 2007.

[64] Y. Ji and H. J. Chizeck. Controllability, stabilizability, and continuous-time markovianjump linear quadratic control. *IEEE Transactions on Automatic Control*, 35(7):777–778, July 1990.

[65] S. Johannessen. Time synchronization in a local area network. *IEEE Control Systems Magazine*, 24(2):61–69, Apr 2004.

[66] C.Y. Jung, H.Y. Hwang, D.K. Sung, and G.U. Hwang. Enhanced Markov chain model and throughput analysis of the slotted CSMA/CA for IEEE 802.15.4 under unsaturated traffic conditions. *IEEE Transactions on Vehicular Technology*, 58(1):473–478, Jan. 2009.

[67] M. Kijima. *Markov proceses for stochastic modeling.* Chapman & Hall, London, 1997.

[68] S. A. Koubias and G. D. Papadopoulos. Modern fieldbus communication architectures for real-time industrial applications. *Computers in Industry*, 26(3):243 – 252, 1995.

[69] N. N. Krasovskii and E. A. Lidskii. Analysis design of controller in systems with random attributes-part 1. *Automatic remote control*, 22:1021–1025, 1961.

[70] N. N. Krasovskii and E. A. Lidskii. Analysis design of controller in systems with random attributes-part 2. *Automatic remote control*, 22:1141–1146, 1961.

[71] H. J. Kushner. *Stochastic stability and control.* Academic Press Inc., New York, 1967.

[72] W. H. Kwon, Y. S. Lee, and S. H. Han. General receding horizon control for linear time-delay systems. *Automatica*, 40(9):1603–1161, 2004.

[73] F.-L. Lian, J. Moyne, and D. Tilbury. Network design consideration for distributed control systems. *IEEE Transactions on Control Systems Technology*, 10(2):297–307, Mar 2002.

[74] F.-L. Lian, J. R. Moyne, and D. M. Tilbury. Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet. *IEEE Control Systems Magazine*, 21(1):66–83, 2001.

[75] F.-L. Lian, J. K. Yook, D. M. Tilbury, and J. Moyne. Network architecture and communication modules for guaranteeing acceptable control and communication performance for networked multi-agent systems. *IEEE Transactions on Industrial Electronics*, 2(1):12–24, 2006.

[76] D. Liberzon and J. P. Hespanha. Stabilization of nonlinear systems with limited information feedback. *IEEE Transaction on Automatic Control*, 50(6):910–915, 2005.

[77] L. Litz, T. Gabriel, M. Groß, and O. Gabel. Networked control systes (NCS) - stand und ausblick. *Automatisierungstechnik*, 56(4-19), 2008.

[78] G. P. Liu, J. X. Mu, D. Rees, and S. C. Chai. Design and stability of networked control systems with random communication time delay using the modified MPC. *International Journal of Control*, 79(4):288–297, 2006.

[79] X. Liu and A. Goldsmith. Wireless network design for distributed control. In *43rd IEEE Conference on Decision and Control*, 2004.

[80] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[81] D.G Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 74(1):59–73, 2007.

[82] R. Lozano, P. Castillo, P. Carcia, and A. Dzul. Robust prediction-based control for unstable delay systems: Application to the yaw control of a mini-helicopter. *Automatica*, 40(4):603–612, 2004.

[83] M. S. Mahmoud and N. F. Al-Muthairi. Design of robust controller for time-delay systems. *IEEE Transactions on Automatic Control*, 39:995–999, 1984.

[84] M. S. Mahmoud and P. Shi. *Methodologies for control of jump time-delay systems.* Kluwer Academic Publishers, Boston, 2003.

[85] X. Mao. Exponential stability of stochastic delay interval systems with Markovian switching. *IEEE Transactions on Automatic Control*, 47(10):1604–1612, 2002.

[86] E. Marchand and F. Chaumette. Virtual visual servoing: A framework for realtime augmented reality. *Computer Graphics Forum*, 21(3):289–298, 2002.

[87] Michel Mariton. *Jump linear systems in automatic control.* Marcel Dekker, Inc, New York, 1990.

[88] P. Martí, J. Yépez, M. Velasco, R. Villà, and J. M. Fuertes. Managing quality-of-control in network-based control systems by controller and message scheduling co-design. *IEEE Transactions on Industrial Electronics*, 51(6):1159–1167, 2004.

[89] T. Matiakis. *Stability and performance of networked control systems with distributed controller approach.* PhD thesis, Technische Universität München, 2009.

[90] P. Metzner, E. Dittmer, T. Jahnke, and Ch. Schütte. Generator estimation of Markov jump processes. *Journal of Computational Physics*, 227(1):353 – 375, 2007.

[91] P. Metzner, I. Horenko, and C. Schütte. Generator estimation of Markov jump processes based on incomplete observations nonequidistant in time. *Physical Review E*, 76(6):066702 [1–8], 2007.

[92] L. Montestruque and P. Antsaklis. Stability of model-based networked control systems with time-varying transmission times. *IEEE Transactions on Automatic Control*, 49(9):1562–1572, 2004.

[93] P. Naghshtabrizi, J. P. Hespanha Hespanha, and A. R. Teel. Exponential stability of impulsive systems with application to uncertain sampled-data systems. *Systems & Control Letters*, 57(5):378–385, 2008.

[94] G. N. Nair and R. J. Evans. Exponential stabilisability of finite-dimensional linear systems with limited data rates. *Automatica*, 39:585–593, 2003.

[95] G. N. Nair, R. J. Evans, I. M. Y. Mareels, and W. Moran. Topological feedback entropy and nonlinear stabilization. *IEEE Transactions on Automatic Control*, 49(9):1585–1597, 2004.

[96] G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans. Feedback control under data rate constraints: An overview. *Proceedings of the IEEE*, 95:108–137, 2007.

[97] D. Nešić and A. R. Teel. Input-to-state stability of networked control systems. *Automatica*, 40:2121–2128, 2004.

[98] H. M. Newman. Integrating building automation and control products using the BACnet protocol. *ASHRAE journal*, 38(11):36–42, 1996.

[99] R. Niemann. *Hardware/Software Co-design for Data Flow Dominated Embedded Systems.* Kluwer Academic Publishers, Boston, 1998.

[100] J. Nilsson. *Real-time control systems with delay.* PhD thesis, Lund Institute of Technology, 1998.

[101] P. G. Otanez, J. T. Parrott, J. R. Moyne, and D. M. Tilbury. The implications of ethernet as a control network. In *Proceedings of Global Powertrain Conference*, 2002.

[102] N. Özdemir and S. Townley. Integral control by variable sampling based on steady-state data. *Automatica*, 39:135–140, 2003.

[103] Athanasios Papoulis and S. Unnikrisha Pillai. *Probability, Random Variables and Stochastic Processes.* McGraw-Hill, New York, 4th edition, 2002.

[104] V.N. Phat. Robust stability and stabilizability of uncertain linear hybrid systems with state delays. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 52(2):94–98, Feb. 2005.

[105] N. J. Ploplys, P. A. Kawka, and A. G. Alleyne. Closed-loop control over wireless networks. *IEEE Control Systems Magazine*, 24(3):58–71, 2004.

[106] A. Ray and Y. Galevi. Integrated communication and control systems: Part II-design considerations. *Journal of Dynamic Systems, Measurements, and Control*, 110:374–381, 1988.

[107] J.-P. Richard. Time-delay systems: an overview of some recent advances and open problems. *Automatica*, 39:1667–1694, 2003.

[108] S. M. Ross. *Applied probability models with optimization applications.* Holden Day, San Francisco, CA, 1970.

[109] Sheldon M. Ross. *Stochastic processes.* John Wiley & Sons, Inc., 2 edition, 1996.

[110] A. Sala. Computer control under time-varying sampling period: An LMI gridding approach. *Automatica*, 41:2007–2082, 2005.

[111] G. Schickhuber and O. McCarthy. Distributed fieldbus and control network systems. *Computing & Control Engineering Journal*, 8(1):21–32, Feb 1997.

[112] D.C. Schuurman and D.W. Capson. Robust direct visual servo using network-synchronized cameras. *IEEE Transactions on Robotics and Automation*, 20(2):319–334, April 2004.

[113] L. Sciavicco and B. Siciliano. *Modelling and Control of Robot Manipulators.* Springer, London, 2 edition, 2001.

[114] P. Seiler. *Coordinated control of unmanned aerial vehicles.* PhD thesis, Univ. California, Berkeley, 2001.

[115] L. Socha. Some remarks about stochastic controllability for linear composite systems. *IEEE Transactions on Automatic Control*, 29(1):60–63, 1984.

[116] Y. S. Suh. Stability and stabilization of nonuniform sampling systems. *Automatica*, 44:3222–3226, 2008.

[117] M. Sun, J. Lam, S. Xu, and Y. Zou. Robust exponential stabilization for Markovian jump systems with mode-dependent input delay. *Automatica*, 43:1799–1807, 2007.

[118] B. Sundararaman, U. Buy, and A. D. Kshemkalyani. Clock synchronization for wireless sensor networks: a survey. *Ad Hoc Networks*, 3(3):281 – 323, 2005.

[119] V. Suplin, E. Fridman, and U. Shaked. Sampled-data H_∞ control and filtering: Nonuniform uncertain sampling. *Automatica*, 43:1072–1083, 2007.

[120] M. Tabbara and D. Nešić. Input–output stability of networked control systems with stochastic protocols and channels. *IEEE Transaction on Automatic Control*, 53(5):1160–1175, 2008.

[121] P. L. Tang and C. W. de Silva. Compensation for transmission delays in an Ethernet-based control network using variable-horizon predictive control. *IEEE Transactions on Control Systems Technology*, 14(4):707–718, 2006.

[122] S. Tatikonda and S. Mitter. Control under communication constraints. *IEEE Transactions on Automatic Control*, 49(7):1056–1068, 2004.

[123] Y. Tipsuwan and M.-Y. Chow. Control methodologies in networked control systems. *Control Engineering Practice*, 11:1099–1111, 2003.

[124] Y. Tipsuwan and M.-Y. Chow. Gain scheduler middleware: a methodology to enable existing controllers for networked control and teleoperation – part i: networked control. *IEEE Transactions on Industrial Electronics*, 51(6):1218–1227, 2004.

[125] M. Ueberle. *Design, Control, and Evaluation of a Family of Kinematic Haptic Interfaces.* PhD thesis, Technische Universität München, Institute of automatic control engineering, 2006.

[126] M. Vidyasagar. *Nonlinear systems analysis.* SIAM, Philadelphia, 2002.

[127] G. C. Walsh, O. Beldiman, and L. G. Bushnell. Asymptotic behavior of nonlinear networked control system. *IEEE Transactions on Automatic Control*, 46(7):1093–1097, 2001.

[128] G. C. Walsh and H. Ye. Scheduling of networked control systems. *IEEE Control Systems Magazine*, 21(1):57–65, February 2001.

[129] G. C. Walsh, H. Ye, and L. G. Bushnell. Stability analysis of networked control systems. *IEEE Transactions and Automatic Control*, 10(3):438–446, 2002.

[130] F.Y. Wang and D. Liu. *Networked control systems: theory and applications.* Springer-Verlag, London, 2008.

[131] W. Wei, B. Wang, and D. Towsley. Continuous-time hidden Markov models for network performance evaluation. *Performance Evaluation*, 49(1-4):129 – 146, 2002.

[132] J. D. Wheelis. Process control communications: Token bus, CSMA/CD, or token ring? *ISA Transactions*, 32(2):193–198, 1993.

[133] W. H. Wolf. Hardware-software co-design of embedded systems. *Proceedings of the IEEE*, 82(7):967–989, 1994.

[134] H. Wu, C.-C. Chen, J. Feng, K. Kühnlenz, and S. Hirche. A switching control law for a networked visual servo control system. In *IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, 2010.

[135] L. Xiao, A. Hassibi, and Jonathan P. How. Control with random communication delays via a discrete-time jump system approach. In *Proceedings of the American Control Conference*, Chicago, Illinois, June 2000.

[136] L. Xiao, M. Johansson, H. Hindi, S. Boyd, and A. Goldsmith. Joint optimization of communication rates and linear systems. *IEEE Transactions on Automatic Control*, 48(1):148–153, Jan. 2003.

[137] F. Yang, Z. Wang, Y. S. Hung, and Mahbub Gani. H_∞ control for networked control systems with random communication delays. *IEEE Transactions on Automatics Control*, 51(3):511–518, March 2006.

[138] T.-C. Yang. Networked control system: a brief survey. *IEE Proceedings - Control Theory and Applications*, 153(4):403–412, 2006.

[139] K. Yogesan, S. Kumar, L. Goldschmidt, and J. Cuadros. *Teleophthalmology.* Springer-Verlag, Berlin Heidelberg, 2006.

[140] C. Yuan and X. Mao. Robust stability and controllability of stochastic differential delay equations with Markovian switching. *Automatica*, 40:343–354, 2004.

[141] L. Zhang and D. Hristu-Varsakelis. Communication and control co-design for networked control systems. *Automatica*, 42:953–958, 2006.

[142] L. Zhang, Y. Shi, T. Chen, and B. Huang. A new method for stabilization of networked control systems with random delays. *IEEE Transactions on Automatic Control*, 20(8):117–1181, 2005.

[143] W. Zhang, M. S. Branicky, and S. M. Philips. Stability of network control systems. *IEEE Control Systems Magazine*, 21(1):84–99, February 2001.

[144] Y.B. Zhao, G.P. Liu, and D. Rees. Integrated predictive control and scheduling co-design for networked control systems. *IET Control Theory* & *Applications*, 2(1):7–15, 2008.

[145] H. Zimmermann. OSI reference model — the ISO model of architecture for open systems interconnection. *IEEE Transactions on Communications*, 28(4):425–432, 1980.