Lecture Notes
in Control and Information Sciences 387

Yuanqing Xia, Mengyin Fu, and Peng Shi

Analysis and Synthesis of Dynamical Systems with Time-Delays

Springer

Authors

Prof. Yuanqing Xia
Department of Automatic Control
Beijing Institute of Technology
Beijing 100081
China
E-mail: xia_yuanqing@bit.edu.cn

Prof. Peng Shi
School of Science and Engineering
Victoria University
P.O. Box 14428
Melbourne, VIC 8001
Australia
E-mail: peng.shi@vu.edu.au

Prof. Mengyin Fu
Department of Automatic Control
Beijing Institute of Technology
Beijing 100081
China
E-mail: fumy@bit.edu.cn

ISBN 978-3-642-02695-9 e-ISBN 978-3-642-02696-6

DOI 10.1007/978-3-642-02696-6

Lecture Notes in Control and Information Sciences ISSN 0170-8643

Library of Congress Control Number: Applied for

Typeset & Cover Design: Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed in acid-free paper

5 4 3 2 1 0

springer.com

For our families and friends with gratitude

Preface

Time-delay occurs in many dynamical systems such as biological systems, chemical systems, metallurgical processing systems, nuclear reactor, long transmission lines in pneumatic, hydraulic systems and electrical networks. Especially, in recent years, time-delay which exists in networked control systems has brought more complex problem into a new research area. Frequently, it is a source of the generation of oscillation, instability and poor performance. Considerable effort has been applied to different aspects of linear time-delay systems during recent years. Because the introduction of the delay factor renders the system analysis more complicated, in addition to the difficulties caused by the perturbation or uncertainties, in the control of time-delay systems, the problems of robust stability and robust stabilization are of great importance.

This book presents some basic theories of stability and stabilization of systems with time-delay, which are related to the main results in this book. More attention will be paid on synthesis of systems with time-delay. That is, sliding mode control of systems with time-delay; networked control systems with time-delay; networked data fusion with random delay.

This book contains three parts:

Part I: basic theory of systems with time-delay. There are three chapters in this part. In Chapter 1, recent developments on stability analysis of systems with time-delay are presented. Traditional methods are introduced for stability analysis of delayed continuous systems. In Chapter 2, new stability conditions for continuous-time systems with interval time-varying delay are given. In Chapter 3, new stability and stabilization conditions for discrete systems with time-delay are derived. Part II: sliding mode control of systems with time-delay are investigated. Both delay-independent and delay-dependent conditions for existence of sliding mode surface and controller are given in Chapter 4 and Chapter 5. In Chapter 6, robust adaptive control for uncertain systems with time-delay is presented. In Chapter 7, sliding mode control for systems with input delay are presented. In Chapter 8, sliding mode control for systems with state delay and input delay is given. In Chapter 9, based on the delta operator approach, the reaching law proposed is able to reduce the chattering, and possesses the desired characteristics of robustness and good performance. The proposed method can unify some previous related results of the continuous and discrete sliding mode control for systems into the delta operator systems framework. In Chapter 10, active disturbance rejection controller which consists of the tracking differentiator, the extended state observer and the nonlinear proportional derivative (PD) controller, is proposed to deal with both robust stability and performance specifications for a multivariable process with time delay in the input. Since parts of the idea of ADRC originate from sliding mode control, Chapter 10 is also included in Part II.

Part III: networked control systems with time-delay are studied. Much research work has been done in networked control systems, most work has ignored a very important feature of networked control systems. This feature is that the communication networks can transmit a packet of data at the same time, which is not done in traditional control systems. The method in this part makes full use of this network feature and proposes a new networked control scheme–networked predictive control, which can overcome the effects caused by network delay. Furthermore, three different ways to choose control input are discussed and the performances have been compared and analyzed in Chapter 11. The performance analysis of systems with random delay in feedback channel, forward channels and both feedback and forward channels are presented in Chapter 12, Chapter 13 and Chapter 14, respectively. In Chapter 15, a practical architecture and some algorithms for the networked data fusion system with packets losses and variable delays are presented. The data fusion system proposed in this chapter is based on the well-known Federated Filter. The algorithm in this chapter is optimal, and the system considered in this chapter is equi-interval sampled and there is completely no prior information about when packet delays occur. And also, the advantages of the proposed algorithm are analyzed.

We would like to acknowledge the collaborations with Professor Guo-Ping Liu on some of the research works reported in the monograph and Dr. Jinhui Zhang for his help in checking and correcting some errors in this monograph.

Haidian District, Beijing, China, *Yuanqing Xia*
Haidian District, Beijing, China, *Mengyin Fu*
Trefforest, Glamorgan, UK, *Peng Shi*
December 2008

Symbols and Acronyms

C	field of complex numbers
R	field of real numbers
R^n	n-dimensional real Euclidean space
$R^{n\times m}$	space of $n \times m$ real matrices
I	identity matrix
$0_{n\times m}$	zero matrix of dimension $n \times m$
A^T	transpose of matrix A
A^{-1}	inverse of matrix A
$A > 0$	symmetric positive definite
$A \geq 0$	symmetric positive semi-definite
$A < 0$	symmetric negative definite
$A \leq 0$	symmetric negative semi-definite
$\text{tr}(A)$	trace of matrix A
$\text{rank}(A)$	rank of matrix A
$\det(A)$	determinant of matrix A
$\text{diag}(X_1, X_2, \cdots, X_m)$	diagonal matrix with X_i as its ith diagonal element
$\lambda_{\min}(A)$	minimum eigenvalue of matrix A
$\lambda_{\max}(A)$	maximum eigenvalue of matrix A
$\sigma_{\min}(A)$	minimum singular value of matrix A
$\sigma_{\max}(A)$	maximum singular value of matrix A
LMI	linear matrix inequality
$A \perp B$	$E\{AB^T\} = 0$
$\text{sgn}(x)$	the sign of x
$\|x\|$	absolute value (or modulus) of x
$\|x\|$	Euclidean norm
$\|P\|$	induced norm $\sup_{\|x\|=1} \|Px\|$
$\forall$	for all
$\in$	belong to
$\sum$	sum
lim	limit
$\rightarrow$	tend to, or mapping to (case sensitive)

$\otimes$	matrix Kronecker product
$\mathcal{E}\{\cdot\}$	mathematical expectation operator
δ_{kj}	Kronecker delta function
sup	supremum
inf	infimum

Contents

Part II: Sliding Mode Control (SMC) for Systems with Time-Delay

Chapter 1
Recent Results on Analysis of Systems with Time-Delay

1.1 Introduction

It is well known that time delays as a source of the generation of oscillation and a source of instability are frequently encountered in various engineering systems such as long transmission lines in pneumatic systems, nuclear reactors, rolling mills, hydraulic systems and manufacturing processes. Therefore, the problem of stability analysis and control of time-delay systems has attracted much attention, and considerable effort has been applied to different aspects of linear time-delay systems during recent years [153], [128], [141], [64], [112], [140], [206], [66], [65], [204], [87], [39], [214], [199], [125], [157], [155], [156], [158], [159], [193], [192]. It has been proven that the linear time delay system $\dot{x}(t) = -bx(t-\tau), b > 0, \tau > 0$ is stable if $\tau < \frac{\pi}{2b}$, and the system is unstable if τ is too large. Existing criteria for asymptotic stability of time-delay systems can be classified into two types, that is, delay-independent stability and delay-dependent stability; the former does not include any information on the size of delay while the latter employs such information. It is known that delay-dependent stability conditions are generally less conservative than delay-independent ones especially when the size of the delay is small.

In the literature, many kinds of approaches have been proposed to obtain delay-dependent stability conditions based on linear matrix inequality (LMI) approach. LMI approach is the most popular and has played an important role, which can be efficiently solved via standard numerical software. The main objective of this chapter is to review the recent development on the LMI techniques in deriving both delay-independent and delay-dependent stability results for systems with constant and time-varying delays. Our attention will be paid on issues concerned with the conservatism of the results. First, some basic results on delay-independent will be recalled. Then, we will pay more attention on discussing delay-dependent results, and four approaches will be introduced, they are: Model Transformation approach, Bounding Techniques, Descriptor System approach and Free-Weighting Matrix approach.

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 3–16.
springerlink.com

Then, the more recent results on delay-dependent stability conditions will be introduced. It will be seen that, on one hand, the free-weighting matrix approach is effective in reducing conservatism in existing delay dependent results. On the other hand, since many matrix variables will be introduced, the stability criteria obtained are very complicated. Thus, based on Jensen's inequality, some new delay-dependent stability conditions are provided recently. The advantages associated to the new approach are that it involves fewer matrix variables and has less conservatism.

The chapter is organized as follows. Section 1.2 presents the problem formulation and some preliminaries. Recent developments in LMI techniques for deriving delay-independent stability conditions are given in Section 1.3. Four approaches have been introduced for developing delay-dependent stability conditions in Section 1.4. Some conclusion remarks are given in Section 1.5.

1.2 Problem Formulation

Consider the following two kinds of time-delay systems:

$$(\mathcal{S}_1) : \dot{x}(t) = Ax(t) + A_d x(t-d) \tag{1.1}$$

$$x(t) = \phi(t), \quad \forall t \in [-d, 0] \tag{1.2}$$

and

$$(\mathcal{S}_2) : \dot{x}(t) = Ax(t) + A_d x(t-d(t)) \tag{1.3}$$

$$x(t) = \phi(t), \quad \forall t \in [-d_M, 0] \tag{1.4}$$

where $x(t) \in R^n$ is the state; $\phi(t)$ is the continuous initial condition. In system $(\mathcal{S}_1)$, the scalar $d > 0$ denotes the constant delay, while for the time-varying delay $d(t)$ in system $(\mathcal{S}_2)$, generally, there are three assumptions given as follows:

Assumption 1.2.1 *$d(t)$ is a continuous function satisfying $0 \le d(t) \le d_M$, where d_M is a constant positive scalar representing the upper bound of $d(t)$.*

Assumption 1.2.2 *$d(t)$ is a differentiable function satisfying $0 \le d(t) \le d_M$ and $\dot{d}(t) \le \rho < 1$, where d_M is given in Assumption 1.2.1, and ρ is the upper bound of $\dot{d}(t)$.*

Assumption 1.2.3 *$d(t)$ is a differentiable function satisfying $0 \le d_m \le d(t) \le d_M$ and $\dot{d}(t) \le \rho$, where d_m and d_M are given positive scalars representing the lower and upper bounds of interval time-varying delay $d(t)$, and ρ is given in Assumption 1.2.2.*

Remark 1.1. The difference between Assumption 1.2.2 and Assumption 1.2.3 is that the constraint $\rho < 1$ is not required in Assumption 1.2.3.

1.3 Delay-Independent Conditions

For the time-delay system $(\mathcal{S}_1)$, the initial stability condition is provided in [177].

Theorem 1.2. *The time-delay system $(\mathcal{S}_1)$ is asymptotically stable if there exist matrices $P > 0$ and $Q > 0$ such that*

$$\begin{bmatrix} A^T P + PA + Q & PA_d \\ A_d^T P & -Q \end{bmatrix} < 0. \tag{1.5}$$

In Theorem 1.2, the following Lyapunov-Krasovskii functional is used:

$$V(t) = x^T(t)Px(t) + \int_{t-d}^{t} x^T(s)Qx(s)ds.$$

For the time-delay system $(\mathcal{S}_2)$, the counterpart of Theorem 1.2 is given in the following theorem under Assumption 1.2.2.

Theorem 1.3. *Under Assumption 1.2.2, the time-delay system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$ and $Q > 0$ such that*

$$\begin{bmatrix} A^T P + PA + Q & PA_d \\ A_d^T P & -(1-\rho)Q \end{bmatrix} < 0. \tag{1.6}$$

In Theorem 1.3, the following Lyapunov-Krasovskii functional is used:

$$V(t) = x^T(t)Px(t) + \int_{t-d(t)}^{t} x^T(s)Qx(s)ds.$$

1.4 Delay-Dependent Conditions

It is well known that delay-independent stability conditions are simpler to apply, while delay-dependent stability conditions are less conservative especially in the case when the time delay is small. Therefore, recent years have witnessed a resurgence of research interests in developing delay-dependent stability conditions of time-delay systems.

1.4.1 Basic Approaches

Approach I: Model Transformation Approach

By using the Newton-Leibniz formula, we have $x(t-d) = x(t) - \int_{t-d}^{t} \dot{x}(s)ds$, thus, system $(\mathcal{S}_1)$ can be rewritten as following two forms

$$\dot{x}(t) = (A + A_d)x(t) - A_d \int_{t-d}^{t} [Ax(s) + A_d x(s-d)]ds \tag{1.7}$$

$$\dot{x}(t) = (A + A_d)x(t) - A_d \int_{t-d}^{t} \dot{x}(s)ds. \tag{1.8}$$

Accordingly, for system $(\mathcal{S}_2)$, we have

$$\dot{x}(t) = (A + A_d)x(t) - A_d \int_{t-d(t)}^{t} [Ax(s) + A_d x(s-d(s))]ds \tag{1.9}$$

$$\dot{x}(t) = (A + A_d)x(t) - A_d \int_{t-d(t)}^{t} \dot{x}(s)ds. \tag{1.10}$$

It should be pointed out that (1.7)-(1.10) are transformed from the time-delay system in (1.1) or (1.3) by using the Newton-Leibniz formula, respectively. Based on the transformed systems (1.7)-(1.10), a lot of delay-dependent stability results have been obtained, see [18, 19, 113, 123] for example, and the references cited therein. However, the drawback associated to this approach is that all of transformed systems are not equivalent to (1.1) or (1.3).

Approach II: Bounding techniques

It is well known that some weighted cross products may arise in the analysis of the delay-dependent stability problem, thus, getting better bounds of them will play a key role in reducing conservatism. Note that in [18, 19, 113], the following inequality was used,

$$-2a^T b \leq a^T X a + b^T X^{-1} b$$

where $a \in R^n$, $b \in R^n$ and $X > 0$.

To reduce the conservativeness, Park presents the following lemma.

Lemma 1.4. *(Park's Inequality) Assume that $a(s) \in R^n$, and $b(s) \in R^n$ are given for $s \in \Omega$. Then, for any $X > 0$ and any matrix M, the following inequality holds*

$$-2\int_{\Omega} a(s)^T b(s)ds \leq \int_{\Omega} \begin{bmatrix} a(s) \\ b(s) \end{bmatrix}^T \begin{bmatrix} X & XM \\ M^T X & (2,2) \end{bmatrix} \begin{bmatrix} a(s) \\ b(s) \end{bmatrix} ds$$

where $(2,2) = (M^T X + I)X^{-1}(M^T X + I)^T$.

Then, by using this inequality, an improved delay-dependent stability condition was reported in [145], which is stated in the following theorem.

Theorem 1.5. [145] The time-delay system $(\mathcal{S}_1)$ is asymptotically stable for any delay d, $0 < d \leq d_M$ if there exist matrices $P > 0$, $Q > 0$, $V > 0$, and W such that

$$\begin{bmatrix} \Psi & -W^T A_d & A^T A_d^T V & d_M(W^T + P) \\ * & -Q & A_d^T A_d^T V & 0 \\ * & * & -V & 0 \\ * & * & * & -V \end{bmatrix} < 0$$

where $\Psi = (A + A_d)^T P + P(A + A_d) + W^T A_d + A_d^T W + Q$.

It should be noted that Park's inequality was further improved in [138] by using the following Moon's inequality.

Lemma 1.6. *(Moon's Inequality) Assume that $a(s) \in R^n$, $b(s) \in R^m$ and $N(s) \in R^{n\times m}$ are given for $s \in \Omega$. Then, for any $X > 0$, $Z > 0$ and any matrix Y, the following inequality holds*

$$-2\int_{\Omega} a(s)^T N(s) b(s) ds \leq \int_{\Omega} \begin{bmatrix} a(s) \\ b(s) \end{bmatrix}^T \begin{bmatrix} X & Y - N(s) \\ Y^T - N(s)^T & Z \end{bmatrix} \begin{bmatrix} a(s) \\ b(s) \end{bmatrix} ds$$
$$\begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \geq 0.$$

Theorem 1.7. [138] The time-delay system $(\mathcal{S}_1)$ is asymptotically stable for any delay d, $0 < d \leq d_M$ if there exist matrices $P > 0$, $Q > 0$, X, Y and Z such that

$$\begin{bmatrix} (1,1) & PA_d - Y & d_M A^T Z \\ * & -Q & d_M A_d^T Z \\ * & * & d_M Z \end{bmatrix} < 0$$
$$\begin{bmatrix} X & Y \\ * & Z \end{bmatrix} \geq 0$$

where $(1,1) = PA + A^T P + d_M X + Y + Y^T + Q$.

Based on Moon's inequality, a great amount of delay-dependent conditions for various system with time delay were derived out, see [50, 209, 115, 144] for example, and the references therein.

By applying Moon's inequality, in [216], the following integral inequality was given for system with time-varying delay.

Lemma 1.8. *Let $x(t) \in R^n$ be a vector-valued function with first-order continuous-derivative entries. Then, the following integral inequality holds for any matrices M_1, $M_2 \in R^{n\times n}$ and $X > 0$,*

$$-\int_{t-d(t)}^{t} \dot{x}^T(s)X\dot{x}(s)ds \leq \xi^T(t)\begin{bmatrix} M_1^T + M_1 & -M_1^T + M_2 \\ -M_1 + M_2^T & -M_2^T - M_2 \end{bmatrix}\xi(t)$$
$$+d(t)\xi^T(t)\begin{bmatrix} M_1^T \\ M_2^T \end{bmatrix} X^{-1}\begin{bmatrix} M_1 \ M_2 \end{bmatrix}\xi(t)$$

where $\xi(t) = \begin{bmatrix} x(t) \\ x(t-d(t)) \end{bmatrix}$.

Besides of Park's inequality and Moon's inequality, another one named Jensen's inequality, which is stated as follows.

Lemma 1.9. (Jensen's Inequality) [67] For any constant matrix $M > 0$, scalars $b > a$, vector function $w : [a, b] \to R^m$ such that the integrations in the following are well defined, then

$$(b-a)\int_a^b w(s)^T M w(s)ds \geq \left[\int_a^b w(s)ds\right]^T M \left[\int_a^b w(s)ds\right].$$

Lemma 1.10. [74] For any constant matrix $W \in R^{n\times n}$, $W = W^T > 0$, scalar $\gamma > 0$, then

$$-\gamma\int_{t-\gamma}^{t} \dot{x}^T(s)W\dot{x}(s)ds \leq \begin{bmatrix} x(t) \\ x(t-\gamma) \end{bmatrix}^T \begin{bmatrix} -W & W \\ W & -W \end{bmatrix}\begin{bmatrix} x(t) \\ x(t-\gamma) \end{bmatrix}.$$

Theorem 1.11. [62] The time-delay system $(\mathcal{S}_1)$ is asymptotically stable for any delay d, $0 < d \leq d_M$ if there exist matrices $P > 0$, $Q > 0$ and $Z > 0$ such that

$$\begin{bmatrix} PA + A^T P + Q - Z & PA_d + Z & d_M A^T Z \\ * & -Q - Z & d_M A_d^T Z \\ * & * & -Z \end{bmatrix} < 0.$$

Theorem 1.12. [62] The time-delay system $(\mathcal{S}_2)$ is asymptotically stable for any delay $d(t)$ satisfying Assumption 1.2.1 if there exist matrices $P > 0$ and $Z > 0$ such that

$$\begin{bmatrix} PA + A^T P - Z & PA_d + Z & d_M A^T Z \\ * & -Z & d_M A_d^T Z \\ * & * & -Z \end{bmatrix} < 0.$$

The Jensen's inequality has been used to deal with different kinds of time-delay systems in order to obtain delay-dependent results, see [67, 183, 74] for example, and the reference therein.

Approach III: Descriptor system approach

It is well known the transformed model is not equivalent to the original time-delay system, and may lead to conservatism. In order to reduce such potential conservatism, a method based on descriptor systems has been introduced in [43] to perform the stability analysis and controller design for system with time-varying delays. This method uses a descriptor system model to derive delay-dependent stability conditions, which is equivalent to the original time-delay system.

To introduce the descriptor system approach, we rewrite system $(\mathcal{S}_2)$ in the following form:

$$\begin{aligned}\dot{x}(t) &= y(t)\\ 0 &= -y(t) + (A + A_d)x(t) - A_d \int_{t-d(t)}^{t} y(s)ds\end{aligned}$$

which can be rewritten as

$$E\dot{\bar{x}}(t) = \bar{A}\bar{x}(t) + \bar{A}_d \int_{t-d(t)}^{t} y(s)ds$$

where $\bar{x}(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$, $E = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}$, $\bar{A} = \begin{bmatrix} 0 & I \\ A + A_d & -I \end{bmatrix}$ and $\bar{A}_d = \begin{bmatrix} 0 \\ A_d \end{bmatrix}$.

The Lyapunov-Krasovskii functional associated to this approach can be chosen as

$$V(t) = \bar{x}^T(t)EP\bar{x}(t) + \int_{t-d(t)}^{t} x^T(s)Qx(s)ds + \int_{-d_M}^{0} \int_{t+\theta}^{t} y^T(s)Ry(s)dsd\theta$$

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

Then, the following theorem can be obtained.

Theorem 1.13. [45] Under Assumption 1.2.2, the time delay system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P_1 > 0$, P_2, P_3, $R_1 > 0$, S_1, Y_{11}, Y_{12}, Z_{11}, Z_{12} and Z_{13}, such that

$$\begin{bmatrix} \Omega + d_M Z_1 & P^T \begin{bmatrix} 0 \\ A_d \end{bmatrix} - Y_1^T \\ \begin{bmatrix} 0 & A_d^T \end{bmatrix} P - Y_1 & -(1-\rho)S_1 \end{bmatrix} < 0$$

and

$$\begin{bmatrix} R_1 & Y_1 \\ Y_1^T & Z_1 \end{bmatrix} \geq 0$$

where

$$\Omega = P^T \begin{bmatrix} 0 & I \\ A & -I \end{bmatrix} + \begin{bmatrix} 0 & I \\ A & -I \end{bmatrix}^T P$$

$$+\begin{bmatrix} S_1 & 0 \\ 0 & d_M R_1 \end{bmatrix} + \begin{bmatrix} Y_1 \\ 0 \end{bmatrix} + \begin{bmatrix} Y_1 \\ 0 \end{bmatrix}^T$$
$$Y_1 = \begin{bmatrix} Y_{11} \ Y_{12} \end{bmatrix}, Z_1 = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{13} \end{bmatrix}.$$

Corollary 1.14. [45] Under Assumption 1.2.1, the time delay system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P_1 > 0$, P_2, P_3, $R_1 > 0$, S_1, Z_{11}, Z_{12} and Z_{13}, such that

$$\Psi_1 + d_M Z_1 < 0$$

and

$$\begin{bmatrix} R_1 & [0 \ \ A_d^T]P^T \\ * & Z_1 \end{bmatrix} \geq 0$$

where

$$\Psi_1 = P^T \begin{bmatrix} 0 & I \\ A & -I \end{bmatrix} + \begin{bmatrix} 0 & I \\ A & -I \end{bmatrix}^T P + \begin{bmatrix} 0 & 0 \\ 0 & d_M R_1 \end{bmatrix}, Z_1 = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12}^T & Z_{13} \end{bmatrix}.$$

By the descriptor system approach together with the following functional

$$V(t) = x(t)^T P x(t) + \int_0^{d_M} (d_M - \sigma)\dot{x}^T(t-\sigma) X_{33} \dot{x}(t-\sigma) d\sigma$$
$$\int_0^t \int_{\sigma - d(t)}^{\sigma} e^T X e ds d\sigma + \int_{t-d(t)}^t x^T(s) Q x(s) ds$$

where $e = \begin{bmatrix} x(\beta) \\ x(\sigma - d) \\ \dot{x}(s) \end{bmatrix}$ and $X = \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{12}^T & X_{22} & X_{23} \\ X_{13}^T & X_{23}^T & X_{33} \end{bmatrix}$, Jing, et al., present the following delay-dependent stability condition:

Theorem 1.15. [95] Under Assumption 1.2.3 with $d_m = 0$, the time delay system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$, P_1, P_2, and X_{ij}, $i \leq j, i, j = 1, 2, 3$ such that

$$\begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{12}^T & X_{22} & X_{23} \\ X_{13}^T & X_{23}^T & X_{33} \end{bmatrix} > 0$$
$$\begin{bmatrix} (1,1) & P - P_1^T + A^T P_2 & (1,3) \\ * & d_M X_{33} - P_2 - P_2^T & P_2^T A_d \\ * & * & (3,3) \end{bmatrix} < 0$$

where $(1,1) = A^T P_1 + P_1^T A + d_M X_{11} + X_{13} + X_{13}^T + Q$, $(1,3) = P_1^T A_1 + d_M X_{12} - X_{13} + X_{23}^T$ and $(3,3) = d_M X_{22} - X_{23} - X_{23}^T - (1-\rho)Q$.

Corollary 1.16. [95] Under Assumption 1.2.1, the time delay system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$, P_1, P_2, $Q > 0$, and X_{ij}, $i \le j, i, j = 1, 2, 3$ such that

$$\begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{12}^T & X_{22} & X_{23} \\ X_{13}^T & X_{23}^T & X_{33} \end{bmatrix} > 0$$

$$\begin{bmatrix} (1,1) & P - P_1^T + A^T P_2 & (1,3) \\ * & d_M X_{33} - P_2 - P_2^T & P_2^T A_d \\ * & * & (3,3) \end{bmatrix} < 0$$

where $(1,1) = A^T P_1 + P_1^T A + d_M X_{11} + X_{13} + X_{13}^T + Q$, $(1,3) = P_1^T A_1 + d_M X_{12} - X_{13} + X_{23}^T$ and $(3,3) = d_M X_{22} - X_{23} - X_{23}^T$.

In [108], an alternative Lyapunov-Krasovskii functional was defined,

$$V(t) = \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix}^T EP \begin{bmatrix} x(t) \\ \dot{x}(t) \end{bmatrix} + \int_{t-d}^{t} x^T(s) Q x(s) ds$$
$$\int_0^t \int_{\beta-d}^{\beta} \begin{bmatrix} x(\beta) \\ \dot{x}(\beta) \\ \dot{x}(s) \end{bmatrix}^T \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{12}^T & X_{22} & X_{23} \\ X_{13}^T & X_{23}^T & X_{33} \end{bmatrix} \begin{bmatrix} x(\beta) \\ \dot{x}(\beta) \\ \dot{x}(s) \end{bmatrix} ds d\beta.$$

Theorem 1.17. [108] The time-delay system $(\mathcal{S}_1)$ is asymptotically stable for any delay d, $0 < d \le d_M$ if there exist matrices $P_1 > 0$, P_2, P_3, Q, X_{11}, X_{12}, X_{22}, Y_1, Y_2 and $Z > 0$ such that the following LMIs hold:

$$\begin{bmatrix} X_{11} & X_{12} & Y_1 \\ X_{12}^T & X_{22} & Y_2 \\ Y_1^T & Y_2^T & Z \end{bmatrix} \ge 0$$

$$\begin{bmatrix} (1,1) & (1,2) & P_2^T A_d - Y_1 \\ * & -P_3 - P_3^T + d_M Z & P_3^T A_d - Y_2 \\ * & * & -Q \end{bmatrix} < 0$$

where $(1,1) = P_2^T A + A^T P_2 + d_M X_{11} + Q + Y_1 + Y_1^T$ and $(1,2) = P_1 - P_2^T + A^T P_3 + d_M X_{12} + Y_2^T$.

For system $(\mathcal{S}_2)$, we can get the descriptor system only by changing d into $d(t)$.

The descriptor system approach has been widely used to deal with various problems of time delay systems in order to provide delay-dependent results, see [43, 45, 202, 95, 108] for example.

Approach IV: Free-weighting matrix approach

From the Newton-Leibniz formula, the following equations are true for any matrices Y and W, with appropriate dimensions:

$$2x^T(t)Y\left[x(t) - x(t-d) - \int_{t-d}^{t} \dot{x}(s)ds\right] = 0$$

and

$$2x^T(t-d)W\left[x(t) - x(t-d) - \int_{t-d}^{t} \dot{x}(s)ds\right] = 0.$$

On the other hand, the following equation is also true:

$$d\zeta^T(t)X\zeta(t) - \int_{t-d}^{t} \zeta^T(t)X\zeta(t)ds = 0$$

where $\zeta(t) = \left[\, x^T(t)\ x^T(t-d) \,\right]^T$, $X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix}$.

With the above observations, the following delay-dependent condition can be gotten.

Theorem 1.18. [184] The time-delay system $(\mathcal{S}_1)$ is asymptotically stable for any delay d, $0 < d \le d_M$ if there exist matrices $P > 0$, $Q > 0$, $Z > 0$, X_{11}, X_{12}, X_{22}, Y and W such that the following LMIs

$$\begin{bmatrix} (1,1) & PA_d - Y + W^T + d_M X_{12} & d_M A^T Z \\ * & -Q - W - W^T + d_M X_{22} & d_M A_d^T Z \\ * & * & -d_M Z \end{bmatrix} < 0$$

$$\begin{bmatrix} X_{11} & X_{12} & Y \\ * & X_{22} & W \\ * & * & Z \end{bmatrix} \ge 0$$

where $(1,1) = PA + A^T P + Y + Y^T + Q + d_M X_{11}$.

Based on free-weighting matrix approach, a great amount of delay-dependent stability conditions has been derived, see [77, 79, 48, 208] for example.

1.4.2 Recent Results on Interval Time-Varying Delay System

1.4.2.1 FWM-based results

Since the free-weighting matrix (FWM) method was proposed in [79], it plays an important role in deriving delay-dependent stability conditions. But there is room for further investigation. First, when estimating the upper bound of the derivative of Lyapunov functional, some useful terms are ignored. For example, in [46], [79] and [184], the derivative of $\int_{-d_M}^{0}\int_{t+\theta}^{t} \dot{x}^T(s)Z\dot{x}(s)dsd\theta$ is often estimated as $d_M\dot{x}^T(t)Z\dot{x}(t) - \int_{t-d(t)}^{t} \dot{x}^T(s)Z\dot{x}(s)ds$ and the term

$-\int_{t-d_M}^{t-d(t)} \dot{x}^T(s)Z\dot{x}(s)ds$ is ignored, which may lead to considerable conservativeness.

Consider system $(\mathcal{S}_2)$, and the time-varying delay satisfying Assumption 1.2.3. Define the following Lyapunov functional candidate:

$$\begin{aligned} V(t) &= x^T(t)Px(t) + \int_{t-d_m}^{t} x^T(s)Q_1x(s)ds + \int_{t-d_M}^{t} x^T(s)Q_2x(s)ds \\ &\quad + \int_{t-d(t)}^{t} x^T(s)Q_3x(s)ds + \int_{-d_M}^{0}\int_{t+\theta}^{t} \dot{x}^T(s)Z_1\dot{x}(s)dsd\theta \\ &\quad + \int_{-d_M}^{-d_m}\int_{t+\theta}^{t} \dot{x}^T(s)Z_2\dot{x}(s)dsd\theta. \end{aligned}$$

By FWM approach, the following equations are true for any matrices N_1, S_i and M_i, $i = 1, 2$, with appropriate dimensions:

$$2[x^T(t)N_1 + x^T(t-d(t))N_2]\left[x(t) - x(t-d(t)) - \int_{t-d(t)}^{t} \dot{x}(s)ds\right] = 0$$

$$2[x^T(t)S_1 + x^T(t-d(t))S_2]\left[x(t-d(t)) - x(t-d_M) - \int_{t-d_M}^{t-d(t)} \dot{x}(s)ds\right] = 0$$

$$2[x^T(t)W_1 + x^T(t-d(t))W_2]\left[x(t-d_m) - x(t-d(t)) - \int_{t-d(t)}^{t-d_m} \dot{x}(s)ds\right] = 0.$$

Calculating the derivative of $V(t)$, and noticing the following equations:

$$-\int_{t-d_m}^{t} \dot{x}^T(s)Z_1\dot{x}(s)ds = -\int_{t-d(t)}^{t} \dot{x}^T(s)Z_1\dot{x}(s)ds - \int_{t-d_m}^{t-d(t)} \dot{x}^T(s)Z_1\dot{x}(s)ds$$

$$-\int_{t-d_M}^{t-d_m} \dot{x}^T(s)Z_2\dot{x}(s)ds = -\int_{t-d_M}^{t-d(t)} \dot{x}^T(s)Z_2\dot{x}(s)ds - \int_{t-d(t)}^{t-d_m} \dot{x}^T(s)Z_1\dot{x}(s)ds$$

yields the following theorem.

Theorem 1.19. [76] Under Assumption 1.2.3, system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$, $Q_i \geq 0$, $i = 1, 2, 3$, $Z_j > 0$, $j = 1, 2$, N_i, M_i and S_i, $i = 1, 2$ such that the following LMI holds:

$$\begin{bmatrix} \Phi_{11} & \Phi_{12} & M_1 & -S_1 & d_MN_1 & \delta S_1 & \delta M_1 & A^TU \\ * & \Phi_{22} & M_2 & -S_2 & d_MN_2 & \delta S_2 & \delta M_2 & A_d^TU \\ * & * & -Q_1 & 0 & 0 & 0 & 0 & 0 \\ * & * & * & -Q_2 & 0 & 0 & 0 & 0 \\ * & * & * & * & -d_MZ_1 & 0 & 0 & 0 \\ * & * & * & * & * & -\delta(Z_1+Z_2) & 0 & 0 \\ * & * & * & * & * & * & -\delta Z_2 & 0 \\ * & * & * & * & * & * & * & -U \end{bmatrix} < 0$$

where $\delta = d_M - d_m$ and

$$\begin{aligned}
\Phi_{11} &= PA + A^T P + \sum_{i=1}^{3} Q_i + N_1 + N_1^T \\
\Phi_{12} &= PA_d + N_2^T - N_1 + S_1 - M_1 \\
\Phi_{22} &= -(1-\rho)Q_3 + S_2 + S_2^T - N_2 - N_2^T - M_2 - M_2^T \\
U &= d_M Z_1 + \delta Z_2.
\end{aligned}$$

Similarly, the authors also notice the conservatism led by using the over bounding 0 to bound the term $-\int_{t-d_M}^{t-d(t)} \dot{x}^T(s)Z\dot{x}(s)ds$. By using Lemma 1.10, and the following Lyapunov functional,

$$\begin{aligned}
V(t) = x^T(t)Px(t) &+ \frac{d_m}{2}\int_{-\frac{d_m}{2}}^{0} ds \int_{t+s}^{t} \dot{x}^T(\theta)R_1\dot{x}(\theta)d\theta \\
&+ \int_{t-\frac{d_m}{2}}^{t} \begin{bmatrix} x(s) \\ x(s-\frac{d_m}{2}) \end{bmatrix}^T \begin{bmatrix} Q_1 & Q_2 \\ Q_2^T & Q_3 \end{bmatrix} \begin{bmatrix} x(s) \\ x(s-\frac{d_m}{2}) \end{bmatrix} ds \\
&+ \int_{t-\frac{d_M}{2}}^{t} \begin{bmatrix} x(s) \\ x(s-\frac{d_M}{2}) \end{bmatrix}^T \begin{bmatrix} Q_4 & Q_5 \\ Q_5^T & Q_6 \end{bmatrix} \begin{bmatrix} x(s) \\ x(s-\frac{d_M}{2}) \end{bmatrix} ds \\
&+ \frac{d_M}{2}\int_{-\frac{d_M}{2}}^{0} ds \int_{t+s}^{t} \dot{x}^T(\theta)R_2\dot{x}(\theta)d\theta \\
&+ (d_M - d_m)\int_{-d_M}^{-d_m} ds \int_{t+s}^{t} \dot{x}^T(\theta)S\dot{x}(\theta)d\theta
\end{aligned}$$

where $P > 0$, $\begin{bmatrix} Q_1 & Q_2 \\ Q_2^T & Q_3 \end{bmatrix} > 0$, $\begin{bmatrix} Q_4 & Q_5 \\ Q_5^T & Q_6 \end{bmatrix} > 0$, $R_1 > 0$, $R_2 > 0$ and $S > 0$ with appropriate dimensions, the following delay-dependent stability condition is given in [93], which is less conservative than [76].

Theorem 1.20. [93] Under Assumption 1.2.3, system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$, $\begin{bmatrix} Q_1 & Q_2 \\ Q_2^T & Q_3 \end{bmatrix} > 0$, $\begin{bmatrix} Q_4 & Q_5 \\ Q_5^T & Q_6 \end{bmatrix} > 0$, $R_1 > 0$, $R_2 > 0$ and $S > 0$ such that

$$\begin{bmatrix}
\Xi_{11} & \Xi_{12} & \Xi_{13} & 0 & \Xi_{15} & 0 \\
* & \Xi_{22} & 0 & S & 0 & S \\
* & * & \Xi_{33} & -Q_2 & 0 & 0 \\
* & * & * & -Q_3 - S & 0 & 0 \\
* & * & * & * & \Xi_{55} & -Q_5 \\
* & * & * & * & * & -Q_6 - S
\end{bmatrix} < 0$$

where

$$\begin{aligned}
\Theta &= \frac{1}{4}(d_m^2 R_1 + d_M^2 R_2 + 4(d_M - d_m)^2 S) \\
\Xi_{11} &= PA + A^T P + A^T \Theta A + Q_1 + Q_4 - R_1 - R_2
\end{aligned}$$

$$\Xi_{12} = PB + A^T\Theta B, \Xi_{13} = Q_2 + R_1, \Xi_{15} = Q_5 + R_2, \Xi_{22} = B^T\Theta B - 2S.$$

1.4.2.2 Jensen's Inequality-Based Results

It is obvious that free-weighting matrix approach is effective in reducing conservatism in existing delay dependent results, however, since many matrix variables will be introduced, which makes stability criteria very complicated. Thus, based on Jensen's inequality, the author of [151] provided the following delay-dependent condition for system $(\mathcal{S}_2)$. The resulting criterion has advantages over some previous ones in that it involves fewer matrix variables and has less conservatism, which is established theoretically.

Theorem 1.21. [151] Under Assumption 1.2.3, system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$, $Q_i > 0$, $i = 1, 2, 3$ and $Z_j > 0$, $j = 1, 2$, such that the following LMIs hold:

$$\begin{bmatrix} \Upsilon & PA_d & Z_1 & 0 & d_m A^T Z_1 & \delta A^T Z_2 \\ * & -(1-\rho)Q_3 - 2Z_2 & Z_2 & Z_2 & d_m A_d^T Z_1 & \delta A_d^T Z_2 \\ * & * & -Q_1 - Z_1 - Z_2 & 0 & 0 & 0 \\ * & * & * & -Q_2 - Z_2 & 0 & 0 \\ * & * & * & * & -Z_1 & 0 \\ * & * & * & * & * & -Z_2 \end{bmatrix} < 0$$

where $\Upsilon = PA + A^T P + Q_1 + Q_2 + Q_3 - Z_1$.

Following a similar line and using a new method to get a tight upper bound of the derivative of Lyapunov functional, the following new result is obtained in [152].

Theorem 1.22. [152] Under Assumption 1.2.3, system $(\mathcal{S}_2)$ is asymptotically stable if there exist matrices $P > 0$, $Q_i > 0$, $i = 1, 2, 3$ and $Z_j > 0$, $j = 1, 2$, such that the following LMIs hold:

$$\Phi_1 = \Phi - \begin{bmatrix} 0 & -I & I & 0 \end{bmatrix}^T Z_2 \begin{bmatrix} 0 & -I & I & 0 \end{bmatrix} < 0$$

and

$$\Phi_2 = \Phi - \begin{bmatrix} 0 & I & 0 & -I \end{bmatrix}^T Z_2 \begin{bmatrix} 0 & I & 0 & -I \end{bmatrix} < 0$$

where

$$\Phi = \begin{bmatrix} \Upsilon & PA_d & Z_1 & 0 \\ * & -(1-\rho)Q_3 - 2Z_2 & Z_2 & Z_2 \\ * & * & -Q_1 - Z_1 - Z_2 & 0 \\ * & * & * & -Q_2 - Z_2 \end{bmatrix} + \begin{bmatrix} A & A_d & 0 & 0 \end{bmatrix}^T (d_m^2 Z_1 + \delta^2 Z_2) \begin{bmatrix} A & A_d & 0 & 0 \end{bmatrix}$$

with

$$\Upsilon = PA + A^T P + \sum_{i=1}^{3} Q_i - Z_1.$$

1.5 Conclusion

This chapter has reviewed certain stability analysis of time-delay systems. LMI techniques in deriving both delay-independent and delay-dependent stability conditions have been reviewed. Firstly, some basic delay-independent stability conditions are recalled. Secondly, four approaches have been introduced for developing delay-dependent stability conditions, they are: Model Transformation approach, Bounding Techniques, Descriptor System approach and Free-Weighting Matrix approach. Finally, the more recent results on delay-dependent stability conditions have been given.

Chapter 2
New Results on Stability of Systems With Interval Time-Varying Delay

2.1 Introduction

Very recently, systems with time-varying delay in an interval have been studied in [76, 93, 151, 152]. In [76], delay-range-dependent stability criteria have been proposed by preserving some useful terms when estimating the upper bound of the derivative of Lyapunov functional. By using a new Lyapunov functional and a tighter bounding technique, some delay-dependent stability criteria have been provided, which are less conservative than [76]. It is worthy mentioning that in [76] and [93], to obtain less conservative results, free matrix variables are introduced during calculating the derivative of Lyapunov functional, which makes stability criteria complicated. With the observations, in [151], Jensen's Inequality is used to derive an improved delay-dependent stability criteria for systems with a delay varying in a range. Compared with previous results, the stability criteria in [151] are with fewer matrix variables and less conservatism. In [152], the author further improve the results in [151] by using the idea of convex combination. Nevertheless, the criteria still have room for further improvement in accuracy as well as complexity reduction.

In this chapter, our attention will be focused on delay-dependent stability for systems with interval time-varying delay. By choosing an appropriate Lyapunov functional and using Finsler's Lemma, some new delay-dependent stability criteria are obtained via linear matrix inequality approach. It is worthy mentioning that the developed results in this chapter, compared with recently published results, such as [92, 76, 93, 151, 152], involve fewer matrix variables but have less conservatism. Numerical examples are provided to demonstrate the advantages of the proposed stability criteria.

The chapter is organized as follows. Section 2.2 gives the problem formulation and some preliminaries. The new sufficient conditions for system with interval time-varying delay are presented in Section 2.3. Numerical simulations are provided in Section 2.4 and some conclusion remarks are given in Section 2.5.

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 17–23.
springerlink.com

2.2 Problem Formulation

Consider the following linear system with interval time-varying delay:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + A_d x(t - d(t)), \quad t > 0 \\ x(t) &= \phi(t), \quad t \in [-d_M, 0] \end{aligned} \tag{2.1}$$

where $x(t) \in R$ is the state vector, A and A_d are constant matrices with appropriate dimensions, $d(t)$ is an interval time-varying delay in the state, $\phi(t)$ is a continuous vector-valued initial function of $t \in [-d_M, 0]$. A natural assumption on $d(t)$ is made as follows.

Assumption 2.2.1 *The time delays $d(t)$ are assumed to be a uniformly continuous time-varying function satisfying $d_m \le d(t) \le d_M$ and $\dot{d}(t) \le \rho$, where d_m and d_M are constant positive scalars representing the lower and upper bounds of $d(t)$, respectively, and ρ is the upper bound of $\dot{d}(t)$.*

Before ending this section, we introduce the following well known lemma, which will be used in the derivation of our main results.

Lemma 2.1. (Finsler's Lemma): Let $x \in R^n$, $P = P^T \in R^{n \times n}$ and $H \in R^{m \times n}$ such that $\text{rank}(H) = r < n$. The following two statements are equivalent:

1. $x^T P x < 0$, $\forall Hx = 0, x \neq 0$,
2. $\exists X \in R^{n \times m}$ such that $P + XH + H^T X^T < 0$.

2.3 Main Results

The main objective of this section is to derive a new delay-interval dependent stability sufficient condition for system (2.1).

Theorem 2.2. *Under Assumption 2.2.1, system (2.1) is asymptotically stable if there exist matrices $P > 0$, $Q > 0$, $Z_1 > 0$ and $Z_2 > 0$ such that the following LMI holds*

$$\begin{bmatrix} \Psi_{11} & PA_d - Z_2 & d_M A^T Z_1 & (d_M - d_m) A^T Z_2 \\ A_d^T P - Z_2 & -(1-\rho)Q - 2Z_2 & d_M A_d^T Z_1 & (d_M - d_m) A_d^T Z_2 \\ d_M Z_1 A & d_M Z_1 A_d & -Z_1 & 0 \\ (d_M - d_m) Z_2 A & (d_M - d_m) Z_2 A_d & 0 & -Z_2 \end{bmatrix} < 0 \tag{2.2}$$

where $\Psi_{11} = A^T P + PA + Q - Z_1 - Z_2$.

Proof. Choose a Lyapunov functional candidate to be

$$V(x(t)) = x^T(t)Px(t) + \int_{t-d(t)}^{t} x^T(s)Qx(s)ds$$
$$+d_M \int_{-d_M}^{0}\int_{t+s}^{t} \dot{x}^T(\alpha)Z_1\dot{x}(\alpha)d\alpha ds$$
$$+(d_M - d_m)\int_{-d_M}^{-d_m}\int_{t+s}^{t} \dot{x}^T(\alpha)Z_2\dot{x}(\alpha)d\alpha ds. \tag{2.3}$$

Calculating the derivative of $V(x(t))$ along the trajectory of system (2.1) yields that

$$\begin{aligned}
\dot{V}(x(t)) &= 2x^T(t)P\dot{x}(t) + x^T(t)Qx(t) - (1-\dot{d}(t))x^T(t-d(t))Qx(t-d(t)) \\
&\quad +d_M^2\dot{x}^T(t)Z_1\dot{x}(t) - d_M\int_{t-d_M}^{t}\dot{x}^T(\alpha)Z_1\dot{x}(\alpha)d\alpha \\
&\quad +(d_M-d_m)^2\dot{x}^T(t)Z_2\dot{x}(t) - (d_M-d_m)\int_{t-d_M}^{t-d_m}\dot{x}^T(\alpha)Z_2\dot{x}(\alpha)d\alpha \\
&\le 2x^T(t)P\dot{x}(t) + x^T(t)Qx(t) - (1-\rho)x^T(t-d(t))Qx(t-d(t)) \\
&\quad +\dot{x}^T(t)\left[d_M^2Z_1 + (d_M-d_m)^2Z_2\right]\dot{x}(t) \\
&\quad -d_M\int_{t-d_M}^{t}\dot{x}^T(\alpha)Z_1\dot{x}(\alpha)d\alpha \\
&\quad -(d_M-d(t))\int_{t-d_M}^{t-d(t)}\dot{x}^T(\alpha)Z_2\dot{x}(\alpha)d\alpha \\
&\quad -(d(t)-d_m)\int_{t-d(t)}^{t-d_m}\dot{x}^T(\alpha)Z_2\dot{x}(\alpha)d\alpha.
\end{aligned} \tag{2.4}$$

According to Jensen's Inequality, the following three inequalities are true:

$$-d_M\int_{t-d_M}^{t}\dot{x}^T(\alpha)Z_1\dot{x}(\alpha)d\alpha \le -\xi_1^T(t)Z_1\xi_1(t) \tag{2.5}$$
$$-(d_M-d(t))\int_{t-d_M}^{t-d(t)}\dot{x}^T(\alpha)Z_2\dot{x}(\alpha)d\alpha \le -\xi_2^T(t)Z_2\xi_2(t) \tag{2.6}$$
$$-(d(t)-d_m)\int_{t-d(t)}^{t-d_m}\dot{x}^T(\alpha)Z_2\dot{x}(\alpha)d\alpha \le -\xi_3^T(t)Z_2\xi_3(t) \tag{2.7}$$

where $\xi_1(t) = x(t) - x(t-d_M)$, $\xi_2(t) = x(t-d(t)) - x(t-d_M)$ and $\xi_3(t) = x(t-d_m) - x(t-d(t))$.

Then, taking into account (2.4)-(2.7) gives that

$$\dot{V}(x(t)) \le \xi^T(t)\Sigma\xi(t) \tag{2.8}$$

where $\xi(t) = [\dot{x}^T(t), x^T(t), x^T(t-d(t)), \xi_1^T(t), \xi_2^T(t), \xi_3^T(t)]^T$ and

$$\Sigma = \begin{bmatrix} d_M^2 Z_1 + (d_M - d_m)^2 Z_2 & P & 0 & 0 & 0 & 0 \\ P & Q & 0 & 0 & 0 & 0 \\ 0 & 0 & -(1-\rho)Q & 0 & 0 & 0 \\ 0 & 0 & 0 & -Z_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -Z_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -Z_2 \end{bmatrix}.$$

It follows from (2.1) that

$$\overline{A}\xi(t) \equiv 0$$

where $\bar{A} = \begin{bmatrix} I & -A & -A_d & 0 & 0 & 0 \\ 0 & -I & I & I & -I & 0 \end{bmatrix}$.

Then, system (2.1) is asymptotically stable if for all $\xi(t)$ subject to $\bar{A}\xi(t) = 0$, $\xi^T(t)\Sigma\xi(t) < 0$. Now, from Finsler's Lemma, $\xi^T(t)\Sigma\xi(t) < 0$ is equivalent to $\bar{A}^{\perp T}\Sigma\bar{A}^{\perp} < 0$, where $\bar{A}^{\perp} = \begin{bmatrix} A^T & I & 0 & I & 0 & I \\ A_d^T & 0 & I & 0 & I & I \end{bmatrix}^T$, which can be rewritten as

$$\begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{12}^T & \Phi_{22} \end{bmatrix} < 0 \tag{2.9}$$

where

$$\begin{aligned} \Phi_{11} &= d_M^2 A^T Z_1 A + (d_M - d_m)^2 A^T Z_2 A + A^T P + PA + Q - Z_1 - Z_2 \\ \Phi_{12} &= d_M^2 A^T Z_1 A_d + (d_M - d_m)^2 A^T Z_2 A_d + PA_d - Z_2 \\ \Phi_{22} &= d_M^2 A_d^T Z_1 A_d + (d_M - d_m)^2 A_d^T Z_2 A_d - (1-\rho)Q - 2Z_2 \end{aligned}$$

which in turn equivalent to (2.2) by Schur complements. Then, $\dot{V}(x(t)) < -\epsilon\|x(t)\|^2$ for a sufficiently small $\epsilon > 0$, which ensures the asymptotic stability of system (2.1), see e.g. [68].

Remark 2.3. It should be pointed out that although free-weighting matrix method is helpful to reduce the conservatism of stability criteria, the criteria are very complex since many free matrix variables are introduced. Based on Finsler's lemma, new delay-interval-dependent stability condition is obtained by appropriately choosing Lyapunov functional. Compared with existing results, such as [92, 76, 93, 151, 152], the newly developed condition in this chapter possesses some advantages. On the top of this, Theorem 2.2 involves fewer matrix variables, i.e., only four positive define matrices in the Lyapunov functional to be determined, thus Theorem 2.2 can be carried out more efficiently. Table 2.1 provides a comparison of the numbers of the variables involved among recently published papers and this chapter. The another advantage associated with Theorem 2.2 is its less conservatism, which can be seen from the numerical examples clearly.

When the information of the time derivative of delay is unknown, by setting $Q = 0$ in the Lyapunov functional (2.3), we have the following result from Theorem 2.2 immediately.

Table 2.1 Comparison of the Numbers of the Variables Involved

Methods	Number of variables involved
[76]	$9n^2 + 3n$
[93]	$6n^2 + 4n$
[151]	$3n^2 + 3n$
[152]	$3n^2 + 3n$
This chapter	$2n^2 + 2n$

Corollary 2.4. *Under Assumption 2.2.1, system (2.1) is asymptotically stable if there exist matrices $P > 0$, $Z_1 > 0$ and $Z_2 > 0$ such that the following LMI holds*

$$\begin{bmatrix} \tilde{\Psi}_{11} & PA_d - Z_2 & d_M A^T Z_1 & (d_M - d_m)A^T Z_2 \\ A_d^T P - Z_2 & -2Z_2 & d_M A_d^T Z_1 & (d_M - d_m)A_d^T Z_2 \\ d_M Z_1 A & d_M Z_1 A_d & -Z_1 & 0 \\ (d_M - d_m)Z_2 A & (d_M - d_m)Z_2 A_d & 0 & -Z_2 \end{bmatrix} < 0 \quad (2.10)$$

where $\tilde{\Psi}_{11} = A^T P + PA - Z_1 - Z_2$.

2.4 Numerical Example

Example 2.5. Consider system (2.1) with

$$A = \begin{bmatrix} -2 & 0 \\ 0 & -0.9 \end{bmatrix}, A_d = \begin{bmatrix} -1 & 0 \\ -1 & -1 \end{bmatrix}.$$

For various ρ, the admissible upper bound d_M of the delay, which guarantee the stability of system (2.1) are listed with given d_m in Table 2.2, 2.3, and 2.4.

Table 2.2 Admissible upper bound d_M for various d_m and $\rho = 0.5$

	$d_m = 1$	$d_m = 2$	$d_m = 3$	$d_m = 4$	$d_m = 4.4697$
[76]	2.07	2.43	3.22	4.07	4.47
[151]	2.07	2.44	3.22	4.06	4.47
[152]	2.12	2.50	3.25	4.07	4.47
Theorem 2.2	2.63	3.63	4.63	5.63	6.10

It is easy to see that, in the above tables, the criteria derived in this chapter improve over some recently published ones in that the computed admissible upper bound of time delay is larger.

Example 2.6. Consider system (2.1) with

Table 2.3 Admissible upper bound d_M for various d_m and $\rho = 0.9$

	$d_m = 1$	$d_m = 2$	$d_m = 3$	$d_m = 4$	$d_m = 4.4697$
[76]	1.74	2.43	3.22	4.07	4.47
[151]	1.76	2.44	3.22	4.06	4.47
[152]	1.87	2.50	3.25	4.07	4.47
Theorem 2.2	2.44	3.44	4.44	5.44	5.91

Table 2.4 Admissible upper bound d_M for various d_m and Unknown ρ

	$d_m = 1$	$d_m = 2$	$d_m = 3$	$d_m = 4$	$d_m = 4.4697$
[92]	1.64	2.39	3.20	4.06	–
[76]	1.7424	2.4328	3.2234	4.0644	4.47
[152]	1.8737	2.5049	3.2591	4.0744	4.4700
[93]	1.8043	2.5213	3.3311	4.1880	4.6009
Corollary 2.4	2.41	3.41	4.41	5.41	5.88

$$A = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix}, A_d = \begin{bmatrix} 0 & 0 \\ -1 & 1 \end{bmatrix}.$$

For $\rho = 0.3$ and unknown ρ, the admissible upper bound d_M of the delay are shown with given d_m in Tables 2.5 and 2.6, respectively. From Table 2.5 and 2.6, it can be seen that with fewer matrix variables the stability results obtained in the chapter are less conservative than those in existing references.

Table 2.5 Admissible upper bound d_M for various d_m and $\rho = 0.3$

	$d_m = 1$	$d_m = 2$	$d_m = 3$	$d_m = 4$	$d_m = 5$
[76]	2.2125	2.4091	3.3342	4.2799	5.2393
[151]	2.2128	2.4179	3.3382	4.2819	5.2403
[152]	2.2474	2.4798	3.3893	4.325	5.2773
Theorem 2.2	3.88	4.88	5.88	6.88	7.88

Table 2.6 Admissible upper bound d_M for various d_m and unknown ρ

	$d_m = 0.3$	$d_m = 0.5$	$d_m = 0.8$	$d_m = 1$	$d_m = 2$
[92]	0.91	1.07	1.33	1.50	2.39
[76]	0.9431	1.0991	1.3476	1.5187	2.4000
[151]	0.9806	1.1325	1.3733	1.5401	2.4100
[152]	1.0715	1.2191	1.4539	1.6169	2.4798
Corollary 2.4	1.71	1.91	2.21	2.41	3.41

2.5 Conclusion

In this chapter, the problem of stability analysis has been performed for systems with interval time-varying delay. By constructing a novel Lyapunov functional and using Finsler's Lemma, a new delay-interval-dependent stability criterion is obtained. The advantage of the resulting criterion lies in its simplicity and less conservatism. The numerical results seem to suggest that the proposed methods may improve the results in recently published papers.

Chapter 3
Stability and Stabilization for Discrete Systems with Time-Delay

3.1 Introduction

Recently, much attention has been given to the delay-dependent stability and stabilization of continuous systems with time-delay, see for example, [146, 139, 99, 44, 188, 53, 109, 185, 78, 198, 203]. In those papers, many kinds of Lyapunov-Krasovskii functional are proposed in order to derive less conservative stability conditions. Correspondingly, some of the techniques adopted in the above papers have been extended to the stability and stabilization problem for discrete systems with time-delay. Less conservative stability conditions for discrete systems with time-delay are also derived, see, [53, 201] for example. However, discrete system with time-delay has an important feature, that is, it can be transformed into augmented delay-free system [3, 170]. Then, the stability and stabilization of such a system can be solved with a simple Lyapunov function. Although the dimensions of augmented systems could be larger if the time-delay is large, the stability conditions are simple and convex, which can be checked easily by today's fast developing computing techniques.

In this chapter, the problems of stability and stabilization for discrete systems with time-delay are considered. Necessary and sufficient stability and stabilization conditions for systems with constant time-delay are presented. The system with time-varying delay is equivalent to a kind of switched systems. Sufficient stability and stabilization conditions are derived.

The chapter is organized as follows. Section 3.2 gives the problem formulation and some preliminaries. Section 3.3 presents stability and stabilization conditions for systems with both constant and time-varying delays. Numerical simulations are presented in Section 3.4 and some conclusion remarks are given in Section 3.5.

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 25–34.
springerlink.com

3.2 Problem Formulation

Consider the following uncertain systems in discrete-time described by

$$\begin{aligned} x(k+1) &= (A + \Delta A(k))x(k) + (A_d + \Delta A_d(k))x(k-\tau(k)) \\ &\quad + (B + \Delta B(k))u(k) \\ x(k) &= \varphi(k), k = -\bar{\tau}, -\bar{\tau}+1, \cdots, 0 \end{aligned} \tag{3.1}$$

where $x(k) \in R^n$ and $u(k) \in R^m$ are the states and system inputs, respectively; A, A_d and B are matrices of appropriate dimensions with $\text{rank}(B) = m$, integer $\tau(k) \geq 0$ denotes the amount of time delay, and assumed to be constant or time-varying. It satisfies $0 \leq \tau(k) \leq \bar{\tau}$ when it is time-varying, and $\bar{\tau}$ is a positive integer; $\varphi(\cdot)$ denotes the initial condition. $\Delta A(k)$, $\Delta A_d(k)$ and $\Delta B(k)$ are time-varying uncertain matrices of the following form

$$[\Delta A(k) \;\; \Delta A_d(k) \;\; \Delta B(k)] = DF(k)[E_a \;\; E_d \;\; E_b] \tag{3.2}$$

where D, E_a, E_d and E_b are constant matrices with appropriate dimensions and $F(k)$ satisfies $F^T(k)F(k) \leq I$. In the following sections, the problems of stability and stabilization for system (3.1) will be investigated.

3.3 Main Results

3.3.1 Stability analysis

It is well known that a discrete-time delay system can be lifted to a non-delay system, so the lifted system now is explored.

First, we consider the case of an unforced system (3.1) where system matrices are certain and the time-delay is constant, that is, $\tau(k) = \tau$, then let

$$z(k) = \begin{bmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(k-\tau+1) \\ x(k-\tau) \end{bmatrix}, z(0) = \begin{bmatrix} \varphi(0) \\ \varphi(-1) \\ \vdots \\ \varphi(-\tau+1) \\ \varphi(-\tau) \end{bmatrix}, \bar{A} = \begin{bmatrix} A & 0 & \cdots & 0 & A_d \\ I & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & I & 0 \end{bmatrix} \tag{3.3}$$

then, system (3.1) without control can be written by the following system without delay:

$$z(k+1) = \bar{A}z(k). \tag{3.4}$$

As (3.4) is a non-delayed system, then the techniques for stability of a non-delayed system can be used to check the stability and stabilization conditions.

From well-known stability results, we have the following result immediately.

Lemma 3.1. *System (3.1) with constant delay τ is stable if and only if there exists a positive definite matrix $P \in R^{(\tau+1)n \times (\tau+1)n}$ such that the following inequality holds*:

$$\bar{A}^T P \bar{A} - P < 0. \tag{3.5}$$

Remark 3.2. Note that inequality (3.5) in Lemma 3.1 is a necessary and sufficient condition for system (3.1) to be stable, any other existing delay-dependent conditions could not be better than this one. It should be mentioned that the dimensions of the augmented system may be larger when the time-delay is longer, however, this convex condition can be easily checked due to the availability of fast computing techniques. This will be demonstrated in Example 3.3.

Example 3.3. To show the effectiveness of Lemma 3.1, let us introduce one example, which is considered by [201, 107], and will be used to compare our results. Consider system (3.1) with

$$A = \begin{bmatrix} 0.8 & 0 \\ 0 & 0.91 \end{bmatrix}, A_d = \begin{bmatrix} -0.1 & 0 \\ -0.1 & -0.1 \end{bmatrix}.$$

By the delay-dependent stability condition in [107], it was given that the maximum allowed delay is 41. By the method provided in [201], it was reported that the maximum allowed delay is 42. By the condition in Lemma 3.1, it is found that system is still stable when the time-delay is 55, and could be even larger.

Next, we consider the case for systems with uncertainties, the unforced system (3.1) can be written as

$$z(k+1) = (\bar{A} + \bar{D}F(k)\bar{E})z(k) \tag{3.6}$$

where

$$\bar{A} = \begin{bmatrix} A & 0 & \cdots & 0 & A_d \\ I & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & I & 0 \end{bmatrix}, \bar{D} = \begin{bmatrix} D \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}, \bar{E}^T = \begin{bmatrix} E_a^T \\ 0 \\ \vdots \\ 0 \\ E_d^T \end{bmatrix}. \tag{3.7}$$

With the definition of quadratic stability for discrete system (3.6) referred in [13], the following result is given.

Theorem 3.4. *System (3.1) with uncertainty is quadratically stable if and only if there exist a definite positive matrix P and a positive scalar ϵ satisfying the following inequality:*

$$\begin{bmatrix} -P+\epsilon\bar{E}^T\bar{E} & \bar{A}^T P & 0 \\ P\bar{A} & -P & P\bar{D} \\ 0 & \bar{D}^T P & -\epsilon I \end{bmatrix} < 0.$$

Proof. From Definition 2.1 in [13], system (3.1) with uncertainty is quadratically stable if and only if the following inequality holds

$$\begin{bmatrix} -P & \bar{A}^T P \\ P\bar{A} & -P \end{bmatrix} < 0$$

which can be written as

$$\begin{bmatrix} -P & \bar{A}^T P \\ P\bar{A} & -P \end{bmatrix} + \begin{bmatrix} 0 \\ P\bar{D} \end{bmatrix} F(k) \begin{bmatrix} \bar{E} & 0 \end{bmatrix} + (\begin{bmatrix} 0 \\ P\bar{D} \end{bmatrix} F(k) \begin{bmatrix} \bar{E} & 0 \end{bmatrix})^T < 0.$$

By Lemma 1 in [197], the above inequality holds if and only if the following inequality is satisfied for any $\epsilon > 0$

$$\begin{bmatrix} -P & \bar{A}^T P \\ P\bar{A} & -P \end{bmatrix} + \epsilon^{-1} \begin{bmatrix} 0 & 0 \\ 0 & P\bar{D}\bar{D}^T P \end{bmatrix} + \epsilon \begin{bmatrix} \bar{E}^T\bar{E} & 0 \\ 0 & 0 \end{bmatrix} < 0.$$

Consequently,

$$\begin{bmatrix} -P+\epsilon\bar{E}^T\bar{E} & \bar{A}^T P \\ P\bar{A} & -P+\epsilon^{-1}P\bar{D}\bar{D}^T P \end{bmatrix} < 0.$$

Using Schur complement formula with respect to $-P+\epsilon^{-1}P\bar{D}\bar{D}^T P$, it leads to

$$\begin{bmatrix} -P+\epsilon\bar{E}^T\bar{E} & \bar{A}^T P & 0 \\ P\bar{A} & -P & P\bar{D} \\ 0 & \bar{D}^T P & -\epsilon I \end{bmatrix} < 0$$

which completes the proof.

Finally, we consider the case that the system matrices of unforced system (3.1) are certain, the delay is time-varying and satisfies $0 \le \tau(k) \le \bar{\tau}$. Then, let

$$z(k) = \begin{bmatrix} x(k) \\ x(k-1) \\ \vdots \\ x(k-\bar{\tau}+1) \\ x(k-\bar{\tau}) \end{bmatrix}, z(0) = \begin{bmatrix} \varphi(0) \\ \varphi(-1) \\ \vdots \\ \varphi(-\bar{\tau}+1) \\ \varphi(-\bar{\tau}) \end{bmatrix}$$

$$\bar{A}_0 = \begin{bmatrix} A + A_d & 0 \cdots 0 & 0 \cdots 0 & 0 \\ I & 0 \cdots 0 & 0 \cdots 0 & 0 \\ \vdots & \vdots \; \vdots & \vdots \; \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 \cdots 0 & 0 \\ 0 & 0 \cdots I & 0 \cdots 0 & 0 \\ \vdots & \vdots \; \vdots & \vdots \; \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 \cdots I & 0 \end{bmatrix}, \bar{A}_j = \begin{bmatrix} A & \overbrace{0 \cdots}^{(j-1)n} & A_d & 0 \cdots 0 & 0 \\ I & 0 \cdots & 0 & 0 \cdots 0 & 0 \\ \vdots & \vdots \; \vdots & \vdots & \vdots \; \vdots & \vdots \\ 0 & 0 \cdots & 0 & 0 \cdots 0 & 0 \\ 0 & 0 \cdots & I & 0 \cdots 0 & 0 \\ \vdots & \vdots \; \vdots & \vdots & \vdots \; \vdots & \vdots \\ 0 & 0 \cdots & 0 & 0 \cdots I & 0 \end{bmatrix} \tag{3.8}$$

for $j = 1, 2, \cdots, \bar{\tau}$. Consequently, system (3.1) with time-varying delay is equivalent to the following switched system:

$$z(k+1) = \bar{A}_\sigma z(k) \tag{3.9}$$

where σ is the piecewise constant switching signal taking value from the finite index set $\mathcal{F} = \{0, 1, \cdots, \bar{\tau}\}$.

According to the theory of switched systems [172], we have the following result.

Corollary 3.5. *System (3.1) with time-varying delay $\tau(k)$ is stable if there exists a positive definite matrix $P \in R^{(\bar{\tau}+1)n \times (\bar{\tau}+1)n}$ such that the following linear inequalities hold for any $i \in \{0, 1, \cdots, \bar{\tau}\}$:*

$$\bar{A}_i^T P \bar{A}_i - P < 0. \tag{3.10}$$

To show the less conservatism of the stability result in Corollary 3.5, the following example, which is taken from [51], is investigated.

Example 3.6. Consider system (3.1) with

$$A = \begin{bmatrix} 0.8 & 0 \\ 0.05 & 0.9 \end{bmatrix}, A_d = \begin{bmatrix} -0.1 & 0 \\ -0.2 & -0.1 \end{bmatrix}$$

where $\tau(k)$ is a bounded time-varying delay. It was obtained in [51] that system (3.1) is asymptotically stable for all $2 \le \tau(k) \le 7$, that is, the maximum delay is $\bar{\tau} = 7$. However, using the method proposed in Corollary 3.5, it is found that system (3.1) is stable for $\tau(k) \le 20$. In fact, we can search for even more larger delays $\tau(k)$ with LMI Tool-box such that (3.10) still holds.

3.3.2 Controller Design

As the lifting method is used, it seems that the controller design for discrete systems with time-delay will be very complex. However, the following method will make it much easier for controller design based on LMIs.

3.3.2.1 Controller Design For Certain Systems with Constant-Delay

Let $u(k) = Kx(k)$, then the closed-loop system is

$$x(k+1) = (A + BK)x(k) + A_d x(k - \tau). \tag{3.11}$$

Defining $z(k)$ as in (3.3), it follows from the lifting method that

$$\begin{aligned} \tilde{A} &= \begin{bmatrix} A & 0 & \cdots & 0 & A_d \\ I & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & I & 0 \end{bmatrix} + \begin{bmatrix} B \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} K \begin{bmatrix} I & 0 & \cdots & 0 & 0 \end{bmatrix} \\ &= \bar{A} + \bar{B} K I_1 \end{aligned} \tag{3.12}$$

then

$$z(k+1) = \tilde{A} z(k). \tag{3.13}$$

The following result will present an easy way to design the controller matrix gain K.

Theorem 3.7. *System (3.11) is stable if there exist a positive definite matrices* $Q = \begin{bmatrix} Q_{11} & 0 \\ 0 & Q_{22} \end{bmatrix}$ *with* $Q_{11} \in R^{n \times n}, Q_{22} \in R^{\tau n \times \tau n}$ *and a matrix* $Y \in R^{m \times n}$ *such that the following inequalities holds*

$$\begin{bmatrix} -Q & Q\bar{A}^T + I_1^T Y^T \bar{B}^T \\ \bar{A}Q + \bar{B} Y I_1 & -Q \end{bmatrix} < 0. \tag{3.14}$$

Then, the state feedback can be chosen as $K = Y Q_{11}^{-1}$ with which the resulting closed-loop system is stable.

Proof. According to Lemma 3.1, system (3.11) is stable if and only if there exists positive definite matrix P such that

$$\tilde{A}^T P \tilde{A} - P < 0. \tag{3.15}$$

By Schur complement, the above inequality is equivalent to the following inequality,

$$\begin{bmatrix} -P & \tilde{A}^T \\ \tilde{A} & -P^{-1} \end{bmatrix} < 0.$$

Pre- and post-multiplying the above inequality by $\begin{bmatrix} P^{-1} & 0 \\ 0 & I \end{bmatrix}$, and let $Q = P^{-1}$, results in

$$\begin{bmatrix} -Q & Q\tilde{A}^T \\ \tilde{A}Q & -Q \end{bmatrix} < 0.$$

It follows from equation (3.12) that

$$\begin{bmatrix} -Q & Q\bar{A}^T + I_1^T Q_{11} K^T \bar{B}^T \\ \bar{A}Q + \bar{B}KQ_{11}I_1 & -Q \end{bmatrix} < 0.$$

Let $Y = KQ_{11}$, then the above inequality is equivalent to (3.14).

Remark 3.8. It is easy to see that inequality (3.14) is LMI with variables Q and Y, then K can be designed easily with LMI Tool-box.

Note that in the above result, the variable Q has a special structure, which may lead to some conservativeness. In order to reduce the conservativeness, an extra variable can be introduced, and Q is still in a general form. To this end, we recall the following lemma, which is useful for us to design the controller for the above systems:

Lemma 3.9. [30] The following statements are equivalent

i There exists a positive definite matrix P such that

$$A^T P A - P < 0. \tag{3.16}$$

ii There exist a positive definite matrix P and a matrix V such that

$$\begin{bmatrix} -Q & V^T A^T \\ AV & Q - V - V^T \end{bmatrix} < 0. \tag{3.17}$$

The following result is an extension of Theorem 3.7 with less conservatism.

Corollary 3.10. *System (3.11) is stable if there exist a positive definite matrix* $Q \in R^{(\tau+1)n\times(\tau+1)n}$ *and matrices* $V = \begin{bmatrix} V_{11} & 0 \\ V_{21} & V_{22} \end{bmatrix}$ *with* $V_{11} \in R^{n\times n}, V_{21} \in R^{\tau n\times n}, V_{22} \in R^{\tau n\times \tau n}$ *and* $Y \in R^{m\times n}$ *such that the following inequality holds*

$$\begin{bmatrix} -Q & V^T \bar{A}^T + I_1^T Y^T \bar{B}^T \\ \bar{A}V + \bar{B}YI_1 & Q - V - V^T \end{bmatrix} < 0. \tag{3.18}$$

Then, the closed-loop system is stable with the state feedback $K = YV_{11}^{-1}$.

Remark 3.11. From inequality (3.18), it can be deduced that V is invertible, otherwise, there exists $\eta \neq 0$ such that $V\eta = 0$, multiplying $Q - V^T - V < 0$

from both sides by η^T and η, respectively, leads to $\eta^T Q\eta < 0$, which contradicts the fact that $Q > 0$. Therefore, V_{11} is invertible. Inequality (3.18) is LMI and Lyapunov matrix Q is in a general form although extra variable V has a special structure.

3.3.2.2 Quadratic Stabilization for Uncertain Systems with Constant Delay

When there are uncertainties, system (3.1) can be written as

$$\begin{aligned} z(k+1) &= (\bar{A} + \bar{B}K \begin{bmatrix} I \; 0 \cdots 0 \; 0 \end{bmatrix} + \bar{D}F(k)(\tilde{E} + \bar{E}))z(k) \\ &= (\tilde{A} + \bar{D}F(k)(\tilde{E} + \bar{E}))z(k) \end{aligned} \tag{3.19}$$

where $\bar{A}$, $\bar{D}$ and $\bar{E}$ are defined in (3.7) and $\bar{B} = \begin{bmatrix} B^T \; 0 \cdots 0 \; 0 \end{bmatrix}^T$, $\tilde{E} = \begin{bmatrix} E_b K \; 0 \cdots 0 \; 0 \end{bmatrix}$. Then, we have the following result.

Corollary 3.12. *System (3.1) is quadratically stabilizable if there exist a positive definite matrix* $Q \in R^{(\tau+1)n\times(\tau+1)n}$ *and matrix* $V = \begin{bmatrix} V_{11} & 0 \\ V_{21} & V_{22} \end{bmatrix}$ *with* $V_{11} \in R^{n\times n}$, $V_{21} \in R^{\tau n\times n}$, $V_{22} \in R^{\tau n\times \tau n}$, $Y \in R^{m\times n}$ *and a scalar* λ *satisfying the following LMI*

$$\begin{bmatrix} -Q & V^T\bar{A}^T + I_1^T Y^T \bar{B}^T & 0 & V^T\bar{E}^T + I_1^T Y^T E_b^T \\ \bar{A}V + \bar{B}YI_1 & Q - V - V^T & \lambda\tilde{D} & 0 \\ 0 & \lambda\bar{D}^T & -\lambda I & 0 \\ \bar{E}V + E_bYI_1 & 0 & 0 & -\lambda I \end{bmatrix} < 0. \tag{3.20}$$

Then, the state feedback $K = YV_{11}^{-1}$ will result in the closed-loop system being quadratically stable.

Proof. From Lemma 3.9, system (3.1) is quadratically stable if (3.17) is satisfied, that is,

$$\begin{bmatrix} -Q & V^T(\tilde{A} + \bar{D}F(k)(\tilde{E} + \bar{E}))^T \\ (\tilde{A} + \bar{D}F(k)(\tilde{E} + \bar{E}))V & Q - V - V^T \end{bmatrix} < 0$$

which can be written as

$$\begin{aligned} &\begin{bmatrix} -Q & V^T\tilde{A}^T \\ \tilde{A}V & Q - V - V^T \end{bmatrix} + \begin{bmatrix} 0 \\ \bar{D} \end{bmatrix} F(k) \begin{bmatrix} (\tilde{E} + \bar{E})V \; 0 \end{bmatrix} \\ &+ [\begin{bmatrix} 0 \\ \bar{D} \end{bmatrix} F(k) \begin{bmatrix} (\tilde{E} + \bar{E})V \; 0 \end{bmatrix}]^T < 0. \end{aligned}$$

Using the same techniques as those in Corollary 3.4, the above inequality is satisfied if and only if the following inequality holds for any $\epsilon > 0$,

$$\begin{bmatrix} -Q + \epsilon V(\tilde{E}+\bar{E})^T(\tilde{E}+\bar{E})V & V^T\tilde{A}^T & 0 \\ \tilde{A}V & Q - V - V^T & \epsilon^{-1}\bar{D} \\ 0 & \epsilon^{-1}\bar{D}^T & -\epsilon^{-1}I \end{bmatrix} < 0.$$

Using Schur complement with respect to the term $\epsilon V(\tilde{E}+\bar{E})^T(\tilde{E}+\bar{E})V$, it yields

$$\begin{bmatrix} -Q & V^T\tilde{A}^T & 0 & V(\tilde{E}+\bar{E})^T \\ \tilde{A}V & Q - V - V^T & \epsilon^{-1}\bar{D} & 0 \\ 0 & \epsilon^{-1}\bar{D}^T & -\epsilon^{-1}I & 0 \\ (\tilde{E}+\bar{E})V & 0 & 0 & -\epsilon^{-1}I \end{bmatrix} < 0.$$

Let $\lambda = \epsilon^{-1}$ and $Y = KV_{11}$, the above inequality is equivalent to (3.20). The proof is completed.

3.3.2.3 Controller Design for Systems with Time-Varying Delay

When the time-delay is time-varying, the following stabilization condition can be derived based on Corollary 3.5.

Corollary 3.13. *The certain system (3.1) with time-varying delay $\tau(k)$ is stabilizable if there exist a positive definite matrix $Q \in R^{(\tau+1)n\times(\tau+1)n}$, matrices $V = \begin{bmatrix} V_{11} & 0 \\ V_{21} & V_{22} \end{bmatrix}$ with $V_{11} \in R^{n\times n}, V_{21} \in R^{\tau n\times n}, V_{22} \in R^{\tau n\times \tau n}$ and $Y \in R^{m\times n}$ such that the following inequalities hold for any $i \in \{0, 1, 2, \cdots, \bar{\tau}\}$:*

$$\begin{bmatrix} -Q & V^T\bar{A}_i^T + I_1^TY^T\bar{B}^T \\ \bar{A}_iV + \bar{B}YI_1 & Q - V - V^T \end{bmatrix} < 0. \tag{3.21}$$

Then, the closed-loop system is stable with the state feedback $K = YV_{11}^{-1}$.

Our last result in this chapter deals with the uncertain system (3.1) with time-varying delays. With the notations $\bar{A}_i$, $i = 0, 1, 2, \cdots, \bar{\tau}$ defined in (3.8), the following result is presented:

Corollary 3.14. *System (3.1) with time-varying delay $\tau(k)$ is quadratically stabilizable if there exists a positive definite matrix $Q \in R^{(\tau+1)n\times(\tau+1)n}$ and matrices $V = \begin{bmatrix} V_{11} & 0 \\ V_{21} & V_{22} \end{bmatrix}$ with $V_{11} \in R^{n\times n}$, $V_{21} \in R^{\tau n\times n}$, $V_{22} \in R^{\tau n\times \tau n}$, $Y \in R^{m\times n}$ and a scalar λ satisfying the following LMIs for $i = 0, 1, 2, \cdots, \bar{\tau}$*

$$\begin{bmatrix} -Q & V^T\bar{A}_i^T + I_1^TY^T\bar{B}^T & 0 & V^T\bar{E}^T + I_1^TY^TE_b^T \\ \bar{A}_iV + \bar{B}YI_1 & Q - V - V^T & \lambda\tilde{D} & 0 \\ 0 & \lambda\bar{D}^T & -\lambda I & 0 \\ \bar{E}V + E_bYI_1 & 0 & 0 & -\lambda I \end{bmatrix} < 0. \tag{3.22}$$

Then, the closed-loop system is quadratically stabilizable with the state feedback $K = YV_{11}^{-1}$.

Remark 3.15. The techniques developed in the chapter can also be easily extended to the problem of H_∞ and H_2 control of discrete-time systems with time-delay, and could be further extended to investigate the filtering problem and model reduction problem for discrete-time delay systems, such as those considered in [54, 52] and singular systems with time-delay [12, 35].

3.4 Numerical Example

Example 3.16. Consider system (3.1) with following system matrices and uncertain parameters ([200]):

$$\begin{aligned} A &= \begin{bmatrix} 1 & -0.6 \\ 0.4 & 0.5 \end{bmatrix}, A_d = \begin{bmatrix} 0.5 & 0.2 \\ 0.6 & 0.4 \end{bmatrix}, B = \begin{bmatrix} 0.1 & 0.2 \\ 0 & 0.1 \end{bmatrix}, M = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} \\ E_a &= \begin{bmatrix} 0.02 & 0.03 \end{bmatrix}, E_d = \begin{bmatrix} 0.02 & 0.01 \end{bmatrix}, E_b = \begin{bmatrix} 2 & 1.5 \end{bmatrix} \end{aligned} \tag{3.23}$$

which is unstable for the unforced nominal system ([200]). It is assumed time-delay $\tau(k)$ is constant, and $\tau = 2$ in [200]. When $\tau = 20$, based on Corollary 3.12, LMI (3.20) has a feasible solution

$$Y = \begin{bmatrix} 1.0343 & -0.2163 \\ -1.3821 & 0.2839 \end{bmatrix}, V_{11} = \begin{bmatrix} 0.2220 & -0.0295 \\ -0.0170 & 0.2727 \end{bmatrix}.$$

Then, the closed-loop system (3.1) is quadratically stable with $K = YV_{11}^{-1} = \begin{bmatrix} 4.6376 & -0.2918 \\ -6.1985 & 0.3708 \end{bmatrix}$ at least for any constant $\tau \leq 20$.

3.5 Conclusion

This chapter has considered the problems of stability and stabilization of discrete system with time-delay. A lifting method has been proposed to transform the discrete system with time-delay to a delay-free system. Stability and stabilization conditions have been established for constant and time-varying delay systems with or without uncertainties. Numerical examples have been included to demonstrate the advantage of the theoretic results obtained.

Chapter 4
Robust SMC for Uncertain Time-Delay Systems

4.1 Introduction

The control problem of time-delay systems has received considerable attention over the past years, and different design approaches have been proposed in [24], [128], [89], [31], [39], [136]. However, they are sensitive to the uncertainty, which directly affects the control systems.

An alternative approach is sliding mode control (SMC), which has many attractive features such as a fast response with asymptotic stability. The salient advantages of this method are: (i) when the state is constrained to the sliding surface, SMC can completely reject uncertainties which satisfy the matching condition; and (ii) the high possibility of stabilizing some complex non-linear systems which are difficult to stabilize by state feedback laws. Because of these advantages, variable structure control theory has found applications to various kinds of plants ([85]). The existence condition of a linear sliding surface for systems with mismatched uncertainties was given in [25], [26], [100], but their methods cannot be applied to systems with time-delay. In [161], a new robust stability criterion for uncertain time-delay systems is given and the SMC is proved to be applicable. There, due to use of matrix norm, the result is more or less conservative and complicated which leads to inconvenience in designing the sliding surface for uncertain time-delay systems.

In this chapter, we consider how to design sliding surface and reaching motion controller for a class of time-delay systems with mismatched uncertainties and matched exogenous disturbance. An LMI condition for the existence of linear sliding surfaces is derived. The solution to the condition can be used to characterize linear sliding surfaces, and by selecting suitable reaching law the reaching motion controller is designed. Our methods have the advantages in computation since the given stability condition is represented by the LMI which can be very efficiently solved by using powerful LMI algorithm [15]. Finally, we extend our results to the interval systems with time-delay.

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 37–48.
springerlink.com

The chapter is organized as follows. Section 4.2 gives the problem formulation and some preliminaries. In section 4.3, the results on designing sliding surface and reaching motion controller are established, and then, those results are extended to interval systems with time-delay. Numerical simulations are presented in Section 4.4 and some conclusion remarks are given in Section 4.5.

4.2 Problem Formulation

Consider the uncertain time-delay system of the form

$$\begin{aligned} \dot{x}(t) &= (A + \Delta A)x(t) + (A_d + \Delta A_d)x(t-\tau) \\ &\quad + B(u(t) + Fw(t)) \\ x(t) &= \varphi(t), t \in \left[-\tau, 0\right] \end{aligned} \tag{4.1}$$

where $x(t) \in R^n$ is the state, $w(t) \in R^l$ is the disturbance whose each component is bounded by the known $\overline{w}_i(t)$, i.e., $w_i(t) \leq \overline{w}_i(t)$, $i = 1, 2, \cdots, l$, $u(t) \in R^m$ is the control input, A, A_d, B and F are real constant matrices with appropriate dimensions and $\text{rank}(B) = m$. The model uncertainties are described by

$$\begin{aligned} \Delta A &= \textstyle\sum_{i=1}^{p} \alpha_i(t) A_i, \ |\alpha_i(t)| \leq 1 \\ \Delta A_d &= \textstyle\sum_{i=1}^{q} \beta_i(t) A_{di}, \ |\beta_i(t)| \leq 1 \end{aligned} \tag{4.2}$$

where the matrices $A_i, i = 1, \cdots, p$ and $A_{dj}, j = 1, \cdots, q$ are known with $\text{rank}(A_i) = a_i$ and $\text{rank}(A_{di}) = a_{di}$, $\alpha_i(t)$ and $\beta_i(t)$ are Lebesgue-measurable. Suppose A_i and A_{di} have full rank factorization of $A_i = G_i H_i$ and $A_{d_i} = G_{di} H_{di}$, respectively, where $G_i \in R^{n \times a_i}$, $H_i \in R^{a_i \times n}$, $G_{di} \in R^{n \times a_{di}}$ and $H_{di} \in R^{a_{di} \times n}$. Then

$$\begin{aligned} \Delta A &= \textstyle\sum_{i=1}^{p} \alpha_i(t) A_i = \left[\, G_1 \ G_2 \cdot\cdot G_p \,\right] D \left[\, H_1 \ H_2 \cdots \ H_p \,\right]^T = GDH \\ \Delta A_d &= \textstyle\sum_{i=1}^{q} \beta_i(t) A_{di} \\ &= \left[\, G_{d1} \ G_{d2} \cdot\cdot G_{dq} \,\right] D_d \left[\, H_{d1} \ H_{d2} \cdots \ H_{dq} \,\right]^T = G_d D_d H_d \end{aligned} \tag{4.3}$$

where

$$\begin{aligned} D &= \text{diag}\left(\alpha_1(t) I_{a_1 \times a_1} \ \alpha_2(t) I_{a_2 \times a_2} \ \cdots \ \alpha_p(t) I_{a_p \times a_p} \right) \\ D_d &= \text{diag}\left(\beta_1(t) I_{a_{d1} \times a_{d1}} \ \beta_2(t) I_{a_{d2} \times a_{d2}} \ \cdots \ \beta_q(t) I_{a_{dq} \times a_{dq}} \right). \end{aligned} \tag{4.4}$$

According to the structure of D and D_d, the following scaling matrices are defined as

$$\begin{aligned} S_D &= \{Y | Y = \text{diag}\left(Y_1 \ Y_2 \cdots Y_p \right) \\ &\quad 0 < Y_i = {Y_i}^T \in R^{a_i \times a_i}\} \\ S_{D_d} &= \{Y_d | Y_d = \text{diag}\left(Y_{d1} \ Y_{d2} \cdots Y_{dq} \right) \\ &\quad 0 < Y_{di} = {Y_{di}}^T \in R^{a_{di} \times a_{di}}\}. \end{aligned} \tag{4.5}$$

To get a regular form of the systems (4.1), a nonsingular matrix T can be chosen such that

$$TB = \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix}$$

where $B_2 \in R^{m\times m}$ is nonsingular. For convenience, let us choose

$$T = \begin{bmatrix} {U_2}^T \\ {U_1}^T \end{bmatrix}$$

where $U_1 \in R^{n\times m}$ and $U_2 \in R^{n\times(n-m)}$ are two sub-blocks of a unitary matrix resulting from the singular value decomposition of B, i.e.,

$$B = \begin{bmatrix} U_1 \; U_2 \end{bmatrix} \begin{bmatrix} \Sigma \\ 0_{(n-m)\times m} \end{bmatrix} V^T$$

where $\Sigma \in R^{m\times m}$ is a diagonal positive-definite matrix and $V \in R^{m\times m}$ is a unitary matrix. By the state transformation $z = Tx$, system (4.1) has the regular form

$$\begin{aligned} \dot{z}(t) &= (\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d)z(t-\tau) \\ &\quad + \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + Fw(t)) \\ z(t) &= \overline{\varphi}(t), t \in \begin{bmatrix} -\tau, \, 0 \end{bmatrix} \end{aligned} \tag{4.6}$$

where $\overline{A} = TAT^{-1}$, $\overline{A}_d = TA_dT^{-1}$, $\Delta\overline{A} = T\Delta AT^{-1}$, $\Delta\overline{A}_d = T\Delta A_dT^{-1}$ and $\overline{\varphi}(t) = T\varphi(t)$. System (4.6) can be written as:

$$\begin{aligned} \dot{z}_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11})z_1(t) + (\overline{A}_{d11} + \Delta\overline{A}_{d11})z_1(t-\tau) \\ &\quad +(\overline{A}_{12} + \Delta\overline{A}_{12})z_2(t) + (\overline{A}_{d12} + \Delta\overline{A}_{d12})z_2(t-\tau) \\ \dot{z}_2(t) &= (\overline{A}_{21} + \Delta\overline{A}_{21})z_1(t) + (\overline{A}_{d21} + \Delta\overline{A}_{d21})z_1(t-\tau) \\ &\quad +(\overline{A}_{22} + \Delta\overline{A}_{22})z_2(t) + (\overline{A}_{d22} + \Delta\overline{A}_{d22})z_2(t-\tau) \\ &\quad +B_2(u + Fw(t)) \\ z_1(t) &= \overline{\varphi}_1(t), t \in \begin{bmatrix} -\tau, \, 0 \end{bmatrix} \\ z_2(t) &= \overline{\varphi}_2(t), t \in \begin{bmatrix} -\tau, \, 0 \end{bmatrix} \end{aligned} \tag{4.7}$$

where $z_1 \in R^{n-m}$, $z_2 \in R^m$, $B_2 = \Sigma V^T$, $\overline{A}_{11} = U_2^T AU_2$, $\overline{A}_{12} = U_2^T AU_1$, $\overline{A}_{d11} = U_2^T A_dU_2$, $\overline{A}_{d12} = U_2^T A_dU_1$, $\Delta\overline{A}_{11} = {U_2}^T GDHU_2$, $\Delta\overline{A}_{12} = {U_2}^T GDHU_1$, $\Delta\overline{A}_{d11} = {U_2}^T G_dD_dH_dU_2$, $\Delta\overline{A}_{d12} = {U_2}^T G_dD_dH_dU_1$, $\overline{\varphi}_1(t) \in R^{(n-m)}$ and $\overline{\varphi}_2(t) \in R^m$ are the sub-blocks of $\overline{\varphi}(t)$.

It is obvious that the first equation of system (4.7) represents the sliding motion dynamics of system (4.6), and hence the corresponding sliding surface can be chosen as follows:

$$S = \begin{bmatrix} C \; I \end{bmatrix} z = Cz_1 + z_2 = 0 \tag{4.8}$$

where $C \in R^{m\times(n-m)}$. Substituting $z_2 = -Cz_1$ to the first equation of system (4.7) gives the sliding motion

$$\begin{aligned}\dot{z}_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C)z_1(t)\\ &\quad +(\overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C)z_1(t-\tau)\\ z_1(t) &= \overline{\varphi}_1(t), t \in \left[-\tau, 0\right].\end{aligned} \tag{4.9}$$

Definition 4.1. [128] The uncertain sliding motion (4.9) is said to be quadratically stable if there exist symmetric positive-definite matrices $P,\ Q \in R^{(n-m)\times(n-m)}$ and a constant $\xi > 0$ such that for any admissible uncertainty the derivative of the Lyapunov functional

$$V(z_1(t),t) = z_1^T(t)Pz_1(t) + \int_{t-\tau}^{t} z_1^T(s)Qz_1(s)ds \tag{4.10}$$

with respect to time t satisfies

$$L(z_1(t),t) = \dot{V}(z_1(t),t) \leq -\xi||z_1||^2 \tag{4.11}$$

for all pairs $(z_1(t),t) \in R^{n-m} \times R$.

The objective in this chapter is how to design constant gain $C \in R^{m\times(n-m)}$ and a reaching motion control law $u(t)$ such that

1) Sliding motion (4.9) is quadratically stable;

2) System (4.7) is asymptotically stable with the reaching control law $u(t)$.

To this end, the following lemmas are necessary.

Lemma 4.2. [100] Let $D \in S_D,\ D_d \in S_{D_d}$. Then for any $X \in S_D$ and $X_d \in S_{D_d}$, the following inequalities

$$GDH + (GDH)^T \leq GXG^T + H^TX^{-1}H \tag{4.12}$$

$$G_dD_dH_d + (G_dD_dH_d)^T \leq G_dX_dG_d{}^T + H_d{}^TX_d{}^{-1}H_d \tag{4.13}$$

hold.

Lemma 4.3. [102] Let $Q = Q^T,\ S,\ R = R^T$ be real matrices of appropriate dimensions, then

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} < 0 \tag{4.14}$$

is equivalent to

$$R < 0,\ Q - SR^{-1}S^T < 0 \tag{4.15}$$

which is usually called the Schur complement theorem.

4.3 Main Results

The first result of designing sliding surface can be stated as follows.

Theorem 4.4. *The reduced order system (4.9) is quadratically stable if there exist symmetric positive-definite matrices* $J \in R^{m\times m}$*,* $Z \in R^{m\times m}$*,* $X \in S_D$*,* $X_d \in S_{D_d}$ *and general matrix* $Y \in R^{m\times(n-m)}$ *such that*

$$\begin{bmatrix} -J & N^T & 0 & L^T \\ N & M & K^T & 0 \\ 0 & K & -X & 0 \\ L & 0 & 0 & -X_d \end{bmatrix} < 0 \tag{4.16}$$

where $L = H_dU_2Z - H_dU_1Y$, $M = J + Z\overline{A}_{11}^T + \overline{A}_{11}Z - Y^T\overline{A}_{12}^T - \overline{A}_{12}Y + U_2^TGXG^TU_2 + U_2^TG_dX_dG_d^TU_2$, $N = \overline{A}_{d11}Z - \overline{A}_{d12}Y$, $K = HU_2Z - HU_1Y$. Moreover, the sliding surface of the system (4.7) is

$$S(t) = YZ^{-1}z_1(t) + z_2(t) = 0. \tag{4.17}$$

Proof. Take symmetric positive-definite matrix variables $P, Q \in R^{(n-m)\times(n-m)}$ and choose a Lyapunov functional as

$$V(z_1,t) = z_1^T(t)Pz_1(t) + \int_{t-\tau}^{t} z_1^T(s)Qz_1(s)ds \tag{4.18}$$

which is positive-define for all $z_1(t) \neq 0$, it follows that the Lyapunov derivative corresponding to the system (4.9) is given by

$$\dot{V}(z_1,t) = \begin{bmatrix} Pz_1 \\ Pz_1(t-\tau) \end{bmatrix}^T W \begin{bmatrix} Pz_1 \\ Pz_1(t-\tau) \end{bmatrix} \tag{4.19}$$

where $\tilde{A}_{11} = \overline{A}_{11} - \overline{A}_{12}C$, $\Delta\tilde{A}_{11} = \Delta\overline{A}_{11} - \Delta\overline{A}_{12}C = U_2^TGDH(U_2 - U_1C)$, $\tilde{A}_{d11} = \overline{A}_{d11} - \overline{A}_{d12}C$, $\Delta\tilde{A}_{d11} = \Delta\overline{A}_{d11} - \Delta\overline{A}_{d12}C = U_2^TG_dD_dH_d(U_2 - U_1C)$, $Z = P^{-1}$, $J = ZQZ$ and

$$\begin{aligned} W = &\begin{bmatrix} Z\tilde{A}_{11}^T + \tilde{A}_{11}Z + J & \tilde{A}_{d11}Z \\ Z\tilde{A}_{d11}^T & -J \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} Z\Delta\tilde{A}_{11}^T \begin{bmatrix} I & 0 \end{bmatrix} \\ &+ \begin{bmatrix} I \\ 0 \end{bmatrix} \Delta\tilde{A}_{11}Z \begin{bmatrix} I & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ I \end{bmatrix} Z\Delta\tilde{A}_{d11}^T \begin{bmatrix} I & 0 \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} \Delta\tilde{A}_{d11}Z \begin{bmatrix} 0 & I \end{bmatrix} \end{aligned}$$

it follows from Lemma 4.2 that for any matrices $X \in S_D$ and $X_d \in S_{D_d}$

$$\begin{aligned}&\begin{bmatrix} I \\ 0 \end{bmatrix} Z\Delta\tilde{A}_{11}^{T} \begin{bmatrix} I \ 0 \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} \Delta\tilde{A}_{11} Z \begin{bmatrix} I \ 0 \end{bmatrix} \\ &\leq \begin{bmatrix} I \\ 0 \end{bmatrix} (U_2^T G X G^T U_2 + Z(U_2 - U_1 C)^T H^T \\ &\quad \times X^{-1} H (U_2 - U_1 C) Z) \begin{bmatrix} I \ 0 \end{bmatrix}\end{aligned} \tag{4.20}$$

and

$$\begin{aligned}&\begin{bmatrix} 0 \\ I \end{bmatrix} Z\Delta\tilde{A}_{11}^{T} \begin{bmatrix} I \ 0 \end{bmatrix} + \begin{bmatrix} I \\ 0 \end{bmatrix} \Delta\tilde{A}_{d11} Z \begin{bmatrix} 0 \ I \end{bmatrix} \\ &\leq \begin{bmatrix} 0 \\ I \end{bmatrix} Z(U_2 - U_1 C)^T H_d^T X_d^{-1} H_d (U_2 - U_1 C) Z \begin{bmatrix} 0 \ I \end{bmatrix} \\ &\quad + \begin{bmatrix} I \\ 0 \end{bmatrix} U_2^T G_d X_d G_d^T U_2 \begin{bmatrix} I \ 0 \end{bmatrix}\end{aligned} \tag{4.21}$$

and, hence, by defining $Y = CZ$, it can be shown that

$$W \leq \begin{bmatrix} V_{11} & \overline{A}_{d11} Z - \overline{A}_{d12} Y \\ Z\overline{A}_{d11}^T - Y^T \overline{A}_{d12}^T & V_{22} \end{bmatrix} \tag{4.22}$$

where $V_{11} = J + Z\overline{A}_{11}^T - Y^T \overline{A}_{12}^T + \overline{A}_{11} Z - \overline{A}_{12} Y + U_2^T G X G U_2^T + (ZU_2^T - Y^T U_1^T) \times H^T X^{-1} H (U_2 Z - U_1 Y) + U_2^T G_d X_d G_d^T U_2$, $V_{22} = -J + (ZU_2^T - Y^T U_1^T) H_d^T X_d^{-1} H_d (U_2 Z - U_1 Y)$.

Note that Lemma 4.3 implies the equivalence of (4.16) and

$$\begin{bmatrix} V_{11} & \overline{A}_{d11} Z - \overline{A}_{d12} Y \\ Z\overline{A}_{d11}^T - Y^T \overline{A}_{d12}^T & V_{22} \end{bmatrix} < 0.$$

Hence, $W < 0$, which also implies that there exists a sufficiently small $\xi_1 > 0$ such that

$$W + \begin{bmatrix} \xi_1 I_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} \\ 0_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} \end{bmatrix} < 0.$$

It follows from (4.19) and above inequality that $\dot{V}(z_1(t), t) \leq -\xi_1 ||Pz_1||^2 \leq -\xi_1 \lambda_{\min}(P) ||z_1||^2$ for all $(z_1(t), t) \in R^{n-m} \times R$. Therefore inequality (4.11) is satisfied with $\xi = \xi_1 \lambda_{min}(P) > 0$, and the reduced system (4.9) is quadratically stable with $C = Y^{-1} Z$. Moreover the sliding surface of the system (4.7) is

$$S(t) = \begin{bmatrix} C \ I \end{bmatrix} z = YZ^{-1} z_1(t) + z_2(t) = 0. \tag{4.23}$$

The proof is completed.

Next the result of designing of reaching motion controller is given.

Theorem 4.5. *Suppose (4.16) have solutions J, Z, Y, X, X_d and the linear sliding surface is given by (4.17). Then the trajectory of the closed-loop system*

(4.7) can be driven onto the sliding surface in limited time with the control

$$\begin{aligned}u = &-B_2^{-1}[KS + \epsilon\mathrm{sgn}(S) + \overline{CA}z(t) + \overline{CA}_d z(t-\tau) \\ &+ \mathrm{diag}\left(\mathrm{sgn}(s_1)\ \mathrm{sgn}(s_2)\ \cdots\ \mathrm{sgn}(s_m)\right)(N_1 + N_2 + N_3)]\end{aligned} \tag{4.24}$$

where

$$\begin{aligned}
&S^T = \left[\, s_1\ s_2\ \cdots\ s_m \,\right]^T, \mathrm{sgn}^T(S) = \left[\, \mathrm{sgn}(s_1)\ \mathrm{sgn}(s_2)\ \cdots\ \mathrm{sgn}(s_m) \,\right]^T \\
&w^T = \left[\, w_1\ w_2\ \cdots\ w_m \,\right], (B_2F)^T = \left[\, b_1\ b_2\ \cdots\ \ b_m \,\right]^T \\
&N_1^T = \left[\, N_{11}\ N_{12}\ \cdots\ N_{1m} \,\right]^T, N_2^T = \left[\, N_{21}\ N_{22}\ \cdots\ N_{2m} \,\right]^T \\
&N_3^T = \left[\, N_{31}\ N_{32}\ \ \cdots\ N_{3m} \,\right]^T, \overline{C}^T = \left[\, \overline{c}_1\ \overline{c}_2\ \cdots\ \overline{c}_m \,\right]^T \\
&\overline{C} = \left[\, C\ I \,\right], N_{1i} = \sum_{j=1}^{p} |\overline{c}_i T A_j T^{-1} z(t)|, N_{2i} = \sum_{j=1}^{q} |\overline{c}_i T A_{dj} T^{-1} z(t-\tau)| \\
&N_{3i} = \sum_{j=1}^{m} |b_{ij}|\overline{w}_j, K = \mathrm{diag}(k_i), \epsilon = \mathrm{diag}(\epsilon_i)
\end{aligned}$$

in which k_i and ϵ_i are positive constants.

Proof. We will complete the proof by showing that the control law (4.24) not only can drive the system trajectory onto the linear sliding surface, but also keep it there for all subsequent time. From the sliding surface

$$S = \left[\, C\ I \,\right] z$$

we have

$$\begin{aligned}\dot{S} = \left[\, C\ I \,\right] \dot{z}\ &= \overline{C}(\overline{A} + \Delta\overline{A})z(t) + \overline{C}(\overline{A}_d + \Delta\overline{A}_d)z(t-\tau) \\ &+ B_2(u(t) + Fw(t)).\end{aligned} \tag{4.25}$$

and from the inequalities

$$\overline{c}_i \Delta\overline{A}z(t) = \overline{c}_i \sum_{j=1}^{p} \alpha_j(t) T A_j T^{-1} z(t) \le \sum_{j=1}^{p} |\overline{c}_i T A_j T^{-1} z(t)| = N_{1i}$$

$$\begin{aligned}\overline{c}_i \Delta\overline{A}_d z(t-\tau) &= \overline{c}_i \sum_{j=1}^{q} \beta_j(t) T A_{dj} T^{-1} z(t-\tau) \\ &\le \sum_{j=1}^{q} |\overline{c}_i T A_j T^{-1} z(t-\tau)| = N_{2i}\end{aligned}$$

and

$$b_i w = \sum_{j=1}^{m} b_{ij} w_j \leq \sum_{j=1}^{m} |b_{ij}| \overline{w}_j = N_{3i}$$

we deduce that each element of (4.25) with the control law (4.24), i.e.,

$$\begin{aligned} \dot{s}_i = & -k_i s_i - \epsilon_i \text{sgn}(s_i) - (N_{i1} \text{sgn}(s_i) - \overline{c}_i \Delta \overline{A} z(t)) \\ & -(N_{2i} \text{sgn}(s_i) - \overline{c}_i \Delta \overline{A}_d z(t-\tau)) - (N_{3i} \text{sgn}(s_i) - b_i w) \end{aligned} \tag{4.26}$$

satisfies the reaching condition:

$$\begin{cases} \dot{s}_i < 0, & \text{if } s_i > 0 \\ \dot{s}_i > 0, & \text{if } s_i < 0 \end{cases}$$

which shows that the trajectory of the system (4.7) can be driven onto the sliding surface in limited time by the control law (4.24) and be maintained there. Thus the proof is completed.

When time-delay τ is an unknown constant, the control law (4.24) is not applicable for the terms of $\overline{CA}_d z(t-\tau)$ and $N_i (i = 1, 2, 3)$ can not be obtained in practice. The following control law will solve this problem.

Theorem 4.6. *Assume that time-delay is an unknown constant, but it is bounded by the known constant $\overline{\tau}$. Write the solutions of (4.16) as J, Z, Y, X, X_d and let the linear sliding surface be given by the equation (4.17). Then the trajectory of the closed-loop system (4.7) can be driven onto the sliding surface in limited time with the control*

$$\begin{aligned} u = & -B_2^{-1} [KS + \epsilon \text{sgn}(S) + \overline{CA} z(t) \\ & + \text{diag} \left(\text{sgn}(s_1) \ \text{sgn}(s_2) \ \cdots \ \text{sgn}(s_m) \right) (N_1 + N_2 + N_3)] \end{aligned} \tag{4.27}$$

where $N_{2i} = q|\overline{c}_i|(|TA_d T^{-1}| + \sum_{j=1}^{q} |TA_{dj} T^{-1}|)|z(t)|$, $S, w, B_2 F, N_1, N_2, N_3$, $\overline{C}$, N_{1i}, N_{3i}, K and ϵ are defined as in Theorem 4.5.

Proof. It follows from the Razumikhin theorem [68] that for any solution $z(t + \theta)$ of system (4.7), there exists a constant $q > 1$ such that

$$|z(t+\theta)| \leq q|z(t)|, -\overline{\tau} \leq \theta \leq 0 \tag{4.28}$$

which leads to the following inequality:

$$\begin{aligned} \overline{c}_i \overline{A}_d z(t-\tau) & = \overline{c}_i T (A_d + \sum_{j=1}^{q} \beta_j A_{dj}) T^{-1} z(t-\tau) \\ & \leq q |\overline{c}_i| (|TA_d T^{-1}| + \sum_{j=1}^{q} |TA_{dj} T^{-1}|) |z(t)| \\ & = N_{2i} \end{aligned}$$

The rest of the proof is similar to that of Theorem 4.5, and omitted here. The proof is completed.

Finally, we remark the results in Theorems 4.4-4.5 can be extended to the time-delay interval systems of the form

$$\begin{aligned} \dot{x}(t) &= Ax(t) + A_d x(t-\tau) + B(u(t) + Fw(t)) \\ x(t) &= \varphi(t), t \in \left[-\tau \ 0 \right] \end{aligned} \tag{4.29}$$

where $w(t)$ is defined as in (4.1), and A and A_d are such matrices whose entries vary in the prescribed ranges:

$$A = (a_{ij}),\ \underline{a}_{ij} \le a_{ij} \le \overline{a}_{ij}, A_d = (a_{dij}),\ \underline{a}_{dij} \le a_{dij} \le \overline{a}_{dij} \tag{4.30}$$

Let

$$A_0 = (a_{0ij}) = (\frac{\underline{a}_{ij} + \overline{a}_{ij}}{2}), D = (\overline{a}_{0ij}) = (\frac{\overline{a}_{ij} - \underline{a}_{ij}}{2}) \tag{4.31}$$

$$A_{d0} = (a_{d0ij}) = (\frac{\underline{a}_{dij} + \overline{a}_{dij}}{2}), D_d = (\overline{a}_{d0ij}) = (\frac{\overline{a}_{dij} - \underline{a}_{dij}}{2}) \tag{4.32}$$

then A and A_d can be re-expressed as

$$\begin{aligned} A &= A_0 + \textstyle\sum_{i,j=1}^{n} k_{ij} e_i e_i^T D e_j e_j^T,\ (|k_{ij}| \le 1) \\ A_d &= A_{d0} + \textstyle\sum_{i,j=1}^{n} k_{dij} e_i e_i^T D_d e_j e_j^T,\ (|k_{dij}| \le 1) \end{aligned} \tag{4.33}$$

where $e_i = \underbrace{[\ 0 \cdots 0}_{i-1}\ 1\ 0 \cdots 0\ ]^T \in R^n$.

If $k_{ij} e_i e_i^T D e_j e_j^T = 0$ and $e_l e_l^T D_d e_m e_m^T = 0$ occur for some i, j, l, m, then A and A_d can be also written as

$$A = A_0 + \sum_{i}^{p} k_i A_i; \quad A_d = A_{d0} + \sum_{i}^{q} k_{di} A_{di} \tag{4.34}$$

where $|k_i| \le 1$, $|k_{di}| \le 1$, p and q are some positive integers. Thus, the system (20) is transformed into

$$\begin{aligned} \dot{x}(t) &= (A_0 + \textstyle\sum_{i}^{p} k_i A_i) x(t) + (A_{d0} + \textstyle\sum_{i}^{q} k_{di} A_{di}) x(t-\tau) \\ &\quad + B(u(t) + Fw(t)) \\ x(t) &= \varphi(t), t \in \left[-\tau, 0 \right] \end{aligned} \tag{4.35}$$

which is just the form of system (4.7), so the previous result can be applied to deal with the interval time-delay system (4.29).

4.4 Numerical Example

In this section an illustrative example is given for testing the design method developed in this chapter.

Example 4.7. Consider the interval time-delay system (4.29) with

$$A = \left[\begin{bmatrix} -5 & -3 \\ -3.5 & 4.5 \end{bmatrix} \ \begin{bmatrix} 1 & 2 \\ -3 & -1 \end{bmatrix} \right],\ B = \begin{bmatrix} 1 \\ 0 \end{bmatrix},\ A_d = \left[\begin{bmatrix} 0.5 & 0.9 \\ 0.4 & 0.8 \end{bmatrix} \ \begin{bmatrix} 0.4 & 1.2 \\ 0.4 & 1.4 \end{bmatrix} \right]$$

$F = 1$, $\tau = 0.5$, $\varphi(t) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, for $t \in \begin{bmatrix} -\tau & 0 \end{bmatrix}$, $|w(t)| \leq 0.8|\sin t|$.

Write A and A_d as

$$A = A_0 + k_1 A_1 + k_2 A_2 + k_3 A_3 + k_4 A_4$$
$$A_d = A_{d0} + k_{d1} A_{d1} + k_{d2} A_{d2} + k_{d3} A_{d3} + k_{d4} A_{d4}$$

where $|k_i| \leq 1$ and $|k_{di}| \leq 1$, $i = 1, 2, 3, 4$. A_i and A_{di} are factorized as follows:

$$A_0 = \begin{bmatrix} -4 & 1.5 \\ 4 & -2 \end{bmatrix},\ A_{d0} = \begin{bmatrix} 0.7 & 0.8 \\ 0.6 & 0.9 \end{bmatrix}$$
$$A_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix},\ A_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0.5 \end{bmatrix}$$
$$A_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0.5 & 0 \end{bmatrix},\ A_4 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix}$$
$$A_{d1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0.2 & 0 \end{bmatrix},\ A_{d2} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0.4 \end{bmatrix}$$
$$A_{d3} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0.2 & 0 \end{bmatrix},\ A_{d4} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 0.5 \end{bmatrix}.$$

According to (4.3), it can be shown that

$$G = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix},\ H = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \\ 0.5 & 0 \\ 0 & 1 \end{bmatrix}$$
$$G_d = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix},\ H_d = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.4 \\ 0.2 & 0 \\ 0 & 0.5 \end{bmatrix}.$$

Taking $T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, and LMI (4.16) has feasible solutions:

$$J = 1.484, Z = 0.3230, Y = 0.9971$$
$$X = \text{diag}(1.5910, 0.9600, 1.1177, 1.0027)$$
$$X_d = \text{diag}(1.0886, 0.9950, 0.9912, 0.9524).$$

It follows from Theorem 4.4 that $C = 3.0874$, the linear sliding surface is $S(t) = \begin{bmatrix} 1 & 3.0874 \end{bmatrix} x = 0$.

From Theorem 4.5, the reaching control law can be taken as follows

$$u(t) = -[Ks + \epsilon \mathrm{sgn}(s) \begin{bmatrix} 8.3496 & -4.6748 \end{bmatrix} x(t) + \begin{bmatrix} 2.5524 & 3.5787 \end{bmatrix} x(t-0.5) \\ +\mathrm{sgn}(s)(N_1 + N_2 + N_3)]$$

where

$N_1 = 2.5437|x_1(t)| + 3.5874|x_2(t)|$, $N_2 = 0.8175|x_1(t-0.5)| + 1.9437|x_2(t-0.5)|$, $N_3 = 0.8|\sin t|$.

The parameter K and ϵ can be tuned to reduce the chattering on the sliding surface. Fig. 4.1 is simulation result when choosing $K = 15.5$ and $\epsilon = 1$. Obviously the system is asymptotically stable and the sliding motion trends to the origin in finite time in spite of time-delay and uncertainties.

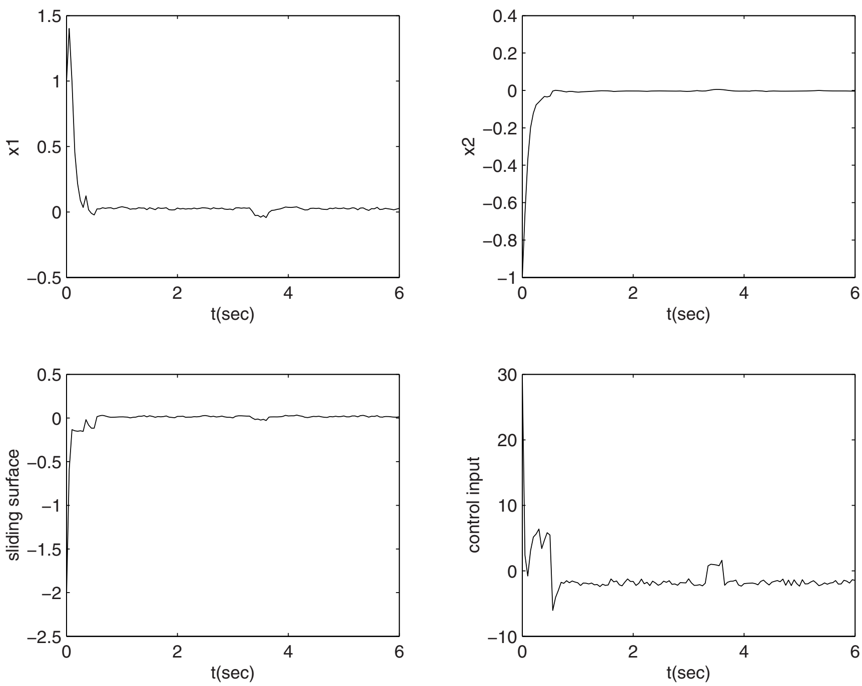

Fig. 4.1 States(x_1, x_2), sliding surface (S) and control input

4.5 Conclusion

In this chapter, the problem of designing robust sliding surfaces based on quadratic stability for a class of uncertain time-delay systems has been considered in which no matching condition is assumed for the state uncertainties. In terms of LMI, sufficient condition is derived for the existence of a linear sliding surface guaranteeing quadratic stability of the reduced-order equivalent system restricted to the sliding surface. A new reaching motion controller is proposed for uncertain time-delay systems by the reaching law. Both the sliding motion and the reaching motion are robust against the mismatched uncertainties and matched external disturbance. The results are also extended to the interval systems with time-delay. The simulation results show that the proposed methods are amenable and generalize previous results available in the literature to date.

Chapter 5
Robust Delay-Dependent SMC for Uncertain Time-Delay Systems

5.1 Introduction

Also, other SMC schemes were proposed for linear systems with state delay ([161, 100]), and systems with input delay ([147, 148]). In the work of [147, 148], methods were proposed for uncertain linear systems with input delay, where the nonlinear parametric perturbations contained in the systems are assumed to satisfy the matching conditions. Recently, Kim and Park [100] presented delay-independent conditions for the existence of sliding mode for systems with time-delay. The effect of uncertainties in the sliding mode, including matching and unmatching conditions, has been considered in [161], in which a variable structure controller is obtained. Based on Lyapunov functional method, the conditions for the existence of sliding mode for the nominal system are independent of the size of the delay. In the above proposed conditions, the time delay is allowed to be arbitrarily large and thus, in general, the results obtained by these conditions are conservative, especially when the system stability depends on the size of the time delay.

In this chapter, the problem of designing both a linear sliding surface and reaching motion controller for a class of uncertain time-delay systems is considered. The uncertainties of the linear portion are assumed to have the matrix polytope structure while the input matrix uncertainty is matched. A delay-dependent LMI existence condition of a linear sliding surface for such systems is derived. The solution obtained can be used to characterize linear sliding surfaces, and by selecting a suitable reaching law, the reaching motion controller is designed. The methods proposed can be very efficiently solved by the LMI toolbox ([15]) since the given stability condition is represented by the LMI. The results obtained in this chapter have extended the work in [189]. More precisely, different types of Lyapunov function have been proposed to analyze the stability of the sliding motion on the sliding surface. Consequently, the conditions for the existence of sliding surface are delay-dependent while they are delay-independent in [189]. Since the delay-dependent criterion makes use of information on the length of delays, they

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 49–65.
springerlink.com

are less conservative than delay-independent ones. A numerical example has been presented to show the advantages and the applicability of the developed techniques.

The chapter is organized as follows. Section 5.2 gives the problem formulation and some preliminaries. In section 5.3, the results on designing sliding surface and reaching motion controller are established. Numerical simulations are presented in Section 5.4 and some conclusion remarks are given in Section 5.5.

5.2 Problem Formulation

Consider the following uncertain time-delay system described by

$$\begin{array}{l} \dot{x}(t) = (A + \Delta A)x(t) + (A_d + \Delta A_d)x(t - \tau) + B(u(t) + Fw(t)) \\ x(t) = \varphi(t),\ t \in [-\tau, \quad 0] \end{array} \tag{5.1}$$

where $x(t) \in R^n$, $u(t) \in R^m$, and $w(t) \in R^l$ are the states, system inputs, and external disturbances, respectively, $A,\ A_d,\ B$ and F are matrices of appropriate dimensions and $\text{rank}(B) = m$, $\tau \geq 0$ is a constant time delay (known or unknown), and $\varphi(\cdot)$ denotes the initial condition. The uncertainty and disturbance are assumed to satisfy the following conditions:

$$\begin{array}{l} 1)\ \Delta A = \sum_{i=1}^{p} \alpha_i A_i, \quad |\alpha_i| \leq 1; \qquad \Delta A_d = \sum_{i=1}^{q} \beta_i A_{di}, \quad |\beta_i| \leq 1 \\ 2)\ |w_i(t)| \leq \overline{w}_i(t) \end{array} \tag{5.2}$$

where the constant matrices $A_i{}'s$, $A_{di}{}'s$ are known, $\text{rank}(A_i) = a_i$, $\text{rank}(A_{di}) = a_{di}$, α_i and β_i are unknown scalars, and $\overline{w}_i(t)$ $(i = 1, 2, \cdots, p)$ are known non-negative functions, $w_i(t)$ is the i-th entry of $w(t) \in R^l$.

Suppose that $A_i = G_i H_i$ and $A_{d_i} = G_{di} H_{di}$ are full rank factorizations of A_i and A_{di}, respectively, and $G_i \in R^{n \times n_{a_i}}$, $H_i \in R^{n_{a_i} \times n}$, $G_{di} \in R^{n \times n_{a_{di}}}$ and $H_{di} \in R^{n_{a_{di}} \times n}$, then

$$\begin{array}{c} \Delta A = \sum_{i=1}^{p} \alpha_i A_i = GDH \\ \Delta A_d = \sum_{i=1}^{q} \beta_i A_{di} = G_d D_d H_d \end{array} \tag{5.3}$$

where

$$\begin{array}{r} G = [G_1 \ \cdots \ G_p], G_d = [G_{d1} \ \cdots \ G_{dq}] \\ H = [H_1^T \ \cdots \ H_p^T]^T, H_d = [H_{d1}^T \ \cdots \ H_{dq}^T]^T \\ D = \text{diag}(\alpha_1 I_{n_{a_1} \times n_{a_1}} \ \cdots \ \alpha_p I_{n_{a_p} \times n_{a_p}}) \\ D_d = \text{diag}(\beta_1 I_{n_{a_{d1}} \times n a_{d1}} \ \cdots \ \beta_q I_{n_{a_{dq}} \times n_{a_{dq}}}). \end{array} \tag{5.4}$$

To obtain a regular form of system (5.1), many methods can be used, such as QR reduction (see, [37, 221]). Here, a nonsingular matrix T is chosen such that

$$TB = \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} \tag{5.5}$$

where $B_2 \in R^{m\times m}$ is nonsingular. For convenience, choose the state transformation $z(t) = Tx(t)$, where

$$T = \begin{bmatrix} {U_2}^T \\ {U_1}^T \end{bmatrix}$$

$U_1 \in R^{n\times m}$ and $U_2 \in R^{n\times(n-m)}$ are two sub-blocks of an unitary matrix resulting from the singular value decomposition of B, i.e.,

$$B = \begin{bmatrix} U_1 \ U_2 \end{bmatrix} \begin{bmatrix} \Sigma \\ 0_{(n-m)\times m} \end{bmatrix} \Gamma^T. \tag{5.6}$$

Then, system (5.1) can be written as

$$\begin{aligned} &\dot{z}(t) = (\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d)z(t-\tau) + \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + Fw(t)) \\ &z(t) = \overline{\varphi}(t),\ t \in [-\tau, \quad 0] \end{aligned} \tag{5.7}$$

where

$$\begin{aligned} &\overline{A} = TAT^{-1},\ \overline{A}_d = TA_dT^{-1},\ \Delta\overline{A} = T\Delta AT^{-1} \\ &\Delta\overline{A}_d = T\Delta A_dT^{-1},\ B_2 = \Sigma\Gamma^T,\ \overline{\varphi}(t) = T\varphi(t). \end{aligned}$$

Then, the first $(n-m)$ entries of $z(t)$ of system (5.7) can be written as

$$\begin{aligned} \dot{z}_1(t) = {} & (\overline{A}_{11} + \Delta\overline{A}_{11})z_1(t) + (\overline{A}_{d11} + \Delta\overline{A}_{d11})z_1(t-\tau) \\ & + (\overline{A}_{12} + \Delta\overline{A}_{12})z_2(t) + (\overline{A}_{d12} + \Delta\overline{A}_{d12})z_2(t-\tau) \end{aligned} \tag{5.8}$$

$$z_1(t) = \overline{\varphi}_1(t), t \in \begin{bmatrix} -\tau, 0 \end{bmatrix} \tag{5.9}$$

where $z_1(t) \in R^{n-m}$, $\overline{\varphi}_1(t) = U_2^T\varphi(t) \in R^{(n-m)}$ is the sub-blocks of $\overline{\varphi}(t)$,

$$\begin{aligned} &\overline{A}_{11} = U_2^TAU_2,\ \Delta\overline{A}_{11} = {U_2}^TGDHU_2,\ \overline{A}_{d11} = U_2^TA_dU_2 \\ &\overline{A}_{12} = U_2^TAU_1,\ \Delta\overline{A}_{12} = {U_2}^TGDHU_1,\ \overline{A}_{d12} = U_2^TA_dU_1 \\ &\Delta\overline{A}_{d11} = {U_2}^TG_dD_dH_dU_2,\ \Delta\overline{A}_{d12} = {U_2}^TG_dD_dH_dU_1. \end{aligned}$$

Without loss of generality, define the following sliding surface:

$$S = \begin{bmatrix} C \ I \end{bmatrix} z = Cz_1 + z_2 = 0 \tag{5.10}$$

where the gain $C \in R^{m\times(n-m)}$. Substituting $z_2 = -Cz_1$ to (5.8) gives the sliding motion

$$\begin{aligned}\dot{z}_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C)z_1(t) \\ &\quad +(\overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C)z_1(t-\tau) \\ z_1(t) &= \overline{\varphi}_1(t),\ t \in [-\tau, \quad 0]\end{aligned} \tag{5.11}$$

which can also be written as

$$\begin{aligned}\dot{z}_1(t) &= \tilde{A}_1 z_1(t) + \tilde{A}_{d1} z_1(t-\tau) \\ z_1(t) &= \overline{\varphi}_1(t),\ t \in [-\tau, \quad 0]\end{aligned} \tag{5.12}$$

where

$$\tilde{A}_1 = \overline{A}_1 + \Delta\overline{A}_1, \overline{A}_1 = U_2^T A(U_2 - U_1 C), \Delta\overline{A}_1 = U_2^T GDH(U_2 - U_1 C) \tag{5.13}$$

$$\tilde{A}_{d1} = \overline{A}_{d1} + \Delta\overline{A}_{d1}, \overline{A}_{d1} = U_2^T A_d(U_2 - U_1 C), \Delta\overline{A}_{d1} = U_2^T G_d D_d H_d(U_2 - U_1 C). \tag{5.14}$$

Now, we recall the following definition:

Definition 5.1. [31] The uncertain sliding motion (5.12) is said to be robustly stable if the equilibrium solution $z_1(t) = 0$ of the functional differential equation associated to sliding motion (5.12) is globally uniformly asymptotically stable for all admissible uncertainties $\Delta\overline{A}_1$ and $\Delta\overline{A}_{d1}$.

The objective in this chapter is to design a constant gain C and a reaching motion control law $u(t)$ such that

1) The sliding motion (5.12) is robustly stable; and
2) System (5.7) is asymptotically stable with the reaching control law $u(t)$.

5.3 Main Results

This section will present the main results on designing a sliding surface and reaching control law.

It is difficult to calculate the gain C which guarantees robust stability of (5.12), because the gain C appears not only in the system matrix $\tilde{A}_1$, but also in the delayed system matrix $\tilde{A}_{d1}$. Many techniques have been proposed to deal with the problem of stability, such as robust stability and robust stabilization. These techniques depend on the size of time-delay ([146, 101, 98, 84, 214]), but they cannot be applied to solve this problem directly. In this chapter, the approach developed makes use of some appropriate Lyapunov-Krasovskii functionals combined with some appropriate matrix inequalities, and the results obtained are expressed in terms of linear matrix inequalities which can be solved in a numerical way very efficiently using LMI algorithms ([15, 14, 12]).

The design consists of two stages. Firstly, a linear sliding surface is designed, then in the second stage, the robust reaching motion control law is derived. To facilitate the presentation of the main results, the following preliminary result is needed.

According to the structure of D and D_d defined in (5.3)-(5.4), we define the following scaling matrices which can be used as design parameters.

$$S_D = \{Y|Y = \mathrm{diag}\left(Y_1 \; Y_2 \; \cdots \; Y_p \right), 0 < Y_i = Y_i^T \in R^{a_i \times a_i}\} \tag{5.15}$$

$$S_{D_d} = \{Y_d|Y_d = \mathrm{diag}\left(Y_{d1} \; Y_{d2} \; \cdots \; Y_{dq} \right), 0 < Y_{di} = {Y_{di}}^T \in R^{a_{di} \times a_{di}}\}. \tag{5.16}$$

Let us recall the following lemma which will be used in the proof of our main results.

Lemma 5.2. [100] Let G, G_d and H be real matrices of appropriate dimensions and $D \in S_D$, $D_d \in S_{D_d}$. Then we have
(a) for any $X \in S_D$,

$$GDH + (GDH)^T \leq GXG^T + H^T X^{-1} H \tag{5.17}$$

(b) for any $X_d \in S_{D_d}$,

$$G_d D_d H_d + (G_d D_d H_d)^T \leq G_d X_d {G_d}^T + {H_d}^T {X_d}^{-1} H_d. \tag{5.18}$$

5.3.1 Design of Linear Sliding Surface

Now, we are ready to present our first result in this chapter.

Theorem 5.3. *Consider the reduced-order system (5.12). Given a scalar* $\overline{\tau} > 0$*, this system is robustly stable for any constant time delay* τ *satisfying* $0 \leq \tau \leq \overline{\tau}$ *if there exist symmetric positive-definite matrices* P*,* Q*,* S_1 *and* S_2*,* $X_1, X_2 \in S_D$ $X_{d1}, X_{d2}, X_{d3} \in S_{Dd}$*,* $V \in R^{m\times(n-m)}$ *satisfying the following LMIs:*

$$\begin{bmatrix} \Pi_{11} & \Pi_{12}^T & \Pi_{13}^T & \Pi_{14}^T & \Pi_{15}^T & S_1 & S_2 \\ \Pi_{12} & -\Pi_{22} & 0 & 0 & 0 & 0 & 0 \\ \Pi_{13} & 0 & -X_1 & 0 & 0 & 0 & 0 \\ \Pi_{14} & 0 & 0 & -X_{d1} & 0 & 0 & 0 \\ \Pi_{15} & 0 & 0 & 0 & -X_{d2} & 0 & 0 \\ S_1 & 0 & 0 & 0 & 0 & -\tau^{-1}S_1 & 0 \\ S_2 & 0 & 0 & 0 & 0 & 0 & -\tau^{-1}S_2 \end{bmatrix} < 0 \tag{5.19}$$

$$\begin{bmatrix} \Theta_{11} & \Theta_{12}^T & \Theta_{13}^T & \Theta_{14}^T & \Theta_{15}^T \\ \Theta_{12} & -\Theta_{22} & 0 & 0 & 0 \\ \Theta_{13} & 0 & -\Theta_{33} & 0 & 0 \\ \Theta_{14} & 0 & 0 & -X_2 & 0 \\ \Theta_{15} & 0 & 0 & 0 & -X_{d3} \end{bmatrix} < 0 \tag{5.20}$$

where

$$
\begin{aligned}
\Pi_{11} =& \; U_2^T A U_2 P - U_2^T A U_1 V + U_2^T A_d U_2 P - U_2^T A_d U_1 V \\
&+ P U_2^T A^T U_2 - V^T U_1^T A^T U_2 + P U_2^T A_d^T U_2 - V^T U_1^T A_d^T U_2 \\
&+ U_2^T G X_1 G^T U_2 + U_2^T G_d X_{d1} G_d^T U_2 \\
\Pi_{12} =& \; U_2^T A_d U_2 P - U_2^T A_d U_1 V \\
\Pi_{13} =& \; H U_2 P - H U_1 V \\
\Pi_{14} =& \; H_d U_2 P - H_d U_1 V \\
\Pi_{15} =& \; H_d U_2 P - H_d U_1 V \\
\Pi_{22} =& \; \tau^{-1} Q - U_2^T G_d X_{d2} G_d^T U_2 \\
\Theta_{11} =& \; -2P + Q \\
\Theta_{12} =& \; U_2^T A U_2 P - U_2^T A U_1 V \\
\Theta_{13} =& \; U_2^T A_d U_2 P - U_d^T A_d U_1 V \\
\Theta_{14} =& \; H U_2 P - H U_1 V \\
\Theta_{15} =& \; H_d U_2 P - H_d U_1 V \\
\Theta_{22} =& \; S_1 - U_2^T G X_2 G^T U_2 \\
\Theta_{33} =& \; S_2 - U_2^T G_d X_{d3} G_d^T U_2 .
\end{aligned} \tag{5.21}
$$

Moreover, the gain $C = VP^{-1}$ and the sliding surface is $S(t) = VP^{-1} z_1(t) + z_2(t)$.

Proof. Since the characteristic equation of system (5.12) is

$$\det\left(sI - \tilde{A}_1 - \tilde{A}_{d1} e^{-\tau s}\right) = 0 \tag{5.22}$$

whose solutions are the same as those of

$$\det\left(sI - \tilde{A}_1^T - \tilde{A}_{d1}^T e^{-\tau s}\right) = 0. \tag{5.23}$$

Therefore, we will consider the following system instead of system (5.12)

$$
\begin{aligned}
\dot{y}(t) &= \tilde{A}_1^T y(t) + \tilde{A}_{d1}^T y(t-\tau) \\
y(t) &= \Psi_y(t), \;\; t \in [-\tau, \;\; 0]
\end{aligned} \tag{5.24}
$$

where $\Psi_y(t)$ is an appropriate initial function for system (5.24). For $t \geq \tau$

$$
\begin{aligned}
y(t-\tau) &= y(t) - \textstyle\int_{-\tau}^{0} \dot{y}(t+\theta) d\theta \\
&= y(t) - \textstyle\int_{-\tau}^{0} [\tilde{A}_1^T y(t+\theta) + \tilde{A}_{d1}^T y(t+\theta-\tau)] d\theta .
\end{aligned} \tag{5.25}
$$

Substituting $y(t-\tau)$ in (5.24) gives

$$\dot{y}(t) = [\tilde{A}_1 + \tilde{A}_{d1}]^T y(t) - \tilde{A}_{d1}^T \int_{-\tau}^{0} [\tilde{A}_1^T y(t+\theta) + \tilde{A}_{d1}^T y(t+\theta-\tau)] d\theta . \tag{5.26}$$

In view of the above, consider the following time-delay system:

$$
\begin{aligned}
&\dot{\eta}(t) = [\tilde{A}_1 + \tilde{A}_{d1}]^T \eta(t) - \tilde{A}_{d1}^T \textstyle\int_{-\tau}^{0} [\tilde{A}_1^T \eta(t+\theta) + \tilde{A}_{d1}^T \eta(t+\theta-\tau)] d\theta \\
&\eta(t) = \psi(t), \forall t \in [-2\tau, 0]
\end{aligned} \tag{5.27}
$$

where $\psi(t)$ is the initial condition and τ is the time delay of system (5.24). Observe that equation (5.27) requires initial data on $[-2\tau, 0]$.

Note that system (5.24) is a special case of system (5.27) and thus any solution of system (5.24) is also a solution of system (5.27) (See [68], p.156). Hence, the global uniform asymptotic stability of system (5.27) will ensure the global uniform asymptotic stability of system (5.24). In the sequel, we will study the stability of system (5.27) in order to ascertain the stability of system (5.24).

We first show that system (5.24) is stable under the conditions (5.19)-(5.20). Using the following Lyapunov functional candidate:

$$V(\eta, t) = V_1(\eta(t), t) + V_2(\eta(t), t) + V_3(\eta(t), t) \tag{5.28}$$

where

$$V_1(\eta(t), t) = \eta^T(t) P \eta(t) \tag{5.29}$$

$$V_2(\eta(t), t) = \int_{-\tau}^{0} \left(\int_{t+\theta}^{t} \eta^T(\lambda) S_1 \eta(\lambda) d\lambda \right) d\theta \tag{5.30}$$

$$V_3(\eta(t), t) = \int_{-2\tau}^{-\tau} \left(\int_{t+\theta}^{t} \eta^T(\lambda) S_2 \eta(\lambda) d\lambda \right) d\theta \tag{5.31}$$

with P, S_1 and S_2 are symmetric positive-definite matrices to be chosen. It is easy to see that there exist positive scalars β_1 and β_2 such that

$$\beta_1 ||\eta(t)||^2 \leq V(\eta(t), t) \leq \beta_2 \sup_{\theta \in [-2\tau, 0]} ||\eta(t+\theta)||^2. \tag{5.32}$$

The time derivative along the state trajectory of system (5.27) is

$$\begin{aligned} \dot{V}(\eta(t), t) = \eta^T(t)[(\tilde{A}_1 + \tilde{A}_{d1})P + P(\tilde{A}_1 + \tilde{A}_{d1})^T]\eta(t) + \mu_1(\eta(t), t) \\ + \mu_2(\eta(t), t) + \dot{V}_2(\eta(t), t) + \dot{V}_3(\eta(t, t) \end{aligned} \tag{5.33}$$

where

$$\mu_1(\eta(t), t) = -2 \int_{-\tau}^{0} \eta^T(t) P \tilde{A}_{d1}^T \tilde{A}_1^T \eta(t+\theta) d\theta \tag{5.34}$$

$$\mu_2(\eta(t), t) = -2 \int_{-\tau}^{0} \eta^T(t) P \tilde{A}_{d1}^T \tilde{A}_{d1}^T \eta(t+\theta-\tau) d\theta \tag{5.35}$$

$$\dot{V}_2(\eta(t), t) = \tau \eta^T(t) S_1 \eta(t) - \int_{-\tau}^{0} \eta^T(t+\theta) S_1 \eta(t+\theta) d\theta \tag{5.36}$$

$$\dot{V}_3(\eta(t), t) = \tau \eta^T(t) S_2 \eta(t) - \int_{-\tau}^{0} \eta^T(t+\theta-\tau) S_2 \eta(t+\theta-\tau) d\theta. \tag{5.37}$$

Since for any vectors u, v and any matrix $S > 0$ of appropriate dimension, the following inequality holds:

$$-2u^T v \leq u^T S u + v^T S^{-1} v. \tag{5.38}$$

Thus, we have

$$\begin{aligned}\mu_1(\eta(t),t) &\leq \tau\eta^T(t)P\tilde{A}_{d1}^T\tilde{A}_1^T S_1^{-1}\tilde{A}_1\tilde{A}_{d1}\eta(t)\\ &\quad+\int_{-\tau}^{0}\eta^T(t+\theta)S_1\eta(t+\theta)d\theta \\ \mu_2(\eta(t),t) &\leq \tau\eta^T(t)P\tilde{A}_{d1}^T\tilde{A}_1^T S_2^{-1}\tilde{A}_1\tilde{A}_{d1}P\eta(t)\\ &\quad+\int_{-\tau}^{0}\eta^T(t+\theta-\tau)S_2\eta(t+\theta-\tau).\end{aligned} \tag{5.39}$$

Hence, it follows that

$$\begin{aligned}&\dot{V}(\eta(t),t)\\ &\leq \eta^T(t)[(\tilde{A}_1+\tilde{A}_{d1})P+P(\tilde{A}_1+\tilde{A}_{d1})^T]+\tau(S_1+S_2\\ &\quad+P\tilde{A}_{d1}^T\tilde{A}_1^T S_1^{-1}\tilde{A}_1\tilde{A}_{d1}P+P\tilde{A}_{d1}^T\tilde{A}_1^T S_2^{-1}A_1\tilde{A}_{d1}P)]\eta(t)\\ &\leq \eta^T(t)[(\tilde{A}_1+\tilde{A}_{d1})P+P(\tilde{A}_1+\tilde{A}_{d1})^T+\tau(S_1+S_2\\ &\quad+P\tilde{A}_{d1}^T[\tilde{A}_1^T\ \ \tilde{A}_{d1}^T]\mathrm{diag}(S_1^{-1},S_2^{-1})[\tilde{A}_1^T\ \ \tilde{A}_{d1}^T]^T\tilde{A}_{d1}P)]\eta(t)\\ &= \eta^T(t)[(\tilde{A}_1+\tilde{A}_{d1})P+P(\tilde{A}_1+\tilde{A}_{d1})^T\\ &\quad+\tau(S_1+S_2+P\tilde{A}_{d1}^T Q^{-1}\tilde{A}_{d1}P)\\ &\quad-\tau(P\tilde{A}_{d1}^T(Q^{-1}-[\tilde{A}_1^T\ \ \tilde{A}_{d1}^T]\mathrm{diag}(S_1^{-1},S_2^{-1}))[\tilde{A}_1^T\ \ \tilde{A}_{d1}^T]^T)\tilde{A}_{d1}P)]\eta(t)\\ &\leq \eta^T(t)[(\tilde{A}_1+\tilde{A}_{d1})P+P(\tilde{A}_1+\tilde{A}_{d1})^T\\ &\quad+\tau(S_1+S_2)+\tau P\tilde{A}_{d1}^T Q^{-1}\tilde{A}_{d1}P]\eta(t)\end{aligned} \tag{5.40}$$

where Q is a positive-definite matrix which satisfies

$$Q^{-1} > [\tilde{A}_1^T\ \ \tilde{A}_{d1}^T]\mathrm{diag}(S_1^{-1},S_2^{-1})[\tilde{A}_1^T\ \ \tilde{A}_{d1}^T]^T. \tag{5.41}$$

The matrix in (5.40) is negative definite if (5.41) and the following inequality:

$$(\tilde{A}_1+\tilde{A}_{d1})P+P(\tilde{A}_1+\tilde{A}_{d1})^T+\tau(S_1+S_2)+\tau P\tilde{A}_{d1}^T Q^{-1}\tilde{A}_{d1}P<0 \tag{5.42}$$

is satisfied.

Using the Schur complement argument ([103]), inequalities (5.41) and (5.42) are equivalent to the following inequalities, respectively

$$\begin{bmatrix} -Q^{-1} & \tilde{A}_1^T & \tilde{A}_{d1}^T \\ \tilde{A}_1 & -S_1 & 0 \\ \tilde{A}_{d1} & 0 & -S_2 \end{bmatrix} < 0 \tag{5.43}$$

and

$$\begin{bmatrix} (\tilde{A}_1+\tilde{A}_{d1})P+P(\tilde{A}_1+\tilde{A}_{d1})^T+\tau(S_1+S_2) & P\tilde{A}_{d1}^T \\ \tilde{A}_{d1}P & -\tau^{-1}Q \end{bmatrix} < 0. \tag{5.44}$$

Pre- and postmultiplying (5.43) by $\mathrm{diag}(P, I_{n-m}, I_{n-m})$ yields

$$\begin{bmatrix} -PQ^{-1}P & P\tilde{A}_1^T & P\tilde{A}_{d1}^T \\ \tilde{A}_1 P & -S_1 & 0 \\ \tilde{A}_{d1}P & 0 & -S_2 \end{bmatrix} < 0. \tag{5.45}$$

From the equality

$$PQ^{-1}P - 2P + Q = (P - Q)Q^{-1}(P - Q) \geq 0 \tag{5.46}$$

we conclude that

$$-2P + Q \geq -PQ^{-1}P. \tag{5.47}$$

Thus, (5.45) will hold if the following inequality is satisfied for some P, Q, S_1 and S_2

$$\begin{bmatrix} -2P + Q & P\tilde{A}_1^T & P\tilde{A}_{d1}^T \\ \tilde{A}_1 P & -S_1 & 0 \\ \tilde{A}_{d1}P & 0 & -S_2 \end{bmatrix} < 0. \tag{5.48}$$

On the other hand, it follows from (5.13) and (5.14) that

$$(\Delta\overline{A}_1 + \Delta\overline{A}_{d1})P = \begin{bmatrix} U_2^T G & U_2^T G_d \end{bmatrix} \begin{bmatrix} D & 0 \\ 0 & D_d \end{bmatrix} \begin{bmatrix} H(U_2 - U_1 C)P \\ H_d(U_2 - U_1 C)P \end{bmatrix}. \tag{5.49}$$

Hence, from Lemma 5.2, we have

(a) for any $X_1 \in S_D$ and for any $X_{d1} \in S_{D_d}$,

$$\begin{aligned} &(\Delta\overline{A}_1 + \Delta\overline{A}_{d1})P + [(\Delta\overline{A}_1 + \Delta\overline{A}_{d1})P]^T \\ &\leq [U_2^T G X_1 G^T U_2 + U_2^T G_d X_{d1} G_d^T U_2] + [(H(U_2 - U_1 C)P)^T X_1^{-1} \\ &\times H(U_2 - U_1 C)P + (H_d(U_2 - U_1 C)P)^T X_{d1}^{-1} H_d(U_2 - U_1 C)P] \end{aligned} \tag{5.50}$$

(b) for any $X_2 \in S_D$,

$$\begin{aligned} &\begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix} \Delta\overline{A}_1 P \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix}^T + \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix} (\Delta\overline{A}_1 P)^T \begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix}^T \\ &\quad \leq \begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix} U_2^T G X_2 G^T U_2 \begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix}^T \\ &\quad + \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix} [(H(U_2 - U_1 C)P)^T X_2^{-1} H(U_2 - U_1 C)P \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix}^T \end{aligned}$$

(c) for any $X_{d2} \in S_{D_d}$,

$$\begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix} \Delta\overline{A}_{d1} P \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix}^T + \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix} P\Delta\overline{A}_{d1}^T \begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix}^T$$

$$\leq \begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix} U_2^T G_d X_{d2} G_d^T U_2 \begin{bmatrix} 0_{n\times n} \\ I_{n\times n} \end{bmatrix}^T + \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix} (H_d(U_2 - U_1 C)P)^T X_{d2}^{-1} H_d(U_2 - U_1 C)P \begin{bmatrix} I_{n\times n} \\ 0_{n\times n} \end{bmatrix}^T$$

(d) for any $X_{d3} \in S_{D_d}$,

$$\begin{aligned} &\begin{bmatrix} 0_{2n\times n} \\ I_{n\times n} \end{bmatrix} \Delta\overline{A}_{d1} P \begin{bmatrix} I_{n\times n} \\ 0_{2n\times n} \end{bmatrix}^T + \begin{bmatrix} I_{n\times n} \\ 0_{2n\times n} \end{bmatrix} (\Delta\overline{A}_{d1} P)^T \begin{bmatrix} 0_{2n\times n} \\ I_{n\times n} \end{bmatrix}^T \\ &\quad \leq \begin{bmatrix} 0_{2n\times n} \\ I_{n\times n} \end{bmatrix} U_2^T G_d X_{d3} G_d^T U_2 \begin{bmatrix} 0_{2n\times n} \\ I_{n\times n} \end{bmatrix}^T + \begin{bmatrix} I_{n\times n} \\ 0_{2n\times n} \end{bmatrix} \\ &\quad \times (H_d(U_2 - U_1 C)P)^T X_{d3}^{-1} H_d(U_2 - U_1 C)P \begin{bmatrix} I_{n\times n} \\ 0_{2n\times n} \end{bmatrix}^T . \end{aligned} \tag{5.51}$$

Now, applying the inequalities (5.50)-(5.51) to the right-hand sides of (5.44) and (5.48) results in

$$\begin{bmatrix} (\tilde{A}_1 + \tilde{A}_{d1})P + P(\tilde{A}_1 + \tilde{A}_{d1})^T + \tau(S_1 + S_2) & P\tilde{A}_{d1}^T \\ \tilde{A}_{d1} P & -\tau^{-1} Q \end{bmatrix} \leq \begin{bmatrix} \Phi_{11} & P\overline{A}_{d1}^T \\ \overline{A}_{d1} P & \Phi_{22} \end{bmatrix} \tag{5.52}$$

$$\begin{bmatrix} -2P + Q & P\tilde{A}_1^T & P\tilde{A}_{d1}^T \\ \tilde{A}_1 P & -S_1 & 0 \\ \tilde{A}_{d1} P & 0 & -S_2 \end{bmatrix} \leq \begin{bmatrix} \Psi_{11} & P\overline{A}_1^T & P\overline{A}_{d1}^T \\ \overline{A}_1 P & \Psi_{22} & 0 \\ \overline{A}_{d1} P & 0 & \Psi_{33} \end{bmatrix} \tag{5.53}$$

where

$$\begin{aligned} \Phi_{11} =\ & (\overline{A}_1 + \overline{A}_{d1})P + P(\overline{A}_1 + \overline{A}_{d1})^T + \tau(S_1 + S_2) \\ & + U_2^T G X_1 G^T U_2 + U_2^T G_d X_{d1} G_d^T U_2 \\ & + (H(U_2 - U_1 C)P)^T X_1^{-1} H(U_2 - U_1 C)P \\ & + (H_d(U_2 - U_1 C)P)^T X_{d1}^{-1} H_d(U_2 - U_1 C)P \\ & + (H_d(U_2 - U_1 C)P)^T X_{d2}^{-1} H_d(U_2 - U_1 C)P \\ \Phi_{22} =\ & -\tau^{-1} Q + U_2^T G_d X_{d2} G_d^T U_2 \\ \Psi_{11} =\ & (H(U_2 - U_1 C)P)^T X_2^{-1} H(U_2 - U_1 C)P \\ & + (H_d(U_2 - U_1 C)P)^T X_{d3}^{-1} H_d(U_2 - U_1 C)P - 2P + Q \\ \Psi_{22} =\ & -S_1 + U_2^T G X_2 G^T U_2 \\ \Psi_{33} =\ & -S_2 + U_2^T G_d X_{d3} G_d^T U_2 . \end{aligned} \tag{5.54}$$

Thus, the derivative of the Lyapunov function candidate $V(\eta(t), t)$ will be negative if the following conditions hold

$$\begin{bmatrix} \Phi_{11} & P\overline{A}_{d1}^T \\ \overline{A}_{d1} P & \Phi_{22} \end{bmatrix} < 0 \tag{5.55}$$

$$\begin{bmatrix} \Psi_{11} & P\overline{A}_1^T & P\overline{A}_{d1}^T \\ \overline{A}_1 P & \Psi_{22} & 0 \\ \overline{A}_{d1} P & 0 & \Psi_{33} \end{bmatrix} < 0. \tag{5.56}$$

By using Schur complement and let $V = CP$, we obtain that (5.55) and (5.56) are equivalent to (5.19) and (5.20), respectively.

Thus, we can conclude that if (5.19) and (5.20) are satisfied, then (5.55) and (5.56) hold for any constant time delay τ satisfying $0 \le \tau \le \overline{\tau}$. Hence, $\dot{V}(\eta(t), t) < 0$ for any non-zero $\eta(t) \in R^{(n-m)}$ and any constant time delay τ satisfying $0 \le \tau \le \overline{\tau}$. Therefore, system (5.27) is robustly stable, which implies that system (5.12) is robustly stable for any constant time delay τ satisfying $0 \le \tau \le \overline{\tau}$ with the sliding function

$$s(t) = [C\ I]z = VP^{-1}z_1(t) + z_2(t). \tag{5.57}$$

The proof is completed.

Remark 5.4. It is worth mentioning that in the literature, some similar techniques (see [187, 188]) have been used to transform nonlinear matrix inequality into LMIs, as we did in (5.46) and (5.47). Note that the approach employed here adds no restrictions to the results obtained in this chapter, since only a new unknown variable (another LMI) needs to be solved. The imposed condition $Q \le 2P$ is to ensure that the LMI toolbox can be used to solve inequality (5.48). This is a relaxed condition comparing with the assumption $Q = P$ made in the work of [61]. The method proposed in [185] can also be used to derive delay-dependent condition for the existence of linear sliding surfaces

Based on the assumption that the time delay $\overline{\tau}$ is constant, Theorem 5.3 establishes an LMI-based stability condition for system (5.1). Now, we come to consider the problem of determining the upper bound for the time-delay τ. Obviously, when τ is constant but unknown, (5.19) is nonlinear in $\overline{\tau}$. Let $\nu = \frac{1}{\overline{\tau}}$, and note that when using the LMI toolbox to solve the generalized eigenvalue problem (GEVP)

$$\min_x \nu$$

subject to

$$A(x) < \nu B(x)$$

the matrix $B(x)$ must be positive-definite. Thus, to cast the problem of maximizing the time-delay $\overline{\tau}$, i.e., minimizing ν, into the framework of GEVP, let us introduce some auxiliary matrices $W_1 > 0$, $W_2 > 0$, $W_3 > 0$ and rewrite (5.19) as follows:

$$\begin{bmatrix} \Pi_{11} & \Pi_{12}^T & \Pi_{13}^T & \Pi_{14}^T & \Pi_{15}^T & S_1 & S_2 \\ \Pi_{12} & -W_1 + U_2^T G_d X_{d2} G_d^T U_2 & 0 & 0 & 0 & 0 & 0 \\ \Pi_{13} & 0 & -X_1 & 0 & 0 & 0 & 0 \\ \Pi_{14} & 0 & 0 & -X_{d1} & 0 & 0 & 0 \\ \Pi_{15} & 0 & 0 & 0 & -X_{d2} & 0 & 0 \\ S_1 & 0 & 0 & 0 & 0 & -W_2 & 0 \\ S_2 & 0 & 0 & 0 & 0 & 0 & -W_3 \end{bmatrix} < 0 \quad (5.58)$$

$$\begin{bmatrix} W_1 & 0 & 0 \\ 0 & W_2 & 0 \\ 0 & 0 & W_3 \end{bmatrix} < \nu \begin{bmatrix} Q & 0 & 0 \\ 0 & S_1 & 0 \\ 0 & 0 & S_2 \end{bmatrix}. \quad (5.59)$$

We then have the following result:

Algorithm 5.3.1 *The optimization problem that determines the upper bound of the delay can be formulated as*:

$$\min_{P,Q,V,X_1,X_2X_{d1},X_{d2},X_{d3},,W_1,W_2,W_3} \quad \nu$$

$$\text{s.t.} \quad (5.20), (5.58), (5.59).$$

Remark 5.5. The above optimization problem consists of minimizing a generalized eigenvalue problem which is a quasi-convex optimization problem. It is important to notice that this algorithm can be numerically solved with very efficient methods ([15]).

5.3.2 Design of A Reaching Motion Controller

Now we are in the position to present our last result in this chapter, namely, designing of reaching motion controller.

Theorem 5.6. *When the time-delay, τ, is a known constant and $C = VP^{-1}$ is given in Theorem 5.3, the trajectory of the closed-loop system (5.7) can be driven onto the sliding surface in finite time by the control law*

$$\begin{aligned} u = -B_2^{-1}[&KS + \epsilon \text{sgn}(S) + \overline{CA}z(t) + \overline{CA}_d z(t-\tau) \\ &+ \text{diag}\left(\text{sgn}(s_1)\ \text{sgn}(s_2)\ \cdots\ \text{sgn}(s_m)\right)(N_1 + N_2 + N_3)] \end{aligned} \quad (5.60)$$

where

$$
\begin{aligned}
&S=\begin{bmatrix} s_1\\ \vdots \\ s_m\end{bmatrix},\ \mathrm{sgn}(S)=\begin{bmatrix} \mathrm{sgn}(s_1)\\ \vdots \\ \mathrm{sgn}(s_m)\end{bmatrix},\ w=\begin{bmatrix} w_1\\ \vdots \\ w_m\end{bmatrix},\ B_2F=\begin{bmatrix} b_1\\ \vdots \\ b_m\end{bmatrix}\\
&N_1=\begin{bmatrix} N_{11}\\ \vdots \\ N_{1m}\end{bmatrix},\ N_2=\begin{bmatrix} N_{21}\\ \vdots \\ N_{2m}\end{bmatrix},\ N_3=\begin{bmatrix} N_{31}\\ \vdots \\ N_{3m}\end{bmatrix},\ \overline{C}=\begin{bmatrix} \overline{c}_1\\ \vdots \\ \overline{c}_m\end{bmatrix}\\
&N_{1i}=\textstyle\sum_{j=1}^{p}|\overline{c}_iTA_jT^{-1}z(t)|,\ N_{2i}=\sum_{j=1}^{q}|\overline{c}_iTA_{dj}T^{-1}z(t-\tau)|\\
&N_{3i}=\textstyle\sum_{j=1}^{m}|b_{ij}|\overline{w}_j,\quad i=1,\cdots,m\\
&\overline{C}=\begin{bmatrix} C\ I\end{bmatrix},\ K=\mathrm{diag}(k_i),\epsilon=\mathrm{diag}(\epsilon_i)
\end{aligned}
\tag{5.61}
$$

where k_i and ϵ_i are positive constants.

Proof. It will be shown that control law (5.60) drives the system trajectory onto the linear sliding surface and maintains the trajectory on the sliding surface for all subsequent time, i.e. the reachability condition is satisfied. Then from the sliding surface

$$
S=\begin{bmatrix} C\ I\end{bmatrix} z
$$

one can obtain

$$
\begin{aligned}
\dot{S}&=\ \begin{bmatrix} C\ I\end{bmatrix}\dot{z}\\
&=\ \overline{C}(\overline{A}+\Delta\overline{A})z(t)+\overline{C}(\overline{A}_d+\Delta\overline{A}_d)z(t-\tau)+B_2(u(t)+Fw(t)).
\end{aligned}
\tag{5.62}
$$

On the other hand, from the inequalities

$$
\overline{c}_i\Delta\overline{A}z(t)=\overline{c}_i\sum_{j=1}^{p}\alpha_jTA_jT^{-1}z(t)\le\sum_{j=1}^{p}|\overline{c}_iTA_jT^{-1}z(t)|=N_{1i}
\tag{5.63}
$$

$$
\overline{c}_i\Delta\overline{A}_dz(t-\tau)=\overline{c}_i\sum_{j=1}^{q}\beta_jTA_{dj}T^{-1}z(t-\tau)\le\sum_{j=1}^{q}|\overline{c}_iTA_{dj}T^{-1}z(t-\tau)|=N_{2i}
\tag{5.64}
$$

and

$$
b_iw=\sum_{j=1}^{m}b_{ij}w_j\le\sum_{j=1}^{m}|b_{ij}|\overline{w}_j|=N_{3i}
\tag{5.65}
$$

it follows that each element of (5.62) with control law (5.60)

$$
\begin{aligned}
\dot{s_i}=&-ks_i-\epsilon_i\mathrm{sgn}(s_i)-(N_{i1}\mathrm{sgn}(s_i)-\overline{c}_i\Delta\overline{A}z(t))\\
&-(N_{2i}\mathrm{sgn}(s_i)-\overline{c}_i\Delta\overline{A}_dz(t-\tau))-(N_{3i}\mathrm{sgn}(s_i)-b_iw)
\end{aligned}
\tag{5.66}
$$

satisfies the reaching condition:

$$\begin{cases} \dot{s}_i < 0 \text{ if } s_i > 0; \\ \dot{s}_i > 0 \text{ if } s_i < 0. \end{cases} \tag{5.67}$$

Thus the trajectory of system (5.7) is driven onto the sliding surface in finite time by the controller (5.60), which completes the proof.

Note that the time-delay τ is constant but unknown, the control law (5.60) is not applicable for the terms of $\overline{CA}_d z(t-\tau)$ and N_2 cannot be obtained in practice. Therefore this needs further consideration which results in the following theorem.

Theorem 5.7. *Assume that time-delay τ is an unknown constant, but bounded by $\overline{\tau}$, i.e., $0 \le \tau \le \overline{\tau}$. Given $C = VP^{-1}$ obtained in Theorem 5.3, then the trajectory of system (5.7) can be driven onto the sliding surface in finite time by the control law*

$$\begin{aligned} u = & -B_2^{-1}[KS + \epsilon \text{sgn}(S) + \overline{CA}z(t) \\ & + \text{diag}\left(\text{sgn}(s_1) \; \text{sgn}(s_2) \; \cdots \; \text{sgn}(s_m) \right) (N_1 + N_2 + N_3)] \end{aligned} \tag{5.68}$$

where $S, w, B_2F, N_1, N_2, N_3, \overline{C}, \overline{C}, N_{1i}, N_{3i}, K, \epsilon$ are defined as in (5.61), $N_{2i} = q|\overline{c}_i|(|TA_dT^{-1}| + \sum_{j=1}^{q} |TA_{dj}T^{-1}|)|z(t)|$, and $q > 1$ is a constant to be determined.

Proof. Following the Razumikhin Theorem (see [68], p.152), it is assumed that for any positive scalar $q > 1$, the following relation holds:

$$|z(t+\theta)| \le q|z(t)|, \; -\overline{\tau} \le \theta \le 0 \tag{5.69}$$

which leads to the following inequality:

$$\begin{aligned} \overline{c}_i \overline{A}_d z(t-\tau) \; &= \overline{c}_i T(A_d + \sum_{j=1}^{q} \beta_j A_{dj})T^{-1} z(t-\tau) \\ &\le q|\overline{c}_i|(|TA_dT^{-1}| + \sum_{j=1}^{q} |TA_{dj}T^{-1}|)|z(t)| \\ &= N_{2i}. \end{aligned} \tag{5.70}$$

The rest of the proof can be carried out along the same lines as those in Theorem 5.6.

5.4 Numerical Example

In this section, we demonstrate the theory developed in this chapter by a numerical example.

Example 5.8. Consider the interval time-delay system with

$$A = \begin{bmatrix} \begin{bmatrix} -5 & -3 \\ -0.5 & 0.5 \end{bmatrix} & \begin{bmatrix} 1 & 2 \\ -1 & 1 \end{bmatrix} \end{bmatrix}, \; A_d = \begin{bmatrix} \begin{bmatrix} -1.7 & -1.3 \\ 0.4 & 0.8 \end{bmatrix} & \begin{bmatrix} 0.4 & 1.2 \\ -3.4 & -2.4 \end{bmatrix} \end{bmatrix}$$

$$B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \; F = 1, \; \varphi(t) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \text{ for } t \in \begin{bmatrix} -\tau, 0 \end{bmatrix}, \; |w(t)| \leq 0.05|sint|.$$

Write A and A_d as

$$\begin{aligned} A &= A_0 + \alpha_1 A_1 + \alpha_2 A_2 + \alpha_3 A_3 + \alpha_4 A_4 \\ &= \begin{bmatrix} -4 & 1.5 \\ 0 & 0 \end{bmatrix} + \alpha_1 \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \alpha_2 \begin{bmatrix} 0 & 0.5 \\ 0 & 0 \end{bmatrix} + \alpha_3 \begin{bmatrix} 0 & 0 \\ 0.5 & 0 \end{bmatrix} + \alpha_4 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \\ A_d &= A_{d0} + \beta_1 A_{d1} + \beta_2 A_{d2} + \beta_3 A_{d3} + \beta_4 A_{d4} \\ &= \begin{bmatrix} -1.5 & 0.8 \\ 0.6 & -2.9 \end{bmatrix} + \beta_1 \begin{bmatrix} 0.2 & 0 \\ 0 & 0 \end{bmatrix} + \beta_2 \begin{bmatrix} 0 & 0.4 \\ 0 & 0 \end{bmatrix} \\ &\quad + \beta_3 \begin{bmatrix} x0 & 0 \\ 0.2 & 0 \end{bmatrix} + \beta_4 \begin{bmatrix} 0 & 0 \\ 0 & 0.5 \end{bmatrix} \end{aligned} \tag{5.71}$$

where $|\alpha_i| \leq 1$ and $|\beta_i| \leq 1$, $i = 1, 2, 3, 4$. We factorize A_i and A_{di} as follows:

$$\begin{aligned} A_1 &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix}, \; A_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0.5 \end{bmatrix}, \; A_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0.5 & 0 \end{bmatrix} \\ A_4 &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix}, A_{d1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0.2 & 0 \end{bmatrix}, \; A_{d2} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 0.4 \end{bmatrix} \\ A_{d3} &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0.2 & 0 \end{bmatrix}, A_{d4} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 0 & 0.5 \end{bmatrix}. \end{aligned} \tag{5.72}$$

According to (5.3) and (5.4), we can obtain

$$\begin{aligned} G &= \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \; H = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \\ 0.5 & 0 \\ 0 & 1 \end{bmatrix} \\ G_d &= \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \; H_d = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.4 \\ 0.2 & 0 \\ 0 & 0.5 \end{bmatrix}. \end{aligned} \tag{5.73}$$

Taking $T = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, we have the following sliding motions:

$$\begin{aligned} \dot{z_1}(t) &= (-0.5\alpha_3 c + \alpha_4) z_1(t) + (-2.9 - 0.6c - 0.2\beta_3 c + 0.5\beta_4) z_1(t - \tau) \\ z_1(t) &= \overline{\varphi}_1(t), \; t \in [-\tau, \;\; 0]. \end{aligned} \tag{5.74}$$

Note that α_3 and α_4 are uncertain parameters, and the gain c can not be obtained using the delay-independent criteria proposed in [161] and [100] as (A_0, B) is not stabilizable. The delay-independent condition obtained in [189] also can not be used to design c such that the uncertain sliding motion is quadratically stable. Based on Definition 1 in [189], sliding motion is quadratically stable if there exist symmetric positive-definite matrices $P, Q \in R^{(n-m)\times(n-m)}$ such that the following inequality is satisfied:

$$\begin{bmatrix} P\tilde{A}_1 + \tilde{A}_1^T P + Q & P\tilde{A}_{d1} \\ \tilde{A}_{d1}^T P & -Q \end{bmatrix} < 0 \tag{5.75}$$

where $\tilde{A}_1 = (-0.5\alpha_3 c + \alpha_4)$, $\tilde{A}_{d1} = (-2.9 - 0.6c - 0.2\beta_3 c + 0.5\beta_4)$. This is the delay-independent condition used in [189]. Obviously, the parameter c can not be found such that (5.75) is satisfied since α_3, α_4, β_3 and β_4 are uncertain parameters, for example, if α_3 and α_4 are zero, that means $\tilde{A}_1 = 0$, then the condition

$$\begin{bmatrix} Q & P\tilde{A}_{d1} \\ \tilde{A}_{d1}^T P & -Q \end{bmatrix} < 0 \tag{5.76}$$

can not be satisfied, no matter what c is.

Applying Algorithm 5.3.1 derived in Section 5.3.1 gives that system (5.74) is robustly stable for $0 \leq \tau \leq 0.0920$. According to Theorem 5.3 and using the LMI-toolbox, we obtain that $c = 0.0037$ for $\tau = 0.05$. Then the linear sliding surface is taken as

$$S(t) = \begin{bmatrix} 1 \ 0.0037 \end{bmatrix} x. \tag{5.77}$$

From Theorem 5.3, we can take

$$\begin{aligned} u(t) = -[&Ks + \epsilon \text{sgn}(s) + \begin{bmatrix} -5 \ 1.5 \end{bmatrix} x(t) + \begin{bmatrix} -1.4978 \ 0.7893 \end{bmatrix} x(t-0.05) \\ &+\text{sgn}(s)(N_1 + N_2 + N_3)] \end{aligned} \tag{5.78}$$

where $N_1 = 2.5437|x_1(t)| + 3.5874|x_2(t)|$, $N_2 = 0.8175|x_1(t-0.05)| + 1.9437|x_2(t-0.05)|$, $N_3 = 0.05|sint|$.

The parameters K and ϵ can be tuned to reduce the chattering on the sliding surface. Choosing $K = 1, \epsilon = 0.1$, the simulation results shown in Fig. 5.1 are obtained. It can be seen that the system is asymptotically stable and the sliding surface dynamics reaches zero in finite time and thereafter maintains the condition regardless of the delay and the uncertainties.

5.5 Conclusion

In this chapter, the problem of designing robust sliding surfaces for a class of uncertain time-delay systems has been considered, where the uncertainty in the states is not required to satisfy the usual matching condition. In terms of LMIs, the delay-dependent sufficient condition for the existence of a linear sliding surface guaranteeing quadratic stability of the reduced-order equivalent system restricted to the sliding surface was derived. Furthermore, the

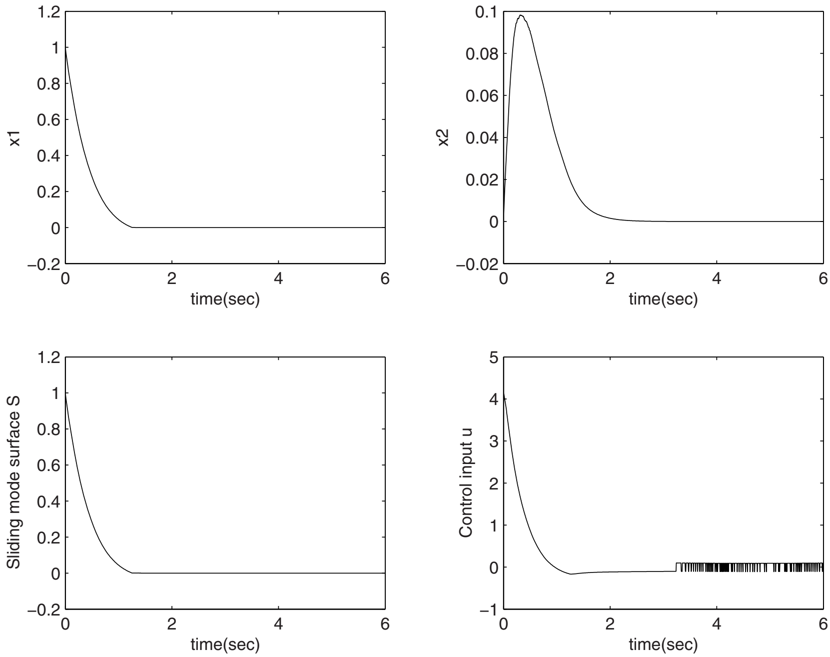

Fig. 5.1 States (x_1, x_2), sliding surface S and control input u

reaching motion control problem was also studied. A new reaching motion controller has been designed for uncertain time-delay systems by means of both the reaching law and the inequality approaches. Both the sliding motion and the reaching motion are robust against the mismatched uncertainties and matched external disturbance. A numerical example was presented to demonstrate the effectiveness and the potential of the developed techniques.

Chapter 6
Robust Adaptive SMC for Uncertain Time-Delay Systems

6.1 Introduction

In this chapter, we consider how to design a sliding surface and reaching motion controller for a class of time-delay systems with mismatched uncertainties and exogenous disturbance, where the parameters of mismatched uncertainties are known to lie in a priori specified intervals and the matched uncertainties and exogenous disturbance are assumed to be bounded with unknown bound. The aim of this chapter is to combine adaptive control and variable structure control to realize their individual advantages. The use of variable structure control improves the transient response of the overall system significantly. As a result, the adaptive law proposed is able to reduce the chattering due to the implementation of variable structure controller, and possesses the desired characteristics of robustness and good performance. Sufficient conditions for the existence of linear sliding surfaces are derived. The solution to the condition can be used to characterize linear sliding surfaces, and by using smooth projection method for adaptive control law the reaching motion controller is designed.

The chapter is organized as follows. Section 6.2 gives the problem formulation and some preliminaries. The main results on designing robust adaptive sliding surface and a reaching motion control law is presented in Section 6.3 such that the closed-loop system is convergent into a residual set of the origin with the proposed reaching control law. Numerical simulations are presented in Section 6.4 and some conclusion remarks are given in Section 6.5.

6.2 Problem Formulation

Consider the uncertain time-delay system of the form

$$\begin{aligned} \dot{x}(t) &= (A + \Delta A)x(t) + (A_d + \Delta A_d)x(t - \tau(t)) \\ &\quad + B(u(t) + w(x, x(t - \tau(t)), t)) \\ x(t) &= \varphi(t), t \in \left[-\tau, 0\right] \end{aligned} \tag{6.1}$$

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 67–83.
springerlink.com

where $x(t) \in R^n$ is the state, $w \in R^l$ is the matched uncertainty and disturbance. ΔA and ΔA_d are unmatched uncertainties, τ is a constant which is the upper bound of $\tau(t)$. $u(t) \in R^m$ is the control input, A, A_d and B are real constant matrices with appropriate dimensions and rank$(B) = m$.

Assumption 6.2.1 *The uncertainties ΔA and ΔA_d in (6.1) are assumed to have the following form*

$$\begin{aligned} \Delta A &= \textstyle\sum_{i=1}^{p} \theta_i A_i \\ \theta &= [\theta_1 \ \theta_2 \ \cdots \ \theta_p]^T \in \Omega = \{\theta | \ |\theta_i| \le 1\} \\ \Delta A_d &= \textstyle\sum_{i=1}^{q} \beta_i A_{di} \\ \beta &= [\beta_1 \ \beta_2 \ \cdots \ \beta_q]^T \in \Omega_d = \{\beta | \ |\beta_i| \le 1\} \end{aligned} \tag{6.2}$$

where the matrices $A_i, i = 1, \cdots, p$ and $A_{dj}, j = 1, \cdots, q$ are known with rank$(A_i) = a_i$ and rank$(A_{di}) = a_{di}$, θ_i and β_i are unknown constants. Suppose that A_i and A_{di} have full rank factorization of $A_i = G_i H_i$ and $A_{d_i} = G_{di} H_{di}$, respectively, where $G_i \in R^{n \times a_i}$, $H_i \in R^{a_i \times n}$, $G_{di} \in R^{n \times a_{di}}$ and $H_{di} \in R^{a_{di} \times n}$. Then

$$\begin{aligned} \Delta A &= \textstyle\sum_{i=1}^{p} \theta_i A_i = \left[G_1 \ G_2 \cdots G_p \right] D \left[H_1 \ H_2 \ \cdots \ H_p \right]^T = GDH \\ \Delta A_d &= \textstyle\sum_{i=1}^{q} \beta_i A_{di} = \left[G_{d1} \ G_{d2} \cdots G_{dq} \right] D_d \left[H_{d1} \ H_{d2} \cdots H_{dq} \right]^T \\ &= G_d D_d H_d \end{aligned} \tag{6.3}$$

where

$$\begin{aligned} D &= \mathrm{diag} \left(\theta_1 I_{a_1 \times a_1} \ \theta_2 I_{a_2 \times a_2} \ \cdots \ \theta_p I_{a_p \times a_p} \right) \\ D_d &= \mathrm{diag} \left(\beta_1 I_{a_{d1} \times a_{d1}} \ \beta_2 I_{a_{d2} \times a_{d2}} \ \cdots \ \beta_q I_{a_{dq} \times a_{dq}} \right). \end{aligned} \tag{6.4}$$

Remark 6.1. The parameters θ_i and β_i are unknown, which will be taken as uncertainties when sliding surfaces are designed, and will be estimated by an adaptive control law when reaching mode control is proposed.

Assumption 6.2.2 *The matched uncertainty $w(x, x(t-\tau(t)), t)$ is assumed to satisfy the following condition*

$$||w(x, x(t-\tau(t)), t)|| \le c + k||x(t)|| = \rho \tag{6.5}$$

where c and k are constants, but it may not be easily obtained due to the complexity of the structure of the uncertainty.

Assumption 6.2.3 *It is assumed that the bounded $\tau(t)$ is time-varying and differentiable, and $\dot{\tau}(t) \le d < 1$.*

To obtain a regular form of the systems (6.1), a nonsingular matrix T can be chosen such that

$$TB = \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix}$$

where $B_2 \in R^{m \times m}$ is nonsingular. For convenience, choose

$$T = \begin{bmatrix} {U_2}^T \\ {U_1}^T \end{bmatrix}$$

where $U_1 \in R^{n\times m}$ and $U_2 \in R^{n\times(n-m)}$ are two sub-blocks of a unitary matrix resulting from the singular value decomposition of B, i.e.,

$$B = \begin{bmatrix} U_1 \ U_2 \end{bmatrix} \begin{bmatrix} \Sigma \\ 0_{(n-m)\times m} \end{bmatrix} J^T$$

where $\Sigma \in R^{m\times m}$ is a diagonal positive-definite matrix and $J \in R^{m\times m}$ is a unitary matrix. By state transformation $z = Tx$, system (6.1) has the regular form

$$\begin{aligned} \dot{z}(t) &= (\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d)z(t-\tau(t)) \\ &\quad + \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + w(z(t), z(t-\tau(t)), t)) \\ z(t) &= \overline{\varphi}(t), t \in \left[-\tau, 0\right] \end{aligned} \tag{6.6}$$

where $\overline{A} = TAT^{-1}$, $\overline{A}_d = TA_dT^{-1}$, $\Delta\overline{A} = T\Delta AT^{-1}$, $\Delta\overline{A}_d = T\Delta A_dT^{-1}$ and $\overline{\varphi}(t) = T\varphi(t)$, $w(z(t), z(t-\tau(t)), t) = w(T^{-1}z(t), T^{-1}z(t-\tau(t)), t)$. It is obvious that $||w(z(t), z(t-\tau(t)), t)|| \le c + k||T^{-1}z(t)|| = \rho$. System (6.6) can be written as:

$$\begin{aligned} \dot{z}_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11})z_1(t) + (\overline{A}_{d11} + \Delta\overline{A}_{d11})z_1(t-\tau(t)) \\ &\quad + (\overline{A}_{12} + \Delta\overline{A}_{12})z_2(t) + (\overline{A}_{d12} + \Delta\overline{A}_{d12})z_2(t-\tau(t)) \\ \dot{z}_2(t) &= (\overline{A}_{21} + \Delta\overline{A}_{21})z_1(t) + (\overline{A}_{d21} + \Delta\overline{A}_{d21})z_1(t-\tau(t)) \\ &\quad + (\overline{A}_{22} + \Delta\overline{A}_{22})z_2(t) + (\overline{A}_{d22} + \Delta\overline{A}_{d22})z_2(t-\tau(t)) \\ &\quad + B_2(u + w(z(t), z(t-\tau(t)), t)) \\ z_1(t) &= \overline{\varphi}_1(t), t \in \left[-\tau, 0\right] \\ z_2(t) &= \overline{\varphi}_2(t), t \in \left[-\tau, 0\right] \end{aligned} \tag{6.7}$$

where $z_1 \in R^{n-m}$, $z_2 \in R^m$, $B_2 = \Sigma J^T$, $\overline{A}_{11} = U_2^T AU_2$, $\overline{A}_{12} = U_2^T AU_1$, $\overline{A}_{d11} = U_2^T A_d U_2$, $\overline{A}_{d12} = U_2^T A_d U_1$, $\Delta\overline{A}_{11} = U_2^T GDHU_2$, $\Delta\overline{A}_{12} = U_2^T GDHU_1$ $\Delta\overline{A}_{d11} = U_2^T G_d D_d H_d U_2$, $\Delta\overline{A}_{d12} = U_2^T G_d D_d H_d U_1$, $\overline{\varphi}_1(t) \in R^{(n-m)}$ and $\overline{\varphi}_2(t) \in R^m$ are the sub-blocks of $\overline{\varphi}(t)$.

It is obvious that the first equation of system (6.7) represents the sliding motion dynamics of system (6.6), hence the corresponding sliding surface can be chosen as follows:

$$\delta(t) = \begin{bmatrix} C \ I \end{bmatrix} z(t) = Cz_1(t) + z_2(t) = 0 \tag{6.8}$$

where $C \in R^{m\times(n-m)}$. Substituting $z_2 = -Cz_1$ into the first equation of system (6.7) gives the sliding motion

$$\begin{aligned}\dot{z}_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C)z_1(t) \\ &\quad +(\overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C)z_1(t-\tau(t)) \\ z_1(t) &= \overline{\varphi}_1(t), t \in \left[-\tau, 0\right].\end{aligned} \tag{6.9}$$

For simplicity, the sliding motion can be written as

$$\begin{aligned}\dot{z}_1(t) &= \tilde{A}z_1(t) + \tilde{A}_d z_1(t-\tau(t)) \\ z_1(t) &= \overline{\varphi}_1(t), t \in \left[-\tau, 0\right]\end{aligned} \tag{6.10}$$

where

$$\begin{aligned}\tilde{A} &= \overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C \\ \tilde{A}_d &= \overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C.\end{aligned} \tag{6.11}$$

To facilitate control design, scaling matrices and structural properties of the system are introduced below.

In accordance to the structures of D and D_d defined in (6.4), the following scaling matrices are defined and used as design parameters latter:

$$\begin{aligned}S_D = \{Y | Y &= \mathrm{diag}(Y_1, \cdots, Y_p) \\ 0 &< Y_i = Y_i^T \in R^{n_{a_i} \times n_{a_i}}\}\end{aligned} \tag{6.12}$$

$$\begin{aligned}S_{D_d} = \{Y_d | Y_d &= \mathrm{diag}(Y_{d1}, \cdots, Y_{dq}) \\ 0 &< Y_{di} = {Y_{di}}^T \in R^{n_{a_{di}} \times n_{a_{di}}}\}.\end{aligned} \tag{6.13}$$

To this end, the following lemma is recalled.

Lemma 6.2. [100] Let $D \in S_D$, $D_d \in S_{D_d}$. Then for any $X \in S_D$ and $X_d \in S_{D_d}$, the following inequalities

$$GDH + (GDH)^T \leq GXG^T + H^T X^{-1} H \tag{6.14}$$

$$G_d D_d H_d + (G_d D_d H_d)^T \leq G_d X_d {G_d}^T + {H_d}^T {X_d}^{-1} H_d \tag{6.15}$$

hold.

6.3 Main Results

The objective in this chapter is to a design constant gain $C \in R^{m \times (n-m)}$ and a reaching motion control law $u(t)$ such that

1) The sliding motion (6.10) is quadratically stable; and

2) The trajectory of the closed-loop system (6.7) is convergent into a residual set of the origin with the reaching control law $u(t)$.

Definition 6.3. [128]: The uncertain sliding motion (6.10) is said to be robustly stable if the equilibrium solution $z_1(t) = 0$ of the functional differential equation associated to sliding motion (6.10) is globally uniformly asymptotically stable for all admissible uncertainty $\Delta\overline{A}_1$ and $\Delta\overline{A}_{d1}$.

Now, we present our first result in this chapter.

Theorem 6.4. *For given scalar $\lambda > 0$, the reduced order system (6.10) is quadratically stable if there exist symmetric positive-definite matrix $P \in R^{(n-m)\times(n-m)}$, symmetric semi-positive matrix $Z \in R^{(n-m)\times(n-m)}$, general matrices $V \in R^{m\times(n-m)}$, and $Y, M, N, Q \in R^{(n-m)\times(n-m)}$, $X \in S_D$, and $X_d \in S_{D_d}$ such that*

$$\begin{bmatrix} \Theta_{11} & \Theta_{12} & \Theta_{13} & \Theta_{14} & 0 \\ \Theta_{12}^T & \Theta_{22} & \Theta_{23} & 0 & \Theta_{25} \\ \Theta_{13}^T & \Theta_{23}^T & \Theta_{33} & 0 & 0 \\ \Theta_{14}^T & 0 & 0 & \Theta_{44} & 0 \\ 0 & \Theta_{25}^T & 0 & 0 & \Theta_{55} \end{bmatrix} < 0 \tag{6.16}$$

and

$$\Omega = \begin{bmatrix} X & Y & M \\ Y^T & Z & N \\ M^T & N^T & \lambda P \end{bmatrix} \geq 0 \tag{6.17}$$

where

$$\begin{aligned}
\Theta_{11} &= \overline{A}_{11}P - \overline{A}_{12}V + P\overline{A}_{11}^T - V^T\overline{A}_{12}^T + M + M^T + Q \\
&\quad + \overline{\tau}X + U_2^T GXG^T U_2 + U_2^T G_d X_d G_d^T U_2 \\
\Theta_{12} &= \Theta_{21}^T = \overline{A}_{d11}P - \overline{A}_{d12}^T V - M + N^T + \overline{\tau}Y \\
\Theta_{13} &= \Theta_{31}^T = \overline{\tau}\lambda(P\overline{A}_{11}^T - V^T\overline{A}_{12}^T + U_2^T GXG^T U_2 \\
&\quad + U_2^T G_d X_d G_d^T U_2) \\
\Theta_{14} &= \Theta_{41}^T = PU_2^T H^T - V^T U_1^T H^T \\
\Theta_{22} &= -N - N^T - (1-d)Q + \overline{\tau}Z \\
\Theta_{23} &= \Theta_{32}^T = \overline{\tau}\lambda(P\overline{A}_{d11}^T - V^T\overline{A}_{d12}^T) \\
\Theta_{25} &= \Theta_{52}^T = PU_2^T H_d^T - V^T U_1^T H_d^T \\
\Theta_{33} &= -\overline{\tau}\lambda P + \overline{\tau}^2\lambda^2(U_2^T GXG^T U_2 + U_2^T G_d X_d G_d^T U_2) \\
\Theta_{44} &= -X \\
\Theta_{55} &= -X_d
\end{aligned}$$

and the rest of entries are zero.
Moreover, gain $C = VP^{-1}$ and the sliding surface of system (6.7) is

$$\delta(t) = VP^{-1}z_1(t) + z_2(t) = 0. \tag{6.18}$$

Proof. Taking a scalar λ, symmetric positive-definite matrix variables $P, Q \in R^{(n-m)\times(n-m)}$ and choosing the Lyapunov functional candidate as

$$\begin{aligned}\mathcal{V}(t) = x^T(t)Px(t) + \textstyle\int_{t-\tau(t)}^{t} x^T(s)Qx(s)ds \\ + \lambda \textstyle\int_{-\overline{\tau}}^{0}\int_{t+\theta}^{t} \dot{x}^T(s)P\dot{x}(s)dsd\theta\end{aligned} \tag{6.19}$$

it follows that the Lyapunov derivative corresponding to system (6.10) is given by

$$\begin{aligned}\dot{\mathcal{V}}(t) &= x^T(t)[P\tilde{A} + \tilde{A}^T P]x(t) + 2x^T(t)P\tilde{A}_d x(t-\tau(t)) \\ &\quad + x^T(t)Qx(t) - (1-\dot{\tau}(t))x^T(t-\tau(t))Qx(t-\tau(t)) \\ &\quad + \lambda\overline{\tau}\dot{x}^T(t)P\dot{x}(t) - \lambda \textstyle\int_{t-\overline{\tau}}^{t} \dot{x}^T(s)P\dot{x}(s)ds \\ &\leq x^T(t)[P\tilde{A} + \tilde{A}^T P]x(t) + 2x^T(t)P\tilde{A}_d x(t-\tau(t)) \\ &\quad + x^T(t)Qx(t) \\ &\quad -(1-d)x^T(t-\tau(t))Qx(t-\tau(t)) + \lambda\overline{\tau}\dot{x}(t)P\dot{x}(t) \\ &\quad -\lambda \textstyle\int_{t-\tau(t)}^{t} \dot{x}^T(s)P\dot{x}(s)ds.\end{aligned} \tag{6.20}$$

It is easy to see that

$$x(t-\tau(t)) = x(t) - \int_{t-\tau(t)}^{t} \dot{x}(s)ds \tag{6.21}$$

then, for any matrices M and N with appropriate dimensions, the following equation is derived:

$$2[x^T(t)M + x^T(t-\tau(t))N][x(t) - x(t-\tau(t)) - \int_{t-\tau(t)}^{t} \dot{x}(s)ds] = 0. \tag{6.22}$$

For any semi-positive definite matrices X, Z and general matrix Y such that $W = \begin{bmatrix} X & Y \\ Y^T & Z \end{bmatrix} \geq 0$, the following result is obvious

$$\begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}^T W \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix} \geq 0 \tag{6.23}$$

then,

$$\begin{aligned}&\overline{\tau}\begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}^T W \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix} \\ &- \int_{t-\tau(t)}^{t} \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}^T W \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix} ds \\ &= (\overline{\tau} - \tau(t))\begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}^T W \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix} \geq 0.\end{aligned}$$

For simplicity, let

$$\alpha(t) = \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}, \beta(t,s) = \begin{bmatrix} x(t) \\ x(t-\tau(t)) \\ \dot{x}(s) \end{bmatrix}. \tag{6.24}$$

Then, adding nonnegative terms into the right side of (6.20) results in

$$\begin{aligned} \dot{\mathcal{V}}(t) \leq\ & x^T(t)[P\tilde{A} + \tilde{A}^T P]x(t) + 2x^T(t)P\tilde{A}_d x(t-\tau(t)) \\ & + x^T(t)Qx(t) - (1-d)x^T(t-\tau(t))Qx(t-\tau(t)) \\ & + \lambda\overline{\tau}\dot{x}(t)P\dot{x}(t) - \lambda\int_{t-\tau(t)}^{t} \dot{x}^T(s)P\dot{x}(s)ds + 2[x^T(t)M \\ & + x^T(t-\tau(t))N][x(t) - x(t-\tau(t)) - \int_{t-\tau(t)}^{t} \dot{x}(s)ds] \\ & + \overline{\tau}\alpha^T(t)W\alpha(t) - \int_{t-\tau(t)}^{t} \alpha^T(t)W\alpha(t)ds \\ =\ & \alpha^T(t)\Phi\alpha(t) - \int_{t-\tau(t)}^{t} \beta^T(t,s)\Omega\beta(t,s)ds \end{aligned} \tag{6.25}$$

where

$$\begin{aligned} \Phi &= \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{12}^T & \Phi_{22} \end{bmatrix} \\ \Phi_{11} &= P\tilde{A} + \tilde{A}^T P + \overline{\tau}\lambda\tilde{A}^T P\tilde{A} + M + M^T + Q + \overline{\tau}X \\ \Phi_{12} &= P\tilde{A}_d + \overline{\tau}\lambda A^T P\tilde{A}_d - M + N^T + \overline{\tau}Y \\ \Phi_{22} &= \overline{\tau}\lambda\tilde{A}_d^T P\tilde{A}_d - N - N^T - (1-d)Q + \overline{\tau}Z \end{aligned} \tag{6.26}$$

$$\Omega = \begin{bmatrix} X & Y & M \\ Y^T & Z & N \\ M^T & N^T & \lambda P \end{bmatrix} \geq 0. \tag{6.27}$$

Then, if $\Phi < 0$ and $\Omega \geq 0$, $\dot{\mathcal{V}}(t) < 0$ for any $\alpha(t) \neq 0$. By Schur complement, $\Phi < 0$ is equivalent to the following inequality:

$$\begin{bmatrix} \Phi_{11} - \overline{\tau}\lambda\tilde{A}^T P\tilde{A} & \Phi_{12} - \overline{\tau}\lambda A^T P\tilde{A}_d & \overline{\tau}\lambda\tilde{A}^T P \\ (\Phi_{12} - \overline{\tau}\lambda A^T P\tilde{A}_d)^T & \Phi_{22} - \overline{\tau}\lambda\tilde{A}_d^T P\tilde{A}_d & \overline{\tau}\lambda\tilde{A}_d^T P \\ \overline{\tau}\lambda P\tilde{A} & \overline{\tau}\lambda P\tilde{A}_d & -\overline{\tau}\lambda P \end{bmatrix} < 0. \tag{6.28}$$

Multiplying both sides of inequalities (6.27) and (6.28) with diag(P^{-1}, P^{-1}, P^{-1}), and let $\tilde{P} = P^{-1}$, $\tilde{M} = P^{-1}MP^{-1}$, $\tilde{N} = P^{-1}NP^{-1}$,$\tilde{X} = P^{-1}XP^{-1}$, $\tilde{Q} = P^{-1}QP^{-1}$, $\tilde{Y} = P^{-1}YP^{-1}$, yields

$$\begin{bmatrix} \tilde{X} & \tilde{Y} & \tilde{M} \\ \tilde{Y}^T & \tilde{Z} & \tilde{N} \\ \tilde{M}^T & \tilde{N}^T & \lambda\tilde{P} \end{bmatrix} \geq 0 \tag{6.29}$$

$$\begin{bmatrix} \tilde{\Phi}_{11} & \tilde{\Phi}_{12} & \tilde{\Phi}_{13} \\ \tilde{\Phi}_{12}^T & \tilde{\Phi}_{22} & \tilde{\Phi}_{23} \\ \tilde{\Phi}_{13}^T & \tilde{\Phi}_{23}^T & \tilde{\Phi}_{33} \end{bmatrix} < 0 \tag{6.30}$$

where $\tilde{\Phi}_{11} = \tilde{A}\tilde{P} + \tilde{P}\tilde{A}^T + \tilde{M} + \tilde{M}^T + \tilde{Q} + \overline{\tau}\tilde{X}$, $\tilde{\Phi}_{12} = \tilde{A}_d\tilde{P} - \tilde{M} + \tilde{N}^T + \overline{\tau}\tilde{Y}$, $\tilde{\Phi}_{13} = \overline{\tau}\lambda\tilde{P}\tilde{A}^T$, $\tilde{\Phi}_{22} = -\tilde{N} - \tilde{N}^T - (1-d)\tilde{Q} + \overline{\tau}\tilde{Z}$,$\tilde{\Phi}_{23} = \overline{\tau}\lambda\tilde{P}\tilde{A}_d^T$, $\tilde{\Phi}_{33} = -\overline{\tau}\lambda\tilde{P}$.

Note that (6.11), and let $V = C\tilde{P}$, $\Omega_{111} = \overline{A}_{11}\tilde{P} - \overline{A}_{12}V + \tilde{P}\overline{A}_{11}^T - V^T\overline{A}_{12}^T + \tilde{M} + \tilde{M}^T + \tilde{Q} + \overline{\tau}\tilde{X}$, $\Omega_{121} = \tilde{P}\overline{A}_{d11}^T - V^T\overline{A}_{d12}^T - \tilde{M}^T + \tilde{N} + \overline{\tau}\tilde{Y}^T$, $\Omega_{122} = -\tilde{N} - \tilde{N}^T - (1-d)\tilde{Q} + \overline{\tau}\tilde{Z}$,$\Omega_{131} = \overline{\tau}\lambda(\overline{A}_{11}\tilde{P} - \overline{A}_{12}V)$, $\Omega_{132} = \overline{\tau}\lambda(\overline{A}_{d11}\tilde{P} - \overline{A}_{d12}V)$, $\Omega_{133} = -\overline{\tau}\lambda\tilde{P}$ and

$$\Omega_1 = \begin{bmatrix} \Omega_{111} & \Omega_{121}^T & \Omega_{131}^T \\ \Omega_{121} & \Omega_{122} & \Omega_{132}^T \\ \Omega_{131} & \Omega_{132} & \Omega_{133} \end{bmatrix}$$

$$\Omega_2 = \begin{bmatrix} U_2^T G & U_2^T G_d \\ 0 & 0 \\ \overline{\tau}\lambda U_2^T G & \overline{\tau}\lambda U_2^T G_d \end{bmatrix}$$

$$\Omega_3 = \begin{bmatrix} HU_2P - HU_1V & 0 \\ 0 & H_dU_2P - H_dU_1V \end{bmatrix}$$

$$\overline{D} = \begin{bmatrix} D & 0 \\ 0 & D_d \end{bmatrix} \tag{6.31}$$

then, (6.30) can be written as

$$\Omega_1 + \Omega_2\overline{D}\begin{bmatrix} \Omega_3 & 0 \end{bmatrix} + \begin{bmatrix} \Omega_3 & 0 \end{bmatrix}^T \overline{D}^T \Omega_2^T < 0. \tag{6.32}$$

By Lemma 6.2, equation (6.32) holds if there exist $X \in S_D$ and $X_d \in S_{D_d}$ such that

$$\Omega_1 + \Omega_2 \begin{bmatrix} X & 0 \\ 0 & X_d \end{bmatrix} \Omega_2^T + \begin{bmatrix} \Omega_3 & 0 \end{bmatrix}^T \begin{bmatrix} X^{-1} & 0 \\ 0 & X_d^{-1} \end{bmatrix} \begin{bmatrix} \Omega_3 & 0 \end{bmatrix} < 0. \tag{6.33}$$

By Schur complement formula, inequality (6.33) is equivalent to

$$\begin{bmatrix} \Omega_1 + \Omega_2 \begin{bmatrix} X & 0 \\ 0 & X_d \end{bmatrix} \Omega_2^T & \begin{bmatrix} \Omega_3 & 0 \end{bmatrix}^T \\ \begin{bmatrix} \Omega_3 & 0 \end{bmatrix} & -\begin{bmatrix} X & 0 \\ 0 & X_d \end{bmatrix} \end{bmatrix} < 0. \tag{6.34}$$

Remark 6.5. Theorem 6.4 presents a sufficient condition for the problem of sliding motion stability, described by (6.10). A desired sliding surface can be constructed by solving the matrix inequalities in (6.16) and (6.17), which can be implemented by employing some standard numerical algorithms, see for example, [15]. It should be noted that the underlying problem may not always have feasible solutions. Fortunately, there is an adjustable scalar λ in the inequalities. If the unsolvable case appears, then the feasible solutions could still be obtained by appropriately adjusting the scalar.

Remark 6.6. When $\lambda = 0$, the condition is delay independent. When $\lambda \neq 0$, inequalities (6.16) and (6.17) are nonlinear since there are products of two variables. They can also be solved using LMI toolbox by fixing λ first, then searching for feasible solutions using different values of λ. The introduction equality (6.22) can give less conservative results than the method in [189], the details can be found in [78] and [207].

The severe drawback of SMC is that it is discontinuous across sliding surfaces. The discontinuity leads to control chattering in practice, which involves high-frequency dynamics. To remove control chattering, the following smooth projection will be introduced.

Definition 6.7. [58] Let $\theta = [\theta_1 \ \theta_2 \ \cdots \ \theta_p]^T \in \Omega$ be an unknown parameter vector, $\hat{\theta}$ be the estimate, and $\Omega \in R^p$ be a closed ball of known radius r_Ω. The projection algorithm $\mathrm{Proj}(y, \hat{\theta})$ is given by

$$\mathrm{Proj}(y,\hat{\theta}) = \begin{cases} y, & \text{if } p(\hat{\theta}) \leq 0 \\ y, & \text{if } p(\hat{\theta}) \geq 0 \text{ and } \frac{\partial p(\hat{\theta})}{\partial \hat{\theta}} y \leq 0 \\ y - \frac{p(\hat{\theta}) \frac{\partial p(\hat{\theta})}{\partial \hat{\theta}} y}{\|\frac{\partial p(\hat{\theta})}{\partial \hat{\theta}}\|^2} \frac{\partial p^T(\hat{\theta})}{\partial \hat{\theta}}, & \text{otherwise} \end{cases} \tag{6.35}$$

where $p(\hat{\theta}) = \frac{\|\hat{\theta}\|^2 - r_\Omega^2}{\epsilon^2 + 2\epsilon r_\Omega}$, ϵ is an arbitrary positive real scalar. From (6.35), if $\hat{\theta}(0) \in \Omega$, the following nice properties follow immediately:

1) $\|\hat{\theta}(t)\| \leq r_\Omega + \epsilon, \forall t \geq 0$;
2) $\mathrm{Proj}(y, \hat{\theta})$ is Lipschitz continuous;
3) $\|\mathrm{Proj}(y, \hat{\theta})\| \leq \|y\|$;
4) $\tilde{\theta}^T \mathrm{Proj}(y, \hat{\theta}) \geq \tilde{\theta}^T y, (\tilde{\theta} = \theta - \hat{\theta})$.

Definition 6.8. [97] Consider the nonlinear system, $\dot{x} = f(x, u)$, $y = h(x)$ where x is a state vector, u is the input vector and y is the output vector. The solution is uniformly ultimately bounded (UUB) if for all $x(t_0) = x_0$, there exists $\varepsilon > 0$ and $T(\varepsilon, x_0)$, such that $\|x(t)\| < \varepsilon$, for all $t \geq t_0 + T$.

Based on Theorem 6.4 and the Definition 6.3, that is, sliding motion enters a neighborhood of equilibrium in finite time and remains within it thereafter [28], the result of designing the reaching motion controller is given in the next theorem.

Theorem 6.9. *With the gain C obtained in Theorem 6.4 and the linear sliding surface is given by (6.18). Then the trajectory of the closed-loop system (6.7) can be driven onto the sliding surface in finite time with the control (6.36) and evolves in a neighborhood around the sliding surface, and finally, converges into a residual set at the origin.*

$$\begin{aligned} u = & -B_2^{-1}[\Pi\delta(t) + \overline{CA}z(t) + \overline{CA}_d z(t-\tau(t)) + f_1(z(t))\hat{\theta} \\ & + f_2(z(t-\tau(t)))\hat{\beta}] + u_N \end{aligned} \tag{6.36}$$

$$u_N = \begin{cases} -\frac{B_2^T\delta(t)}{||B_2^T\delta(t)||}\hat{\rho}, & \text{if } \hat{\rho}||B_2^T\delta(t)|| > \epsilon \\ -\frac{B_2^T\delta(t)}{\epsilon}\hat{\rho}^2, & \text{if } \hat{\rho}||B_2^T\delta(t)|| \le \epsilon \end{cases} \tag{6.37}$$

and the adaptation laws are

$$\dot{\hat{\theta}} = q_1 \text{Proj}(f_1^T(z(t))\delta(t), \hat{\theta}) \tag{6.38}$$

$$\dot{\hat{\beta}} = q_2 \text{Proj}(f_2^T(z(t-\tau(t)))\delta(t), \hat{\beta}) \tag{6.39}$$

$$\dot{\hat{c}}(t,z) = q_3(-\epsilon_0\hat{c} + ||B_2^T\delta(t)||) \tag{6.40}$$

$$\dot{\hat{k}}(t,z) = q_4(-\epsilon_1\hat{k} + ||B_2^T\delta(t)||||z||) \tag{6.41}$$

where Π is a positive definite matrix, q_1, q_2, q_3, q_4, ϵ_0 and ϵ_1 are design parameters, and $\overline{C} = [C \ I]$,

$$\begin{aligned} &\hat{\rho} = \hat{c}(z(t),t) + \hat{k}(z(t),t)||z(t)|| \\ &f_1(z(t)) = [\overline{C}TA_1T^{-1}z(t) \cdots \ \overline{C}TA_pT^{-1}z(t)] \\ &f_2(z(t-\tau(t))) = [\overline{C}TA_{d1}T^{-1}z(t-\tau(t)) \cdots \ \overline{C}TA_{dq}T^{-1}z(t-\tau(t))]. \end{aligned} \tag{6.42}$$

Proof. We will complete the proof by showing that via the control law (6.36)-(6.41), the trajectory of the closed-loop system (6.7) can be driven onto the sliding surface in finite time and evolves in a neighborhood around the sliding surface. In the steady state, it is convergent into a residual set at the origin. Consider the following Lyapunov function:

$$\mathcal{V} = \frac{1}{2}[\delta^T(t)\delta(t) + \frac{1}{q_1}\tilde{\theta}^T\tilde{\theta} + \frac{1}{q_2}\tilde{\beta}^T\tilde{\beta} + \frac{1}{q_3}\tilde{c}^2 + \frac{1}{q_4}\tilde{k}^2] \tag{6.43}$$

where $\tilde{\theta} = \theta - \hat{\theta}$, $\tilde{\beta} = \beta - \hat{\beta}$, $\tilde{c} = c - \hat{c}(z(t),t)$ and $\tilde{k} = k - \hat{k}(z(t),t)$. Its time derivation is

$$\dot{\mathcal{V}} = \ \delta^T(t)\dot{\delta}(t) - \frac{1}{q_1}\tilde{\theta}^T\dot{\hat{\theta}} - \frac{1}{q_2}\tilde{\beta}\dot{\hat{\beta}} - \frac{1}{q_3}\tilde{c}\dot{\hat{c}} - \frac{1}{q_4}\tilde{k}\dot{\hat{k}}. \tag{6.44}$$

From the sliding surface

$$\delta(t) = \begin{bmatrix} C \ I \end{bmatrix} z$$

we have

$$\begin{aligned} \dot{\delta}(t) \ &= \begin{bmatrix} C \ I \end{bmatrix} \dot{z} = \overline{C}(\overline{A} + \Delta\overline{A})z(t) + \overline{C}(\overline{A}_d + \Delta\overline{A}_d)z(t-\tau(t)) \\ &\quad + B_2(u(t) + w(z, z(t-\tau(t)), t)). \end{aligned} \tag{6.45}$$

If $||B_2^T\delta||\hat{\rho} > \epsilon$, with the control law defined in (6.36) and adaptation laws defined in (6.38)-(6.41), we have

$$\begin{aligned}\dot{\mathcal{V}} &= \delta^T(t)[-\Pi\delta(t) + \overline{C}\Delta\overline{A}z(t) + \overline{C}\Delta\overline{A}_d z(t-\tau(t)) - f_1(z(t))\hat{\theta} \\ &-f_2(z(t-\tau(t)))\hat{\beta}] - \tilde{\theta}^T\mathrm{Proj}(f_1^T(z(t))\delta(t),\hat{\theta}) \\ &-\tilde{\beta}^T\mathrm{Proj}(f_2^T(z(t-\tau(t)))\delta(t),\hat{\beta}) + \delta^T(t)B_2u_N \\ &+\delta^T(t)B_2w(z(t),z(t-\tau(t)),t) - \tilde{c}(-\epsilon_0\hat{c} + ||B_2^T\delta(t)||) \\ &-\tilde{k}(-\epsilon_1\hat{k} + ||B_2^T\delta(t)||||z||)\end{aligned}$$

where $\delta^T B_2 u_N = -\delta^T B_2 \frac{B_2^T\delta}{||B_2^T\delta(t)||}\hat{\rho} = -||B_2^T\delta(t)||(\hat{k}||z|| + \hat{c})$.

From (6.2), we have

$$\begin{aligned}\bar{C}\Delta\overline{A}z(t) &= \bar{C}\textstyle\sum_{i=1}^p \theta_i\bar{A}_i z(t) = \sum_{i=1}^p \theta_i\bar{C}TA_iT^{-1}z(t) \\ &= [\overline{C}TA_1T^{-1}z(t) \quad \overline{C}TA_2T^{-1}z(t) \\ &\qquad \cdots \ \overline{C}TA_pT^{-1}z(t)]\theta \\ &= f_1(z(t))\theta.\end{aligned} \tag{6.46}$$

Similarly, we have

$$\bar{C}\Delta\overline{A_d}z(t-\tau(t)) = f_2(z(t-\tau(t)))\beta. \tag{6.47}$$

Then we get

$$\begin{aligned}\dot{\mathcal{V}} &= -\delta^T(t)\Pi\delta(t) - \delta^T(t)f_1(z(t))\hat{\theta} - \delta^T(t)f_2(z(t-\tau(t)))\hat{\beta} \\ &+\delta^T(t)f_1(z(t))\theta + \delta^T(t)f_2(z(t-\tau(t)))\beta \\ &-\tilde{\beta}^T\mathrm{Proj}(f_2^T(z(t-\tau(t)))\delta(t),\hat{\beta}) - \tilde{\theta}^T\mathrm{Proj}(f_1^T(z(t))\delta(t),\hat{\theta}) \\ &-||B_2^T\delta(t)||(\hat{k}||z|| + \hat{c}) + \delta^T(t)B_2w(z(t),z(t-\tau(t)),t) \\ &-\tilde{c}(-\epsilon_0\hat{c} + ||B_2^T\delta(t)||) - \tilde{k}(-\epsilon_1\hat{k} + ||B_2^T\delta(t)||||z||).\end{aligned} \tag{6.48}$$

It follows from $\tilde{\theta} = \theta - \hat{\theta}$, and $\tilde{\beta} = \beta - \hat{\beta}$ that

$$\begin{aligned}\dot{\mathcal{V}} &= -\delta^T(t)\Pi\delta(t) - \delta^T(t)f_1(z(t))\tilde{\theta} - \tilde{\theta}^T\mathrm{Proj}(f_1^T(z(t))\delta(t),\hat{\theta}) \\ &-\delta(t)f_2^T(z(t-\tau(t)))\tilde{\beta} + \tilde{\beta}^T\mathrm{Proj}(f_2^T(z(t-\tau(t)))\delta(t),\hat{\beta}) \\ &-||B_2^T\delta(t)||(\hat{k}||z|| + \hat{c}) + \delta^T(t)B_2w(z(t),z(t-\tau(t)),t) \\ &-\tilde{c}(-\epsilon_0\hat{c} + ||B_2^T\delta(t)||) - \tilde{k}(-\epsilon_1\hat{k} + ||B_2^T\delta(t)||||z||)\end{aligned} \tag{6.49}$$

$$\begin{aligned}\delta^T(t)f_1(z(t))\tilde{\theta} &= \delta^T(t)\textstyle\sum_{i=1}^p \tilde{\theta}_i\bar{C}TA_iT^{-1}z(t) \\ &= \textstyle\sum_{i=1}^p \tilde{\theta}_i\delta^T(t)\bar{C}TA_iT^{-1}z(t).\end{aligned} \tag{6.50}$$

It follows $\delta^T(t)\overline{C}TA_1T^{-1}z(t) \in R$ that

$$\delta^T(t)\overline{C}TA_1T^{-1}z(t) = (\overline{C}TA_1T^{-1}z(t))^T\delta(t)$$

$$\begin{aligned}\delta^T(t)f_1(z(t))\tilde{\theta} &= \textstyle\sum_{i=1}^{p}\tilde{\theta}_i(\bar{C}TA_iT^{-1}z(t))^T\delta(t)\\ &= \tilde{\theta}^T[\overline{C}TA_1T^{-1}z(t) \quad \overline{C}TA_2T^{-1}z(t)\\ &\qquad \cdots \ \overline{C}TA_pT^{-1}z(t)]^T\delta(t)\\ &= \tilde{\theta}^T f_1^T(z(t))\delta(t).\end{aligned} \tag{6.51}$$

Similarly we have

$$\delta^T(t)f_2(z(t-\tau(t)))\tilde{\beta} = \tilde{\beta}^T f_2^T(z(t-\tau(t)))\delta(t). \tag{6.52}$$

It follows from (6.42) and Assumption 6.2.3 that

$$\begin{aligned}\dot{\mathcal{V}} \ &\le -\delta^T(t)\Pi\delta(t) + \tilde{\theta}^T(f_1(z(t))\delta(t) - \mathrm{Proj}(f_1^T(z(t)\delta(t),\hat{\theta}))\\ &+\tilde{\beta}^T(t)(f_2(z(t-\tau(t))\delta(t) - \mathrm{Proj}(f_2^T(z(t-\tau(t)))\delta(t),\hat{\beta})\\ &+ ||B_2^T\delta(t)||(k||z||+c) - ||B_2^T\delta(t)||(\hat{k}||z||+\hat{c})\\ &-\tilde{c}(-\epsilon_0\hat{c} + ||B_2^T\delta(t)||) - \tilde{k}(-\epsilon_1\hat{k} + ||B_2^T\delta(t)||||z||).\end{aligned} \tag{6.53}$$

From the Property 4 of the operator $\mathrm{Proj}(y,\hat{\theta})$ and completing the squares we obtain

$$\begin{aligned}\dot{\mathcal{V}} \ &\le -\delta^T(t)\Pi\delta(t) + \epsilon_0\tilde{c}\hat{c} + \epsilon_1\tilde{k}\hat{k}\\ &= -\delta^T(t)\Pi\delta(t) - \epsilon_0(\hat{c} - \frac{1}{2}c)^2 - \epsilon_1(\hat{k} - \frac{1}{2}k)^2 + \frac{1}{4}(\epsilon_0c^2 + \epsilon_1k^2)\\ &\le -\delta^T(t)\Pi\delta(t) + \frac{1}{4}(\epsilon_0c^2 + \epsilon_1k^2).\end{aligned} \tag{6.54}$$

If $||B_2^T\delta(t)||\hat{\rho}^2 \le \epsilon$, with the control law defined in (6.37) and adaptation laws defined in (6.38)-(6.41), we obtain

$$\begin{aligned}\dot{\mathcal{V}} \ &\le -\delta^T(t)\Pi\delta(t) - \frac{||B_2^T\delta(t)||^2}{\epsilon}\hat{\rho}^2 + ||B_2^T\delta(t)||(k||z||+c)\\ &\quad - \tilde{c}(-\epsilon_0\hat{c} + ||B_2^T\delta(t)||) - \tilde{k}(-\epsilon_1\hat{k} + ||B_2^T\delta(t)||||z||)\\ &= -\delta^T(t)\Pi\delta(t) - \frac{||B_2^T\delta(t)||^2}{\epsilon}\hat{\rho}^2 + ||B_2^T\delta(t)||\hat{\rho} + \epsilon_0\tilde{c}\hat{c} + \epsilon_1\tilde{k}\hat{k}\\ &= -\delta^T(t)\Pi\delta(t) - (\frac{||B_2^T\delta(t)||}{\sqrt{\epsilon}}\hat{\rho} - \frac{\sqrt{\epsilon}}{2})^2 + \frac{\epsilon}{4}\\ &\quad - \epsilon_0(\hat{c} - \frac{1}{2}c)^2 - \epsilon_1(\hat{k} - \frac{1}{2}k)^2 + \frac{1}{4}(\epsilon_0c^2 + \epsilon_1k^2)\\ &\le -\delta^T(t)\Pi\delta(t) + \frac{\epsilon}{4} + \frac{1}{4}(\epsilon_0c^2 + \epsilon_1k^2).\end{aligned} \tag{6.55}$$

It can be concluded now from (6.54) and (6.55) that all signals are uniformly ultimately bounded.

In order to obtain a continuous control law (6.36), the unknown parameters are estimated with the smooth projection. The continuous positive scalar valued function ρ satisfying Assumption 6.2.3 is estimated by a smoothed SMC control law taking account of the boundary layer effect, that is the part, u_N. The benefits of this kind of smooth techniques have been stated in [179], which offers a continuous approximation to the discontinuous SMC law inside the boundary layer and guarantees the output tracking error within any neighborhood of the sliding surface. However, asymptotic stability is lost and it can only guarantee the bounded motion about the sliding surface. Therefore, we cannot analyze the stability of the dynamics of the sliding mode that is restricted on the sliding surface. In the following, the uniformly ultimately bounded of sliding motion will be investigated. The dynamics of the sliding motion around the sliding surface is described by

$$\begin{aligned} \dot{z_1}(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C)z_1(t) \\ &\quad +(\overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C)z_1(t-\tau(t)) \\ &\quad +(\overline{A}_{12} + \Delta\overline{A}_{12})\delta(t) + (\overline{A}_{d12} + \Delta\overline{A}_{d12})\delta(t-\tau(t)) \\ z_1(t) &= \overline{\varphi}_1(t), t \in \left[-\tau, 0\right]. \end{aligned} \tag{6.56}$$

It has been proved in Theorem 6.9 that $\delta(t)$ is ultimately bounded, the same holds for $\delta(t-\tau(t))$ as $t \to \infty$, then $(\overline{A}_{12} + \Delta\overline{A}_{12})\delta(t) + (\overline{A}_{d12} + \Delta\overline{A}_{d12})\delta(t-\tau(t))$ will be bounded as well, and can be viewed as a bounded disturbance in the dynamics of (6.56). As gain C is designed to guarantee the quadratic stability of system (6.10), which will not converge to zero due to the existence of bounded disturbances, however, it will stay in a domain containing zero within the prescribed precision.

Remark 6.10. When the time-delay is time-varying but unknown, the method proposed in this chapter can also work with the slightly changed algorithm of control law presented in this chapter. As time-delay $\tau(t)$ can not be used in the control law, the terms containing time-delay can be viewed as uncertainties. Then, the terms with time-delay and disturbances can be lumped together and are bounded by a function similar to (6.5). Therefore a similar adaptive control can be used, and the upper bound parameters can be estimated with the adaptive law.

6.4 Numerical Example

In this section an illustrative example is given to verify the effectiveness of the design method developed in this chapter.

Example 6.11. Consider the uncertain system (6.1) with

$$A = \begin{bmatrix} -4 & 1.5 \\ 4 & -2 \end{bmatrix}, A_d = \begin{bmatrix} 0.7 & 0.8 \\ 0.6 & 0.9 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}$$

$\tau(t) = 2.0806(1 + \sin \frac{t}{2\times 2.0806})$, $F = 1$, $\varphi(t) = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$, for $t \in \begin{bmatrix} -2.0806 & 0 \end{bmatrix}$, $w(t) = 0.01 \sin t$.

Simply we consider

$$\begin{aligned} \Delta A &= \theta A_1 \\ \Delta A_d &= \beta A_{d1} \end{aligned}$$

where $|\theta| \leq 1$ and $|\beta| \leq 1$. A_1 and A_{d1} are factorized as follows:

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$A_{d1} = \begin{bmatrix} 0.2 & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0.2 & 0 \end{bmatrix}.$$

According to (6.3), it can be shown that

$$G = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, H = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$G_d = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, H_d = \begin{bmatrix} 0.2 \\ 0 \end{bmatrix}.$$

According to Remark 6.6, when $\lambda = 0$, the condition is delay independent. When $\lambda \neq 0$, inequalities (6.16) and (6.17) are nonlinear. With LMI toolbox by fixing λ first, then feasible solutions can be found using different values of λ and d. The following table shows the upper bound of time-delay varies with λ and d:

	λ									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0	7.0612	4.0350	3.0432	2.5587	2.2729	2.0827	1.9447	1.8377	1.7504	1.6763
0.1	6.9989	3.9939	3.0094	2.5284	2.2443	2.0552	1.9178	1.8114	1.7246	1.6511
0.2	6.9252	3.9455	2.9700	2.4934	2.2119	2.0243	1.8881	1.7827	1.6969	1.6245
0.3	6.8362	3.8876	2.9234	2.4526	2.1746	1.9895	1.8552	1.7514	1.6673	1.5966
0.4	6.7263	3.8170	2.8673	2.4045	2.1314	1.9498	1.8185	1.7173	1.6356	1.5674
d 0.5	6.5865	3.7285	2.7986	2.3466	2.0806	1.9043	1.7773	1.6800	1.6020	1.5373
0.6	6.4015	3.6141	2.7118	2.2755	2.0201	1.8517	1.7311	1.6395	1.5668	1.5070
0.7	6.1426	3.4590	2.5985	2.1860	1.9466	1.7902	1.6793	1.5960	1.5307	1.4777
0.8	5.7467	3.2340	2.4425	2.0692	1.8557	1.7181	1.6219	1.5508	1.4957	1.4517
0.9	5.0319	2.8662	2.2102	1.9097	1.7415	1.6349	1.5615	1.5079	1.4670	1.4348

Taking $T = \begin{bmatrix} -0.4472 & 0.8944 \\ -0.8944 & -0.4472 \end{bmatrix}$, we have $B_2 = -1.1180$, and LMI (6.16) has feasible solutions with $\lambda = 0.5$ and $d = 0.5$:

$$V = 26.6564, P = 29.7355$$

$$X = 9.9225, X_d = 5.6653.$$

It follows from Theorem 6.4 that $C = VP^{-1} = 0.8964$, the linear sliding surface is $\delta(t) = \begin{bmatrix} 0.8964 & 1 \end{bmatrix} x(t)$.

From Theorem 6.9, the reaching control law can be taken as follows

$$\begin{aligned} u = \; & 0.8944[\Pi S(t) + [6.5997 \; -2.6522]]x(t) \\ & +[-0.6940 \; -0.7171]x(t-\tau(t)) + \\ & [0.5793 \; 1.1586]x(t)\hat{\theta} + [0.1159 \; 0.2317]x(t-\tau(t))\hat{\beta}] + u_N \end{aligned}$$

$$u_N = \begin{cases} -\frac{B_2^T S}{||B_2^T S||}\hat{\rho}, & \text{if } \hat{\rho}||B_2^T S|| > \epsilon \\ -\frac{B_2^T S}{\epsilon}\hat{\rho}^2, & \text{if } \hat{\rho}||B_2^T S|| \leq \epsilon \end{cases}. \tag{6.57}$$

The parameter Π and ϵ can be tuned to reduce the chattering on the sliding surface. Fig. 6.1 are simulation results with $\Pi = 1$ and ϵ=0.6. Obviously the sliding mode is asymptotically stable and the trajectories of the system tends to the origin in spite of time-delay and uncertainties.

The closed-loop dynamic responses are given in Fig. 6.1 to Fig. 6.4 with the initial conditions $x(0) = [-1, 1]^T$, $\theta(0) = 0.7$, $\beta(0) = 0.45$, $c(0) = 1, k(0) = 0.75$. Fig. 6.1 shows that the original system states approach to a small bounded region in finite time. Fig. 6.2 depicts the input control signal. Fig. 6.3 shows the sliding mode surface. The estimated parameters are shown in Fig. 6.4.

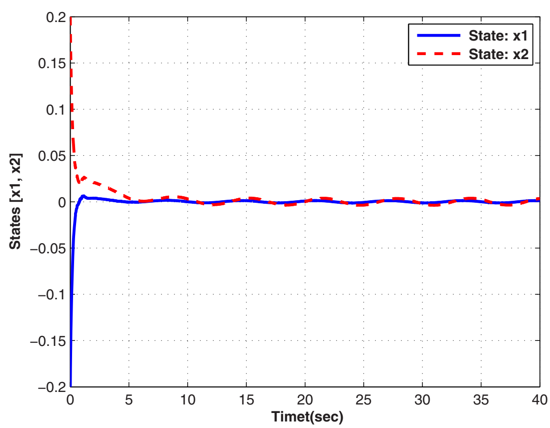

Fig. 6.1 The controlled state trajectories

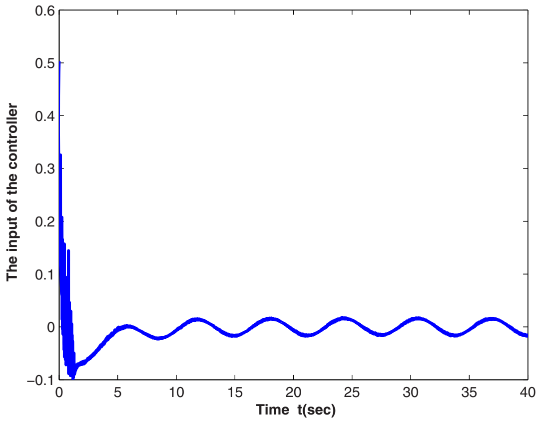

Fig. 6.2 The control signal

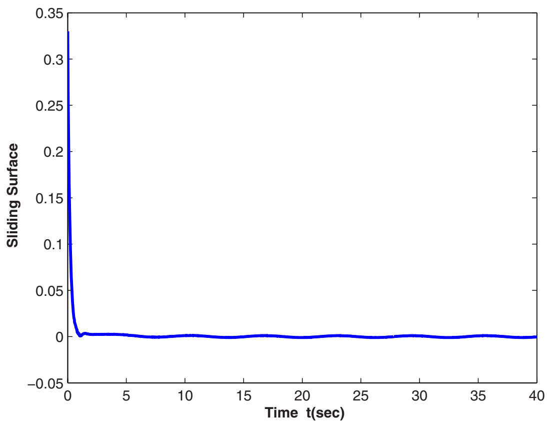

Fig. 6.3 The sliding mode surface

From the above figures, it can be seen that the algorithm proposed in this chapter works well. There is no chattering on the sliding surface and the control input is also very smooth. Parameters $\hat{\theta}$, $\hat{\beta}$, $\hat{c}$ and $\hat{k}$ converge to constants.

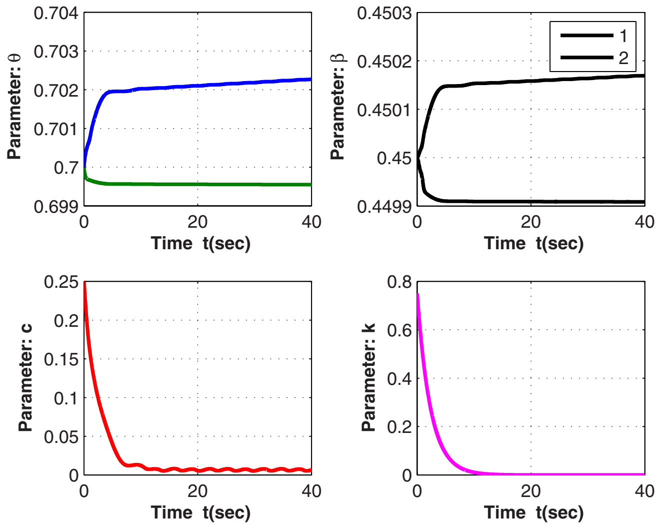

Fig. 6.4 The estimated parameters with initial conditions $q_1 = 0.75; q_2 = 0.31;$ $q_3 = 1.2, \epsilon_0 = 0.5; q_4 = 0.85, \epsilon_1 = 0.5$

6.5 Conclusion

In this chapter, the problem of robust adaptive sliding mode control for a class of uncertain time-delay systems has been considered in which no matching condition is assumed for the state uncertainties. The aim of this study is to attempt to combine the advantages of adaptive control and variable control methods. The resulting combined method makes the system uniformly ultimately bounded, robust in the presence of uncertainties and disturbance and in addition, has a fast transient response.

Chapter 7
SMC of Uncertain Linear Discrete Time Systems with Input Delay

Many papers have considered the problem of SMC for classes of uncertain systems with time-delay [161, 147, 189, 160, 10, 8, 9]. Although much attention has been given to the control problem of discrete systems with time-delay [96, 127, 168, 154, 51], few results have been reported on robust control of systems with time-delay in discrete time with discrete SMC.

A discrete version of SMC is important when the implementation of the control is realized digitally using a relatively slow sampling period. It is worth pointing out that discrete sliding mode control cannot be obtained from their continuous counterpart by means of simple equivalence. In [34], the problem of discrete Variable Structure Control (VSC) was first considered. The concept of the quasi-sliding mode was suggested in [134], and phenomena of switching, reaching, and quasi-sliding mode were investigated in [55].

In continuous time, to compensate for the time delay in the inputs, Smith predictor has been frequently used and is essential in solving the control problem for systems with input delay. Though there are many fundamental contributions made in control system design for systems in discrete time with input delay, there are few results that exploit the idea of applying a predictor in discrete time with an input delay that is in discrete time.

In this chapter, we consider the problem of designing both a linear sliding surface and reaching motion controller for a class of uncertain discrete systems with input delay. For linear systems with input delay, the main contributions of the chapter lie in:

(i) the introduction of a novel predictor in discrete time;
(ii) the design of predictor-based sliding surface $s(k) = 0$ and reaching control law $u(k)$ such that the motion of the closed-loop system satisfies the reaching condition; and
(iii) analysis of the stability of the quasi-sliding mode after it reaches the sliding surface.

This chapter is organized as follows. Section 7.1 shows the control problem and the definitions of quasi-sliding mode and reaching condition. Section 7.2

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 85–97.
springerlink.com

presents the predictor, its property and predictor-based controller without bounded disturbance. In this section, robust predictor-based controller and stability analysis of the quasi-sliding mode are also included. An example is given in Section 7.3 to show the effectiveness of the controller proposed. Section 7.4 presents the conclusion.

7.1 Problem Formulation

For convenience of analysis, we will first consider the linear systems with input delay in discrete time described by

$$\Sigma_1 : \begin{cases} x(k+1) = Ax(k) + Bu(k-\tau) \\ \quad u(k) = \psi(k), k = -\tau, -\tau+1, \cdots, 0 \end{cases} \tag{7.1}$$

where $x(k) \in R^n$ and $u(k) \in R^m$ are the system state and system input, respectively, A and B are matrices of appropriate dimensions and rank$(B) = m$, integer $\tau \geq 0$ denotes the amount of time delay, and $\psi(\cdot)$ denotes the initial condition.

Assumption 7.1.1 The pair, (A, B), is completely controllable.

As we do not need A to be a stable matrix, the control problem is not trivial because of the presence of the input delay. To solve the control problem for linear systems with input time delay, we need the introduction of a novel predictor which converts the control problem for systems with input time delay to a system without the difficulty of time delay under the assumption of known time delay. Then, the predictor based control is extended to uncertain linear systems with input time delay in discrete time as follows:

$$\Sigma_2 : \begin{cases} x(k+1) = Ax(k) + Bu(k-\tau) + d(x(k), x(k-\tau_0), u(k-\tau), k) \\ \quad x(k) = \varphi(k), k = -\tau_0, -\tau_0+1, \cdots, 0 \\ \quad u(k) = \psi(k), k = -\tau, -\tau+1, \cdots, 0 \end{cases} \tag{7.2}$$

where the unknown function $d(x(k), x(k-\tau_0), u(k-\tau), k)$ includes the extraneous disturbance, and non-linear perturbations with respect to the current state and delayed state, respectively, and $\varphi(\cdot)$ denotes the initial condition.

Assumption 7.1.2 There is a positive constant, c_d, such that

$$||d(x(k), x(k-\tau_0), u(k-\tau), k)|| \leq c_d. \tag{7.3}$$

For simplicity, we write $d(k)$ instead of $d(x(k), x(k-\tau_0), u(k-\tau), k)$.

Remark 7.1. The above assumption might restrict the class of systems that the approach can handle. But in reality, there are indeed some systems that fall into this category. For example, there maybe exist bounded disturbances.

It is known that the desired state trajectory of a discrete variable structure control (VSC) system should have the following properties [55]:

P1. Starting from any initial state, the trajectory will move monotonically toward the switching plane and cross it in finite time.
P2. Once the trajectory has crossed the switching plane the first time, it will cross the plane again in every successive sampling period, resulting in a zigzag motion about the switching plane.
P3. The size of each successive zigzagging step is non-increasing and the trajectory stays within a specified band.

The above properties can be extended to multiple input cases by applying the rules to the m entries of the switching function independently. The following definitions are recalled based on the above properties.

Definition 7.2. [55] The motion of a discrete VSC system satisfying properties P2 and P3 is called a quasi-sliding mode (QSM). The specified band which contains the QSM is called the quasi-sliding mode band (QSMB) and is defined by

$$\{x : |s_i(x)| < \delta_i, i = 1, \cdots, m\} \tag{7.4}$$

where $2\delta = 2[\delta_1, \cdots, \delta_m]^T \in R^m$ is the width of the band, $s(x) = 0$ is the sliding surface.

Definition 7.3. [55] The quasi-sliding mode becomes an ideal quasi-sliding mode when $\delta_i = 0, i = 1, \cdots, m$.

Definition 7.4. [55] Discrete VSC is said to satisfy a reaching condition if the resulting system possesses all the three properties: P1, P2, and P3.

7.2 Predictor-Based SMC

SMC design is usually broken down into two phases: (i) the construction of a stable sliding surface, and (ii) the design of the reaching law. However, for systems with input time delay, additional measures are needed. In this chapter, we would like to explore the idea of using a predictor to convert the original system with input time delay to a system where the input time delay does not appear explicitly so that control system design can be constructed with the new states, the states of the predictor.

7.2.1 Predictor in Discrete-Time

In continuous time, the Smith predictor plays a very important role. In this chapter, we would like to introduce a predictor in discrete time, which is essential to solving the control problem due to the input time delay as will be detailed later.

For simplicity, it is assumed that A is nonsingular. Let $\hat{x}(k)$ be the predicted states of system (7.1) and be defined as follows:

$$\hat{x}(k) = A^{\tau}x(k) + \sum_{i=-\tau+1}^{0} A^{-i}Bu(k-1+i) \tag{7.5}$$

where $\hat{x} \in R^n$.

Proposition 7.5. *The dynamics of the proposed predictor (7.5) for system (7.1) can be conveniently described by system matrices, A and B, as*

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k). \tag{7.6}$$

Proof. Noticing system (7.1), the predictor (7.5) can be written as

$$\begin{aligned}\hat{x}(k+1) &= A^{\tau}x(k+1) + \sum_{i=-\tau+1}^{0} A^{-i}Bu(k+i) \\ &= A^{\tau}(Ax(k) + Bu(k-\tau)) + \sum_{i=-\tau+1}^{0} A^{-i}Bu(k+i). \end{aligned} \tag{7.7}$$

Noting (7.5), we have

$$\begin{aligned}\hat{x}(k+1) &= A^{\tau}Ax(k) + A^{\tau}Bu(k-\tau) + \sum_{i=-\tau+1}^{0} A^{-i}Bu(k+i) \\ &= A^{\tau}Ax(k) + [\sum_{i=-\tau+1}^{-1} A^{-i}Bu(k+i) + A^{\tau}Bu(k-\tau)] + Bu(k) \\ &= AA^{\tau}x(k) + A\sum_{i=-\tau+1}^{0} A^{-i}Bu(k-1+i) + Bu(k) \\ &= A[A^{\tau}x(k) + \sum_{i=-\tau+1}^{0} A^{-i}Bu(k-1+i)] + Bu(k) \\ &= A\hat{x}(k) + Bu(k). \end{aligned} \tag{7.8}$$

Remark 7.6. Through the introduction of Predictor (7.5), the original system with input time delay (7.1) has been converted into (7.6) without the explicit appearance of time delay. As such, control system design can be carried out for the equivalent system without the explicit appearance of time delay, i.e., control system design is carried out in the new predicted states rather than the original states. The possible disadvantage of the approach is the need for a known time delay.

7.2.2 Predictor-Based SMC for System Σ_1

The sliding surface is defined as

$$s(k) = C\hat{x}(k) \tag{7.9}$$

where $s = [s_1, \cdots, s_m]^T \in R^m$ and $C = [C_1^T, \cdots, C_m^T]^T \in R^{m\times n}$.

Without losing generality, we assume that the matrix C is of full rank and the matrix CB is nonsingular.

After selecting the sliding surface, the next step is to choose the control law such that the condition is satisfied. This condition ensures that the control law will force system trajectories toward the sliding surface in finite time.

Noting that (7.6) and (7.9), $s(k+1)$ can be expressed as

$$s(k+1) = CA\hat{x}(k) + CBu(k). \tag{7.10}$$

The equivalent control $u_{eq}(k)$ can be obtained by letting

$$s(k+1) = 0 \tag{7.11}$$

which leads to the equivalent control $u_{eq}(k)$ to be defined by

$$u(k) = -(CB)^{-1}CA\hat{x}(k). \tag{7.12}$$

The ideal sliding motion is then described by

$$\hat{x}(k+1) = (I - B(CB)^{-1}C)A\hat{x}(k). \tag{7.13}$$

The above equation also describes the "equivalent system motion". In this chapter, let us consider the following reaching law

$$s(k+1) - s(k) = -wTs(k) - \epsilon T\mathrm{sgn}(s(k)) \tag{7.14}$$

where

$$\begin{aligned}
&\epsilon = \mathrm{diag}\left(\epsilon_1 \cdots \epsilon_m\right)\\
&w = \mathrm{diag}\left(w_1 \cdots w_m\right)\\
&\mathrm{sgn}(s(k)) = [\mathrm{sgn}(s_1(k)), \cdots, \mathrm{sgn}(s_m(k))]^T\\
&\epsilon_i > 0, w_i > 0, 1 - w_iT > 0, T \text{ is the sampling period.}
\end{aligned} \tag{7.15}$$

In [55], it has been shown that the reaching control law can guarantee the trajectory of the closed-loop system so that it is driven onto the sliding surface in finite time, and the chattering is reduced by tuning the parameters, ϵ_i, and w_i properly.

In view of (7.10), the incremental change of $s(k)$ is

$$\begin{aligned} s(k+1) - s(k) &= CA\hat{x}(k) + CBu(k) - C\hat{x}(k) \\ &= -wTs(k) - \epsilon T \mathrm{sgn}(s(k)). \end{aligned} \tag{7.16}$$

Solving for $u(k)$ gives the control law

$$u(k) = -(CB)^{-1}[C(A-I)\hat{x}(k) + \epsilon T \mathrm{sgn}(s(k)) + wTs(k)]. \tag{7.17}$$

When the discrete sliding mode controller is applied to the plant, the state response of the system can also be separated into the following three modes: the reaching, sliding, and steady-state modes. However, in all practical situations, switching seldom occurs on the sliding surface, it will cross the sliding surface in every successive sampling period after the system trajectory passing through the sliding surface. With the control law (7.17), in the steady state, the system trajectory will move within a band δ.

7.2.3 Robust Predictor Based SMC for System Σ_2

To investigate the SMC law of system (7.2), let $\hat{x}(k)$ be the predicted states of system (7.2). After some similar algebraic manipulations in the proof of Proposition 7.5, we have

$$\hat{x}(k+1) \ = A\hat{x}(k) + Bu(k) + A^{\tau} d(k). \tag{7.18}$$

Given the predictor-based sliding surface (7.9), then

$$s(k+1) \ = CA\hat{x}(k) + CBu(k) + CA^{\tau} d(k). \tag{7.19}$$

With the equivalent control (7.12), the ideal sliding motion in the presence of bounded disturbance is then described by

$$\hat{x}(k+1) = (I - B(CB)^{-1}C)[A\hat{x}(k) + A^{\tau} d(k)]. \tag{7.20}$$

Note that $A^{\tau} d(k)$ is not matched in general, i.e., $\mathrm{Im}(A^{\tau} d(k))$ is not necessarily included in $\mathrm{Im}(B)$, thus, the well-known "invariance" property of the ideal sliding motion does not exist. In the presence of bounded uncertainties $d(k)$, it is almost impossible to control the state to approach zero. Instead of controlling the state to zero, bounded motion about the sliding surface can be guaranteed with the reaching law (7.14).

Similar to (7.17), we might consider the following control

$$u(k) = -(CB)^{-1}[C(A-I)\hat{x}(k) + CA^{\tau}d(k) + \epsilon T\text{sgn}(s(k)) + wTs(k)] \quad (7.21)$$

provided that $CA^{\tau}d(k)$ is known. However, it is unknown. As such, we can use the maximum bound of $d(k)$ to suppress the uncertainty in robust control, i.e.,

$$u(k) = -(CB)^{-1}[C(A-I)\hat{x}(k) + \bar{d}(k) + \epsilon T\text{sgn}(s(k)) + wTs(k)] \quad (7.22)$$

where

$$\bar{d}(k) = c_d||CA^{\tau}||\text{sgn}(s(k)). \quad (7.23)$$

The choice of $\bar{d}(k)$ is done to ensure that the sign of the incremental $s(k)$ of (7.16) is opposite to the sign of $s(k)$.

Without loss of generality, it is assumed that $|A| \neq 0$, then we can now conclude our results of robust predictor-based SMC design in the following main theorem.

Theorem 7.7. *If the linear sliding surface is given by (7.9), system (7.2) can be driven onto the sliding surface by the control (7.22) in a finite time.*

Proof. Applying controller (7.22), then reaching condition developed in [55] and [5], we know that system (7.18) will be driven around the sliding surface by the control (7.22) in finite time. From equation (7.5), it is shown that $x(k) = A^{-\tau}[\hat{x}(k) - \sum_{i=-\tau+1}^{0} A^{-i}Bu(k-1+i)]$, then $x(k)$ will also be driven around the sliding surface.

7.2.4 Stability Analysis of Quasi-Sliding Motion

The switching law in a continuous-time sliding mode control can force system trajectories toward the sliding surface in finite time and maintain on the surface thereafter, despite the possible existence of any modelling errors and disturbances with known bound. The underlying motivation is given by the fact that the switching law can instantaneously react to any error such that it is canceled out directly. Physically, we have made the assumption that the slide mode control can switch at infinitely high frequency. This is obviously not possible in discrete-time. The switching function can only change its value at specific time instant dictated by the sampling time, the system will no longer stay on the sliding surface and no 'true' sliding mode will be possible. The concept "discrete quasi-sliding mode" is introduced [55], which means that the system state moves in a neighborhood around the sliding surface $s(k) = 0$ when k is sufficiently large. Therefore, we cannot analyze the stability of the dynamics of sliding mode strict to the sliding surface.

Thus, the stability of sliding motion has to be investigated using Lyapunov synthesis as will be detailed later.

To obtain a regular form of system (7.1), a nonsingular matrix U can be always chosen such that

$$UB = \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} \tag{7.24}$$

where $B_2 \in R^{m\times m}$ is nonsingular. For convenience, choose the state transformation $z(k) = U\hat{x}(k)$, with

$$U = \begin{bmatrix} {U_2}^T \\ {U_1}^T \end{bmatrix}$$

where $U_1 \in R^{n\times m}$ and $U_2 \in R^{n\times(n-m)}$ are two sub-blocks of a unitary matrix resulting from the singular value decomposition of B, i.e.,

$$B = \begin{bmatrix} U_1 \; U_2 \end{bmatrix} \begin{bmatrix} \Lambda \\ 0_{(n-m)\times m} \end{bmatrix} V^T \tag{7.25}$$

where $\Lambda \in R^{m\times m}$ is a diagonal positive-definite matrix and $V \in R^{m\times m}$ is a unitary matrix. With the state transformation $z(k) = U\hat{x}(k)$, system (7.2) can be rewritten as:

$$\begin{aligned} z_1(k+1) &= A_{11}z_1(k) + A_{12}z_2(k) + d_1(k) \\ z_2(k+1) &= A_{21}z_1(k) + A_{22}z_2(k) + B_2u(k) + d_2(k) \end{aligned} \tag{7.26}$$

where $z_1(k) \in R^{n-m}$, $z_2(k) \in R^m$, $A_{11} = U_2^T AU_2$, $A_{12} = U_2^T AU_1$, $A_{21} = U_1^T AU_2$, $A_{22} = U_1^T AU_1$, $d_1(k) = U_2^T A^\tau d(k) \in R^{(n-m)}$ and $d_2(k) = U_1^T A^\tau d(k) \in R^m$ are the sub-blocks of $UA^\tau d(k)$, the term $d_1(k)$ is called unmatched uncertainty since it is in the null space of B, the term $d_2(k)$ is called matched uncertainty since it in the range of B. The sliding surface, in this case, is given by

$$s(k) = C\hat{x}(k) = CU^{-1}U\hat{x}(k) = [C_1, C_2][z_1^T(k), z_2^T(k)]^T = C_1z_1(k) + C_2z_2(k). \tag{7.27}$$

Noting that (7.19) and control law proposed in (7.22), are respectively,

$$s(k+1) \; = CA\hat{x}(k) + CBu(k) + CA^\tau d(k) \tag{7.28}$$

and

$$u(k) = -(CB)^{-1}[C(A-I)\hat{x}(k) + \bar{d}(k) + \epsilon T\mathrm{sgn}(s(k)) + wTs(k)]$$

then

$$s(k+1) = s(k) - \epsilon T\mathrm{sgn}(s(k)) - wTs(k) - \bar{d}(k) + CA^\tau d(k) \tag{7.29}$$

Without loss of generality, we assume that $\det(C_2) \neq 0$. Using the following coordinate transformation

$$\begin{bmatrix} z_1(k) \\ s(k) \end{bmatrix} = \begin{bmatrix} I & 0 \\ C_1 & C_2 \end{bmatrix} \begin{bmatrix} z_1(k) \\ z_2(k) \end{bmatrix} \tag{7.30}$$

we transform system (7.26) into the following system after some algebraic manipulations

$$z_1(k+1) = \bar{A}_{11} z_1(k) + \bar{A}_{12} s(k) + d_1(k) \tag{7.31}$$
$$s(k+1) = s(k) - \epsilon T \text{sgn}(s(k)) - wTs(k) - \bar{d}(k) + CA^{\tau} d(k) \tag{7.32}$$

where

$$\bar{A}_{11} = A_{11} - A_{12} C_2^{-1} C_1, \bar{A}_{12} = A_{12} C_2^{-1}, \tag{7.33}$$

Now we study the stability of quasi-sliding mode around sliding surface.

To facilitate the presentation of the main results, the following lemma is introduced.

Lemma 7.8. [5] Assume that $\epsilon_i T$ is selected such that

$$\epsilon_i T > \frac{2c_d(1 - w_i T)}{w_i T}, i = 1, \cdots, m \tag{7.34}$$

then the properties P1, P2 and P3 are met and system (7.18) will converge in finite time to quasi-sliding mode band $\delta = [\delta_1, \cdots, \delta_m]^T \in R^m$ given by

$$\delta_i = \epsilon_i T + 2|\bar{d}(k)| = \epsilon_i T + 2c_d ||CA^{\tau}||, i = 1, \cdots, m. \tag{7.35}$$

Let $\bar{\delta} = \max_{i=1,\cdots,m}\{\delta_i\}$, $\bar{w} = \max_{i=1,\cdots,m}\{w_i\}$, $\bar{\epsilon} = \max_{i=1,\cdots,m}\{\epsilon_i\}$.

Based on the discrete version of the concept "uniformly ultimately bounded", that is, sliding motion enters a neighborhood of equilibrium in finite time and remains within it thereafter [28], we have the following result:

Theorem 7.9. *System (7.31) will be uniformly ultimately bounded with the control law (7.22) if ϵT is selected such that condition (7.34) is satisfied.*

Proof. Choose the Lyapunov function candidate as

$$V(k) = z_1^T(k) P z_1(k) + s^T(k) s(k) \tag{7.36}$$

where $0 < P = P^T \in R^{(n-m)\times(n-m)}$. The incremental change of Lyapunov function corresponding to system (7.31) is given by

$$\begin{aligned} & V(k+1) - V(k) \\ &= z_1^T(k+1) P z_1(k+1) + s^T(k+1) s(k+1) - z_1^T(k) P z_1(k) - s^T(k) s(k) \\ &= [\bar{A}_{11} z_1(k) + \bar{A}_{12} s(k) + d_1(k)]^T P [\bar{A}_{11} z_1(k) + \bar{A}_{12} s(k) + d_1(k)] \end{aligned}$$

$$
\begin{aligned}
&-z_1^T(k)Pz_1(k) + [s(k) - \epsilon T\text{sgn}(s(k)) - wTs(k) - \bar{d}(k) + CA^\tau d(k)]^T \\
&\times[s(k) - \epsilon T\text{sgn}(s(k)) - wTs(k) - \bar{d}(k) + CA^\tau d(k)] - s^T(k)s(k).
\end{aligned}
$$

Note that from (7.31) and (7.32), we have

$$
\begin{aligned}
&V(k+1) - V(k) \\
&= z_1^T(k)(\bar{A}_{11}^T P\bar{A}_{11} - P)z_1(k) + s^T(k)(\bar{A}_{12}^T P\bar{A}_{12} - 2wTI + T^2w^2I)s(k) \\
&\quad +s^T(k)[2\bar{A}_{12}^T P\bar{A}_{11}z_1(k) + 2\bar{A}_{12}^T Pd_1(k) - 2\epsilon T(1-wT)\text{sgn}(s(k)) \\
&\quad +2(1-wT)(-\bar{d}(k) + CA^\tau d(k))] + d_1^T(k)Pd_1(k) + 2d_1^T(k)P\bar{A}_{11}z_1(k) \\
&\quad +T^2\text{sgn}^T(s(k))\epsilon^2\text{sgn}(s(k)) + [-\bar{d}(k) + CA^\tau d(k)]^T[-\bar{d}(k) + CA^\tau d(k)].
\end{aligned}
$$

Note that (A, B) is completely controllable, then, there exits a matrix C such that $\bar{A}_{11}$ in (7.33) is stable [176]. Therefore, there exists a positive definite matrix P such that the following equation holds for any positive definite matrix Q

$$
\bar{A}_{12}^T P\bar{A}_{12} - P = -Q. \tag{7.37}
$$

By Lemma 7.8, we have

$$
\begin{aligned}
&V(k+1) - V(k) \\
&\leq -\sigma_{\min}(Q)||z_1(k)||^2 + 2(\bar{\delta}||\bar{A}_{12}^T P\bar{A}_{11}|| + c_d||\bar{A}_{11}^T PU_2^T A^\tau||)||z_1(k)|| \\
&\quad +||\bar{A}_{12}^T P\bar{A}_{12} - 2wTI + T^2w^2I||\bar{\delta}^2 + 2c_d||\bar{A}_{12}^T PU_2^T A^\tau||\bar{\delta} - 2m\epsilon T(1-wT)\bar{\delta} \\
&\quad -2(1-wT)c_d||CA^\tau||\bar{\delta} + 2c_d||(1-wT)CA^\tau||\bar{\delta} + c_d^2||(A^\tau)^T U_2U_2^T A^\tau|| \\
&\quad +mT^2\bar{\epsilon}^2 + c_d^2||CA^T||^2 + 2c_d^2||CA^\tau||^2 + c_d^2||(A^\tau)^T C^T CA^\tau||.
\end{aligned} \tag{7.38}
$$

For simplicity, let

$$
\begin{aligned}
\bar{\delta}_1 &= \bar{\delta}||\bar{A}_{12}^T P\bar{A}_{11}|| + c_d||\bar{A}_{11}^T PU_2^T A^\tau|| \\
\bar{\delta}_2 &= ||\bar{A}_{12}^T P\bar{A}_{12} - 2wTI + T^2w^2I||\bar{\delta}^2 + 2c_d||\bar{A}_{12}^T PU_2^T A^\tau||\bar{\delta} - 2m\epsilon T(1-wT)\bar{\delta} \\
&\quad -2(1-wT)c_d||CA^\tau||\bar{\delta} + 2c_d||(1-wT)CA^\tau||\bar{\delta} + c_d^2||(A^\tau)^T U_2U_2^T A^\tau||
\end{aligned}
$$

then

$$
V(k+1) - V(k) \leq -\sigma_{\min}(Q)(||z_1(k)|| - \frac{\bar{\delta}_1}{\sigma_{\min}(Q)})^2 + \frac{\bar{\delta}_1^2}{\sigma_{\min}(Q)} + \bar{\delta}_2.
$$

Therefore,

$$
V(k+1) - V(k) < 0, \quad \text{if}||z_1(k)|| > \frac{\bar{\delta}_1 + \sqrt{\bar{\delta}_1^2 + \sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)}. \tag{7.39}
$$

The above means that the trajectory of $z_1(k)$ will enter into the ball $\Omega(0, \frac{\bar{\delta}_1 + \sqrt{\bar{\delta}_1^2 + \sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)})$ with center at the origin and radius $\frac{\bar{\delta}_1 + \sqrt{\bar{\delta}_1^2 + \sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)}$, i.e.,

$$\{\Omega(0, \frac{\bar{\delta}_1 + \sqrt{\bar{\delta}_1^2 + \sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)}) : ||z_1(k)|| \leq \frac{\bar{\delta}_1 + \sqrt{\bar{\delta}_1^2 + \sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)}\}. \quad (7.40)$$

On the other hand, the trajectory of $z_1(k)$ converges in finite time to quasi-sliding mode band δ. Therefore, the quasi-sliding motion will remain in the domain of the intersection of the ball $\Omega(0, \frac{\bar{\delta}_1+\sqrt{\bar{\delta}_1^2+\sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)})$ and quasi-sliding surface with band δ for all subsequent time. The uniform ultimate boundedness thus follows using the discrete version of the result and terminology in [28].

Remark 7.10. The size of radius of the ball $\Omega(0, \frac{\bar{\delta}_1+\sqrt{\bar{\delta}_1^2+\sigma_{\min}(Q)\bar{\delta}_2}}{\sigma_{\min}(Q)})$ depends on the bound of uncertainty, i.e., the constant c_d, the size of radius of the ball will be small as the bound c_d is. At the same time, the ϵ, sampling time T, w as well as matrix Q can be tuned to reduce the size of radius of the ball. From equation (7.5), it is shown that $x(k) = A^{-\tau}[\hat{x}(k) - \sum_{i=-\tau+1}^{0} A^{-i}Bu(k-1+i)]$, then $x(k)$ is also uniformly ultimately bounded.

7.3 Numerical Example

In this section, an illustrative example is given to verify the effectiveness of the design method developed in this chapter.

Example 7.11. Consider system (7.1) with input-delay

$$A = \begin{bmatrix} 1.2 & 0.1 \\ 0 & 0.6 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
$$d(x(k), x(k-\tau_0), k) = 0.004sin(k(x(k) + x(k-\tau_0))), \tau_0 = 5, \tau = 5$$
$$x(k) = 0, k = -\tau_0, -\tau_0 + 1, \cdots, 0; u(k) = 0, k = -\tau, -\tau + 1, \cdots, 0.$$

It is obvious that the open loop of system (7.1) is unstable. When $\tau = 0, d(x(k), x(k-\tau_0), k) = 0$, it is the same as the example used in [55, 5]. Choose $T = 0.01, \epsilon = 10, \bar{q} = 25$. The sliding surface is taken as

$$s(x) = C[A^{\tau}x + \sum_{i=-\tau+1}^{0} A^{-i}Bu(k-1+i)] \quad (7.41)$$

where $C = [5\ 1]$. Note that the upper bound of $d(x(k), x(k-\tau_0), k)$ is $c_d = 0.004$, the control law is obtained as

$$u(k) = -C(A-I)\hat{x} - c_d\text{sgn}(s)||CA^{\tau}|| - 0.5\text{sgn}(s) - 0.25s.$$

Observe that $c_d||CA^{\tau}|| = 0.024$, inequality (7.34) is satisfied, we have the following simulation results:

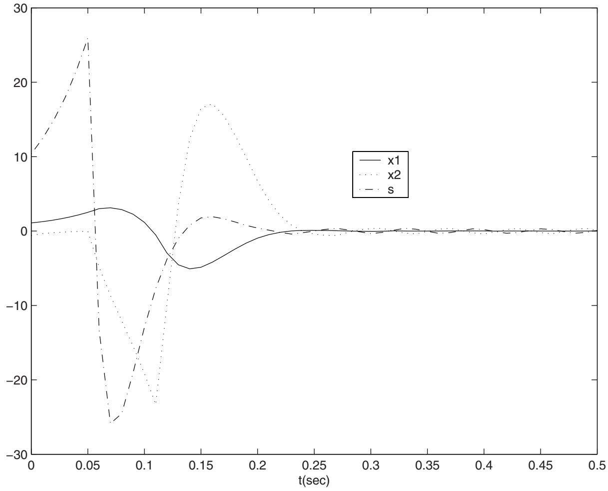

Fig. 7.1 States x_1, x_2 and sliding surface s

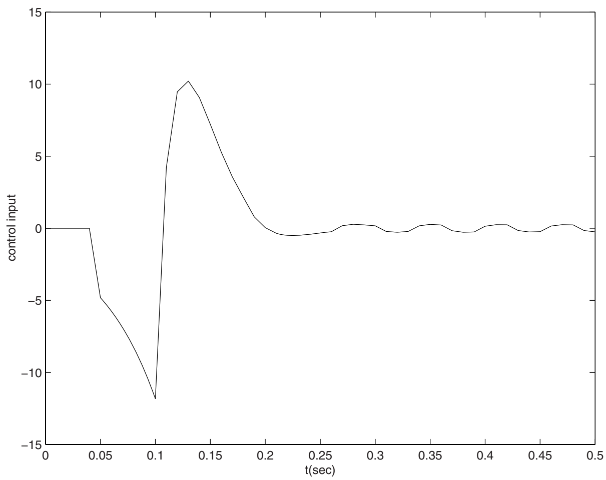

Fig. 7.2 Control input u

From Fig. 7.1 and Fig. 7.2, it is shown that the system is open loop before 0.05 second because $\tau = 5$. During this period, s diverges, so do x_1 and x_2, and they will be in steady state after the control is implemented.

7.4 Conclusion

In this chapter, after the introduction of a new predictor, the original system with input delay was converted to an equivalent system without the explicit appearance of time delay. Based on predictor-based sliding surface, a new quasi-SMC has been proposed. The reaching control law has been designed to ensure the existence of the quasi-sliding mode. The simulation results have demonstrated the effectiveness of the developed control design techniques for systems with input-delay and bounded uncertainties.

Chapter 8
SMC for Linear Systems with Input and State Delays

8.1 Introduction

Mainly due to these advantages variable structure control theory has found applications in various kinds of plants (see, e.g. [86], [175], [33], [85]. Other SMC schemes are proposed for linear systems with state delay [161], [104], [100], and system with input delay [147]. Their methods have the same benefits, but cannot be applied to systems with both state and input delay. In a recent paper [63], the authors propose the design methods of a SMC for systems with both input and state delay. However, the results of [63] involve the tuning and factorizing of a symmetric positive-definite matrix. Because no tuning procedure for such matrix is available, this makes the use of these methods somehow difficult, especially when one wants to find the largest possible bound for the time delay which ensures stabilization.

In this chapter, the problem of SMC for systems with both input and state delay is addressed. Two different design methods are proposed for the sliding mode control of these systems. Delay-independent sufficient conditions as well as delay-dependent ones are given for the existence of sliding manifolds in terms of LMIs. A reaching motion controller is also proposed for such systems by means of both the reaching law and the inequality approach. The proposed methods have the advantage that can be implemented numerically very efficiently using standard LMI algorithms and that the problem of finding the upper bound of time delay which ensures stabilization can be easily solved. Moreover, they can be easily extended to uncertain linear systems as well as to the case of multiple time delays.

The chapter is organized as follows. Section 8.2 gives the problem formulation and some preliminaries. In section 8.3, the problem of compensator-based SMC is considered, and both delay-independent and delay-dependent results on designing sliding surface are presented, and then, the reaching motion controller is established. In section 8.4, SMC for linear systems with time-varying input and state delays is discussed. In section 8.5, compensator-based SMC

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 99–118.
springerlink.com

for systems with time-varying delays is presented. Numerical simulations are presented in Section 8.6 and some conclusion remarks are given in Section 8.7.

8.2 Problem Formulation

Consider the following time-delay system described by

$$\begin{aligned}\dot{x}(t) &= Ax(t) + A_d x(t-\tau) + B_1 u(t) + B_2 u(t-\tau)\\ x(t) &= \varphi(t), t \in \left[-\tau, 0\right]\\ u(t) &= \psi(t), t \in \left[-\tau, 0\right]\end{aligned} \tag{8.1}$$

where $x(t) \in R^n$ is the state, $u(t) \in R^m$ is the control. $A,\ A_d,\ B_1$ and B_2 are with appropriate dimensions. The following assumption is needed.

Assumption 8.2.1
A1 rank$(B_1 + B_2) = m$;
A2 $(A + A_d, B_1 + B_2)$ *is controllable.*

The goal of this chapter is to address the following problems:

- How to design sliding mode and reaching law for system (8.1)?
- How long can the time delay be for the system (8.1) to be stabilized with the designed sliding mode and reaching law?

8.3 Compensator-Based SMC for Systems with Time Delay

The switching manifolds are taken as following

$$S(t) = Cx(t) + \int_{t-\tau}^{t} CA_d x(\xi)d\xi + \int_{t-\tau}^{t} CB_2 u(\xi)d\xi + \Pi(t) \tag{8.2}$$

where $C \in R^{m\times n}$ is a constant matrix satisfying the condition: $C(B_1 + B_2)$ is invertible, according to A1, there always exists such matrix satisfying this condition. $\Pi(t)$ is a sliding mode compensator and taken as

$$\dot{\Pi}(t) = -C[(A + A_d) - (B_1 + B_2)K]x(t) \tag{8.3}$$

where $K \in R^{m\times n}$ is a constant matrix to be found.

After selecting the sliding manifolds, the next step is to choose a control law such that it satisfies the condition for the existence of the sliding mode;

$S^T \dot{S} < 0$. This condition ensures that the control law will force system trajectories toward the sliding manifolds in finite time and maintain them on the manifolds after then. The following control structure of the form is considered:

$$u(t) = u_{eq} + u_N \tag{8.4}$$

where u_{eq} is an equivalent control for the system (8.1), and u_N is a switching control. The equivalent control law u_{eq} is derived by $\dot{S} = 0$ for system (8.1). Differentiating S with respect to time gives

$$\dot{S} = C(A + A_d)x(t) + C(B_1 + B_2)u(t) + \dot{\Pi}(t). \tag{8.5}$$

It follows (8.3) that

$$\dot{S} = C(B_1 + B_2)Kx(t) + C(B_1 + B_2)u(t). \tag{8.6}$$

Then, the equivalent control is obtained by

$$u_{eq} = -Kx(t). \tag{8.7}$$

Now, in order to force the system trajectories toward the designed sliding manifolds, the switching control u_N is chosen by

$$\dot{S} = -gS - \epsilon \text{sgn}(S) \tag{8.8}$$

where g and ϵ are constant, $\text{sgn}(S(z)) = [\text{sgn}(s_1(z)), \cdots, \text{sgn}(s_l(z))]^T$. Then,

$$u_N = -[C(B_1 + B_2)]^{-1}[gS + \epsilon \text{sgn}(S)]. \tag{8.9}$$

Thus, the following control law is obtained

$$u(t) = -Kx(t) - [C(B_1 + B_2)]^{-1}[gS + \epsilon \text{sgn}(S)]. \tag{8.10}$$

After the system trajectories reaching the sliding manifolds, the sliding motion on the sliding manifolds is

$$\dot{x} = (A - B_1K)x(t) + (A_d - B_2K)x(t - \tau). \tag{8.11}$$

Writing $\overline{A} = A - B_1K$, $\overline{A}_d = A_d - B_2K$, then (8.11) can be written as

$$\dot{x} = \overline{A}x(t) + \overline{A}_d x(t - \tau). \tag{8.12}$$

From the above, the design of a SMC is possible if (i) there exists $K \in R^{m \times n}$ which guarantees stability of (8.12), and (ii) there exists control law which makes the sliding function asymptotically stable for a specified sliding manifolds. The remaind of this section is devoted to the design of a sliding manifolds and control law to satisfy these requirements.

To motivate the technique used in this chapter, the following lemma is needed to derive main results of this chapter.

Lemma 8.1. [106] The sliding motion (8.12) is stable for all $\tau \geq 0$ if there exist symmetric positive-definite matrices P and Q which satisfy the following inequality:

$$\overline{A}^T P + P\overline{A} + P\overline{A}_d Q^{-1}\overline{A}_d^T P + Q < 0. \tag{8.13}$$

Theorem 8.2. (delay-independent) The sliding motion (8.12) is stable for all $\tau > 0$ with $K = LY^{-1}$ if there exist symmetric positive-definite matrices $Y \in R^{n\times n}$, $J \in R^{n\times n}$ and matrix $L \in R^{m\times n}$ such that the following LMI holds

$$\begin{bmatrix} \Gamma_1 & A_d Y - B_2 L \\ Y^T A_d^T - L^T B_2^T & -J \end{bmatrix} < 0 \tag{8.14}$$

where $\Gamma_1 = AY + YA^T - B_1 L - L^T B_1^T + J$.

Proof. Assuming (8.14) holds. Pre-and post-multiplying both sides of (8.14) by $\mathrm{diag}(Y^{-1}, Y^{-1})$ and taking $K = LY^{-1}$ yields

$$\begin{bmatrix} \Gamma_2 & Y^{-1}(A_d - B_2 K) \\ (A_d - B_2 K)^T Y^{-1} & -Y^{-1}JY^{-1} \end{bmatrix} < 0 \tag{8.15}$$

where $\Gamma_2 = Y^{-1}(A - B_1 K) + (A - B_1 K)^T Y^{-1} + Y^{-1}JY^{-1}$.

Taking $P = Y^{-1}$ and $Q = Y^{-1}JY^{-1}$, using Schur complement formula, (8.15) is equivalent to

$$\overline{A}^T P + P\overline{A} + P\overline{A}_d Q^{-1}\overline{A}_d^T P + Q < 0. \tag{8.16}$$

From Lemma 8.1, the sliding motion (8.12) is stable for all $\tau > 0$. The proof is completed.

The above results are independent of the length of the time delay, i.e. the time delay is allowed to be arbitrarily large, and thus they cannot handle systems whose stability depends on the size of the time-delay. Thus, in general, they are conservative, especially when practically existing time-delays are small. The following results give delay-dependent criteria.

Theorem 8.3. (delay-dependent) The sliding motion (8.12) is stable for any constant time delay τ satisfying $0 \leq \tau \leq \overline{\tau}$ if there exist symmetric positive-definite matrices $P \in R^{n\times n}$, $W \in R^{n\times n}$, $Q \in R^{n\times n}$ and matrix $V \in R^{m\times n}$ such that the following LMIs hold

$$\begin{bmatrix} \Gamma_3 & \Gamma_4^T & Q \\ \Gamma_4 & -\overline{\tau}^{-1}W & 0 \\ Q & 0 & -\overline{\tau}^{-1}Q \end{bmatrix} < 0 \tag{8.17}$$

$$\begin{bmatrix} -2P+W & PA_d^T - V^T B_2^T \\ A_d P - B_2 V & -Q \end{bmatrix} < 0 \tag{8.18}$$

where $\Gamma_3 = (A+A_d)P + P(A+A_d)^T - (B_1+B_2)V - V^T(B_1+B_2)^T$, $\Gamma_4 = P(A+A_d)^T - V^T(B_1+B_2)^T$.

Moreover, a stabilizing equivalent control law is given by $u_{eq}(t) = VP^{-1}x(t)$.

Proof. Since the solutions of $\det|sI - \overline{A} - \overline{A}_d e^{-\tau s}| = 0$ are the same as those of $\det|sI - \overline{A}^T - \overline{A}_d^T e^{-\tau s}| = 0$, the following system is studied instead of system (8.12)

$$\dot{y} = \overline{A}^T y(t) + \overline{A}_d^T y(t-\tau). \tag{8.19}$$

Writing the system (8.19) in the form:

$$\begin{aligned} z(t) &= y(t) + \textstyle\int_{t-\tau}^{t} \overline{A}_d^T y(w)dw \\ \dot{z}(t) &= (\overline{A} + \overline{A}_d)^T y(t). \end{aligned} \tag{8.20}$$

Next, consider the Lyapunov-Krasovskii functional:

$$\begin{aligned} V(t) &= V_1(t) + V_2(t) \\ V_1(t) &= z^T(t)Pz(t) \\ V_2(t) &= \textstyle\int_{t-\tau}^{t} \int_{w}^{t} y^T(v)Qy(v)dvdw \end{aligned} \tag{8.21}$$

where P and Q are symmetric positive-definite matrices. Then, differentiating the functional along the solutions of (8.19) yields:

$$\begin{aligned} \dot{V}(t) &= \dot{V}_1(t) + \dot{V}_2(t) \\ \dot{V}_1(t) &= y^T(t)(\overline{A} + \overline{A}_d)Pz(t) + z^T(t)P(\overline{A} + \overline{A}_d)^T y(t) \\ &= y^T(t)[(\overline{A} + \overline{A}_d)P + P(\overline{A} + \overline{A}_d)^T]y(t) + W(t) \\ \dot{V}_2(t) &= -\int_{t-\tau}^{t} y^T(w)Qy(w)dw + \int_{t-\tau}^{t} y^T(t)Qy(t)dw \end{aligned}$$

where

$$W(t) = 2\int_{t-\tau}^{t} y^T(t)(\overline{A} + \overline{A}_d)P\overline{A}_d^T y(w)dw.$$

Recalling that for any vectors u, v and any matrix $Q > 0$ of appropriate dimensions

$$2u^T v \le u^T Qu + v^T Q^{-1} v$$

then, for any matrix $Q > 0$

$$W(t) \le \int_{t-\tau}^{t} y^T(t)(\overline{A} + \overline{A}_d)P\overline{A}_d^T Q^{-1} \overline{A}_d P$$

$$\times(\overline{A}+\overline{A}_d)^T y(t)dw + \int_{t-\tau}^{t} y^T(w)Qy(w)dw.$$

Hence, it follows that

$$\begin{aligned}\dot{V}(t) \ &\leq y^T(t)[(\overline{A}+\overline{A}_d)P + P(\overline{A}+\overline{A}_d)^T + \tau Q \\ &+\tau(\overline{A}+\overline{A}_d)P\overline{A}_d^T Q^{-1}\overline{A}_d P(\overline{A}+\overline{A}_d)^T]y(t). \qquad (8.22)\end{aligned}$$

The matrix in (8.22) is negative definite if the following inequality holds:

$$\begin{aligned}&(\overline{A}+\overline{A}_d)P + P(\overline{A}+\overline{A}_d)^T + \tau Q \\ &+\tau(\overline{A}+\overline{A}_d)P\overline{A}_d^T Q^{-1}\overline{A}_d P(\overline{A}+\overline{A}_d)^T < 0. \qquad (8.23)\end{aligned}$$

In order to turn (8.23) into LMI expression, (8.23) is transformed into

$$\begin{aligned}&(\overline{A}+\overline{A}_d)P + P(\overline{A}+\overline{A}_d)^T + \tau Q + \tau(\overline{A}+\overline{A}_d) \\ &\times PW^{-1}P(\overline{A}+\overline{A}_d)^T - \tau(\overline{A}+\overline{A}_d)P \\ &\times(W^{-1} - \overline{A}_d^T Q^{-1}\overline{A}_d)P(\overline{A}+\overline{A}_d)^T < 0 \qquad (8.24)\end{aligned}$$

where W is a symmetric positive-definite matrix which satisfies

$$W^{-1} > \overline{A}_d^T Q^{-1}\overline{A}_d. \qquad (8.25)$$

Thus, (8.23) is negative definite if (8.25) and the following inequality is satisfied.

$$\begin{aligned}&(\overline{A}+\overline{A}_d)P + P(\overline{A}+\overline{A}_d)^T + \tau Q \\ &+\tau(\overline{A}+\overline{A}_d)PW^{-1}P(\overline{A}+\overline{A}_d)^T < 0. \qquad (8.26)\end{aligned}$$

By using the Schur complement argument (8.25) and (8.26) imply that

$$\begin{bmatrix} -W^{-1} & \overline{A}_d^T \\ \overline{A}_d & -Q \end{bmatrix} < 0 \qquad (8.27)$$

and

$$\begin{bmatrix} \Gamma_5 & (\overline{A}+\overline{A}_d)P & Q \\ P(\overline{A}+\overline{A}_d)^T & -\tau^{-1}W & 0 \\ Q & 0 & -\tau^{-1}Q \end{bmatrix} < 0 \qquad (8.28)$$

where $\Gamma_5 = (\overline{A}+\overline{A}_d)P + P(\overline{A}+\overline{A}_d)^T$.

Pre- and post-multiplying (8.27) by $\mathrm{diag}(P, I_n)$, yields

$$\begin{bmatrix} -PW^{-1}P & P\overline{A}_d^T \\ \overline{A}_d P & -Q \end{bmatrix} < 0. \qquad (8.29)$$

From the equality

$$PW^{-1}P - 2P + W = (P - W)W^{-1}(P - W) \geq 0$$

it can be shown that

$$-2P + W \geq -PW^{-1}P.$$

Thus, (8.29) will hold if the following inequality is satisfied for some symmetric positive-definite matrices P, W and Q

$$\begin{bmatrix} -2P + W & P\overline{A}_d^T \\ \overline{A}_d P & -Q \end{bmatrix} < 0. \tag{8.30}$$

Taking $K = VP^{-1}$, then (8.17) and (8.18) will imply (8.28) and (8.30) hold for any constant time delay τ satisfying $0 \leq \tau \leq \overline{\tau}$, respectively. Thus, the LMIs (8.17)-(8.18) will guarantee the negativeness of $\dot{V}(t)$ for any non-zero $y(t) \in R^n$, which immediately implies the stability of the sliding motion (8.12). The proof is completed.

In the following, another technique will be investigated to solve this problem and find some upper bound on the delay up to which the system is stabilized by the proposed control law.

Adding m integrators (see, e.g. [132], [63]), system (8.1) is written as:

$$\begin{aligned} \dot{y}_1(t) &= Ay_1(t) + A_d y_1(t-\tau) + B_1 y_2(t) + B_2 y_2(t-\tau) \\ \dot{y}_2(t) &= \tilde{u}, \quad \tilde{u} \in R^m. \end{aligned} \tag{8.31}$$

Consider the sliding surface

$$s(t) = y_2(t) + Ky_1(t) \tag{8.32}$$

where $K \in R^{m\times n}$.

After the system trajectories reaching the sliding surface, the sliding motion on the sliding surface is

$$\dot{y}_1 = (A - B_1K)y_1 + (A_d - B_2K)y_1(t-\tau). \tag{8.33}$$

Then the conclusion obtained in Theorem 8.3 can be applied to this sliding motion.

Remark 8.4. The stability of the equation $y(t) + \int_{t-\tau}^{t} A_d y(w)dw = 0$ is equivalent to the stability of the corresponding characteristic equation, that is all the solutions λ of the associated characteristic equation

$$\det(I_n + A_d \int_{-\tau}^{0} e^{sw}dw) = 0, \quad s \in C \tag{8.34}$$

have negative real part. One constraint should be added to guarantee the stability of system (8.19). It is omitted here, for more details, see [188].

Remark 8.5. Theorem 8.3 provides delay-dependent condition for stabilization of sliding motion (8.33) in terms of the solvability of linear matrix inequalities. This is in contrast with the results of [60], [63] which are based on the solution of a Lyapunov equation. We note that the result of Gouaisbant *et al.* (1999) involves the tuning and factorizing of a 2×2 symmetric positive-definite matrix. Because no tuning procedure for such matrix is available, this makes the use of these methods somehow difficult, especially when one wants to find the largest possible bound for the time delay which ensures stabilization.

The stabilization criteria of Theorem 8.3 have the advantages that they do not require any parameter tuning and can be tested numerically very efficiently using standard LMI techniques: see, e.g. [15]. Another advantage is that the problem of finding the largest $\overline{\tau}$ which ensures stabilization using the methods of Theorem 8.3 can be easily solved. For example, Let $\nu = \frac{1}{\overline{\tau}}$, then, using gevp of LMI toolbox to solve the generalized eigenvalue problem (GEVP) which can be solved numerically very efficiently using LMI algorithms [15].

$$\min_{P>0, W>0, Q>0, W_1>0, W_2>0,} \nu \tag{8.35}$$

subject to (8.18) and

$$\begin{bmatrix} \Gamma_3 & \Gamma_4 & Q \\ \Gamma_4^T & -W_1 & 0 \\ Q & 0 & -W_2 \end{bmatrix} < 0 \tag{8.36}$$

$$\begin{bmatrix} W_1 & 0 \\ 0 & W_2 \end{bmatrix} < \nu \begin{bmatrix} W & 0 \\ 0 & Q \end{bmatrix} \tag{8.37}$$

where Γ_3, Γ_4 are defined as in Theorem 8.3, W_1 W_2 are symmetric positive-definite matrices.

Now, with $K = VP^{-1}$ obtained in Theorem 8.3, the following control law is designed.

Theorem 8.6. *Given $K = VP^{-1}$ obtained in Theorem 8.3, the trajectory of the closed-loop system (8.31) can be driven onto the sliding surface in limited time with the following control law:*

$$\tilde{u}(t) = \begin{cases} 0 & (||s(t)|| = 0) \\ -[K(Ay_1(t) + B_1 y_2(t)) & \\ +\beta s + N(t)\mathrm{sgn}(s)] & (||s(t)|| > 0) \end{cases} \tag{8.38}$$

where $q_1 > 1$, $q_2 > 1$, $\mathrm{sgn}(s) = [\mathrm{sgn}(s_1), \cdots, \mathrm{sgn}(s_m)]^T$, $N(t) = \mathrm{diag}(n_1, \cdots, n_m)$, $i = 1, \cdots, m$, for n_i defined as

$$n_i = \sum_{j=1}^{n} q_1 |(KA_d)_{ij}||y_{1i}(t)| + \sum_{k=1}^{n} q_2 |(KB_2)_{ik}||y_{2i}(t)|. \tag{8.39}$$

Proof. Applying the control law (8.38) to system (8.31). Consider the following Lyapunov-Krasovskii functional:

$$V = \frac{1}{2} s^T(t) s(t). \tag{8.40}$$

Differentiating $V(t)$ along the solutions of (8.31) with the control law (8.38) is given by:

$$\begin{aligned} \dot{V} &= s^T(y(t))(K(Ay_1(t) + A_d y_1(t-\tau) \\ &\quad + B_1 y_2(t) + B_2 y_2(t-\tau)) + \overline{u}). \end{aligned} \tag{8.41}$$

From the Razumikhin Theorem (Hale and Lunel, 1993),

$$|y_{1i}(t+\theta)| \le q_1 |y_{1i}(t)|, \quad q_1 > 1, -\overline{\tau} \le \theta \le 0 \tag{8.42}$$
$$|y_{2i}(t+\theta)| \le q_2 |y_{2i}(t)|, \quad q_2 > 1, -\overline{\tau} \le \theta \le 0 \tag{8.43}$$

for $i = 1, \cdots, n$.

We have

$$\begin{aligned} s^T K A_d y_1(t-\tau) &= \sum_{i=1}^{m} s_i \sum_{j=1}^{n} (KA_d)_{ij} y_{1i}(t-\tau) \\ &\le \sum_{i=1}^{m} |s_i| \sum_{j=1}^{n} q_1 |(KA_d)_{ij}||y_{1i}(t)| \end{aligned} \tag{8.44}$$

and

$$\begin{aligned} s^T K B_2 y_2(t-\tau) &= \sum_{i=1}^{m} s_i \sum_{k=1}^{m} (KB_2)_{ij} y_{2i}(t-\tau) \\ &\le \sum_{i=1}^{m} |s_i| \sum_{j=1}^{m} q_2 |(KB_2)_{ij}||y_{2i}(t)|. \end{aligned} \tag{8.45}$$

It can be shown from above inequalities that $\dot{V} < -\beta ||s||^2$. Thus, the state trajectories will reach the surface in a finite time with the control (8.38).

8.4 SMC for Linear Systems with Time-varying Input and State Delays

In this section, the time-delays in the state and input are time-varying and different. Consider the following time-delay system described by

$$\dot{x}(t) = Ax(t) + A_d x(t - \tau(t)) + Bu(t) + B_d u(t - d(t)) \tag{8.46}$$

$$x(t) = \varphi(t), \ t \in [-\tau_M, 0] \tag{8.47}$$

$$u(t) = \psi(t), \ t \in [-d_M, 0] \tag{8.48}$$

where $x(t) \in R^n$ is system state, $u(t) \in R^m$ is control input, $\tau(t)$ and $d(t)$ denote state delay and input delay, respectively. Furthermore, τ_M and d_M are two constant delays such that $\tau(t) \leq \tau_M$ and $d(t) \leq d_M$, respectively. For state delay $\tau(t)$, there also exists $\dot{\tau}(t) \leq \rho < 1$. System parameters $A, \ A_d, \ B$ and B_d are constant matrices with appropriate dimensions.

The following assumptions are needed.

A1 $\text{rank}(B + B_d) = m$;
A2 $(A + A_d, \ B + B_d)$ is controllable.

8.5 Compensator-Based SMC for Systems with Time-varying Delays

The switching manifolds are taken as following

$$S(t) = Cx(t) + \int_{t-\tau_0}^{t} CA_d x(\xi) d\xi + \int_{t-d_0}^{t} CB_d u(\xi) d\xi + \Pi(t) \tag{8.49}$$

where $\tau_0 = \frac{\tau_M}{2}$, $d_0 = \frac{d_M}{2}$ and $C \in R^{m \times n}$ is a constant matrix satisfying the condition: $C(B + B_d)$ is invertible, according to A1, there always exists such matrix satisfying this condition. $\Pi(t)$ is a sliding mode compensator and taken as

$$\dot{\Pi}(t) = -C[(A + A_d) - (B + B_d)K]x(t) \tag{8.50}$$

where $K \in R^{m \times n}$ is a constant matrix to be found.

After selecting the sliding manifolds, the next step is to choose a control law such that it satisfies the condition for the existence of the sliding mode; $S^T(t)\dot{S}(t) < 0$. This condition ensures that the control law will force system trajectories toward the sliding manifolds in finite time and maintain them on the manifolds after then. The following control structure of the form is considered:

$$u(t) = u_{eq}(t) + u_N(t) \tag{8.51}$$

where $u_{eq}(t)$ is an equivalent control for system (8.46), and $u_N(t)$ is a switching control. The equivalent control law $u_{eq}(t)$ is derived by $\dot{S}(t) = 0$ for the system (8.46). Differentiating $S(t)$ with respect to time gives

$$\begin{aligned}\dot{S}(t) &= C(A + A_d)x(t) + C(B + B_d)u(t) + CA_d x(t - \tau(t)) - CA_d x(t - \tau_0) \\ &\quad + CB_d u(t - d(t)) - CB_d u(t - d_0) + \dot{\Pi}(t). \end{aligned} \tag{8.52}$$

It follows (8.3) that

$$\begin{aligned}\dot{S}(t) &= C(B + B_d)Kx(t) + C(B + B_d)u(t) + CA_d[x(t - \tau(t)) - x(t - \tau_0)] \\ &\quad + CB_d[u(t - d(t)) - u(t - d_0)]. \end{aligned} \tag{8.53}$$

Let $\Omega(t) = CA_d(x(t - \tau(t)) - x(t - \tau_0)) + CB_d(u(t - \tau(t)) - u(t - \tau(t))$.

Then, the equivalent control is obtained by

$$u_{eq}(t) = -Kx(t). \tag{8.54}$$

Now, in order to force system (8.46) trajectories toward the designed sliding manifolds, the switching control u_N is chosen by

$$\dot{S}(t) = -gS(t) - \epsilon \mathrm{sgn}(S) + \Omega(t) \tag{8.55}$$

where $\mathrm{sgn}(S(z)) = [\mathrm{sgn}(s_1(z)), \cdots, \mathrm{sgn}(s_l(z))]^T$, $g = \mathrm{diag}\{g_i\}$ and $\epsilon = \mathrm{diag}\{\epsilon_i\}$ with two positive-constants g_i and ϵ_i that can be chosen such that $S^T(t)\dot{S}(t) < 0$. Then,

$$u_N = -[C(B + B_d)]^{-1}[gS(t) + \epsilon \mathrm{sgn}(S)]. \tag{8.56}$$

Thus, the following control law is obtained

$$u(t) = -Kx(t) - [C(B + B_d)]^{-1}[gS(t) + \epsilon \mathrm{sgn}(S)]. \tag{8.57}$$

After the system trajectories reaching the sliding manifolds, the sliding motion on the sliding manifolds is

$$\dot{x}(t) = (A - B_1K)x(t) + A_d x(t - \tau(t)) - B_d Kx(t - d(t)). \tag{8.58}$$

Writing $A_k = A - B_1K$, $B_k = -B_dK$, then (8.58) can be written as

$$\dot{x}(t) = A_k x(t) + A_d x(t - \tau(t)) + B_k x(t - d(t)). \tag{8.59}$$

From the above, the design of a sliding mode control is possible if (i) there exists $K \in R^{m\times n}$ which guarantees stability of (8.59), and (ii) there exists control law which makes the sliding function asymptotically stable for a specified sliding manifolds. The remaind of this section is devoted to the design of a sliding manifolds and control law to satisfy these requirements.

Theorem 8.7. *The sliding motion (8.59) is stable for any time-varying delays $\tau(t)$ and $d(t)$, if there exist symmetric positive-definite matrices $\mathcal{P} \in R^{n\times n}$, $\mathcal{Q} \in R^{n\times n}$, $\mathcal{R}_1 \in R^{n\times n}$, $\mathcal{R}_2 \in R^{n\times n}$ and a matrix $Y \in R^{m\times n}$ such that the following LMI holds*

$$\Sigma = \begin{bmatrix} \Xi_{11} & \Xi_{12} & \Xi_{13} & \Xi_{14} & \Xi_{15} & 0 & 0 & \mathcal{P} \\ * & \Xi_{22} & 0 & \Xi_{24} & \Xi_{25} & \tau_M\mathcal{R}_1 & 0 & 0 \\ * & * & -2\beta_2^{-1}\mathcal{P} & \Xi_{33} & \Xi_{34} & 0 & d_M\mathcal{R}_2 & 0 \\ * & * & * & -\tau_M\mathcal{R}_1 & 0 & 0 & 0 & 0 \\ * & * & * & * & -d_M\mathcal{R}_2 & 0 & 0 & 0 \\ * & * & * & * & * & -\tau_M\mathcal{R}_1 & 0 & 0 \\ * & * & * & * & * & * & -d_M\mathcal{R}_2 & 0 \\ * & * & * & * & * & * & * & -\mathcal{Q} \end{bmatrix} < 0$$

with

$$\begin{aligned} \Xi_{11} &= A\mathcal{P} - BY + \mathcal{P}A^T - Y^TB^T - \alpha_1\beta_1^{-1}A_d\mathcal{Q} - \alpha_1\beta_1^{-1}\mathcal{Q}A_d^T \\ &\quad + \alpha_2\beta_2^{-1}B_dY + \alpha_2\beta_2^{-1}Y^TB_d^T - (1-\rho)\alpha_1^2\beta_1^{-2}\mathcal{Q} \\ \Xi_{12} &= \mathcal{P} + \alpha_1\beta_1^{-1}\mathcal{Q} + \beta_1^{-1}A_d\mathcal{Q} + (1-\rho)\alpha_1\beta_1^{-2}\mathcal{Q} \\ \Xi_{13} &= \mathcal{P} + \alpha_2\beta_2^{-1}\mathcal{Q} + \beta_2^{-1}B_dY \\ \Xi_{14} &= \tau_M\mathcal{P}A^T - \tau_MY^TB^T - \tau_M\alpha_1\beta_1^{-1}\mathcal{Q}A_d^T + \tau_M\alpha_2\beta_2^{-1}Y^TB_d^T \\ \Xi_{15} &= d_M\mathcal{P}A^T - d_MY^TB^T - d_M\alpha_1\beta_1^{-1}\mathcal{Q}A_d^T + d_M\alpha_2\beta_2^{-1}Y^TB_d^T \\ \Xi_{22} &= -2\beta_1^{-1}\mathcal{Q} - (1-\rho)\beta_1^{-2}\mathcal{Q} \\ \Xi_{24} &= \tau_M\beta_1^{-1}\mathcal{Q}A_d^T \\ \Xi_{25} &= d_M\beta_1^{-1}\mathcal{Q}A_d^T \\ \Xi_{33} &= -\tau_M\beta_2^{-1}Y^TB_d^T \\ \Xi_{34} &= -d_M\beta_2^{-1}Y^TB_d^T. \end{aligned} \tag{8.60}$$

Moreover, a stabilizing equivalent control law is given by $u_{eq}(t) = -Y\mathcal{P}^{-1}x(t)$.

Proof. Construct the following Lyapunov-Krasovskii functional:

$$V(t) = x^T(t)Px(t) + \int_{t-\tau(t)}^{t} x^T(s)Qx(s)ds + \int_{-\tau_M}^{0}\int_{t+\theta}^{t} \dot{x}^T(s)R_1\dot{x}(s)dsd\theta + \int_{-d_M}^{0}\int_{t+\theta}^{t} \dot{x}^T(s)R_2\dot{x}(s)dsd\theta \tag{8.61}$$

where $P > 0$, $Q > 0$, $R_1 > 0$ and $R_2 > 0$. Then, the time derivative of $V(t)$ along system trajectory (8.59) satisfies

$$\dot{V}(t) = 2x^T(t)P\dot{x}(t) + x^T(t)Qx(t) - (1-\dot{\tau}(t))x^T(t-\tau(t))Qx(t-\tau(t))$$

$$+\tau_M \dot{x}^T(t)R_1\dot{x}(t)$$
$$-\tau_M \int_{t-\tau_M}^{t} \dot{x}^T(s)R_1\dot{x}(s)ds + d_M\dot{x}^T(t)R_2\dot{x}(t)$$
$$-d_M \int_{t-d_M}^{t} \dot{x}^T(s)R_2\dot{x}(s)ds.$$

Using Lemma 1.8, there exists:

$$-\int_{t-\tau_M}^{t} \dot{x}^T(s)R_1\dot{x}(s)ds \le -\int_{t-\tau(t)}^{t} \dot{x}^T(s)R_1\dot{x}(s)ds$$
$$\le \left[\, x^T(t)\ x^T(t-\tau(t)) \,\right] \begin{bmatrix} M_1^T + M_1 & -M_1^T + M_2 \\ -M_1 + M_2^T & -M_2 - M_2^T \end{bmatrix} \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}$$
$$+\tau_M \left[\, x^T(t)\ x^T(t-\tau(t)) \,\right] \begin{bmatrix} M_1^T \\ M_2^T \end{bmatrix} R_1^{-1} \left[\, M_1\ M_2 \,\right] \begin{bmatrix} x(t) \\ x(t-\tau(t)) \end{bmatrix}$$

and

$$-\int_{t-d_M}^{t} \dot{x}^T(s)R_1\dot{x}(s)ds \le -\int_{t-d(t)}^{t} \dot{x}^T(s)R_2\dot{x}(s)ds$$
$$\le \left[\, x^T(t)\ x^T(t-d(t)) \,\right] \begin{bmatrix} N_1^T + N_1 & -N_1^T + N_2 \\ -N_1 + N_2^T & -N_2 - N_2^T \end{bmatrix} \begin{bmatrix} x(t) \\ x(t-d(t)) \end{bmatrix}$$
$$+d_M \left[\, x^T(t)\ x^T(t-d(t)) \,\right] \begin{bmatrix} N_1^T \\ N_2^T \end{bmatrix} R_1^{-1} \left[\, N_1\ N_2 \,\right] \begin{bmatrix} x(t) \\ x(t-d(t)) \end{bmatrix}.$$

By using $\dot{\tau}(t) \le \rho < 1$ and Schur complement, it can be easily shown that

$$V(t) \le \xi^T(t)\Sigma_1\xi(t) < 0 \tag{8.62}$$

where

$$\xi(t) = \left[\, x^T(t)\ x^T(t-\tau(t))\ x^T(t-d(t))\ y_1^T(t)\ y_2^T(t)\ y_3^T(t)\ y_4^T(t) \,\right]^T$$

$$\Sigma_1 = \begin{bmatrix} \Sigma_1(1,1) & \Sigma_1(1,2) & \Sigma_1(1,3) & \tau_M A_k^T & d_M A_k^T & \tau_M M_1^T & d_M N_1^T \\ * & \Sigma_1(2,2) & 0 & \tau_M A_d^T & d_M A_d^T & \tau_M M_2^T & 0 \\ * & * & \Sigma_1(3,3) & \tau_M B_k^T & d_M B_k^T & 0 & d_M N_2^T \\ * & * & * & -\tau_M R_1^{-1} & 0 & 0 & 0 \\ * & * & * & * & -d_M R_2^{-1} & 0 & 0 \\ * & * & * & * & * & -\tau_M R_1 & 0 \\ * & * & * & * & * & * & -d_M R_2 \end{bmatrix}$$

with

$$\Sigma_1(1,1) = PA_k + A_k^T P + Q + M_1^T + M_1 + N_1^T + N_1$$
$$\Sigma_1(1,2) = PA_d - M_1^T + M_2$$

$$\Sigma_1(1,3) = PB_k - N_1^T + N_2$$
$$\Sigma_1(2,2) = -(1-\rho)Q - M_2^T - M_2$$
$$\Sigma_1(3,3) = -N_2^T - N_2$$

and $y_1(t)$, $y_1(t)$, $y_3(t)$ and $y_4(t)$ are arbitrary vectors with appropriate dimensions.

In order to obtain a stabilizing equivalent controller gain K, letting $M_1 = \alpha_1 P$, $M_2 = \beta_1 Q$, $N_1 = \alpha_2 P$, $N_2 = \beta_2 P$ and

$$\Theta_{(\alpha,\beta)} = \begin{bmatrix} P^{-1} & 0 & 0 \\ -\alpha_1\beta_1^{-1}Q^{-1} & \beta_1^{-1}Q^{-1} & 0 \\ -\alpha_2\beta_2^{-1}P^{-1} & 0 & \beta_2^{-1}P^{-1} \end{bmatrix}.$$

Pre-multiplying and post-multiplying $\Sigma_1 < 0$ by the diagonal matrix $\text{diag}\{\Theta_{(\alpha,\beta)}^T \ I \ I \ R_1^{-1} \ R_2^{-1}\}$ and $\text{diag}\{\Theta_{(\alpha,\beta)} \ I \ I \ R_1^{-1} \ R_2^{-1}\}$, respectively, it can be changed to $\Sigma_2 < 0$, where

$$\Sigma_2 = \begin{bmatrix} \Sigma_2(1,1) & \Sigma_2(1,2) & \Sigma_2(1,3) & \Sigma_2(1,4) & \Sigma_2(1,5) & 0 & 0 \\ * & \Sigma_2(2,2) & 0 & \Sigma_2(2,4) & \Sigma_2(2,5) & \tau_M R_1^{-1} & 0 \\ * & * & \Sigma_2(3,3) & \Sigma_2(3,4) & \Sigma_2(3,5) & 0 & d_M R_2^{-1} \\ * & * & * & -\tau_M R_1^{-1} & 0 & 0 & 0 \\ * & * & * & * & -d_M R_2^{-1} & 0 & 0 \\ * & * & * & * & * & -\tau_M R_1^{-1} & 0 \\ * & * & * & * & * & * & -d_M R_2^{-1} \end{bmatrix}$$

with

$$\Sigma_2(1,1) = A_k P^{-1} + P^{-1}A_k^T - \alpha_1\beta_1^{-1}A_d Q^{-1} - \alpha_1\beta_1^{-1}Q^{-1}A_d^T$$
$$-\alpha_2\beta_2^{-1}B_k P^{-1} - \alpha_2\beta_2^{-1}P^{-1}B_k^T - (1-\rho)\alpha_1^2\beta_1^{-2}Q^{-1} + P^{-1}QP^{-1}$$
$$\Sigma_2(1,2) = P^{-1} + \alpha_1\beta_1^{-1}Q^{-1} + \beta_1^{-1}A_d Q^{-1} + (1-\rho)\alpha_1\beta_1^{-2}Q^{-1}$$
$$\Sigma_2(1,3) = P^{-1} + \alpha_2\beta_2^{-1}Q^{-1} + \beta_2^{-1}B_k P^{-1}$$
$$\Sigma_2(1,4) = \tau_M P^{-1}A_k^T - \tau_M\alpha_1\beta_1^{-1}Q^{-1}A_d^T - \tau_M\alpha_2\beta_2^{-1}P^{-1}B_k^T$$
$$\Sigma_2(1,5) = d_M P^{-1}A_k^T - d_M\alpha_1\beta_1^{-1}Q^{-1}A_d^T - d_M\alpha_2\beta_2^{-1}P^{-1}B_k^T$$
$$\Sigma_2(2,2) = -2\beta_1^{-1}Q^{-1} - (1-\rho)\beta_1^{-2}Q^{-1}$$
$$\Sigma_2(2,4) = \tau_M\beta_1^{-1}Q^{-1}A_d^T$$
$$\Sigma_2(2,5) = d_M\beta_1^{-1}Q^{-1}A_d^T$$
$$\Sigma_2(3,3) = -2\beta_2^{-1}P^{-1}$$
$$\Sigma_2(3,4) = \tau_M\beta_2^{-1}P^{-1}B_k^T$$
$$\Sigma_2(3,5) = d_M\beta_2^{-1}P^{-1}B_k^T$$

letting $\mathcal{P} = P^{-1}$, $Y = KP^{-1}$, $\mathcal{Q} = Q^{-1}$, $\mathcal{R}_1 = R_1^{-1}$ and $\mathcal{R}_2 = R_2^{-1}$, the inequality $\Sigma_2 < 0$ is equivalent to $\Sigma < 0$.

Now, we are in a position to prove that the trajectories of the closed-loop systems can reach sliding surface in a finite time.

Theorem 8.8. *Suppose inequality (8.60) in Theorem 8.7 has solutions $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{R}_1$, $\mathcal{R}_2$ and Y and the sliding surface is given by (8.49). Then the trajectory of the closed-loop system (8.46) can be driven onto the sliding surface in finite time with the control*

$$u(t) = -Kx(t) - [C(B + B_d)]^{-1}[gS(t) + \epsilon \mathrm{sgn}(S)] \tag{8.63}$$

where $\mathrm{sgn}((z)) = [\mathrm{sgn}(s_1(z)), \cdots, \mathrm{sgn}(s_l(z))]^T$, $g = diag\{g_i\}$ *and* $\epsilon = diag\{\epsilon_i\}$ *with two positive-constants* g_i *and* ϵ_i.

Proof. Choose the following Lyaponov functional

$$V_s(t) = \frac{1}{2} S^T(t)S(t). \tag{8.64}$$

Based on (8.49), (8.3) and (8.55), differentiating $V_s(t)$ with respect to time along the trajectories of systems (8.53), we have

$$\begin{aligned} \dot{V}_s(t) &= S^T(t)(-gS(t) - \epsilon \mathrm{sgn}(S) + \Omega(t)) \\ &= -gS^T(t)S(t) - \epsilon S^T(t)\mathrm{sgn}(S) + S^T(t)\Omega(t). \end{aligned} \tag{8.65}$$

Since $\Omega(t)$ can be viewed as bounded functional, we can tune two parameters g and ϵ appropriately, such that $\dot{V}_s(t) < 0$. Thus, the trajectories of the closed-loop systems can reach sliding surface in a finite time.

8.6 Numerical Example

In this section, the theory developed in this chapter is demonstrated by means of one example which was presented in [63].

Example 8.9. Consider the following time-delayed system [63]

$$\dot{y}(t) = Ay(t) + A_d y(t - \tau) + B_2 u(t - \tau)$$

with

$$\begin{aligned} &A = \begin{bmatrix} -2 & 3 \\ 0 & 1 \end{bmatrix}, A_d = \begin{bmatrix} 1 & 2 \\ -1 & 3 \end{bmatrix}, B_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &u(t) = 1, t \in [-\tau, 0] \\ &y(t) = [1 \ 1]^T, t \in [-\tau, 0] \\ &y_1(t) = [y_{11} \quad y_{12}]^T. \end{aligned} \tag{8.66}$$

In the sequel, the stabilizability criterion of Theorem 8.3 is compared with those of [60], [63].

In the light of Theorem 8.3 it has been obtained using the software package LMI Lab that the system (8.33) is stabilizable for any τ satisfying $0 \leq \tau \leq 0.3615$. On the other hand, the method of [63] ensures the stabilizability of system (8.33) for $0 \leq \tau \leq 0.2472$, whereas using the result of [60] the time delay must satisfy $0 \leq \tau \leq 0.0338$. It should be noted that the results of [60] and [63] are based on the solution of Lyapunov equation, furthermore, the results of [63] involve the tuning and factorizing of a 2×2 symmetric positive-definite matrix while it is difficult for tuning and factorizing such matrix in order to maximize the bound for the time delay.

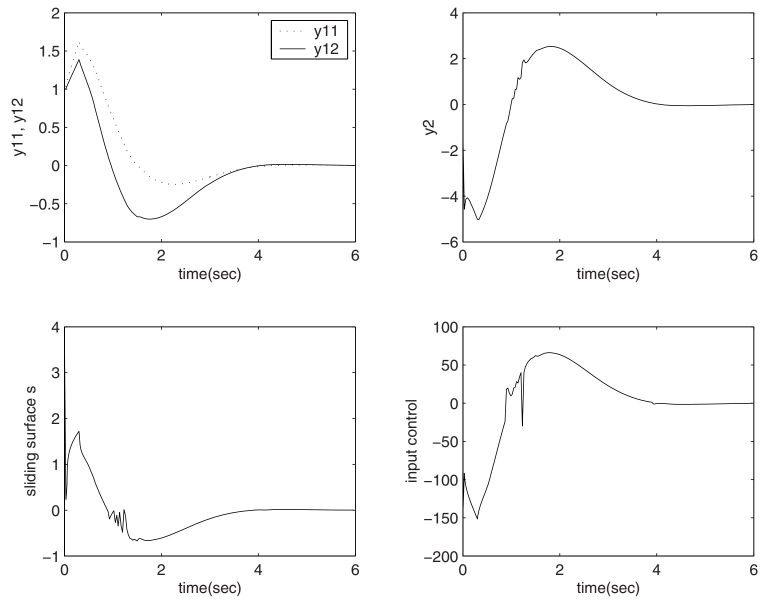

Fig. 8.1 States (y_{11}, y_{12}), y_2 , sliding surface S and control input $(\overline{u})$

Hence, for this example, the stabilizability criterion of the sliding motion of this chapter gives a less conservative result than those obtained by the methods of [60], [63].

Applying the control law (8.38) with $q_1 = 2$, $q_2 = 2$, $\beta = 6$, $K = [0.3434 \;\; 4.4480]$, gives the following simulation using a delay of 0.3 (near optimal one). Fig. 8.1 shows that the system is asymptotically stable in the sliding mode and the chattering of the proposed control with appropriate coefficients is not evident.

Example 8.10. The proposed method will be applied to design a sliding mode controller for an inverted pendulum system to illustrate the results developed. Consider the inverted pendulum on a cart such as in [16]. The physical structure is shown in Fig. 8.2. where M is the mass of the cart, m is the mass of the pendulum rod, b is the friction coefficient of the cart. l is the length

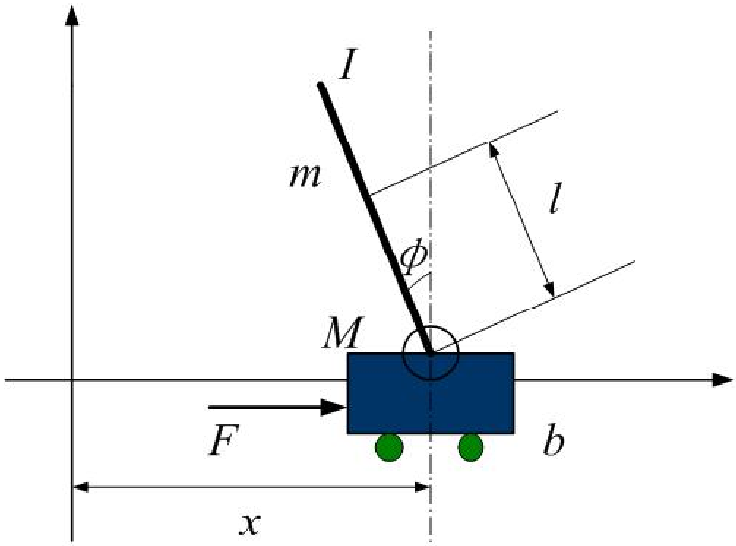

Fig. 8.2 Inverted pendulum Model.

of the correspond rod; F is the force acting on the cart, x is the horizontal displacement of the cart, ϕ is the angle between the pendulum rod and the vertical. Let state variables are x, $\dot{x}$, ϕ and $\dot{\phi}$ which corresponds to the horizontal position, horizontal velocity of the cart, angle and angle velocity of the pole respectively. We are interested only in the linear system about the equilibrium point at the origin. The equations of motion derived by using Newton's Second Law can be obtained as follows:

$$\begin{aligned}
\dot{x} &= \dot{x} \\
\ddot{x} &= \frac{-(I+ml^2)b}{I(M+m)+Mml^2}\dot{x} + \frac{m^2gl^2}{I(M+m)+Mml^2}\phi + \frac{(I+ml^2)b}{I(M+m)+Mml^2}u \\
\dot{\phi} &= \dot{\phi} \\
\ddot{\phi} &= \frac{-mlb}{I(M+m)+Mml^2}\dot{x} + \frac{mgl(M+m)}{I(M+m)+Mml^2}\phi + \frac{ml}{I(M+m)+Mml^2}u
\end{aligned}$$

where $M = 1.096$ Kg, $m = 0.109$ Kg, $b = 0.1$ N/m/sec, $l = 0.25$ m and $I = 0.034$ kg·m·m.

In the following, we investigate the case of control pass networks, the state delay and input delay of inverted pendulum model have to be considered. Let state variables be $x = x_1(t)$, $\dot{x} = x_2(t)$, $\phi = x_3(t)$ and $\dot{\phi} = x_4(t)$, the two delays be $\tau(t) = 0.2|sin(t)|$ and $d(t) = 0.1|cos(t)|$, we can obtained as

$$\dot{x}(t) = Ax(t) + A_d x(t-\tau(t)) + Bu(t) + B_d u(t-d(t)) \tag{8.67}$$

where

$$A = \begin{bmatrix} 0 & 0.7 & 0 & 0 \\ 0 & -0.0618 & 0.4405 & 0 \\ 0 & 0 & 0 & 0.7 \\ 0 & -0.165 & 19.4799 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 0 \\ 0.6182 \\ 0 \\ 1.6496 \end{bmatrix}$$

$$A_d = \begin{bmatrix} 0 & 0.3 & 0 & 0 \\ 0 & -0.0265 & 0.1888 & 0 \\ 0 & 0 & 0 & 0.3 \\ 0 & -0.0707 & 8.3485 & 0 \end{bmatrix}, \; B_d = \begin{bmatrix} 0 \\ 0.265 \\ 0 \\ 0.707 \end{bmatrix}.$$

Then, using the Matlab LMI Control Toolbox to solve the LMIs (8.60) in Theorem 8.7, the solution of it is obtained as follows:

$$\mathcal{P} = \begin{bmatrix} 45.7759 & -11.3435 & 0.1407 & -0.5977 \\ -11.3435 & 7.2390 & 0.1478 & 1.5145 \\ 0.1407 & 0.1478 & 0.3277 & -1.4483 \\ -0.5977 & 1.5145 & -1.4483 & 9.1189 \end{bmatrix}$$

$$\mathcal{Q} = \begin{bmatrix} 80.8251 & -19.8415 & 0.2143 & -1.0993 \\ -19.8415 & 11.3501 & 0.0836 & 1.4963 \\ 0.2143 & 0.0836 & 0.3913 & -1.8627 \\ -1.0993 & 1.4963 & -1.8627 & 10.8990 \end{bmatrix}$$

$$\mathcal{R}_1 = \begin{bmatrix} 37.5651 & -6.7791 & 0.1996 & 0.0872 \\ -6.7791 & 6.5049 & -0.1253 & 1.3955 \\ 0.1996 & -0.1253 & 1.4731 & -6.9796 \\ 0.0872 & 1.3955 & -6.9796 & 40.9957 \end{bmatrix}$$

$$\mathcal{R}_2 = \begin{bmatrix} 68.8773 & -4.0952 & 0.6471 & 0.9340 \\ -4.0952 & 39.0123 & 1.7623 & 2.7297 \\ 0.6471 & 1.7623 & 1.9660 & -9.1944 \\ 0.9340 & 2.7297 & -9.1944 & 51.7105 \end{bmatrix}$$

$$Y = \begin{bmatrix} 2.2859 & 5.0737 & 1.1375 & 5.8852 \end{bmatrix}.$$

The largest state delay is $\tau_M = 0.2$ and the largest input delay is $d_M = 0.1$. Moreover, a stabilizing equivalent control law is given by

$$u_{eq}(t) = -Kx(t) = -Y\mathcal{P}^{-1}x(t) = [0.3217 \; 1.4361 \; -27.0722 \; -5.1627]x(t).$$

It follows from $C(B+B_d)$ is invertible that $C = [0 \; 0.1394 \; 0 \; 0.3721]$, the linear sliding surface is given as (8.49). From Theorem 2, the reaching control law can be taken as follows:

$$u(t) = -Kx(t) - [C(B + B_d)]^{-1}[gS(t) + \epsilon Sign(S)]$$

where parameter g and ϵ can be tuned to reduce the chattering on the sliding surface. Fig. 8.3-Fig. 8.3 are simulation results when choosing $g = 0.4$ and

$\epsilon = 0.02$. $x_1(0) = x_2(0) = x_3(0) = x_4(0) = 1$, $S(0) = 0.5115$ and $u(0) = 0$. Obviously, system (8.67) is asymptotically stable and the sliding motion trends to the origin in finite time.

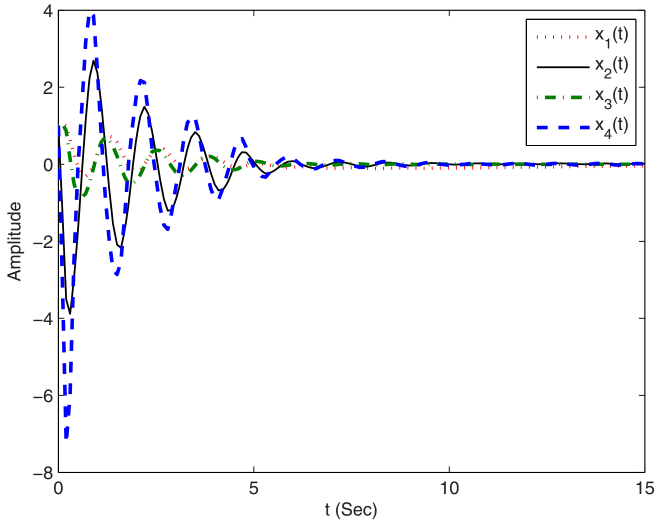

Fig. 8.3 State variable of system (8.67)

8.7 Conclusion

In this chapter, the problem of designing sliding mode controller for systems with input and state delays has been considered. In terms of LMI, sufficient conditions for the existence of two kinds of linear sliding manifolds guaranteeing stability of the sliding motion restricted to the sliding surface are derived. A new reaching motion controller is proposed for these systems by means of the reaching law.

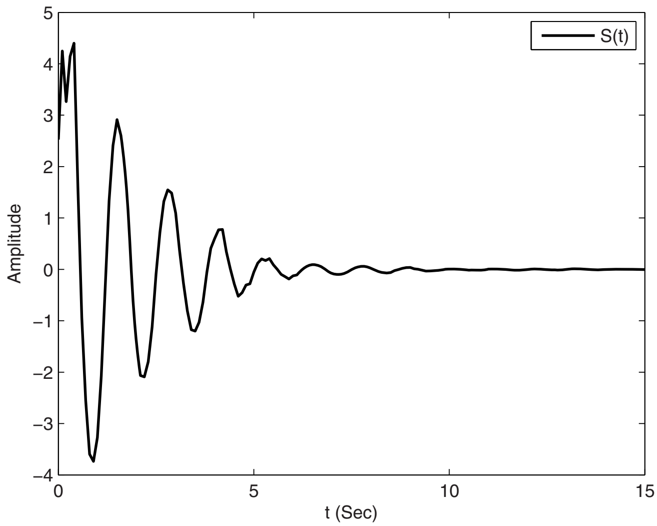

Fig. 8.4 Sliding surface of system (8.67)

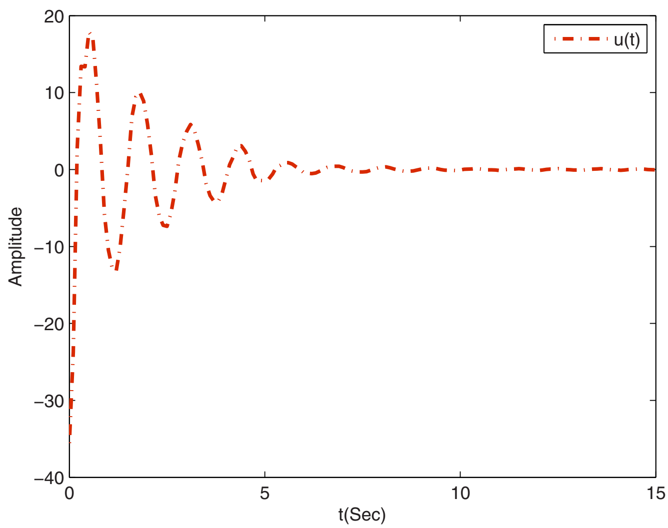

Fig. 8.5 Control input of system (8.67)

Chapter 9
Robust SMC for Uncertain Time-Delay Systems Based on Delta Operator

9.1 Introduction

A discrete version of SMC is important when the implementation of the control is realized digitally using a relatively slow sampling period. In [34], the problem of discrete VSC was first considered. The concept of the quasi-sliding mode was proposed in [134], and phenomena of switching, reaching, and quasi-sliding mode were investigated in [55, 5, 6] and reference therein. It is worth pointing out that discrete SMC cannot be obtained from their continuous counterpart by means of simple equivalence. When the sampling is fast using the traditional shift operator, the poles are located in the stable boundary, the discrete systems will lose stability in finite word length computer. To solve this problem, a delta operator instead of traditional shift operator for sampling continuous systems is constructed in [88]. The delta operator is defined by:

$$\delta x(t) = \begin{cases} \frac{dx(t)}{dt}, & h = 0 \\ \frac{x(t+h)-x(t)}{h}, & h \neq 0 \end{cases}$$

where h is a sampling period. A class of systems in delta domain has been studied in [195], [111], [23]. The problem of system instability in fast sampling is successfully solved by using delta operator model in [36]. The delta operator model also has the advantage of better numerical properties at high sampling rates [169]. In contrast to the discrete shift operator, the delta operator approach means that the Euler derivative can lead to a quasi-continuous time s-domain model for high sampling frequencies, see [41]. The robustness problem for some delta operator systems with parametric uncertainties also has been investigated. The problems of stability of delta operator systems were considered in [166] and [167]. A feedback control approach was reported in [38], which proved that a discrete-time controller is approach to a continuous-time controller using delta operator when $h \to 0$.

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 119–136.
springerlink.com

In the time-domain approach, the direct Lyapunov method is a powerful tool for studying the problems of stability and feedback control for continuous-time systems and discrete systems. The problem of designing state feedback controller for continuous systems with time-varying delay has received considerable attention ([194], [189], [188], [7], [75]). However, there have been few papers on SMC for discrete-time systems with time-varying delay via delta operator approach. As a result, based on the delta operator approach, the reaching law proposed is able to reduce the chattering due to the implementation of variable structure controller, and possesses the desired characteristics of robustness and good performance. The proposed method can unify some previous related results of the continuous and discrete sliding mode control for systems into the delta operator systems framework ([191],[160], [9], [189]).

In this chapter, we focus on the delta operator systems with both linear fractional uncertainties and time-varying delays. The objective is to design a sliding mode controller such that the closed-loop systems is robustly asymptotically stable for all admissible uncertainties. A new delay-dependent approach is developed to design the sliding surface, and all results are obtained by using some new Lyapunov functions and given in terms of LMIs. The sampling-period h is an explicit parameter in our results so that it is convenient to analyze the effect of the sliding mode controller with different sampling periods. It is worth mentioning that we employ a fast sampling method in discrete systems, where the delays are actually time-varying. It is assumed that both the lower delay bound and upper delay bound can be denoted to the summations of the same sampling periods, respectively. Numerical example is given to illustrate the feasibility and effectiveness of the developed technique.

Remark 9.1. Consider the following continuous-time system in s-domain without time delays:

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t).$$

Utilizing traditional shift operator, the above continuous-time system can be changed to the discrete-time system in z-domain as below:

$$x((k+1)h) = A_z x(kh) + B_z u(kh), \quad y(kh) = C_z x(kh)$$

where $A_z = e^{Ah}$, $B_z = \int_0^h e^{A(h-s)}Bds$ and $C_z = C$. When $h \to 0$, there exist $\lim_{h\to 0} A_z = I$ and $\lim_{h\to 0} B_z = 0$. It is shown that A_z and B_z can not approach A and B, respectively, when the sampling period tends to zero. Utilizing delta operator, the discrete-time system in δ-domain can be given as follows:

$$\delta x(kh) = A_\delta x(kh) + B_\delta u(kh), \quad y(kh) = C_\delta x(kh)$$

where $A_\delta = (A_z - I)/h$, $B_\delta = B_z/h$ and $C_\delta = C$. When $h \to 0$, there exist $\lim_{h\to 0} A_\delta = A$ and $\lim_{h\to 0} B_\delta = B$, which is reasonable. The discrete method by using delta operator can possess better numerical properties. In this chapter, let $t = kh$ for the sake of convenience in the following analysis.

The chapter is organized as follows. Section 9.2 gives the problem formulation and some preliminaries. In section 9.3, results on designing sliding surface and the reaching motion controller in δ-domain are established. Numerical simulations are presented in Section 9.4 and some conclusion remarks are given in Section 9.5.

9.2 Problem Formulation

In this chapter, the following uncertain delta operator system with time-varying delays is considered:

$$\begin{aligned} \delta x(t) &= (A + \varDelta A(t))x(t) + (A_d + \varDelta A_d(t))x(t - d(t)) \\ &\quad + B(u(t) + w(t)) \\ x(t) &= \varphi(t) \ \ for \ t \in [-d_M,\ 0] \end{aligned} \tag{9.1}$$

where $x(t) \in R^n$ is the state variable; $u(t) \in R^m$ is control input; $\varphi(t)$ is a initial value at t; $w(t) \in R^m$ is the disturbance with each component bounded by the known $\overline{w}_i(t)$, i.e., $|w_i(t)| \le \overline{w}_i(t)$, $i = 1, 2, \cdots, m$, $|\overline{w}_i(t)| > 0$, $\forall t$; the time delay $d(t)$ is a time-varying function that satisfies $0 \le d_m \le d(t) \le d_M$, with $d_m = n_m h$ and $d_M = n_M h$, n_m and n_M are two known positive and finite integers, from which we let $n_m \le n \le n_M$. A, A_d and B are real constant matrices with appropriate dimensions and $\mathrm{rank}(B) = m$. The linear fractional parametric uncertainties are described by

$$\varDelta A = G\hat{F}(t)H, \varDelta A_d = G_d\hat{F}(t)H_d$$

where

$$\hat{F}(t) = F(t)\left[I - WF(t)\right]^{-1} \tag{9.2}$$

where G, H, G_d, H_d and W are known constant real matrices with appropriate dimensions. $F(t)$ is unknown time-varying matrix satisfying $F^T(t)F(t) \le I$. It is assumed that the matrix $[I - WF(t)]^{-1}$ is invertible for any $F(t)$ and $I - W^T W > 0$.

Remark 9.2. The model (9.2) describes a wider class of parameter uncertainties than norm-bounded parameter uncertainties. It is easy to see that the linear fractional parameter uncertainties can be reduced to norm-bounded parameter uncertainties when $W = 0$.

To get a regular form of systems (9.1), a nonsingular matrix Ψ can be chosen such that

$$\Psi B = \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix}$$

where $B_2 \in R^{m\times m}$ is nonsingular. For convenience, let us choose

$$\Psi = \begin{bmatrix} {U_2}^T \\ {U_1}^T \end{bmatrix}$$

where $U_1 \in R^{n\times m}$ and $U_2 \in R^{n\times(n-m)}$ are two sub-blocks of an unitary matrix resulting from the singular value decomposition of B, i.e.,

$$B = \begin{bmatrix} U_1 \; U_2 \end{bmatrix} \begin{bmatrix} \Sigma \\ 0_{(n-m)\times m} \end{bmatrix} V^T$$

where $\Sigma \in R^{m\times m}$ is a diagonal positive-definite matrix and $V \in R^{m\times m}$ which is an unitary matrix. By the state transformation $z = \Psi x$, system (9.1) has the regular form

$$\begin{aligned} \delta z(t) &= (\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d)z(t-d(t)) + \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix}(u(t)+w(t)) \\ z(t) &= \overline{\varphi}(t) \;\; for \; t \in [-d_M, \; 0] \end{aligned} \tag{9.3}$$

where $\overline{A} = \Psi A \Psi^{-1}$, $\overline{A}_d = \Psi A_d \Psi^{-1}$, $\Delta\overline{A} = \Psi \Delta A \Psi^{-1}$, $\Delta\overline{A}_d = \Psi \Delta A_d \Psi^{-1}$ and $\overline{\varphi}(t) = \Psi\varphi(t)$ is a initial value at t. System (9.3) can be written as:

$$\begin{aligned} \delta z_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11})z_1(t) + (\overline{A}_{d11} + \Delta\overline{A}_{d11})z_1(t-d(t)) \\ &\quad +(\overline{A}_{12} + \Delta\overline{A}_{12})z_2(t) + (\overline{A}_{d12} + \Delta\overline{A}_{d12})z_2(t-d(t)) \\ \delta z_2(t) &= (\overline{A}_{21} + \Delta\overline{A}_{21})z_1(t) + (\overline{A}_{d21} + \Delta\overline{A}_{d21})z_1(t-d(t)) \\ &\quad +(\overline{A}_{22} + \Delta\overline{A}_{22})z_2(t) + (\overline{A}_{d22} + \Delta\overline{A}_{d22})z_2(t-d(t)) + B_2(u+w(t)) \\ z_1(t) &= \overline{\varphi}_1(t) \;\; for \; t \in [-d_M, \; 0] \\ z_2(t) &= \overline{\varphi}_2(t) \;\; for \; t \in [-d_M, \; 0] \end{aligned} \tag{9.4}$$

where $z_1(t) \in R^{n-m}$, $z_2(t) \in R^m$, $B_2 = \Sigma V^T$, $\overline{A}_{11} = U_2^T A U_2$, $\overline{A}_{12} = U_2^T A U_1$, $\overline{A}_{d11} = U_2^T A_d U_2$, $\overline{A}_{d12} = U_2^T A_d U_1$, $\Delta\overline{A}_{11} = {U_2}^T \Delta A U_2$, $\Delta\overline{A}_{d11} = {U_2}^T \Delta A_d U_2$, $\Delta\overline{A}_{12} = {U_2}^T \Delta A U_1$, $\Delta\overline{A}_{d12} = {U_2}^T \Delta A_d U_1$, $\overline{A}_{21} = U_1^T A U_2$, $\overline{A}_{22} = U_1^T A U_1$, $\overline{A}_{d21} = U_1^T A_d U_2$, $\overline{A}_{d22} = U_1^T A_d U_1$, $\Delta\overline{A}_{21} = {U_1}^T \Delta A U_2$, $\Delta\overline{A}_{d21} = {U_1}^T \Delta A_d U_2$, $\Delta\overline{A}_{22} = {U_1}^T \Delta A U_1$, $\Delta\overline{A}_{d22} = {U_1}^T \Delta A_d U_1$; initial values $\overline{\varphi}_1(t) \in R^{(n-m)}$ and $\overline{\varphi}_2(t) \in R^m$ are the sub-blocks of $\overline{\varphi}(t)$.

It is obvious that the first equation of system (9.4) represents the sliding motion dynamics of system (9.3), and hence the corresponding sliding surface can be chosen as follows:

$$S = [\; C \;\; I \;]\, z = Cz_1 + z_2 = 0 \tag{9.5}$$

where $C \in R^{m\times(n-m)}$ is sliding surface parameter matrix. Substituting $z_2 = -Cz_1$ to the first equation of system (9.4) gives the sliding motion

$$\begin{aligned}\delta z_1(t) &= (\overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C)z_1(t)\\ &\quad +(\overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C)z_1(t-d(t))\\ z_1(t) &= \overline{\varphi}_1(t) \ for \ t \in [-d_M,\ 0]\,.\end{aligned} \tag{9.6}$$

Let $\tilde{A} = \overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C$, $\tilde{A}_d = \overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C$, then the above equation can be written as

$$\begin{aligned}\delta z_1(t) &= \tilde{A}z_1(t) + \tilde{A}_d z_1(t-d(t))\\ z_1(t) &= \overline{\varphi}_1(t) \ for \ t \in [-d_M,\ 0]\,.\end{aligned} \tag{9.7}$$

Definition 9.3. The uncertain sliding motion (9.7) is said to be quadratically stable if there exists a constant $\xi > 0$ such that for any admissible uncertainties the delta operator manipulations of the Lyapunov functional $V(z_1, t)$ in delta domain for all pairs $(z_1, t) \in R^{n-m} \times R$, with respect to time t satisfies

$$L(z_1, t) = \delta V(z_1, t) \le -\xi\|z_1\|^2. \tag{9.8}$$

The objective in this chapter is to design sliding surface parameter $C \in R^{m\times(n-m)}$ and a reaching motion control law $u(t)$ such that

1) Sliding motion (9.7) is quadratically stable;

2) System (9.4) is asymptotically stable with the reaching control law $u(t)$.

Before ending this section, the following lemmas will be used to prove our main results in this chapter.

Lemma 9.4. [195] *The property of delta operator: for any time function $x(t)$ and $y(t)$*

$$\delta(x(t)y(t)) = \delta x(t)y(t) + x(t)\delta y(t) + h\delta x(t)\delta y(t)$$

where h is a sampling period.

Lemma 9.5. [94] *For any constant positive semi-definite symmetric matrix Π, two positive integers r and r_0 satisfying $r \ge r_0 \ge 1$, the following inequality holds*

$$\left(\sum_{i=r_0}^{r} x(i)\right)^T \Pi \left(\sum_{i=r_0}^{r} x(i)\right) \le (r - r_0 + 1)\sum_{i=r_0}^{r} x^T(i)\Pi x(i).$$

Lemma 9.6. [197] *For some given matrices Υ, G and H of appropriate dimensions and with Υ symmetric, then*

$$\Upsilon + G\hat{F}(t)H + H^T\hat{F}^T(t)G^T \le 0$$

with $\hat{F}(t)$ is as in (9.2), if and only if there exists a scalar $\varepsilon > 0$ such that

$$\begin{bmatrix} \varepsilon^{-1}H^T & \varepsilon G \end{bmatrix} \begin{bmatrix} I & -W \\ -W^T & I \end{bmatrix}^{-1} \begin{bmatrix} \varepsilon^{-1}H \\ \varepsilon G^T \end{bmatrix} + \Upsilon \leq 0.$$

9.3 Main Results

The first result of designing sliding surface can be stated as follows.

Theorem 9.7. *The reduced order system (9.7) is quadratically stable if there exist symmetric positive-definite matrices $J \in R^{(n-m)\times(n-m)}$, $Z \in R^{(n-m)\times(n-m)}$, $L \in R^{(n-m)\times(n-m)}$ and general matrix $Y \in R^{m\times(n-m)}$, as well as a positive scalar $\alpha > 0$ such that*

$$\begin{bmatrix} \Omega_{11} & \overline{A}_{11}Z - \overline{A}_{12}Y & \overline{A}_{d11}Z - \overline{A}_{d12}Y & 0 & \alpha U_2^T G \\ * & \Omega_{22} & \Omega_{23} & ZU_2^T H^T - Y^T U_1^T H^T & \alpha U_2^T G_d \\ * & * & -J - \frac{1}{d_M}L & ZU_2^T H_d^T - Y^T U_1^T H_d^T & 0 \\ * & * & * & -\alpha I & \alpha W \\ * & * & * & * & -\alpha I \end{bmatrix} < 0 \tag{9.9}$$

where

$$\Omega_{11} = (h-2)Z + d_M L$$
$$\Omega_{22} = Z\overline{A}_{11}^T - Y^T\overline{A}_{12}^T + \overline{A}_{11}Z - \overline{A}_{12}Y + (d_M - d_m + 1)J - \frac{1}{d_M}L$$
$$\Omega_{23} = \overline{A}_{d11}Z - \overline{A}_{d12}Y + \frac{1}{d_M}L.$$

Moreover, the sliding surface of the system (9.4) is

$$S(t) = YZ^{-1}z_1(t) + z_2(t) = 0. \tag{9.10}$$

Proof. Take symmetric positive-definite matrix variables P, Q, R with appropriate dimensions and choose the following Lyapunov functional

$$V(z_1,t) = V_1(z_1,t) + V_2(z_1,t) + V_3(z_1,t) + V_4(z_1,t)$$

which is positive-define for all $z_1(t) \neq 0$ and

$$V_1(z_1,t) = z_1^T(t)Pz_1(t)$$
$$V_2(z_1,t) = h\sum_{i=1}^{n} z_1^T(t-ih)Qz_1(t-ih)$$

$$V_3(z_1,t) = h^2 \sum_{i=n_m+1}^{n_M} \sum_{j=1}^{i} z_1^T(t-jh)Qz_1(t-jh)$$

$$V_4(z_1,t) = \sum_{i=1}^{n} \sum_{j=1}^{i} e^T(t-jh)Re(t-jh)$$

where $e(j) = z_1(j) - z_1(j+h)$, so there exist $\delta z_1(j) = -e(j)/h$ and $e(t-ih) = z_1(t-ih) - z_1(t-(i-1)h)$. Taking the delta operator manipulations of $V(z_1,t)$ along the trajectory of system (9.7), using Lemma 9.4, we can obtain:

$$\begin{aligned} &\delta V_1(z_1,t) = \delta^T(z_1(t))Pz_1(t) + z_1^T(t)P\delta(z_1(t)) + h\delta^T(z_1(t))P\delta(z_1(t)) \\ &= z_1^T(t)\tilde{A}^T Pz_1(t) + z_1^T(t-d(t))\tilde{A}_d^T Pz_1(t) + z_1^T(t)P\tilde{A}z_1(t) \\ &+ z_1^T(t)P\tilde{A}_d z_1(t-d(t)) + h\delta^T(z_1(t))P\delta(z_1(t)). \end{aligned} \tag{9.11}$$

Taking the delta operator manipulations of $V_2(z_1,t)$ and $V_3(z_1,t)$, it can be obtained that

$$\begin{aligned} &\delta V_2(z_1,t) \\ &= \frac{1}{h}\left[h\sum_{i=1}^{n} z_1^T(t-(i-1)h)Qz_1(t-(i-1)h) - h\sum_{i=1}^{n} z_1^T(t-ih)Qz_1(t-ih)\right] \\ &\le z_1^T(t)Qz_1(t) - z_1^T(t-d(t))Qz_1(t-d(t)) + h\sum_{i=n_m+1}^{n_M} z_1^T(t-ih)Qz_1(t-ih) \\ &\delta V_3(z_1,t) \\ &= h\sum_{i=n_m+1}^{n_M}\left(\sum_{j=1}^{i} z_1^T(t-(j-1)h)Qz_1(t-(j-1)h)\right. \\ &\quad \left. -\sum_{j=1}^{i} z_1^T(t-jh)Qz_1(t-jh)\right) \\ &= (d_M - d_m)z_1^T(t)Qz_1(t) - h\sum_{i=n_m+1}^{n_M} z_1^T(t-ih)Qz_1(t-ih). \end{aligned} \tag{9.12}$$

Using Lemma 9.5 and taking the delta operator manipulations of $V_4(z_1,t)$, there exists

$$\begin{aligned} &\delta V_4(z_1,t) \\ &= \frac{1}{h}\left[\sum_{i=1}^{n}\sum_{j=1}^{i} e^T(t-(j-1)h)Re(t-(j-1)h)\right. \end{aligned}$$

$$
\begin{aligned}
& -\sum_{i=1}^{n}\sum_{j=1}^{i} e^T(t-jh)Re(t-jh)\Bigg] \\
&= \frac{1}{h}\left[\sum_{i=1}^{n} e^T(t)Re(t) - \sum_{i=1}^{n} e^T(t-ih)Re(t-ih)\right] \\
&\leq \frac{n}{h}e^T(t)Re(t) - \frac{1}{nh}\left[\sum_{i=1}^{n} e(t-ih)\right]^T R\left[\sum_{i=1}^{n} e(t-ih)\right] \\
&\leq d_M\delta^T(x(t))R\delta(z_1(t)) - \frac{1}{d_M}\left[z_1(t-d(t)) - z_1(t)\right]^T R\left[z_1(t-d(t)) - z_1(t)\right].
\end{aligned}
$$

For the positive definite real matrix P, one has that

$$
\begin{aligned}
0 =& \; -2\delta^T(z_1(t))P[\delta(z_1(t)) - \tilde{A}(t)z_1(t) - \tilde{A}_d(t)z_1(t-d(t))] \\
=& \; -2\delta^T(z_1(t))P\delta(z_1(t)) + 2\delta^T(z_1(t))P\tilde{A}(t)z_1(t) \\
& +2\delta^T(z_1(t))P\tilde{A}_d(t)z_1(t-d(t)).
\end{aligned}
$$

It follows that the Lyapunov derivative corresponding to system (9.7) is given by

$$
\delta V(z_1,t) = \begin{bmatrix} P\delta(z_1(t)) \\ Pz_1(t) \\ Pz_1(t-d(t)) \end{bmatrix}^T \varTheta \begin{bmatrix} P\delta(z_1(t)) \\ Pz_1(t) \\ Pz_1(t-d(t)) \end{bmatrix} \tag{9.13}
$$

where

$$
\varTheta = \varUpsilon + \bar{G}\hat{F}(t)\bar{H} + \bar{H}^T\hat{F}^T(t)\bar{G}^T \tag{9.14}
$$

$$
\varUpsilon = \begin{bmatrix} \varUpsilon(1,1) & \tilde{A}_{11}Z & \tilde{A}_{d11}Z \\ * & \varUpsilon(2,2) & \tilde{A}_{d11}Z + \frac{1}{d_M}L \\ * & * & -J - \frac{1}{d_M}L \end{bmatrix}, \bar{G} = \begin{bmatrix} U_2^T G \\ U_2^T G_d \\ 0 \end{bmatrix}
$$

$$
\bar{H} = \begin{bmatrix} 0 \\ ZU_2^T H^T - Y^T U_1^T H^T \\ ZU_2^T H_d^T - Y^T U_1^T H_d^T \end{bmatrix}^T
$$

with

$$
\begin{aligned}
& \tilde{A}_{11} = \overline{A}_{11} - \overline{A}_{12}C, \tilde{A}_{d11} = \overline{A}_{d11} - \overline{A}_{d12}C \\
& Z = P^{-1}, J = ZQZ, L = ZRZ \\
& \varUpsilon(1,1) = (T-2)Z + d_M L \\
& \varUpsilon(2,2) = Z\tilde{A}_{11}^T + \tilde{A}_{11}Z + (d_M - d_m + 1)J - \frac{1}{d_M}L.
\end{aligned}
$$

By Lemma 9.6, $\Theta < 0$ if and only if there exists a scalar $\varepsilon > 0$ such that the following inequality is satisfied

$$\Sigma_3 = \Upsilon + \left[\varepsilon \bar{H}^T \ \varepsilon^{-1}\bar{G} \right] \begin{bmatrix} -I & W \\ W^T & -I \end{bmatrix}^{-1} \begin{bmatrix} \varepsilon \bar{H} \\ \varepsilon^{-1}\bar{G}^T \end{bmatrix} < 0. \qquad (9.15)$$

Hence, by defining $Y = CZ$ and using the Schur complement, $\Sigma_3 < 0$ can be written as $\Sigma_4 < 0$, where

$$\Sigma_4 = \begin{bmatrix} \Sigma_4(1,1) & \overline{A}_{11}Z - \overline{A}_{12}Y & \overline{A}_{d11}Z - \overline{A}_{d12}Y & 0 & \varepsilon^{-1}U_2^T G \\ * & \Sigma_4(2,2) & \Sigma_4(2,3) & \varepsilon(ZU_2^T - Y^T U_1^T)H^T & \varepsilon^{-1}U_2^T G_d \\ * & * & -J - \frac{1}{d_M}L & \varepsilon(ZU_2^T - Y^T U_1^T)H_d^T & 0 \\ * & * & * & -I & W \\ * & * & * & * & -I \end{bmatrix}$$

where

$$\Sigma_4(1,1) = (T-2)Z + d_M L$$
$$\Sigma_4(2,2) = Z\overline{A}_{11}^T - Y^T\overline{A}_{12}^T + \overline{A}_{11}Z - \overline{A}_{12}Y + (d_M - d_m + 1)J - \frac{1}{d_M}L$$
$$\Sigma_4(2,3) = \overline{A}_{d11}Z - \overline{A}_{d12}Y + \frac{1}{d_M}L.$$

Pre-multiplying and post-multiplying Σ_4 by the diagonal matrix diag$(I, I, I, \varepsilon^{-1}, \varepsilon^{-1})$, and setting $\varepsilon^{-2} = \alpha$, the inequality $\Sigma_4 < 0$ is equivalent to (9.9).

Hence, $\Theta < 0$, which also implies that there exists a sufficiently small $\xi_1 > 0$ such that

$$\begin{bmatrix} 0_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} \\ 0_{(n-m)\times(n-m)} & \xi_1 I_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} \\ 0_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} & 0_{(n-m)\times(n-m)} \end{bmatrix} + \Theta < 0.$$

It follows from (9.13) and above inequality that $\delta V(z_1(t), t) \le -\xi_1 ||Pz_1||^2 \le -\xi_1 \lambda_{min}(P)||z_1||^2$ for all $(z_1(t), t) \in R^{n-m} \times R$. Therefore, inequality (9.8) is satisfied with $\xi = \xi_1 \lambda_{min}(P) > 0$, and the reduced system (9.7) is quadratically stable with $C = YZ^{-1}$. Moreover, the sliding surface of the system (9.4) is

$$S(t) = [\ C \ \ I \ ]\, z = YZ^{-1}z_1(t) + z_2(t) = 0. \qquad (9.16)$$

The proof is completed.

Now, we let $h \to 0$, $d(t) = \tau(t)$, the delta operator systems (9.1) is simplified into the general systems as system (1) in [189]. The simplified results can be given as follows:

$$\dot{x}(t) = (A + \Delta A(t))x(t) + (A_d + \Delta A_d(t))x(t - \tau(t)) + B(u(t) + w(t))$$

$$x(t) = \varphi_1(t) \ for \ t \in [-d_M, \ 0] . \tag{9.17}$$

The delta operator systems (9.7) is simplified into

$$\begin{aligned}
&\dot{z}_1(t) = (\overline{A}_{11} + \Delta\overline{A}_{11} - \overline{A}_{12}C - \Delta\overline{A}_{12}C)z_1(t) \\
&+(\overline{A}_{d11} + \Delta\overline{A}_{d11} - \overline{A}_{d12}C - \Delta\overline{A}_{d12}C)z_1(t-\tau(t)) \\
&z_1(t) = \overline{\varphi}_1(t) \ for \ t \in [-d_M, \ 0] .
\end{aligned} \tag{9.18}$$

The following corollary can be gotten:

Corollary 9.8. *The reduced order system (9.18) is quadratically stable if there exist symmetric positive-definite matrices* $J \in R^{(n-m)\times(n-m)}$, $Z \in R^{(n-m)\times(n-m)}$, $L \in R^{(n-m)\times(n-m)}$ *and general matrix* $Y \in R^{m\times(n-m)}$, *as well as a positive scalar* $\alpha > 0$ *such that*

$$\begin{bmatrix}
\Xi_{11} & \overline{A}_{11}Z - \overline{A}_{12}Y & \overline{A}_{d11}Z - \overline{A}_{d12}Y & 0 & \alpha U_2^T G \\
* & \Xi_{22} & \Xi_{23} & ZU_2^T H^T - Y^T U_1^T H^T & \alpha U_2^T G_d \\
* & * & -J - \frac{1}{d_M}L & ZU_2^T H_d^T - Y^T U_1^T H_d^T & 0 \\
* & * & * & -\alpha I & \alpha W \\
* & * & * & * & -\alpha I
\end{bmatrix} < 0 \tag{9.19}$$

where

$$\begin{aligned}
\Xi_{11} &= d_M L - 2Z \\
\Xi_{22} &= Z\overline{A}_{11}^T - Y^T\overline{A}_{12}^T + \overline{A}_{11}Z - \overline{A}_{12}Y + (d_M - d_m + 1)J - \frac{1}{d_M}L \\
\Xi_{23} &= \overline{A}_{d11}Z - \overline{A}_{d12}Y + \frac{1}{d_M}L.
\end{aligned}$$

Proof. Considering $h \to 0$, the simplified results of Lyapunov function in delta domain can be changed to the ones in s-domain as below:

$$V(z_1, t) = V_1(z_1, t) + V_2(z_1, t) + V_3(z_1, t) + V_4(z_1, t)$$

which is positive-define for all $z_1(t) \neq 0$ and where

$$\begin{aligned}
V_1(z_1, t) &= z_1^T(t)Pz_1(t) \\
V_2(z_1, t) &= \int_{t-\tau(t)}^{t} z_1^T(s)Qz_1(s)ds \\
V_3(z_1, t) &= \int_{t-d_M}^{t-d_{m+1}} \int_s^t z_1^T(v)Qz_1(v)dvds \\
V_4(z_1, t) &= \int_{t-\tau(t)}^{t} \int_s^t \dot{z}_1^T(v)R\dot{z}_1(v)dvds.
\end{aligned}$$

Taking the derivative of $V(t)$ along the trajectory of (9.17), we can get the LMI (9.19).

In the next step, we will design a controller such that any trajectory of the closed-loop system will be driven on to the sliding surface and then maintained on it thereafter, and will be convergent to the origin. Then, without loss of generality [68], we assume that there exists a constant $q > 1$ such that

$$|z(t+\theta)| \leq q|z(t)|, -d_M \leq \theta \leq 0. \tag{9.20}$$

Thus, the result of designing of reaching motion controller is given in the following.

Theorem 9.9. *Suppose inequality (9.9) in Theorem 9.7 has solutions J, Z, L, Y and the linear sliding surface is given by (9.10). Then the trajectory of the closed-loop system (9.4) can be driven onto the sliding surface in finite time with the control*

$$\begin{aligned} u = -B_2^{-1}[&KS(t) + \epsilon \text{sgn}(S(t)) + \bar{C}(\overline{A} + \Psi G W^T H \Psi^{-1})z(t) \\ &+\bar{C}(\overline{A}_d + \Psi G_d W^T H_d \Psi^{-1})z(t-d(t)) + \text{sgn}(S(t))(N_1 + N_2)] \end{aligned} \tag{9.21}$$

where

$$\text{sgn}(S(t)) = \text{diag}\left\{\text{sgn}(s_1)\ \text{sgn}(s_2)\ \cdots\ \text{sgn}(s_m)\right\}$$

$$\text{sgn}(S(t)) = [\text{sgn}(s_1)\ \text{sgn}(s_2)\ \cdots\ \text{sgn}(s_m)]^T$$

$$\begin{aligned} B_2^T &= \begin{bmatrix} b_1\ b_2 \cdots b_m \end{bmatrix}^T \\ b_i w &= \sum_{j=1}^{m} b_{ij} w_j \leq \sum_{j=1}^{m} |b_{ij}| \overline{w}_j = N_{2i} \\ N_1 &= \begin{bmatrix} N_{11} \cdots N_{1m} \end{bmatrix}^T \\ N_2 &= \begin{bmatrix} N_{21} \cdots N_{2m} \end{bmatrix}^T \\ N_{1i} &\geq N_{2i} \end{aligned} \tag{9.22}$$

and $\bar{C} = [\ C\ \ I\ ]$, $K = \text{diag}(k_i)$, $\epsilon = \text{diag}(\epsilon_i)$, k_i and ϵ_i are positive constants.

Proof. Note that

$$\begin{aligned} S(t) &= \begin{bmatrix} C & I \end{bmatrix} z(t) = Cz_1(t) + z_2(t) \\ \delta S(t) &= \begin{bmatrix} C & I \end{bmatrix} \delta z(t). \end{aligned} \tag{9.23}$$

Then,

$$\delta S(t) = \bar{C}\left[(\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d)z(t-d(t))\right.$$

$$+\begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + w(t))].$$

Let $V_s(t) = S^T(t)S(t)$, we have

$$\begin{aligned}
\delta V_s(t) &= S^T(t)\delta S(t) + \delta S^T(t)S(t) + h\delta S^T(t)\delta S(t) \\
&= S^T(t)\bar{C}[(\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d)z(t-d(t)) \\
&+ \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + w(t))] + [(\overline{A} + \Delta\overline{A})z(t) + (\overline{A}_d + \Delta\overline{A}_d) \\
&\times z(t-d(t)) \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + w(t))]^T \bar{C}^T S(t) + h\delta S^T(t)\delta S(t).
\end{aligned} \tag{9.24}$$

It follows from Lemma 9.6 that there exist scalars β_1 and β_2 such that

$$\begin{aligned}
&S^T(t)\bar{C}\Delta\overline{A}z(t) + (\bar{C}\Delta\overline{A}z(t))^T S(t) + S^T(t)\bar{C}\Delta\overline{A}_d z(t-d(t)) \\
&+(\bar{C}\Delta\overline{A}_d z(t-d(t)))^T S(t) \\
&\le \begin{bmatrix} \beta_1^{-1}(H\Psi^{-1}z(t))^T & \beta_1 S^T(t)\bar{C}\Psi G \end{bmatrix} \begin{bmatrix} I & -W \\ -W^T & I \end{bmatrix} \begin{bmatrix} \beta_1^{-1}H\Psi^{-1}z(t) \\ \beta_1(S^T(t)\bar{C}\Psi G)^T \end{bmatrix} \\
&\quad + \begin{bmatrix} \beta_2^{-1}(H_d\Psi^{-1}z(t-d(t)))^T & \beta_2 S^T(t)\bar{C}\Psi G_d \end{bmatrix} \\
&\quad \times \begin{bmatrix} I & -W \\ -W^T & I \end{bmatrix} \begin{bmatrix} \beta_2^{-1}H_d\Psi^{-1}z(t-d(t)) \\ \beta_2(S^T(t)\bar{C}\Psi G_d)^T \end{bmatrix} \\
&= \beta_1^2 S^T(t)\bar{C}\Psi G G^T \Psi^T \bar{C}^T S(t) + \beta_1^{-2} z^T(t)\Psi^{-T}H^T H\Psi^{-1}z(t) \\
&\quad - 2S^T(t)\bar{C}\Psi G W^T H\Psi^{-1}z(t) + \beta_2^2 S^T(t)\bar{C}\Psi G_d G_d^T \Psi^T \bar{C}^T S(t) \\
&\quad + \beta_2^{-2} z^T(t-d(t))\Psi^{-T}H_d^T H_d\Psi^{-1}z(t-d(t)) \\
&\quad - 2S^T(t)\bar{C}\Psi G_d W^T H_d\Psi^{-1}z(t-d(t)) \\
&\le \beta_3 S^T(t)S(t) + \beta_4 z^T(t)z(t) + \beta_5 z^T(t-d(t))z(t-d(t)) \\
&\quad - 2S^T(t)\bar{C}\Psi G W^T H\Psi^{-1}z(t) - 2S^T(t)\bar{C}\Psi G_d W^T H_d\Psi^{-1}z(t-d(t))
\end{aligned} \tag{9.25}$$

where

$$\beta_3 = \lambda_{max}(\beta_1^2\bar{C}\Psi G G^T \Psi^T \bar{C}^T + \beta_2^2\bar{C}\Psi G_d G_d^T \Psi^T \bar{C}^T)$$

$$\beta_4 = \lambda_{max}(\beta_1^{-2}\Psi^{-T}H^T H\Psi^{-1}),\ \beta_5 = \lambda_{max}(\beta_2^{-2}\Psi^{-T}H_d^T H_d\Psi^{-1}).$$

Hence,

$$\begin{aligned}
\delta V_s(t) &\le 2S^T(t)\bar{C}[\overline{A}z(t) + \overline{A}_d z(t-d(t)) + \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + w(t))] \\
&+h\delta S^T(t)\delta S(t) + \beta_3 S^T(t)S(t) + \beta_4 z^T(t)z(t) \\
&+\beta_5 z^T(t-d(t))z(t-d(t)) - 2S^T(t)\bar{C}\Psi G W^T H\Psi^{-1}z(t) \\
&-2S^T(t)\bar{C}\Psi G_d W^T H_d\Psi^{-1}z(t-d(t)) + h\delta S^T(t)\delta S(t) \\
&\le 2S^T(t)\bar{C}[(\overline{A} + \Psi G W^T H\Psi^{-1})z(t) + (\overline{A}_d + \Psi G_d W^T H_d\Psi^{-1}) \\
&\times z(t-d(t)) + \begin{bmatrix} 0_{(n-m)\times m} \\ B_2 \end{bmatrix} (u(t) + w(t))] + \beta_3 S^T(t)S(t) + \\
&\ (\beta_4 + \beta_5 q^2)z^T(t)z(t) + h\delta S^T(t)\delta S(t).
\end{aligned} \tag{9.26}$$

Based on the control input (9.21), we have

$$\begin{aligned}\delta V_s(t) \le & -2S^T(t)KS(t) - 2\epsilon\text{sgn}(S)S(t) + \beta_3 S^T(t)S(t) + (\beta_4 + \\ & \beta_5 q^2)z^T(t)z(t) + 2S(t)(-\text{sgn}(S(t)N_1 - w(t)) + h\delta S^T(t)\delta S(t) \\ = & -\gamma S^T S(t) + z^T(t)(\bar{C}^T(\gamma I - 2K - \beta_3 I)\bar{C} + \beta_4 I + \beta_5 q^2 I)z(t) \\ & -2\epsilon\text{sgn}(S)S(t) + 2S(t)(-\text{sgn}(S(t))N_1 + B_2 w(t)) + h\delta S^T(t)\delta S(t).\end{aligned} \tag{9.27}$$

Note that in the above inequality, the term $h\delta S^T(t)\delta S(t)$ contains the uncertainties, which can be suppressed by properly selected N_1, K, ϵ. When the uncertainties are large, the sampling interval h should be selected small enough, then $h\delta S^T(t)\delta S(t)$ is small, therefore, it can be suppressed by appropriately selected parameters N_1, K, ϵ such that $\lambda_{max}(\bar{C}^T(\gamma I - 2K - \beta_3 I)\bar{C} + \beta_4 I + \beta_5 q^2 I) < -\sigma$, where γ, σ are proper positive scalars. Therefore, we can get

$$\delta V_s(t) < -\gamma S^T(t)S(t) - 2\epsilon\text{sgn}(S)S(t) \tag{9.28}$$

which implies that any trajectory of the system will be driven on to the sliding surface and then maintained on it thereafter.

When time-delay $d(t)$ is an unknown constant, the control law (9.21) is not applicable for the terms of $(\overline{A}_d + \Psi G_d W^T H_d \Psi^{-1})z(t - d(t))$ in the control input can not be obtained in practice. Note that there exists a constant q such that $\|(\overline{A}_d + \Psi G_d W^T H_d \Psi^{-1})z(t - d(t))\| \le q\lambda_{max}((\overline{A}_d + \Psi G_d W^T H_d \Psi^{-1}))\|z(t)\|$. Then we have the following corollary.

Corollary 9.10. *Suppose that time-delay $d(t)$ is unknown and inequality (9.8) have solutions J, Z, L, Y and the linear sliding surface is given by (9.10). Then the trajectory of the closed-loop system (9.4) can be driven onto the sliding surface in finite time with the control*

$$u = -B_2^{-1}[KS + \epsilon\text{sgn}(S) + \overline{C}((\overline{A} + \Psi GW^T H\Psi^{-1})z(t) + \text{sgn}(S)N_1]. \tag{9.29}$$

9.4 Numerical Example

In this section, the first numerical example will show the characteristic difference between discrete-time system and delta operator system in sampling continuous-time system.

Example 9.11. Consider a continuous-time system in s-domain:

$$\dot{x}(t) = \begin{bmatrix} -1 & 0 \\ 1 & -2 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) \tag{9.30}$$

$$y(t) = \begin{bmatrix} 1 & 1 \end{bmatrix} x(t). \tag{9.31}$$

By using shift operator and delta operator in sampling the above continuous-time system, respectively. We get the relevant different discrete-time system in z-domain and δ-domain. When $h = 1$, there exist

$$x((k+1)h) = \begin{bmatrix} 0.3679 & 0 \\ 0.2325 & 0.1353 \end{bmatrix} x(kh) + \begin{bmatrix} 0 \\ 0.4323 \end{bmatrix} u(kh)$$
$$\delta x(kh) = \begin{bmatrix} -0.6321 & 0 \\ 0.2325 & -0.8647 \end{bmatrix} x(kh) + \begin{bmatrix} 0 \\ 0.4323 \end{bmatrix} u(kh).$$

When $h = 0.55$, there exist

$$x((k+1)h) = \begin{bmatrix} 0.5769 & 0 \\ 0.2441 & 0.3329 \end{bmatrix} x(kh) + \begin{bmatrix} 0 \\ 0.3336 \end{bmatrix} u(kh)$$
$$\delta x(kh) = \begin{bmatrix} -0.7692 & 0 \\ 0.4438 & -1.2130 \end{bmatrix} x(kh) + \begin{bmatrix} 0 \\ 0.6065 \end{bmatrix} u(kh).$$

When $h = 0.1$, there exist

$$x((k+1)h) = \begin{bmatrix} 0.9048 & 0 \\ 0.0861 & 0.8187 \end{bmatrix} x(kh) + \begin{bmatrix} 0 \\ 0.0906 \end{bmatrix} u(kh)$$
$$\delta x(kh) = \begin{bmatrix} -0.9516 & 0 \\ 0.8611 & -1.8127 \end{bmatrix} x(kh) + \begin{bmatrix} 0 \\ 0.9063 \end{bmatrix} u(kh).$$

From above results, it is easy to see that why the virtue of delta operator systems in sampling continuous-time systems will come out when the sampling is fast. Two output curve graphs of the above systems with different sampling periods are given as follows:

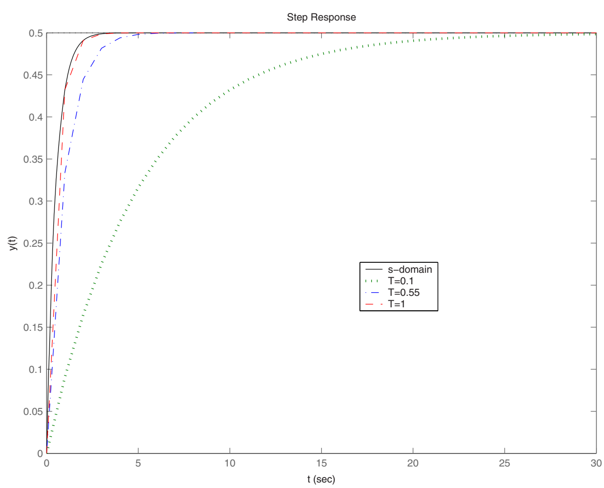

Fig. 9.1 The trajectories of the output in z-domain.

The following illustrative example is given for testing the design method developed in this chapter.

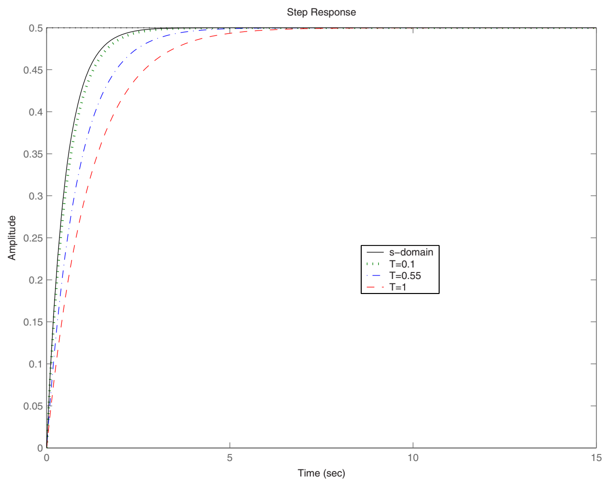

Fig. 9.2 The trajectories of the output in δ-domain.

Example 9.12. The proposed method will be applied to design a robust state feedback controller for truck trailer system with time-delay proposed in [17] and [173]. The time-delay model with uncertainties is given by:

$$\begin{aligned}\dot{x}_1(t) &= M[-ax_1(t) - (1-a)x_1(t-d(t))] + q[u(t)+w(t)]\\ \dot{x}_2(t) &= M[-ax_1(t) + (1-a)x_1(t-d(t))]\\ \dot{x}_3(t) &= qlsin\{x_2(t) + (M/2)[x_1(t) + (1-a)x_1(t-d(t))]\}\end{aligned}$$

where $M = \frac{v\bar{t}}{(L+\Delta L(t))t_0}$, $q = \frac{v\bar{t}}{lt_0}$ with $a = 0.7$, $v = -1.0$, $\bar{t} = 2.0$, $t_0 = 0.5$, $L = 5.5$, $l = 2.8$ and $-0.2519 \leq \Delta L \leq 0.2891$, $w(t) = 0.001sin(t)$. When $x_2(t) + a\frac{v\bar{t}}{2(L+\Delta L(t))}x_1(t) + (1-a)\frac{v\bar{t}}{2(L+\Delta L(t))}x_1(t-d(t))$ is about zero, and it is easy to see that the truck trailer system can be sampled to the class of delta operator system when $h = 0.01$, $d_m = 0.1$ as follows:

$$\begin{aligned}\delta x(t) &= \left(A + GF(t)\left[I - WF(t)\right]^{-1}H\right)x(t)\\ &\quad + \left(A_d + G_dF(t)\left[I - WF(t)\right]^{-1}H_d\right)x(t-d(t)) + B(u(t)+\omega(t)) \quad (9.32)\end{aligned}$$

with

$$A = \begin{bmatrix} 0.5109 & 0 & 0\\ -0.5109 & 0 & 0\\ 0.5212 & -4 & 0\end{bmatrix},\ B = \begin{bmatrix} -1.4338\\ 0.0052\\ -0.0053\end{bmatrix},\ A_d = [a_{d3\times1}\ 0_{3\times2}]$$

$$A_d = [0.219\ \ -0.219\ 0.2234]^T,\ F(t) = \text{diag}\{sin(t)\ cos(t)\ sin(t)\}$$

$$G = G_d = 0.05\cdot I_{3\times3},\ W = 0.01\cdot I_{3\times3},\ H = [h_{3\times1}\ 0_{3\times2}],\ H_d = [h_{d3\times1}\ 0_{3\times2}]$$

$$h = [0.5091 \ -0.5091 \ -0.5091]^T, \ h_d = [0.2182 \ -0.2182 \ -0.2182]^T.$$

Taking $\Psi = \begin{bmatrix} 0.0036 & 1 & 0 \\ -0.0037 & 0 & 1 \\ -1 & 0.0036 & -0.0037 \end{bmatrix}$, such that $\Psi B = \begin{bmatrix} 0 \\ 0 \\ 1.4338 \end{bmatrix}$, then LMI (9.9) has feasible solutions:

$$Z = 10^5 \times \begin{bmatrix} 0.0008 & 0.0145 \\ 0.0145 & 2.5473 \end{bmatrix}, \ L = 10^4 \times \begin{bmatrix} 0.0010 & 0.0185 \\ 0.0185 & 3.0645 \end{bmatrix}$$
$$J = \begin{bmatrix} 0.0008 & 0.0267 \\ 0.0267 & 2.0168 \end{bmatrix}, \ Y = \begin{bmatrix} 12.7483 & 761.3555 \end{bmatrix}, \ \alpha = 692.5605.$$

The largest time delay is $d_M = 13.19$. It follows from Theorem 9.7 that $C = [0.1234, 0.0023]$, the linear sliding surface is $S(t) = [0.1234, 0.0023, 1]\, z = 0$.

From Theorem 9.9, the reaching control law can be taken as follows:

$$\begin{aligned} u(t) = &-0.6974 \times [KS(t) + \epsilon \text{sgn}(S(t)) + \begin{bmatrix} 0.0035 & 0.0021 & 0.5766 \end{bmatrix} x(t) \\ &+\text{sgn}(S(t))(N_1)] \end{aligned}$$

where N_1, the parameter K and ϵ can be tuned to reduce the chattering on the sliding surface. Fig. 9.3, Fig. 9.5 and Fig. 9.5 are simulation results when choosing $x_1(0) = x_2(0) = x_3(0) = 1$, $S(0) = -0.87$, $u(0) = 2.85$, $K = 0.4$, $\epsilon = 0.02$ and $N_1 = 0$. Obviously the system is asymptotically stable and the sliding motion trends to the origin in finite time in spite of time-delay and uncertainties.

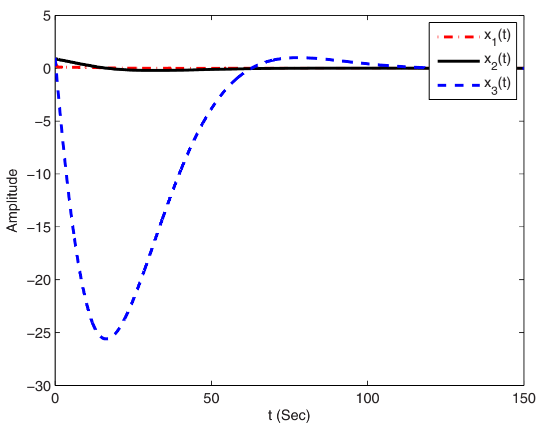

Fig. 9.3 States (x)

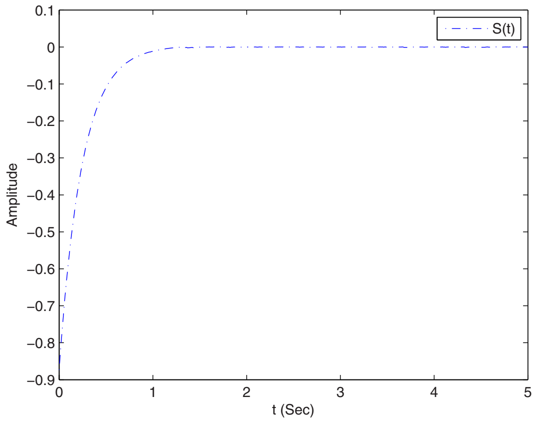

Fig. 9.4 Sliding surface (S)

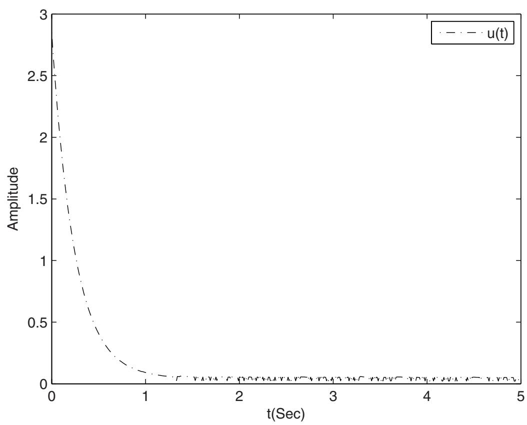

Fig. 9.5 Control input (u)

9.5 Conclusion

In this chapter, the problem of designing robust sliding surfaces based on delta operator for a class of uncertain time-delay systems has been considered in which no matching condition is assumed for the state uncertainties. In terms of LMI, sufficient condition is derived for the existence of a linear sliding surface guaranteeing quadratic stability of the reduced-order equivalent system restricted to the sliding surface. A reaching motion controller is proposed for uncertain time-delay systems by the reaching law. Both the sliding motion and the reaching motion are robust against the mismatched uncertainties and matched external disturbance. The simulation results show that the proposed methods are amenable and generalize previous results available in the literature to date.

Chapter 10
ADRC for Uncertain Systems with Time-Delay

10.1 Introduction

The majority of control systems are operated by proportional-integral- derivative (PID) controllers, which dates back to 1922 ([135], [110], [220]) well before classical and modern control theory were born. Today, much work has been done related to systems with time-delays, see for example, ([188], [189], [154], [126]) and references therein. It should be noted that PID still remains as the preferred controller in over 95% of industrial applications, as generally a good performance can be achieved [22], [162]. Given the wide spread industrial use of PID controllers, it is clear that even a small percentage improvement in the design, a PID controller could have a tremendous impact on practical applications. Despite this, it is unfortunate that currently there is not much theory dealing with PID designs. Indeed, most of the industrial PID designs are carried out using only empirical techniques and the mathematically elegant and sophisticated theories developed in the context of modern optimal control cannot be applied to them. This represents a significant gap between the theory and practice of automatic control.

In the chemical industries, many processes can be modelled as a class of multi-variable time-delay systems. Several control schemes have been proposed to deal with this kind of systems (See for example [90] and references therein). However, the problem of how to compensate for model uncertainties, cross couplings and disturbances existing in the systems has not been fully investigated yet. This problem is important and challenging in both theory and practice, which provides the motivation for this study. In this chapter, we propose a new methodology of control for multi-variable systems with time-delay. It is based upon an unique Active Disturbance Rejection Control (ADRC) concept [194]. In this approach, the systems with time-delay in the input are viewed as higher order systems without time-delay in the input, and the error resulting from unmodelled dynamics and disturbances are estimated using an extended state observer (ESO) and compensated for

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 137–149.
springerlink.com

during each sampling period. Since uncertainties and disturbances are estimated and canceled via the ESO, there is no need for integral control. This method was developed in [57]. The proposed ADRC control system consists of the TD, the ESO and a nonlinear PD controller. It is designed under the assumption of high degree of model uncertainties. The controller is designed to be inherently robust against plant variations. Once it is set up for the problem within a predetermined range of variation in system variables, no tuning is needed for start up, or to compensate for changes in the system dynamics and disturbance. This method, because of its robustness and disturbance rejection capabilities, is particularly suitable for control of systems with time-delay [215], [124]. Since parts of the idea of ADRC originate from sliding mode control [73], it is also included in Part II.

The chapter is organized as follows. Section 10.2 gives the problem formulation and some preliminaries. In section 10.3, the concept of tracking differentiator (TD) and extended state observer (ESO) and nonlinear feedback are given. Numerical simulations are presented in Section 10.4 and some conclusion remarks are given in Section 10.5.

10.2 Problem Formulation

Consider an uncertain multi-variable system with time-delay

$$y = \begin{bmatrix} \frac{k_{11}e^{-L_{11}s}}{t_{11}s+1} & \frac{k_{12}e^{-L_{12}s}}{t_{12}s+1} \\ \frac{k_{21}e^{-L_{21}s}}{t_{21}s+1} & \frac{k_{22}e^{-L_{22}s}}{t_{22}s+1} \end{bmatrix} u \tag{10.1}$$

where $y \in R^2, u \in R^2$, while $k_{11} \in [k_{11}^-, k_{11}^+]$, $k_{12} \in [k_{12}^-, k_{12}^+]$, $k_{21} \in [a_{21}^-, k_{21}^+]$ and $k_{22} \in [k_{22}^-, k_{22}^+]$ are controller gains; $t_{11} \in [t_{11}^-, t_{11}^+]$, $t_{12} \in [t_{12}^-, t_{12}^+]$, $t_{21} \in [t_{21}^-, t_{21}^+]$ and $t_{22} \in [t_{22}^-, t_{22}^+]$ are positive time constants.

Let

$$a_{11} = -\frac{1}{t_{11}}, a_{12} = -\frac{1}{t_{12}}, a_{21} = -\frac{1}{t_{21}}, a_{22} = -\frac{1}{t_{22}}$$

$$b_{11} = \frac{k_{11}}{t_{11}}, b_{12} = \frac{k_{12}}{t_{12}}, b_{21} = \frac{k_{21}}{t_{21}}, b_{22} = \frac{k_{22}}{t_{22}}$$

then

$$a_{11} \in [a_{11}^-, a_{11}^+], a_{12} \in [a_{12}^-, a_{12}^+], a_{21} \in [a_{21}^-, a_{21}^+], a_{22} \in [a_{22}^-, a_{22}^+]$$

$$b_{11} \in [b_{11}^-, b_{11}^+], b_{12} \in [b_{12}^-, b_{12}^+], b_{21} \in [b_{21}^-, b_{21}^+], b_{22} \in [b_{22}^-, b_{22}^+]$$

where a_{ij}^- and a_{ij}^+, b_{ij}^- and b_{ij}^+ are obtained appropriately according to the value of k_{ij}^-, k_{ij}^+, t_{ij}^- and t_{ij}^+, $i, j = 1, 2$, respectively.

Remark 10.1. This model is well known in chemical control systems, such as a class of chemical reactors, distillation columns, and fluidized bed combustor [181]. There exist a lot of methods which have been applied to solve this kind of problem, for example, Smith predictor method [1], decentralized control [150], PI control [91], model reference adaptive control scheme [90], model predictive control [164].

In order to study this kind of system, some transformations and approximations will be introduced.

First, write the transfer matrix (10.1) in the state space form

$$\begin{cases} \dot{x} = Ax + B(u) \\ y = Cx \end{cases} \tag{10.2}$$

where

$$A = \begin{pmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{12} & 0 & 0 \\ 0 & 0 & a_{21} & 0 \\ 0 & 0 & 0 & a_{22} \end{pmatrix}, B(u) = \begin{pmatrix} b_{11}u_1(t - L_{11}) \\ b_{12}u_2(t - L_{12}) \\ b_{21}u_1(t - L_{21}) \\ b_{22}u_2(t - L_{22}) \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}. \tag{10.3}$$

Using state transformation $x = Tz$, where

$$T = \begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{10.4}$$

System (10.2) can be rewritten as

$$\begin{cases} \dot{z} = \tilde{A}z + \tilde{B}(u) \\ y = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} \end{cases} \tag{10.5}$$

where

$$\tilde{A} = \begin{pmatrix} a_{11} & 0 & -a_{11} + a_{12} & 0 \\ 0 & a_{21} & 0 & -a_{21} + a_{22} \\ 0 & 0 & a_{12} & 0 \\ 0 & 0 & 0 & a_{22} \end{pmatrix}$$

$$\tilde{B}(u) = \begin{pmatrix} b_{11}u_1(t-L_{11}) + b_{12}u_2(t-L_{12}) \\ b_{21}u_1(t-L_{21}) + b_{22}u_2(t-L_{22}) \\ b_{12}u_2(t-L_{12}) \\ b_{22}u_2(t-L_{22}) \end{pmatrix}. \tag{10.6}$$

Remark 10.2. Note that model (10.2)-(10.3) is not an unique presentation for system (10.1) and there are other transformations for (10.2). The transformation (10.4) is applied here so that the output of the system is separated and the techniques proposed in this chapter can be applied appropriately, that is, $y_1 = z_1$ and $y_2 = z_2$. Then there is no cross terms between y_1 and y_2. Thus, control inputs u_1 and u_2 will be designed such that outputs y_1 and y_2 can satisfactorily track the set points.

Now, system (10.5) can be treated as two subsystems:

$$\begin{cases} \dot{z}_1 = \overline{a}_{11}z_1 + ff_1 + \overline{b}_{11}u_1(t-L_{11}) + \overline{b}_{12}u_2(t-L_{12}) \\ y_1 = z_1 \end{cases} \tag{10.7}$$

and

$$\begin{cases} \dot{z}_2 = \overline{a}_{21}z_2 + ff_2 + \overline{b}_{21}u_1(t-L_{21}) + \overline{b}_{22}u_2(t-L_{22}) \\ y_2 = z_2 \end{cases} \tag{10.8}$$

where

$\overline{a}_{11} = \frac{a_{11}^- + a_{11}^+}{2}$, $\overline{a}_{21} = \frac{a_{21}^- + a_{21}^+}{2}$, $\overline{b}_{11} = \frac{b_{11}^- + b_{11}^+}{2}$, $\overline{b}_{12} = \frac{b_{12}^- + b_{12}^+}{2}$, $\overline{b}_{21} = \frac{b_{21}^- + b_{21}^+}{2}$, $\overline{b}_{22} = \frac{b_{22}^- + b_{22}^+}{2}$, $ff_1 = (a_{11} - \overline{a}_{11})z_1 + (-a_{11} + a_{12})z_3 + (b_{11} - \overline{b}_{11})u_1(t-L_{11}) + (b_{12} - \overline{b}_{12})u_2(t-L_{12})$, $ff_2 = (a_{21} - \overline{a}_{21})z_2 + (-a_{21} + a_{22})z_4 + (b_{21} - \overline{b}_{21})u_1(t-L_{21}) + (b_{22} - \overline{b}_{22})u_2(t-L_{22})$

To explain the idea in this chapter clearly, the following first-order approximation for the dead time is used

$$e^{-\tau s} \approx \frac{1 - T_N s}{1 + T_D s} = \frac{1 - \mu\tau s}{1 + (1-\mu)\tau s} \tag{10.9}$$

where $\mu = T_N/\tau \in [0, 1]$ is a provisionally parameter left free.

Remark 10.3. The term $e^{-\tau s}$ is not polynomial and cannot be realized in finite dimensions. For small signal analysis, it can be approximated by a polynomial transfer function. Generally, there are several ways to do this approximation, such as Bessel functions, Padè approximations, Laguerre polynomials and hyperbolic functions. Among these alternatives, Padè approximations are more accurate. Here, the first-order Padè approximation is used, the approximation error can be viewed as modeling errors. Since it is assumed that $a_{12} = -\frac{1}{t_{12}} < 0$, $a_{22} = -\frac{1}{t_{22}} < 0$, it follows from the equations in (10.5) that

$$\dot{z}_3 = a_{12}z_3 + b_{12}u_2(t-L_{12})$$
$$\dot{z}_4 = a_{22}z_4 + b_{22}u_2(t-L_{22})$$

and the states z_3 and z_4 are bounded while input u_2 is bounded, therefore, z_3 and z_4 can be viewed as disturbances, then, the modeling errors and disturbances are lumped together, which will be observed by the ESO, the definition of ESO will be given in Section 10.3.

Note that subsystems (10.7)-(10.8) can be written as

$$\begin{cases} sz_1 = \overline{a}_{11}z_1(s) + ff_1(s) + \overline{b}_{11}e^{-L_{11}s}u_1 + \overline{b}_{12}e^{-L_{12}s}u_2 \\ y_1 = z_1 \end{cases} \tag{10.10}$$

and

$$\begin{cases} sz_2 = \overline{a}_{21}z_2(s) + ff_2(s) + \overline{b}_{21}e^{-L_{21}s}u_1 + \overline{b}_{22}e^{-L_{22}s}u_2 \\ y_2 = z_2 \end{cases}. \tag{10.11}$$

According to (10.9), subsystem (10.10)-(10.11) can be taken as

$$\begin{cases} sz_1 = \overline{a}_{11}z_1(s) + \underline{ff}_1(s) + \overline{b}_{11}\frac{1-\mu_{11}L_{11}s}{1+(1-\mu_{11})L_{11}s}u_1 + \overline{b}_{12}\frac{1-\mu_{12}L_{12}s}{1+(1-\mu_{12})L_{12}s}u_2 \\ y_1 = z_1 \end{cases} \tag{10.12}$$

and

$$\begin{cases} sz_2 = \overline{a}_{21}z_2(s) + \underline{ff}_2(s) + \overline{b}_{21}\frac{1-\mu_{21}L_{21}s}{1+(1-\mu_{21})L_{21}s}u_1 + \overline{b}_{22}\frac{1-\mu_{22}L_{22}s}{1+(1-\mu_{22})L_{22}s}u_2 \\ y_2 = z_2 \end{cases} \tag{10.13}$$

where $\mu_{ij} \in [0,1], i,j = 1,2$ are free parameters.

$$\begin{aligned} \underline{ff}_1(s) = &\ ff_1(s) + \overline{b}_{11}e^{-L_{11}s}u_1 + \overline{b}_{12}e^{-L_{12}s}u_2 - (\overline{b}_{11}\frac{1-\mu_{11}L_{11}s}{1+(1-\mu_{11})L_{11}s}u_1 \\ &+\overline{b}_{12}\frac{1-\mu_{12}L_{12}s}{1+(1-\mu_{12})L_{12}s}u_2) \end{aligned}$$

$$\begin{aligned} \underline{ff}_2(s) = &\ ff_2(s) + \overline{b}_{21}e^{-L_{21}s}u_1 + \overline{b}_{22}e^{-L_{22}s}u_2 - (\overline{b}_{21}\frac{1-\mu_{21}L_{21}s}{1+(1-\mu_{21})L_{21}s}u_1 \\ &+\overline{b}_{22}\frac{1-\mu_{22}L_{22}s}{1+(1-\mu_{22})L_{22}s}u_2). \end{aligned}$$

Furthermore, subsystem (10.12)-(10.13) can be viewed as follows

$$\begin{cases} sz_1 = \overline{a}_{11}z_1(s) + \overline{ff}_1(s) + b_{01}\frac{1}{s+\mu_1}U_1 \\ y_1 = z_1 \end{cases} \tag{10.14}$$

and

$$\begin{cases} sz_2 = \overline{a}_{21}z_2(s) + \overline{ff}_2(s) + b_{02}\frac{1}{s+\mu_2}U_2 \\ y_2 = z_2 \end{cases} \tag{10.15}$$

where

$$\overline{ff}_1(s) = \underline{ff}_1(s) + \overline{b}_{11}\frac{1-\mu_{11}L_{11}s}{1+(1-\mu_{11})L_{11}s}u_1 + \overline{b}_{12}\frac{1-\mu_{12}L_{12}s}{1+(1-\mu_{12})L_{12}s}u_2$$
$$-b_{01}\frac{1}{s+\mu_1}U_1$$
$$\overline{ff}_2(s) = \underline{ff}_2(s) + \overline{b}_{21}\frac{1-\mu_{21}L_{21}s}{1+(1-\mu_{21})L_{21}s}u_1 + \overline{b}_{22}\frac{1-\mu_{22}L_{22}s}{1+(1-\mu_{22})L_{22}s}u_2$$
$$-b_{02}\frac{1}{s+\mu_2}U_2.$$

Remark 10.4. Note that U_1 and U_2 are virtual control inputs, which will help us to design the actual controls u_1 and u_2. $\mu_i > 0$, $i = 1, 2$ are free parameters introduced to interpret the principle of the method used in this chapter they are not used in the rest of this chapter. b_{01} and b_{02} are parameters to be determined. Also, (10.14)-(10.15) can be regarded as simplified form of (10.12)-(10.13), in which the modeling errors have been included in $\overline{ff}_1(s)$ and $\overline{ff}_2(s)$ as uncertainties. This kind of uncertainties will be observed by the ESO, and then will be actively compensated, consequently, subsystems (10.14) and (10.15) are decoupled dynamically. The resulting controller can be easily designed.

In order to simplify the structure of the observer of subsystems (10.14)-(10.15), the following transformations are used:

$$\begin{cases} s^2 z_1 = \overline{a}_{11} z_1(s)s + \tilde{ff}_1(s) + b_{01}U_1 \\ y_1 = z_1 \end{cases} \tag{10.16}$$

and

$$\begin{cases} s^2 z_2 = \overline{a}_{21} z_2(s)s + \tilde{ff}_2(s) + b_{02}U_2 \\ y_2 = z_2 \end{cases} \tag{10.17}$$

where $\tilde{ff}_1(s) = (\mu_1 + s)\overline{ff}_1(s) + \mu_1\overline{a}_{11}z_1(s) - \mu_1 z_1(s)s$, $\tilde{ff}_2(s) = (\mu_2 + s)\overline{ff}_2(s) + \mu_2\overline{a}_{21}z_2(s) - \mu_2 z_2(s)s$

Let $p_1 = z_1, p_2 = sz_1, p_3 = z_2, p_4 = sz_2$, then subsystems (10.16)-(10.17) can be written as

$$\begin{cases} \dot{p}_1 = p_2 \\ \dot{p}_2 = \overline{a}_{11}p_2 + L_a^{-1}(\tilde{ff}_1(s)) + b_{01}U_1 \\ y_1 = p_1 \end{cases} \tag{10.18}$$

and

$$\begin{cases} \dot{p}_3 = p_4 \\ \dot{p}_4 = \overline{a}_{21}p_4 + L_a^{-1}(\tilde{ff}_2(s)) + b_{02}U_2 \\ y_2 = p_3 \end{cases} \tag{10.19}$$

where $L_a^{-1}(.)$ denotes the inverse Laplace transformation.

The uncertain dynamics, $L_a^{-1}(\tilde{ff}_1(s))$ and $L_a^{-1}(\tilde{ff}_2(s))$ are difficult to model accurately, which therefore limits the performance of model-based control methods. In addition to these modeling errors, external disturbances are inevitable in real situations and these disturbances also degrade the control performance. Therefore, the controller should have a robust capability to achieve this objective under the above circumstances. To achieve this robustness, several approaches have been proposed, such as adaptive control, and robust control. In adaptive control approaches, where there is no knowledge of the bounding function, the control must be designed to measure the size of uncertainties while compensating for them. This means that the unknown load condition and system parameters are identified on-line. This approach has a heavy computation and is limited to those systems where uncertainties are structured. In the meantime, the so-called robust control technique has been proposed which has the capability to guarantee the stability and a prescribed performance for the control systems with uncertainties. Its design requires some knowledge about norm-bounded functions of the largest possible size of the uncertainties and disturbances. While uncertainties being bounded ensures that a stabilizing control (if it exists) will be of finite magnitude, but in most of cases, it is difficult to estimate the proper norm bound, thus the usual robust control method for systems with uncertainties of this kind can often result in a conservative design.

In the work of [57], the uncertainties are estimated using the extended state observer (ESO), the control signals are then used to actively compensate for its effect and the control problem is thus reduced to a simple one.

The objective in this chapter is to design a control law such that the following goals can be achieved:

a) Robust stability;
b) Set-point following, i.e, the system asymptotically tracks stepwise set point changes and minimize the impact of disturbances;
c) The couplings of the two closed-loop subsystems are to be reduced as far as possible.

In order to understand the key idea of ADRC, which is composed of TD, ESO and nonlinear feedback controller, the design philosophy and the main components of the new paradigm are introduced. As for a class of multivariable time-delay systems, a special control structure is also proposed in the next section. More details of the theoretical results and associated algorithms can be found in [57]. A brief introduction is given below.

10.3 Concepts of TD, ESO and Nonlinear Feedback Controller

In the work of [71], [72], two methods are presented to improve the performance of the closed-loop systems to be designed: the tracking differentiator (TD) and the extended state observer (ESO). Furthermore, the nonlinear feedback controller is composed of a PD controller. They can be described as follows:

First, a transient profile is used. Engineers have long realized that set point q_r, also known as reference, shouldn't be changed suddenly. Instead, it should be changed gradually so that the output of the plant can follow closely. This desired set point changing trajectory, q_r , is denoted as *transient profile*. The following tracking differentiator (TD) is used to arrange a transient profile.

I. The tracking differentiator (TD).

a. TD for non-discrete signal.

$$\begin{cases} \dot{\eta}_1 = \eta_2 \\ \dot{\eta}_2 = -r\text{sgn}(\eta_1 - q_r + \frac{\eta_2|\eta_2|}{2r}) \end{cases} \tag{10.20}$$

where sgn(·) is defined as

$$\text{sgn}(x) = \begin{cases} 1, x > 0 \\ 0, x = 0 \\ -1, x < 0 \end{cases} \tag{10.21}$$

and r represents the maximum actuation available in the system.

b. TD for discrete signal.

$$\begin{cases} v_1(t+h) = v_1(t) + hv_2(t) \\ v_2(t+h) = v_2(t) + hfst(v_1 - q_r(t), v_2(t), r, h_1) \end{cases} \tag{10.22}$$

where v_1 and v_2 are the state variables, $q_r(t)$ is the input signal, h is the step size, h_1 is a tuning parameter and function$fst(v_1, v_2, r, h)$ is defined as:

$$d = rh,\ d_0 = dh,\ g = v_1 + hv_2 \tag{10.23}$$

$$a_0 = \sqrt{d^2 + 8r|g|} \tag{10.24}$$

$$a = \begin{cases} v_2 + \frac{g}{h}, |g| < d_0 \\ v_2 + \frac{\text{sgn}(g)(a_0 - d)}{2}, |g| \geq d_0 \end{cases} \tag{10.25}$$

$$fst = \begin{cases} -r\frac{a}{d}, |a| \leq d \\ -r\text{sgn}(a), |a| > d \end{cases}. \tag{10.26}$$

The impact of the TD is profound. Firstly, as a noise filter, it blocks any part of the signal with an acceleration exceeding r. Secondly, TD has a very desirable frequency filter response characteristic. In particular, it has a much smaller phase shift compared to linear filters, while maintaining an extremely flat gain over the bandwidth. Finally, perhaps the most important role of TD is its ability to obtain the derivative of a noisy signal with a good signal to noise ratio. It is well known that a pure differentiator is not physically implementable. The error signal is often not differentiable in practice due to the noises in the feedback and the discontinuities in the reference signal. However, a discrete time realization of the TD can improve the numerical properties and can avoid high frequency oscillations. Further explanations of this can be found in [56].

II.The extended state observer (ESO).

The ESO was first proposed [71] for on-line estimating the total dynamics, which lumps the internal nonlinear dynamics and the external disturbance. Two three-order ESOs for system (10.7) and (10.8) are proposed in the following:

$$\begin{cases} \epsilon_1 = w_1 - y \\ \dot{w}_1 = w_2 - \beta_{01}\epsilon_1 \\ \dot{w}_2 = w_3 + f_0(w_1, w_2) - \beta_{02}Fal(\epsilon_1, \alpha_1, \delta) + b_0 U \\ \dot{w}_3 = -\beta_{03}Fal(\epsilon_1, \alpha_2, \delta) \end{cases} \tag{10.27}$$

where $\beta_{01}, \beta_{01}, \beta_{03}$, b_0, α_1, α_2 and δ are constants to be determined, $w_1 = \begin{bmatrix} w_{11} \\ w_{12} \end{bmatrix}, w_2 = \begin{bmatrix} w_{21} \\ w_{22} \end{bmatrix}, w_3 = \begin{bmatrix} w_{31} \\ w_{32} \end{bmatrix} \in R^2$ are the states of the observer . $f_0(w_1, w_2) = \begin{bmatrix} \overline{a}_{11}w_{21} \\ \overline{a}_{22}w_{32} \end{bmatrix}$, $b_0 = \begin{bmatrix} b_{01} & 0 \\ 0 & b_{02} \end{bmatrix}$, $U = \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}$, $Fal(\cdot) = \begin{bmatrix} fal(\cdot) \\ fal(\cdot) \end{bmatrix}$

$fal(\cdot)$ is defined as

$$fal(e, \alpha, \delta) = \begin{cases} |\epsilon|^{\alpha}\text{sgn}(e), & |e| > \delta \\ \frac{e}{\delta^{1-\alpha}}, & |e| \le \delta \end{cases}. \tag{10.28}$$

Note that the third formula w_3 in (10.27) is the most important. It shows that w_3 can estimate (or track) the total action of the uncertain models and the external disturbances or the real-time action of the system disturbances. Since w_3 is the estimation for the total action of the unknown disturbances, in the feedback, w_3 is used to compensate for the disturbances. So the resulting systems after compensating for disturbance will be in the type of second order cascade integrators, which will be easy to control.

III. The active disturbance rejection control method (ADRC).

Since uncertainties are estimated and canceled via the ESO, there is no need for integral control. Applying the ESO given above in (10.27), we propose a nonlinear controller with a feedback block containing a filter

$$\begin{aligned} e_1 &= v_1 - w_1 \\ e_2 &= v_2 - w_2 \\ u_0 &= \beta_1 fal(e_1, \alpha_{c1}, \delta_{L_2}) + \beta_2 fal(e_2, \alpha_{c2}, \delta_c) \\ u &= u_0 - b_0^{-1} w_3 \\ U &= -a_f(U - u) \end{aligned} \tag{10.29}$$

where $a_f > 0$ is filter gain, $\beta_1 = \begin{bmatrix} \beta_{11} \\ \beta_{12} \end{bmatrix}$, $\beta_2 = \begin{bmatrix} \beta_{21} \\ \beta_{22} \end{bmatrix}$.

IV. Control input u_1 and u_2

From (10.29), the virtual control input U can be obtain, then the control input $u_1(t)$ and $u_2(t)$ can be designed as follows:

$$u_1(t) = \frac{\bar{b}_{22}U_1 - \bar{b}_{12}U_2}{\bar{b}_{11}\bar{b}_{22} - \bar{b}_{12}\bar{b}_{21}}$$
$$u_2(t) = \frac{\bar{b}_{11}U_2 - \bar{b}_{21}U_1}{\bar{b}_{11}\bar{b}_{22} - \bar{b}_{12}\bar{b}_{21}}. \tag{10.30}$$

To be more precisely, ADRC control scheme is illustrated by the following figure:

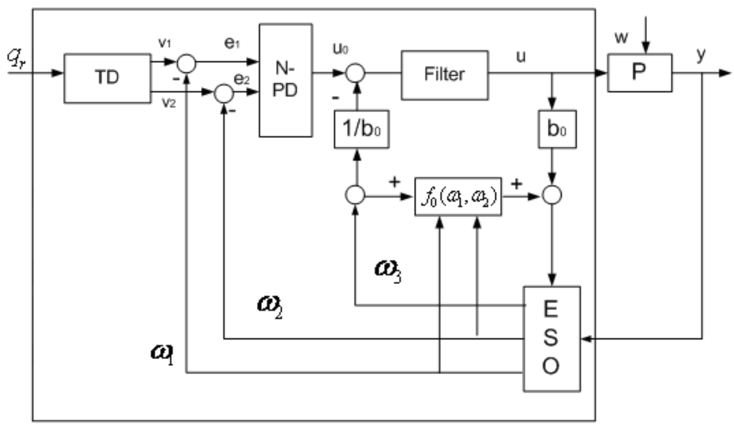

Fig. 10.1 The structure of ADRC control scheme

Remark 10.5. The filter in (10.29) is used to obtain a control of better quality. The stability analysis of the nonlinear continuous extended state observer can be found in [82], [83]. This model can also be extended to multi input and multi output cases with dimension larger than two.

10.4 Example

Example 10.6. In order to demonstrate the performance of the proposed control strategy, we borrow the following example which has been considered in [181].

$$\begin{pmatrix} y_1(s) \\ y_2(s) \end{pmatrix} = \begin{pmatrix} \frac{12.8e^{-2s}}{16.7s+1} & \frac{-18.9e^{-4s}}{21.0s+1} \\ \frac{6.6e^{-10s}}{10.9s+1} & \frac{-19.4e^{-4s}}{14.4s+1} \end{pmatrix} \begin{pmatrix} u_1(s) \\ u_2(s) \end{pmatrix}. \tag{10.31}$$

Using the control scheme proposed above, with reference $q_r(1)=2, q_r(2)=1$

1) Transient process parameters: $h = 0.01$, $r = \begin{pmatrix} 60 \\ 60 \end{pmatrix}$, $h_1 = \begin{pmatrix} 1.5h \\ 1.5h \end{pmatrix}$;
2) The ESO parameters: $\beta_{01} = 110, \beta_{02} = 330, \beta_{03} = 100$, $\alpha_1 = 0.5, \alpha_2 = 0.25$, $\delta = 0.05, b_0 = 50$;
3) Nonlinear feedback parameters: $\beta_{11} = 0.01, \beta_{12} = 0.18$, $\beta_{21} = 0.01$, $\beta_{22} = 0.18$, $\alpha_{c1} = 0.75, \alpha_{c2} = 1.0$, $\delta_c = 0.2$.
4) Filter parameter: $a_f = 0.8$.

1. As for the nominal system with $t_{11} = 16.7$, $t_{12} = 21$, $t_{21} = 10.9$, $t_{22} = 14.4$, $k_{11} = 12.8$, $k_{12} = -18.9$, $k_{21} = 6.6$, $k_{22} = -19.4$, $L_{11} = 2$, $L_{12} = 4$, $L_{21} = 10$, $L_{22} = 4$, the results of a simulation study is as follows:

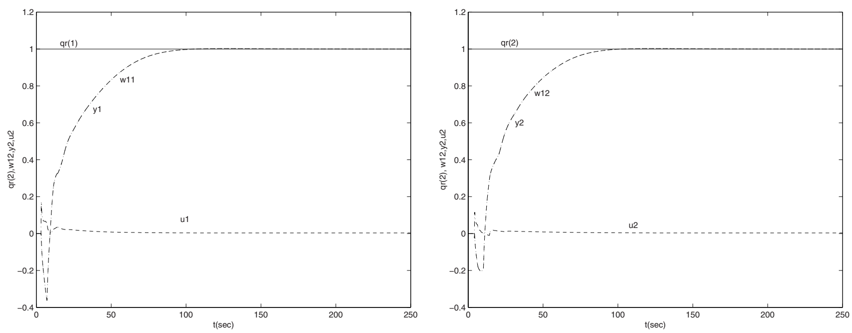

Fig. 10.2 Step response of y_1, y_2, w_{11}, w_{12} and control input u_1, u_2

2. 100% change in time delay, i.e, $L_{11} = 4$, $L_{12} = 8$, $L_{21} = 20$ and $L_{22} = 8$, the results of simulation are

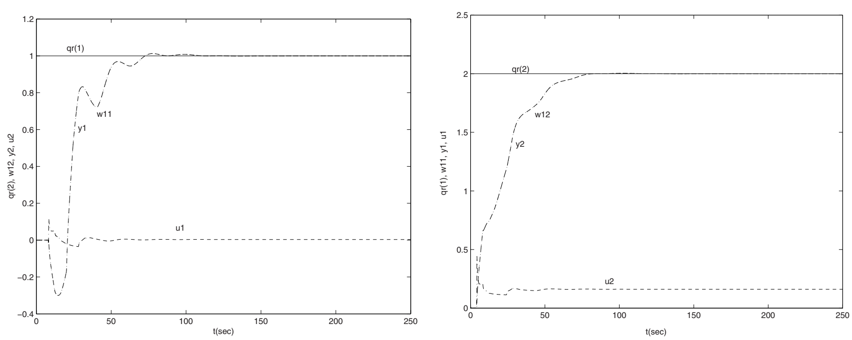

Fig. 10.3 Step response of y_1, y_2, w_{11}, w_{12} and control input u_1, u_2

3. 20% change in coefficients of the system, for example, 20% in t_{11}, the results of simulation are

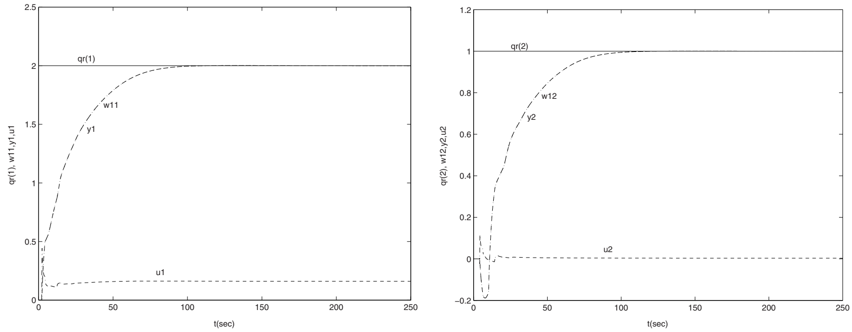

Fig. 10.4 Step response of y_1, y_2, w_{11}, w_{12} and control input u_1, u_2

It can be seen from the above simulations that the states of observer w_{11} and w_{12} are very next to the outputs of the system y_1 and y_2. The control scheme proposed in this chapter is robust against the uncertainties in coefficients of systems and time-delay in the input while preserving a satisfactory transient and steady state performance.

Remark 10.7. A great deal of effort has been devoted to this topic, see [27] and the references therein. However, these methods usually assume the knowledge of the disturbance model and/or the plant model, and usually, a higher order observer or derivatives of the measured signal are used. From the point of ADRC, it regards all factors affecting the plant, including the nonlinear dynamics, uncertainties, the coupling effects and the external disturbances, as a "total disturbance" (extended state) to be observed. This new vision facilitates solution for a series of challenging control problems, such as disturbance rejection, dynamic linearization and decoupling control, in an ingenious way. The proposed technique requires tuning more than three parameters compared with using a standard PID controller. In fact, we use the same set of parameters for the different control channels. Even when a standard PID controller is in cascade with the TD, a better performance can be obtained. Once the sampling time h is given and a predetermined range of variation in system variables are estimated roughly, the parameters of the ESO will be fixed, and the free parameters needed to be tuned are β_1, β_2, b_0. In general, the parameters $\alpha_1 = 0.5$, $\alpha_2 = 0.25$, $\delta = 0.05$, $\delta_c = 0.2$, $\alpha_{c1} = 0.75$ and $\alpha_{c2} = 1.0$ are previously fixed. When the sampling period is h, the parameter of TD is $h_1 = 1.5h$, and the parameters of ESO are $\beta_{01} = \frac{1}{h}$, $\beta_{02} = \frac{2\times4}{27\times h^2}$, $\beta_{03} = \frac{16}{27^2\times h^3}$. If the simulation result is not satisfactory, then β_{01}, β_{02} and β_{03} may need to be suitably tuned.

10.5 Conclusion

In this chapter, an active disturbance rejection controller (ADRC) design method was proposed to deal with both robust stability and performance specifications for a multi-variable process with time-delay in the input. It was shown that a satisfactory performance can be obtained when system contains parameter uncertainties (coefficients, time-delay).

Chapter 11
Analysis and Synthesis of NCSs with Random Forward Delay

11.1 Introduction

With the development of network technology, more and more networks (e.g., Internet) have been applied to distributed control systems, which are termed as networked control systems (NCSs) [143, 137, 2, 70, 180, 47, 49]. Although the networks make it convenient to control large distributed systems, there are many control issues, which occur in conventional control systems, such as network delay and data dropout, sampling and transmitting methods [205]. To solve these problems, various methods have been developed. In the ADDM (augmented deterministic discrete-time model method) [69], an augmented deterministic discrete time model methodology has been proposed to control a linear plant over a periodic delay network. The queuing method [122] has used queuing mechanisms to reshape random network delays in an NCS to deterministic delays such that the NCS becomes time-invariant. The optimal stochastic control method [142] has treated the effects of random network delays in an NCS as a linear quadratic Gaussian problem. The sampling time scheduling method [80] has selected a sampling period for an NCS such that network delays do not significantly affect the control system performance, and the NCS remains stable. The robust control method [59] has designed a networked controller in the frequency domain using robust control theory, and a priori information about the probability distribution of network delays is not required. The hybrid system stability analysis method [214] has modeled networked control systems as hybrid systems. All these methods have put some strict assumptions on NCS, e.g., the network time delay is less than a sampling period [142]. Most of them simply treat the NCS as a system with time delay, which ignores NCS features, e.g., random network delay and data transmitted in packets. In paper [210], the problem of the design of robust H_∞ controllers for uncertain networked control systems (NCSs) with the effects of both the network-induced delay and data dropout is considered, and the memoryless type H_∞ controller is proposed. However, how to compensate for

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 153–176.
springerlink.com

the time-delay and data dropout has not been considered in the papers listed above.

Although much research work has been done in networked control systems, the most work has ignored a very important feature of networked control systems. This feature is that the communication networks can transmit a packet of data at the same time, which is not done in traditional control systems. This chapter makes full use of this network feature and proposes a new networked control scheme—networked predictive control, which can overcome the effects caused by network delay [194, 118]. Furthermore, three different ways to choose control input are discussed in the chapter and the performances have been compared and analyzed.

This chapter is organized as follows: Section 11.2 presents a networked predictive control scheme such that the closed-loop system is asymptotically stable. Section 11.3 details the stability analysis of closed-loop networked predictive control systems. Real-time simulations and practical experiments are presented in Section 11.4. Some conclusion remarks are given in Section 11.5.

11.2 Networked Predictive Control for Systems with Networked Delay

This chapter considers the case where the system controller is far away from the plant but the sensor is next to the plant. So, the network delay in the feedback channel is not considered. A networked predictive control scheme for NCS with random network delay in the forward channel is proposed. The main part of the scheme is the networked predictive controller, which compensates for the network delay in the forward channel and achieves the desired control performance.

Consider a MIMO discrete system described in the following state space form

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t \end{aligned} \tag{11.1}$$

where $x_t \in R^n$, $u_t \in R^m$, and $y_t \in R^l$ are the state, input, and output vectors of the system, respectively, $A \in R^{n\times n}$, $B \in R^{n\times m}$ and $C \in R^{l\times n}$ are the system matrices. For the simplicity of stability analysis, it is assumed that the reference input of the system is zero. Also, the following assumptions are made.

Assumption 11.2.1 *The pair (A, B) is completely controllable, and the pair (A, C) is completely observable.*

Assumption 11.2.2 *The number of consecutive data dropouts is less than N_1, where N_1 is a positive integer.*

Assumption 11.2.3 *The upper bound of the network delay is not greater than N, where N is a positive integer.*

Remark 11.1. In a real NCS, if the data packet does not arrive at a destination in a certain transmission time, it means this data packet is lost, based on the commonly used network protocols. From the physical point of view, it is natural to assume that only a finite number of consecutive data dropouts can be tolerated in order to avoid that the NCS becomes open-loop. Thus, the number of consecutive data dropouts should be less than a finite number N_1. Similarly, the network delay should also be bounded by a finite number N. Clearly, the upper bound number of consecutive data dropouts should not be greater than the upper bound of the network delay (*i.e.* $N_1 \leq N$).

The state observer is designed as

$$\hat{x}_{t+1|t} = A\hat{x}_{t|t-1} + Bu_{ot} + L(y_t - C\hat{x}_{t|t-1}) \tag{11.2}$$

where $\hat{x}_{t+1|t} \in R^n$ and $u_{ot} \in R^m$ are the one-step ahead state prediction and the input of the observer at time t, respectively. The matrix $L \in R^{n\times l}$ can be designed using observer design approaches.

The estimator of the state

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + M(y_t - C\hat{x}_{t|t-1}) \tag{11.3}$$

where $\hat{x}_{t|t-1} \in R^n$ is the one-step prediction, $y_t - C\hat{x}_{t|t-1}$ is the innovation, M is filter gain to be determined.

Following the state observer described by (11.2), based on the output data up to $t-k$, the state predictions from time $t-k+1$ to t are constructed as

$$\begin{aligned}
\hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_{ot} + L(y_t - C\hat{x}_{t|t-1}) \\
\hat{x}_{t+2|t} &= A\hat{x}_{t+1|t} + Bu_{t+1|t} \\
&\vdots \\
\hat{x}_{t+N|t} &= A\hat{x}_{t+N-1|t} + Bu_{t+N-1|t}.
\end{aligned} \tag{11.4}$$

In particular, based on (11.1)-(11.3), the augmented system without time-delay, i.e., $u_t = u_{t|t}$, can be described as follows:

$$\begin{aligned}
&x_{t+1} = (A + BKMC)x_t + (BK - BKMC)\hat{x}_{t|t-1} \\
&\hat{x}_{t+1|t} = (A + BK - LC - BKMC)\hat{x}_{t|t-1} \\
&+(BKMC + LC)x_t.
\end{aligned} \tag{11.5}$$

Where the gain of feedback controller, K, can be designed based on modern control theory in the case of no delay. To overcome unknown network transmission delay, a networked predictive control scheme is proposed. It mainly consists of a control prediction generator and a network delay compensator. The control prediction generator is designed to generate a set of future control predictions. The network delay compensator is used to compensate the unknown random network delay. This chapter considers the case where the system's sensor is next to the plant. So, the transmission delay in the feedback channel is not discussed. In order to compensate the network transmission delay, a network delay compensator is proposed. A very important characteristic of the network is that it can transmit a set of data at the same time. Thus, it is assumed that predictive control sequence at time t is packed and sent to the plant side through a network. The network delay compensator chooses the latest control value from the control prediction sequences available on the plant side. For example, if the following predictive control sequences are received on the plant side:

$$\begin{aligned}
&[u^T_{t-k_1|t-k_1}, u^T_{t-k_1+1|t-k_1}, \cdots, u^T_{t|t-k_1}, \cdots, u^T_{t+N-k_1|t-k_1}]^T \\
&[u^T_{t-k_2|t-k_2}, u^T_{t-k_2+1|t-k_2}, \cdots, u^T_{t|t-k_2}, \cdots, u^T_{t+N-k_2|t-k_2}]^T \\
&\vdots \\
&[u^T_{t-k_t|t-k_t}, u^T_{t-k_t+1|t-k_t}, \cdots, u^T_{t|t-k_t}, \cdots, u^T_{t+N-k_t|t-k_t}]^T
\end{aligned} \tag{11.6}$$

where the control values $u_{t|t-k_i}$ for $i = 1, 2, \cdots, t$, are available to be chosen as the control input of the plant at time t, the output of the network delay compensator, i.e., the input to the actuator will be

$$u_t = u_{t|t-min\{k_1,k_2,\cdots,k_t\}}. \tag{11.7}$$

In fact, using the networked predictive control scheme presented in this section, the control performance of the closed-loop system with network delay is very similar to that of the closed-loop system without network delay.

While designing of the predictor, control input to the observer u_t is needed to improve the predictive precision. The control inputs to the actuator may be different to the control inputs to the observer due to networked induced time-delay and data dropout. There are three ways to obtain control inputs to the observer, which are analyzed as follows:

Case 1: Use predicted value as the control input to actuator shown in Fig. 11.1:

$$u_t = K\hat{x}_{t|t-k_t} \tag{11.8}$$

while the control input to the observer is

$$u_{ot} = u_{t|t} \tag{11.9}$$

where $K \in R^{m\times n}$ is the state feedback control matrix to be determined using modern control theory. Then, the control predictions are generated by

$$u_{ot} = u_{t|t} = K\hat{x}_{t|t}, u_{t+k_t|t} = K\hat{x}_{t+k_t|t},\ for\ k_t = 0,\ 1,\ 2,\ \cdots,\ N, \tag{11.10}$$

Thus, it follows from equation (11.4) that

$$\hat{x}_{t+k_t|t} = (A + BK)^{k_t-1}\hat{x}_{t+1|t}. \tag{11.11}$$

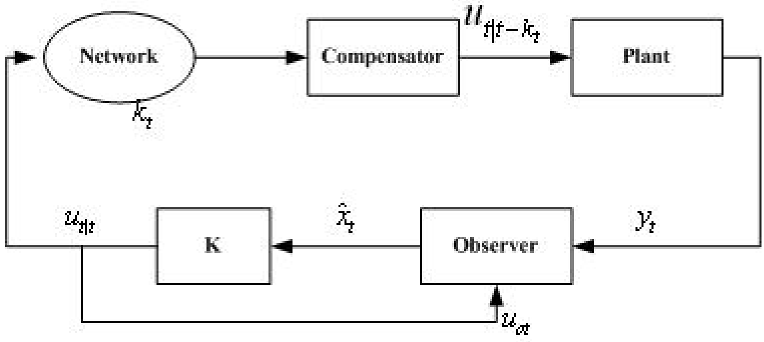

Fig. 11.1 The diagram of Case 1

Based on equations (11.2), (11.3) and (11.9), it can be shown that

$$\begin{aligned}\hat{x}_{t+k_t|t} = & (A+BK)^{k_t-1}(BKMC+LC)x_t \\ & +(A+BK)^{k_t-1}(A+BK-LC-BKMC)\hat{x}_{t|t-1}\end{aligned} \quad (11.12)$$

and

$$\begin{aligned}u_{t+k_t|t} = & K(A+BK)^{k_t-1}(BKMC+LC)x_t \\ & +K(A+BK)^{k_t-1}(A+BK-LC-BKMC)\hat{x}_{t|t-1}.\end{aligned} \quad (11.13)$$

Case 2: The control input to the observer is $u_{t|t-k}$ as u_t which is sent directly to the actuator of plant, as shown in Fig. 11.2:

$$u_t = u_{ot} = u_{t|t-k_t} = K\hat{x}_{t|t-k_t} \quad (11.14)$$

where the state feedback matrix $K \in R^{m\times n}$, and k_t is the network delay in

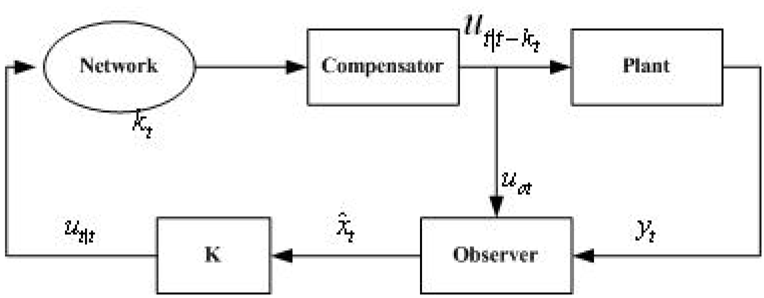

Fig. 11.2 The diagram of Case 2

forward channel. Using (11.4), (11.7) and (11.14), the output of the networked predictive control at time t is determined by

$$\begin{aligned} u_t &= K\hat{x}_{t|t-k_t} \\ &= K(A+BK)^{k_t-1}\hat{x}_{t-k_t+1|t-k_t} \\ &= K(A+BK)^{k_t-1}[A\hat{x}_{t-k_t|t-k_t-1} + Bu_{t-k_t} + L(y_t - C\hat{x}_{t-k_t|t-k_t-1})] \\ &= K(A+BK)^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} \\ &\quad + K(A+BK)^{k_t-1}LCx_{t-k_t} + K(A+BK)^{k_t-1}Bu_{t-k_t}. \end{aligned} \tag{11.15}$$

Case 3: The control input u_t which is sent to the actuator of the plant is transferred to the observer through the network, (shown in Fig. 11.3).

$$u_{at} = u_{t|t-k_t}, u_{ot} = u_{t-f_t|t-k_t-f_t} = K\hat{x}_{t-f_t|t-k_t-f_t} \tag{11.16}$$

where the state feedback matrix $K \in R^{m\times n}$, k_t and f_t are the network

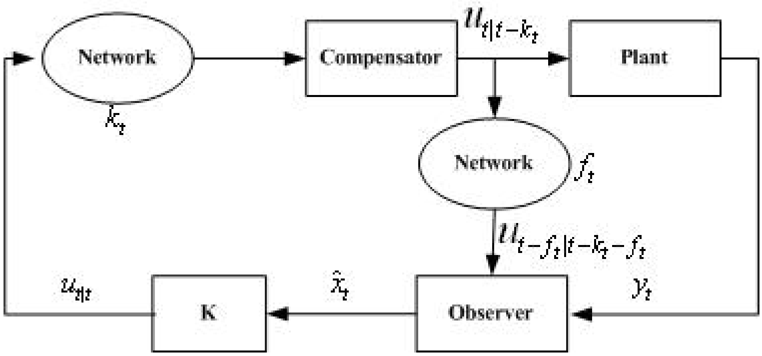

Fig. 11.3 The diagram of Case 3

induced delay in forward channel. Using (11.5) and (11.16), the output of the networked predictive control at time t is determined by

$$\begin{aligned} u_t &= K\hat{x}_{t|t-k_t} \\ &= K(A+BK)^{k_t-1}\hat{x}_{t-k_t+1|t-k_t} \\ &= K(A+BK)^{k_t-1}[A\hat{x}_{t-k_t|t-k_t-1} + Bu_{t-k_t} + L(y_t - C\hat{x}_{t-k_t|t-k_t-1})] \\ &= K(A+BK)^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} \\ &\quad + K(A+BK)^{k_t-1}LCx_{t-k_t} + K(A+BK)^{k_t-1}Bu_{t-k_t}. \end{aligned} \tag{11.17}$$

11.3 Stability Analysis of Closed Networked Predictive Control Systems

Case 1:

Theorem 11.2. *For the networked predictive control systems with random network delay in the forward channel, the closed-loop system (11.1) is stable if there exists a positive definite matrix* $P \in R^{(2N+2)n\times(2N+2)n}$ *such that*

$$\overline{A}^T(k_t)P\overline{A}(k_t) - P < 0 \tag{11.18}$$

for $k_t = 0, 1, 2, \cdots, N$*, where*

$$\overline{A}(k_t) = \begin{bmatrix} A & 0\cdots & M_1(k_t) & 0\cdots & 0 & 0 & 0 & \cdots & M_2(k_t) & 0 & \cdots 0 & 0 \\ I & 0\cdots & 0 & 0\cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots\cdots & \vdots & \vdots\cdots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0\cdots & I & 0\cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots 0 & 0 \\ 0 & 0\cdots & 0 & I\cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots\cdots & \vdots & \vdots\cdots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0\cdots & 0 & 0\cdots & I & 0 & 0 & \cdots & 0 & 0 & \cdots 0 & 0 \\ M_3(k_t) & 0\cdots & 0 & 0\cdots & 0 & 0 & M_4(k_t) & \cdots & 0 & 0 & \cdots 0 & 0 \\ 0 & 0\cdots & 0 & 0\cdots & 0 & 0 & I & \cdots & 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots\cdots & \vdots & \vdots\cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \cdots\vdots & \vdots \\ 0 & 0\cdots & 0 & 0\cdots & 0 & 0 & 0 & \cdots & I & 0 & \cdots 0 & 0 \\ 0 & 0\cdots & 0 & 0\cdots & 0 & 0 & 0 & \cdots & 0 & I & \cdots 0 & 0 \\ \vdots & \vdots\cdots & \vdots & \vdots & \vdots & \cdots\vdots & \vdots & \cdots & \vdots & \vdots & \cdots\vdots & \vdots \\ 0 & 0\cdots & 0 & 0\cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \cdots I & 0 \end{bmatrix} \tag{11.19}$$

with $k_t \in \{1, 2, \cdots, N\}$, $A(k_t) \in R^{2(N+1)n\times 2(N+1)n}$,

$$M_1(k_t) = BK(A+BK)^{k_t-1}(BKMC+LC)$$
$$M_2(k_t) = BK(A+BK)^{k_t-1}(A+BK-LC-BKMC)$$
$$M_3(k_t) = BKMC+LC, \quad M_4(k_t) = A+BK-LC-BKMC$$

$$\overline{A}(0) = \begin{bmatrix} M_1(0) & 0 \cdots 0 & 0 \cdots & 0 & 0 & M_2(0) & \cdots 0 & 0 & \cdots 0 & 0 \\ I & 0 \cdots 0 & 0 \cdots & 0 & 0 & 0 & \cdots 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots \cdots & \vdots & \vdots & \cdots & \vdots & \vdots \cdots & \vdots & \vdots \\ 0 & 0 \cdots I & 0 \cdots & 0 & 0 & 0 & \cdots 0 & 0 & \cdots 0 & 0 \\ 0 & 0 \cdots 0 & I \cdots & 0 & 0 & 0 & \cdots 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots \cdots & \vdots & \vdots & \cdots & \vdots & \vdots \cdots & \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 \cdots & I & 0 & 0 & \cdots 0 & 0 & \cdots 0 & 0 \\ M_3(0) & 0 \cdots 0 & 0 \cdots & 0 & 0 & M_4(0) & \cdots 0 & 0 & \cdots 0 & 0 \\ 0 & 0 \cdots 0 & 0 \cdots & 0 & 0 & I & \cdots 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots \cdots & \vdots & \vdots & \vdots & \cdots \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 \cdots & 0 & 0 & 0 & \cdots I & 0 & \cdots 0 & 0 \\ 0 & 0 \cdots 0 & 0 \cdots & 0 & 0 & 0 & \cdots 0 & I & \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots & \vdots & \vdots & \cdots \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 \cdots & 0 & 0 & 0 & \cdots 0 & 0 & \cdots I & 0 \end{bmatrix}$$

where $A(0) \in R^{2(N+1)n \times 2(N+1)n}$, $M_1(0) = A + BKMC$, $M_2(0) = BK - BKMC$, $M_3(0) = BKMC + LC$, $M_4(0) = A + BK - LC - BKMC$.

Proof. Since the control input to the actuator of the plant is $u_t = K\hat{x}_{t|t-k_t}$, then, based on (11.13), the closed-loop system can be written as

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ &= Ax_t + Bu_{t|t-k_t} \\ &= Ax_t + M_1(k_t)x_{t-k_t} \\ &\quad + M_2(k_t)\hat{x}_{t-k_t|t-k_t-1}. \end{aligned} \tag{11.20}$$

Since the control input to the observer is $u_{ot} = u_{t|t} = K\hat{x}_{t|t}$, then, based on (11.2), the state observer has the following form

$$\begin{aligned} \hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_{t|t} + L(Cx_t - C\hat{x}_{t|t-1}) \\ &= (A - LC)\hat{x}_{t|t-1} + LCx_t + BK(\hat{x}_{t|t-1} \\ &\quad + M(y_t - C\hat{x}_{t|t-1})) \\ &= M_4\hat{x}_{t|t-1} + (BKMC + LC)x_t. \end{aligned} \tag{11.21}$$

Let

$$\begin{aligned} \overline{x}^T(t) = [&x_t^T \; x_{t-1}^T \; \cdots \; x_{t-k_t}^T \; x_{t-(k_t+1)}^T \; \cdots \; x_{t-N}^T \; \hat{x}_{t|t-1}^T \\ &\hat{x}_{t-1|t-2}^T \cdots \; \hat{x}_{t-k_t|t-k_t-1}^T \; \cdots \; \hat{x}_{t-N|t-N-1}^T] \end{aligned} \tag{11.22}$$

then, the augmented system can be expressed as

$$\overline{x}_{t+1} = \overline{A}(k_t)\overline{x}_t.$$

When $k_t = 0$, based on (11.5), the augmented system can be expressed as

$$x_{t+1} = \overline{A}(0)\overline{x}_t. \tag{11.23}$$

It follows that the closed-loop system is a switched system which is composed of $N+1$ discrete-time subsystems, i.e.,

$$\overline{x}_{t+1} = \overline{A}(k_t)\overline{x}_t \tag{11.24}$$

where $k_t = 0, 1, \cdots, N$. The switched system can be described as

$$\overline{x}_{t+1} = \overline{A}_{\delta(k)}\overline{x}_t \tag{11.25}$$

where $\delta(k) : \{0, 1, \cdots\} \to \{0, 1, 2, \cdots, N\}$, $\delta(k)$ is switching signal.

Let $V_t = \overline{x}_t^T P \overline{x}_t$, then

$$\begin{aligned} V_{t+1} - V_t &= \overline{x}_{t+1}^T P \overline{x}_{t+1} - \overline{x}_t^T P \overline{x}_t \\ &= \overline{x}_t^T [\overline{A}_{\delta(k)}^T P \overline{A}_{\delta(k)} - P]\overline{x}_t^T. \end{aligned} \tag{11.26}$$

From (11.18), it follows that $V_{t+1} - V_t < 0$, for $\delta(k)$. Therefore, system (11.25) is stable for all switching sequences $\delta(k)$.

Case 2:

Then, we are in a position to give the results for Case 2.

Theorem 11.3. *For the networked predictive control system with random network delay in the feedback channel, the closed-loop system (11.1) is stable if there exists a positive definite matrix $P \in R^{(2nN+mN+2n)\times(2nN+mN+2n)}$ such that*

$$\Lambda^T(k_t) P \Lambda(k_t) - P < 0 \tag{11.27}$$

for $k_t = 0, 1, 2, \cdots, N$, where

$$\Lambda(0) = \begin{bmatrix} \overbrace{M_1(0)\ 0 \cdots 0\ 0}^{(N+1)n} & \overbrace{0 \cdots 0\ 0}^{Nm} & M_2(0) & 0 \cdots 0\ 0 \\ I\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots\ \vdots & \vdots \cdots \vdots\ \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots I\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & K & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & I \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots\ \vdots & \vdots \cdots \vdots\ \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0\ 0 & 0 \cdots I\ 0 & 0 & 0 \cdots 0\ 0 \\ M_3(0)\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & M_4(0) & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & I & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots\ \vdots & \vdots \cdots \vdots\ \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots I\ 0 \end{bmatrix} \tag{11.28}$$

$\Lambda(0) \in R^{(2nN+mN+2n)\times(2nN+mN+2n)}$, $M_1(0) = A+BKMC$, $M_2(0) = BK-BKMC$, $M_3(0) = BKMC + LC$, $M_4(0) = A + BK - LC - BKMC$.

$$\Lambda(k_t) = \begin{bmatrix} \Lambda_{11}(k_t) & \Lambda_{12}(k_t) & \Lambda_{13}(k_t) \\ \Lambda_{21}(k_t) & \Lambda_{22}(k_t) & \Lambda_{23}(k_t) \\ \Lambda_{31}(k_t) & \Lambda_{32}(k_t) & \Lambda_{33}(k_t) \end{bmatrix} \in R^{(2nN+mN+2n)\times(2nN+mN+2n)} \tag{11.29}$$

for $k_t = 1, 2, \cdots, N$, *where*

$$\Lambda_{11}(k_t) = \begin{bmatrix} A & \overbrace{0 \cdots 0}^{(k_t-1)n} & \Pi(1) & 0 \cdots 0\; 0 \\ I & 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ 0 & 0 \cdots I & 0 & 0 \cdots 0\; 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots I\; 0 \end{bmatrix}, \quad \Lambda_{12}(k_t) = \begin{bmatrix} \overbrace{0\; 0 \cdots \Pi(2)}^{k_t m} & 0\; 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0\; 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots & \vdots\; \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0 & 0\; 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0\; 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots & \vdots\; \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0 & 0\; 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0\; 0 \cdots 0\; 0 \end{bmatrix} \tag{11.30}$$

$$\Lambda_{13}(k_t) = \begin{bmatrix} \overbrace{0\; 0 \cdots 0}^{k_t n} & \Pi(3) & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \end{bmatrix}, \quad \Lambda_{21}(k_t) = \begin{bmatrix} \overbrace{0\; 0 \cdots 0}^{k_t n} & \Pi(4) & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0 & 0 & 0 \cdots 0\; 0 \end{bmatrix} \tag{11.31}$$

$$\Lambda_{22}(k_t)=\begin{bmatrix} \overbrace{0\ 0\ \cdots\ \Pi(5)}^{k_t m} & 0\ 0\ \cdots & 0 & 0 \\ I\ 0\ \cdots\ 0 & 0\ 0\ \cdots & 0 & 0 \\ \vdots\ \vdots\ \cdots\ \vdots & \vdots\ \vdots\ \cdots & \vdots & \vdots \\ 0\ 0\ \cdots\ 0 & 0\ 0\ \cdots & 0 & 0 \\ 0\ 0\ \cdots\ I & 0\ 0\ \cdots & 0 & 0 \\ \vdots\ \vdots\ \cdots\ \vdots & \vdots\ \vdots\ \vdots\ \cdots & \vdots & \vdots \\ 0\ 0\ \cdots\ 0 & 0\ 0\ \cdots & 0 & 0 \\ 0\ 0\ \cdots\ 0 & 0\ 0\ \cdots & I & 0 \end{bmatrix},\quad \Lambda_{23}(k_t)=\begin{bmatrix} \overbrace{0\ 0\ \cdots\ 0}^{k_t n} & \Pi(6) & 0\ \cdots\ 0\ 0 \\ 0\ 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ \vdots\ \vdots\ \cdots\ \vdots & \vdots & \vdots\ \cdots\ \vdots\ \vdots \\ 0\ 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ 0\ 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ \vdots\ \vdots\ \cdots\ \vdots & \vdots & \vdots\ \cdots\ \vdots\ \vdots \\ 0\ 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ 0\ 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \end{bmatrix} \tag{11.32}$$

$$\Lambda_{31}(k_t)=\begin{bmatrix} LC & \overbrace{0\ \cdots\ 0}^{(k_t-1)n} & \Pi(7) & 0\ \cdots\ 0\ 0 \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ \vdots & \vdots\ \cdots\ \vdots & \vdots & \vdots\ \cdots\ \vdots\ \vdots \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ \vdots & \vdots\ \cdots\ \vdots & \vdots & \vdots\ \cdots\ \vdots\ \vdots \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \end{bmatrix},\quad \Lambda_{32}(k_t)=\begin{bmatrix} \overbrace{0\ 0\ \cdots\ \Pi(8)}^{k_t m} & 0\ 0\ \cdots & 0 & 0 \\ I\ 0\ \cdots\ 0 & 0\ 0\ \cdots & 0 & 0 \\ \vdots\ \vdots\ \cdots\ \vdots & \vdots\ \vdots\ \cdots & \vdots & \vdots \\ 0\ 0\ \cdots\ 0 & 0\ 0\ \cdots & 0 & 0 \\ 0\ 0\ \cdots\ I & 0\ 0\ \cdots & 0 & 0 \\ \vdots\ \vdots\ \cdots\ \vdots & \vdots\ \vdots\ \vdots\ \cdots & \vdots & \vdots \\ 0\ 0\ \cdots\ 0 & 0\ 0\ \cdots & 0 & 0 \\ 0\ 0\ \cdots\ 0 & 0\ 0\ \cdots & I & 0 \end{bmatrix} \tag{11.33}$$

$$\Lambda_{33}(k_t)=\begin{bmatrix} \Pi(9) & \overbrace{0\ \cdots\ 0}^{(k_t-1)n} & \Pi(10) & 0\ \cdots\ 0\ 0 \\ I & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ \vdots & \vdots\ \cdots\ \vdots & \vdots & \vdots\ \cdots\ \vdots\ \vdots \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ 0\ 0 \\ 0 & 0\ \cdots\ 0 & I & 0\ \cdots\ 0\ 0 \\ \vdots & \vdots\ \cdots\ \vdots & \vdots & \vdots\ \cdots\ \vdots\ \vdots \\ 0 & 0\ \cdots\ 0 & 0 & I\ \cdots\ 0\ 0 \\ 0 & 0\ \cdots\ 0 & 0 & 0\ \cdots\ I\ 0 \end{bmatrix} \tag{11.34}$$

where

$\Lambda_{11}(k_t)\in R^{(nN+n)\times(nN+n)}$, $\Lambda_{12}(k_t)\in R^{(nN+n)\times(mN)}$
$\Lambda_{13}(k_t)\in R^{(nN+n)\times(nN+n)}$, $\Lambda_{21}(k_t)\in R^{(mN)\times(nN+n)}$
$\Lambda_{22}(k_t)\in R^{(mN)\times(mN)}$, $\Lambda_{23}(k_t)\in R^{(mN)\times(nN+n)}$
$\Lambda_{31}(k_t)\in R^{(nN+n)\times(nN+n)}$, $\Lambda_{32}(k_t)\in R^{(nN+n)\times(mN)}$
$\Lambda_{33}(k_t)\in R^{(nN+n)\times(nN+n)}$

$\Pi(1)=BK(A+BK)^{k_t-1}LC$, $\Pi(2)=BK(A+BK)^{k_t-1}B$, $\Pi(3)=BK(A+BK)^{k_t-1}(A-LC)$, $\Pi(4)=K(A+BK)^{k_t-1}LC$, $\Pi(5)=K(A+BK)^{k_t-1}B$,

$\Pi(6) = K(A+BK)^{k_t-1}(A-LC)$, $\Pi(7) = BK(A+BK)^{k_t-1}LC$, $\Pi(8) = BK(A+BK)^{k_t-1}B$, $\Pi(9) = A-LC$, $\Pi(10) = BK(A+BK)^{k_t-1}(A-LC)$.

Proof. Since the control input to the actuator of the plant is $u_t = K\hat{x}_{t|t-k_t}$, then, based on (11.15), the closed-loop system can be written as

$$
\begin{aligned}
x_{t+1} &= Ax_t + Bu_t \\
&= Ax_t + Bu_{t|t-k_t} \\
&= Ax_t + BK(A+BK)^{k_t-1}LCx_{t-k_t} \\
&\quad + BK(A+BK)^{k_t-1}Bu_{t-k_t} + BK(A+BK)^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1}
\end{aligned}
$$

and the control input to the observer is the same as the one to the actuator the plant, i.e., $u_{ot} = u_t = K\hat{x}_{t|t-k_t}$, then, the state observer has the following form

$$
\begin{aligned}
\hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_{ot} + L(Cx_t - C\hat{x}_{t|t-1}) \\
&= A\hat{x}_{t|t-1} + BK\hat{x}_{t|t-k_t} + L(Cx_t - C\hat{x}_{t|t-1}) \\
&= LCx_t + BK(A+BK)^{k_t-1}LCx_{t-k_t} + BK(A+BK)^{k_t-1}Bu_{t-k_t} \\
&\quad + (A-LC)\hat{x}_{t|t-1} + BK(A+BK)^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1}.
\end{aligned}
$$

Let

$$
\begin{aligned}
X_t = [\, & x_t^T \; x_{t-1}^T \cdots x_{t-k_t+1}^T \; x_{t-k_t}^T \; x_{t-k_t-1}^T \cdots x_{t-N+1}^T \; x_{t-N}^T \\
& u_{t-1}^T \; u_{t-2}^T \cdots u_{t-k_t+1}^T \; u_{t-k_t}^T \; u_{t-k_t-1}^T \cdots u_{t-N+1}^T \; u_{t-N}^T \; \hat{x}_{t|t-1}^T \; x_{t-1|t-2}^T \\
& \cdots \hat{x}_{t-k_t+1|t-k_t}^T \; \hat{x}_{t-k_t|t-k_t-1}^T \; x_{t-k_t-1|t-k_t-2}^T \cdots \\
& \hat{x}_{t-N+1|t-N}^T \hat{x}_{t-N|t-N-1}^T]^T
\end{aligned}
\tag{11.35}
$$

then, the augmented system can be expressed as

$$\overline{x}_{t+1} = \Lambda(k_t)\overline{x}_t.$$

When $k_t = 0$, based on (11.5), the augmented system can be expressed as

$$\overline{x}_{t+1} = \Lambda(0)\overline{x}_t. \tag{11.36}$$

It follows that the closed-loop system is a switched system which is composed of $N+1$ discrete-time subsystems, i.e.,

$$\overline{x}_{t+1} = \Lambda(k_t)\overline{x}_t \tag{11.37}$$

where $k_t = 0, 1, \cdots, N$. The switched system can be described as

$$\overline{x}_{t+1} = \overline{A}_{\delta(k)}\overline{x}_t \tag{11.38}$$

where $\delta(k) : \{0, 1, \cdots\} \to \{0, 1, 2, \cdots, N\}$, $\delta(k)$ is switching signal.

Let $V_t = \overline{x}_t^T P \overline{x}_t$, then

$$V_{t+1} - V_t = \overline{x}_{t+1}^T P \overline{x}_{t+1} - \overline{x}_t^T P \overline{x}_t$$

$$= \overline{x}_t^T [\overline{A}_{\delta(k)}^T P \overline{A}_{\delta(k)} - P] \overline{x}_t^T. \quad (11.39)$$

From (11.27), it follows that $V_{t+1} - V_t < 0$, for $\delta(k)$. Therefore, system (11.37) is stable for all switching sequences $\delta(k)$.

It is assumed that the network delay k_t and f_t in the forward channel are random but bounded by N, that is, $k_t \in \{1, 2, \cdots, N\}$ and $f_t \in \{1, 2, \cdots, N\}$, respectively. The networked predictive controller is in the form of

$$u_t = K\hat{x}_{t-f_t|t-k_t-f_t}, \text{ subject to } k_t \leq k_{t-1} + 1, \ f_t \leq f_{t-1} + 1. \quad (11.40)$$

Under this control scheme, the closed-loop system will be a switched linear system. The theory of switched systems can be used to judge whether the closed-loop system with random time-delay is stable with the observer gain L and feedback gain K ([32]).

Finally, we generalize the results of Case 2 in Case 3, where the input to the observer is sent through network.

Case 3:

Theorem 11.4. *For the networked predictive control systems with random network delay in the feedback channel, the closed-loop system (11.1) is stable if there exists a positive definite matrix* $P \in R^{(2nN+2mN+2n)\times(2nN+2mN+2n)}$ *such that*

$$\Lambda^T(k_t, \ f_t) P \Lambda(k_t, \ f_t) - P < 0 \quad for \quad k_t, f_t = 0, 1, 2, \cdots, N \quad (11.41)$$

where

$$\Lambda(k_t, \ f_t) = \begin{bmatrix} \Lambda_{11}(k_t, \ f_t) & \Lambda_{12}(k_t, \ f_t) & \Lambda_{13}(k_t, \ f_t) \\ \Lambda_{21}(k_t, \ f_t) & \Lambda_{22}(k_t, \ f_t) & \Lambda_{23}(k_t, \ f_t) \\ \Lambda_{31}(k_t, \ f_t) & \Lambda_{32}(k_t, \ f_t) & \Lambda_{33}(k_t, \ f_t) \end{bmatrix}$$
$$\in R^{(2nN+2mN+2n)\times(2nN+2mN+2n)} \quad (11.42)$$

$\Lambda(0,0)$ *is* $\Lambda(0)$ *with the symbol* mN *replaced by* $2mN$*,* $\Lambda(k_t, f_t$ *is* $\Lambda(k_t)$ *with the symbol* mk_t *replaced by* $m(k_t + f_t)$*.*

Proof. If the random network delays are k_t or $f_t \neq 0$, the control input to the actuator of the plant is $u_{at} = K\hat{x}_{t|t-k_t}$, then, based on (11.17), the closed-loop system can be written as

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ &= Ax_t + Bu_{t|t-k_t} \\ &= Ax_t + BK(A+BK)^{k_t-1}LCx_{t-k_t} \\ &\quad + BK(A+BK)^{k_t-1}Bu_{t-k_t} + BK(A+BK)^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} \end{aligned}$$

and the control input to the observer is sent through the network, i.e., $u_{ot} = u_{t-k_t-f_t}$, then, the state observer has the following form

$$\begin{aligned}
\hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_{ot} + L(Cx_t - C\hat{x}_{t|t-1}) \\
&= A\hat{x}_{t|t-1} + BK\hat{x}_{t|t-k_t} + L(Cx_t - C\hat{x}_{t|t-1}) \\
&= A\hat{x}_{t|t-1} + K(A+BK)^{k_t-1}[A\hat{x}_{t-k_t|t-k_t-1} + Bu_{t-k_t-f_t} \\
&\quad + L(y_t - C\hat{x}_{t-k_t|t-k_t-1})] + L(Cx_t - C\hat{x}_{t|t-1}) \\
&= LCx_t + BK(A+BK)^{k_t-1}LCx_{t-k_t} + BK(A+BK)^{k_t-1}Bu_{t-k_t-f_t} \\
&\quad + (A-LC)\hat{x}_{t|t-1} + BK(A+BK)^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1}.
\end{aligned}$$

Combination of above equations, and let

$$\begin{aligned}
X_t = \big[\, & x_t^T \; x_{t-1}^T \cdots x_{t-k_t+1}^T \; x_{t-k_t}^T \; x_{t-k_t-1}^T \cdots x_{t-N+1}^T \; x_{t-N}^T \\
& u_{t-1}^T \; u_{t-2}^T \cdots u_{t-k_t+1}^T \; u_{t-k_t}^T \; u_{t-k_t-1}^T \cdots u_{t-2N+1}^T \; u_{t-2N}^T \; \hat{x}_{t|t-1}^T \; x_{t-1|t-2}^T \\
& \cdots \hat{x}_{t-k_t+1|t-k_t}^T \; \hat{x}_{t-k_t|t-k_t-1}^T \; x_{t-k_t-1|t-k_t-2}^T \cdots \\
& \hat{x}_{t-N+1|t-N}^T \; \hat{x}_{t-N|t-N-1}^T \big]^T .
\end{aligned} \tag{11.43}$$

While there is no time-delay, i.e., $k_t = 0$ and $f_t = 0$, based on (11.5), the augmented system simply becomes

$$X_{t+1} = \Lambda(0,\ 0)X_t. \tag{11.44}$$

As the time-delay is random, the closed-loop system is a switched system which is composed of $(N+1)\times(N+1)$ discrete-time subsystems, i.e.,

$$X_{t+1} = \Lambda(i_t, k_t)X_t \tag{11.45}$$

where $i_t = 0, 1, \cdots, N$ and $k_t = 0, 1, \cdots, N$. The switched system can be described as

$$X_{t+1} = \Lambda_{\delta(k)}X_t \tag{11.46}$$

where $\delta(k) : \{0, 1, \cdots\} \to \{0, 1, 2, \cdots, N\} \times \{0, 1, 2, \cdots, N\}$, $\delta(k)$ is switching signal.

Let $V_{k_t} = X_t^T P X_t$, then

$$\begin{aligned}
V_{t+1} - V_t &= X_{t+1}^T P X_{t+1} - X_t^T P X_t \\
&= X_t^T [\Lambda_{\delta(k)}^T P \Lambda_{\delta(k)} - P] X_t^T.
\end{aligned} \tag{11.47}$$

From (11.41), it follows that $V_{t+1} - V_t < 0$, for $\delta(k)$. Therefore, system (11.45) is stable.

Remark 11.5. Since network delay k_t and f_t randomly vary from 0 to the upper bound N, respectively. And it is necessary to design gain matrices K and L to ensure matrices $\Lambda(k_t,\ f_t)$ are stable (i.e., their eigenvalues are within the unit circle), for $k_t,\ f_t = 0, 1, ..., N$. But, it does not mean that the closed-loop system is stable because the closed-loop system with random network delay is a switched system. Theorem 13.3 shows that the closed-loop system is stable if (11.41) is satisfied. So, when gain matrices K and L are designed, then (11.41) is a set of LMIs, which are only related to matrices

A, B, C and random network delay k_t and f_t. The LMI Toolbox can be used to find feasible solution P ([15]).

11.4 Realtime Simulation and Practical Experiment

11.4.1 Practical Experiments

Example 11.6. To implement networked control systems, a test rig was built, based on an ARM9 embedded system. The forward channel is through a network and the communication protocol between the controller and the sensor is UDP. The kernel chip of the embedded board is ATMEL's AT91RM9200, which is a cost-effective, high-performance 32-bit RISC microcontroller for Ethernet- based embedded systems. A 10M/100M self-adaptive network controller is integrated in the chip and the chip also has a high computing performance and can work at speeds up to 180 MHz. 2-channel 16-bit high speed digital-analog (D/A) converters and 8-channel 16-bit high speed analog-digital (A/D) converters in the controller board provide I/O interfaces for the controlled plant. In order to validate the proposed method, a servo motor control system which consists of a DC motor, peripheral equipment, speed sensors is considered. The model of the motor control plant at sampling period 0.04 second was identified to be

$$G(z^{-1}) = \frac{A(z^{-1})}{B(z^{-1})} = \frac{0.3077z^{-1} - 0.05806z^{-2} - 0.09977z^{-3} - 0.09555z^{-4}}{1 - 0.3538z^{-1} - 0.3059z^{-2} - 0.2932z^{-3} + 0.006748z^{-4}}$$

The system can also be written as the state space form with the following system matrices

$$A = \begin{bmatrix} 0.3538 & 0.3059 & 0.2932 & -0.006748 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{11.48}$$
$$C = \begin{bmatrix} 0.3077 & -0.05806 & -0.09977 & 0.09555 \end{bmatrix}.$$

The matrices K, L and M were designed to be

$$K = \begin{bmatrix} -0.2468 & -0.3097 & -0.2931 & 0.0067 \end{bmatrix}, L = \begin{bmatrix} 3.0459 \\ 2.4805 \\ 2.0067 \\ -0.0515 \end{bmatrix}, M = \begin{bmatrix} 2 & 2 & 2 & 2 \end{bmatrix}$$

which ensure the close-loop system without time delay is stable.

To illustrate the operation of the proposed networked predictive control scheme, three cases were considered:

a) Local control. There is no network in the closed-loop system, i.e., the output signal from the sensor is directly connected to the controller. So, the network delay is zero. The design of matrices K, L and M have ensured that the closed-loop system is stable.

b) Intranet based control. In this case, the output signal of controller was physically transmitted between two Intranet IP addresses 192.168.2.106 and 192.168.2.108 which were both located on the Chinese Academy of Sciences, Beijing. It was measured that the maximum network delay was 0.12 second. As the sampling period is 0.04 second, the upper bound $N = 3$. For $k_t = 0, 1, 2, 3$, in each three cases, all eigenvalues of matrices $\Lambda(k_t)$ or $\Lambda(k_t, f_t)$ are stable and each common positive definite matrix P satisfying inequalities (11.18), (11.27) and (11.41) were found using the LMI toolbox. That means the closed-loop system is stable with K, L and M given above.

c) Internet based control. In this case, the output signal of controller was transmitted between the same two Internet IP addresses 192.168.2.106 and 192.168.2.108 which were both located on the Chinese Academy of Sciences, Beijing. From [13], we get that the maximum internet network delay was measured to be 0.32 second. So we add 0.2 second random delay to simulate the internet delay. The sampling period was still 0.04 second. So, the upper bound $N = 8$. Similar to the Intranet control case, for $k_t = 0, 1, 2, ..., 8$, all eigenvalues of matrices $\Lambda(k_t)$ or $\Lambda(k_t, f_t)$ are stable and also each common positive definite matrix P satisfying inequalities (11.18), (11.27) and (11.41) were found using the LMI toolbox. That implies that the closed-loop system is stable for given K, L and M above.

To evaluate the performance of the networked predictive control scheme, one real-time simulation and one real-time experiment were carried out.

Case 1: $u_t = K\hat{x}_{t|t-k}, u_{ot} = u_{t|t} = K\hat{x}_{t|t}$

1) Real-time simulation. In this simulation, the servo motor plant to be controlled is represented by its model but the network was a real one. The simulations were performed using Matlab/simulink/Real time Workshop. The real-time simulation diagram is shown in Fig. 11.4. The reference input is a square wave generated by the pulse generator block, which changes between 0v to 1000rpm with period 5s. The controller block Netctrl is the networked predictive controller. Blocks Recv and Send are the receiver and sender of the UDP communication protocol. All of them were designed using Matlab S-Functions. The simulated plant and the controller were executed in a ARM 9 embedded system. The real network (Intranet or Internet) was between UDP communication blocks Recv and Send in Fig. 11.4 Four real time simulations were conducted: local control (i.e., no network), Internet based control without delay compensation, Internet based control with delay compensation and Intranet based control with delay compensation. The real-time simulation results are shown in Fig. 11.5 and Fig. 11.6. The Internet based control without delay compensation was also conducted, but it was found

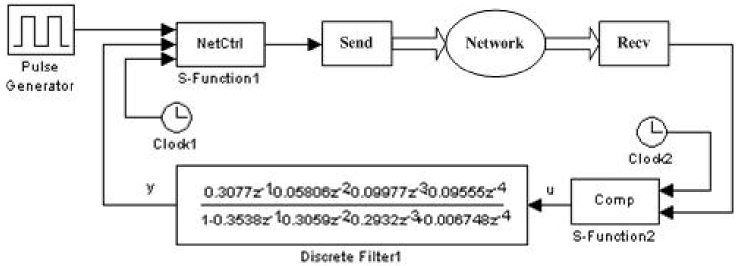

Fig. 11.4 The diagram of the real-time simulation

that the system was no longer stable due to the large network delay, which was between 0.2-0.3s.

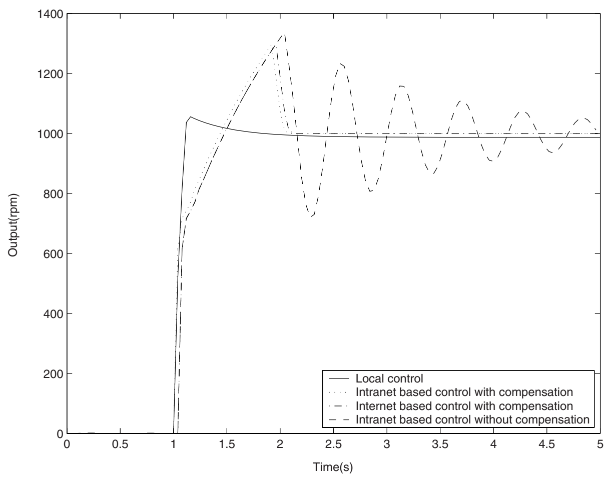

Fig. 11.5 Outputs of servo plant (Simulation).

2) Real-time experiment. The difference between the real-time simulation and real-time experiments is that the plant model of the servo motor in the real-time simulation is replaced by D/A block Dac and A/D block Adc and the real servo motor. The diagram of the real-time experiment is shown in Fig. 11.7. The two blocks Dac and Adc were the driver of the A/D and D/A channels in the embedded system and were designed in Matlab S-Function. Similarly, four real-time experiments were made: local control (i.e., no network), Intranet based control without delay compensation, Intranet based control with delay compensation and Internet based control with delay compensation. The real-time experiment results are shown in Fig. 11.8 and Fig. 11.9.

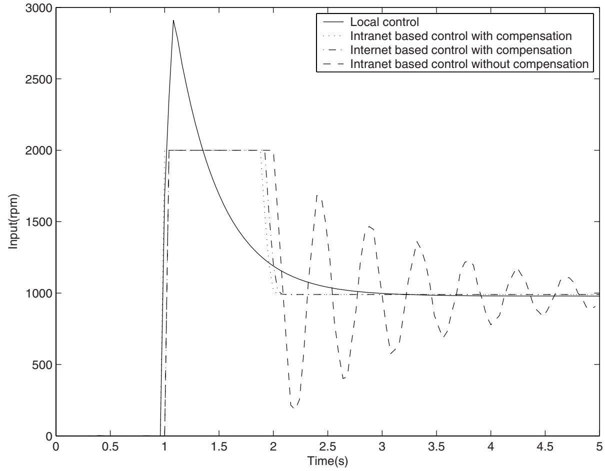

Fig. 11.6 Inputs of servo plant (Simulation).

Also, it was found that the Internet based control without delay compensation was unstable.

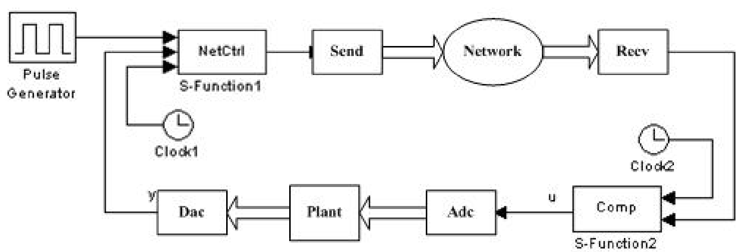

Fig. 11.7 The diagram of the real-time experiment

Case 2: $u_t = u_{t|t-k_t} = K\hat{x}_{t|t-k_t}, u_{ot} = u_{t-k_t}$.
1) Real-time simulation
2) Real-time experiment
Case 3: $u_t = K\hat{x}t|t - k_t, u_{ot} = u_{t-k_t-f_t}$.
1) Real-time simulation
2) Real-time experiment

From the results of real-time simulations and real-time experiments, it is clear that the network transmission delay degrades the performance of NCS. But the networked predictive control scheme proposed in this chapter can compensate for the network delay actively. Its performance is very close to that of the local control scheme (i.e., no network).

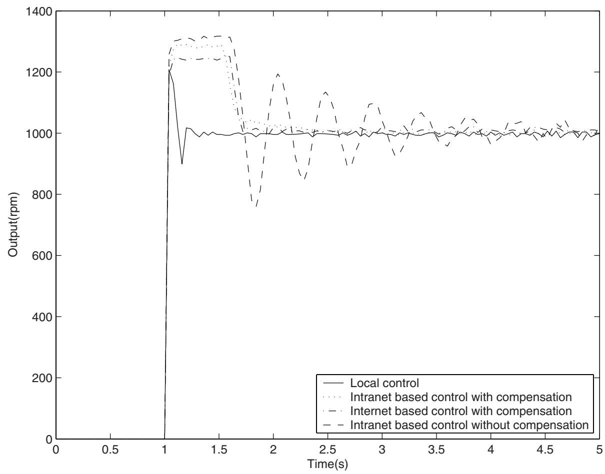

Fig. 11.8 Outputs of servo plant (Experiment).

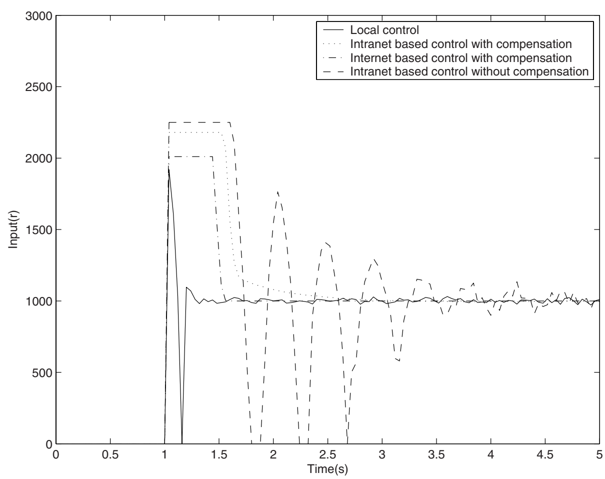

Fig. 11.9 Inputs of servo plant (Experiment)

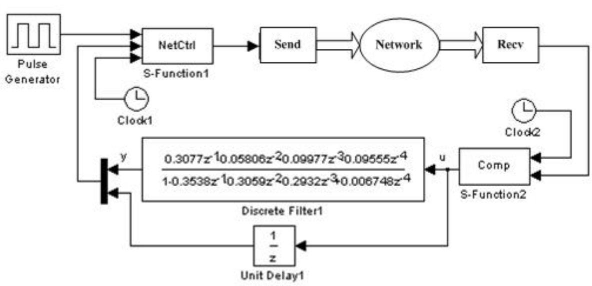

Fig. 11.10 The diagram of the real-time simulation

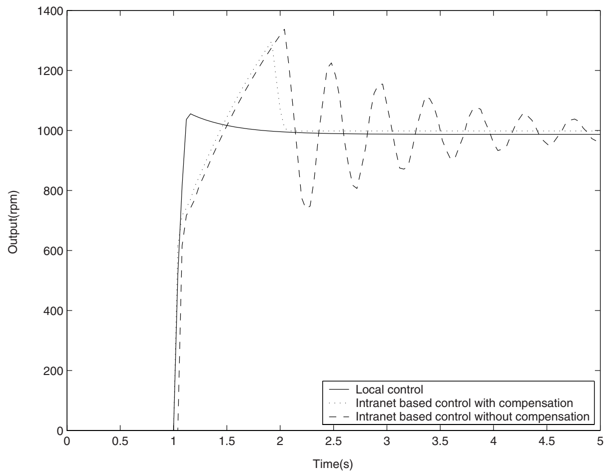

Fig. 11.11 Outputs of servo plant (Simulation).

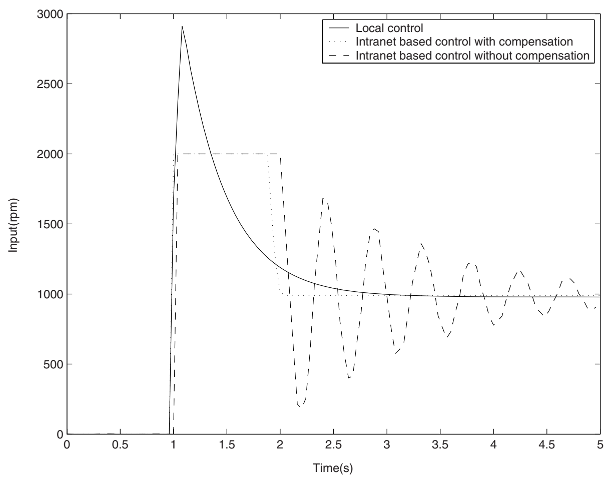

Fig. 11.12 Inputs of servo plant (Simulation).

The results also shows the performances of three different ways to get u_t. We can see that Case 2 is ideal, in which u_t is obtained from the plant directly. However, it is a pity that it is impossible in real-time experiment. Comparing Case 1 with Case 2 and Case 3, we'll find that the effect of Case 1 is very similar to Case 2,which is better than Case 3.

Although it is hard to make the model of the servo motor plant be exactly the same as the real practical one, the results of the real-time experiments are very similar to those of the real-time simulations.

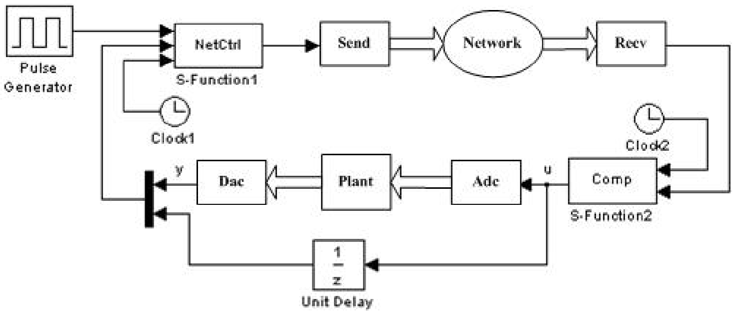

Fig. 11.13 The diagram of the real-time experiment

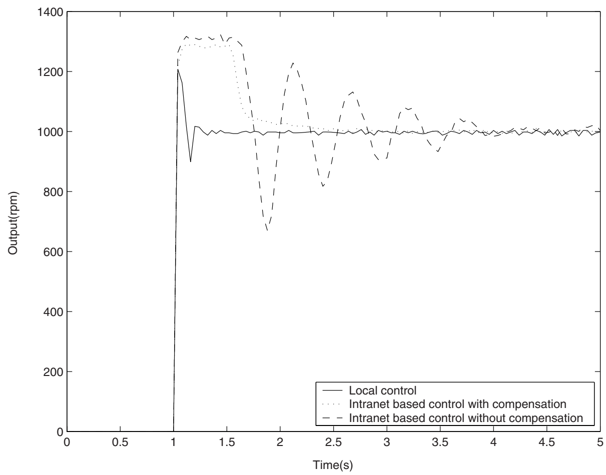

Fig. 11.14 Outputs of servo plant (Experiment).

11.5 Conclusion

A new networked predictive control scheme in a state-space form has been proposed for networked distributed control systems with random network delay and the stability of the closed loop networked predictive control systems has also been discussed in this chapter. Based on the network feature of transmitting a set of data each time, the proposed networked predictive controller consists of the control prediction generator and the network delay compensator. The former provides a set of future control predictions to satisfy the system performance requirements. The latter compensates the random network transmission delay. Three important theorems on the stability of the closed-loop networked predictive control system have given the analytical stability criteria for different u_t, respectively. Also, both real time simulations

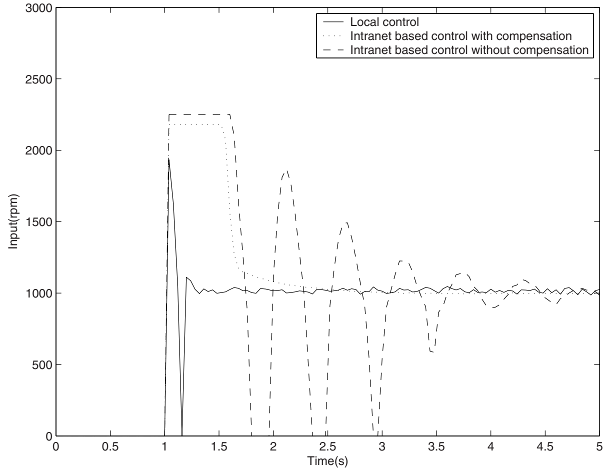

Fig. 11.15 Inputs of servo plant (Experiment).

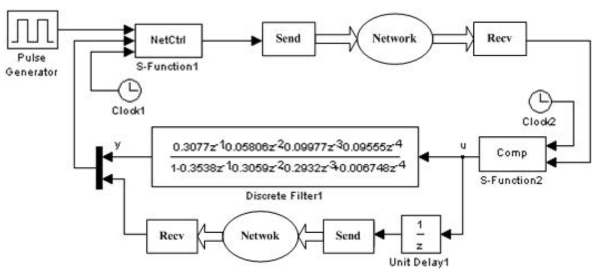

Fig. 11.16 The diagram of the real-time simulation

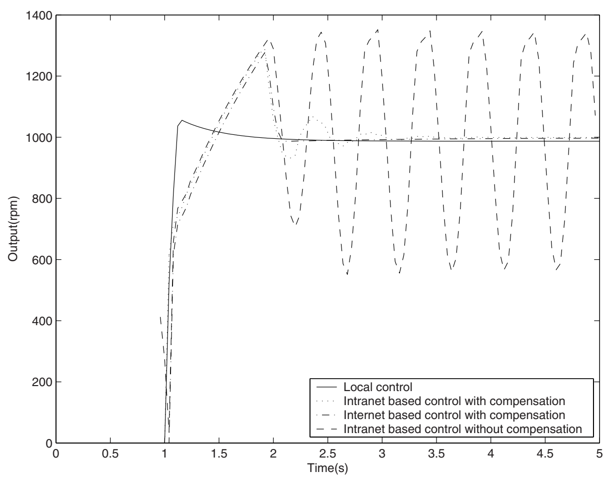

Fig. 11.17 Outputs of servo plant (Simulation).

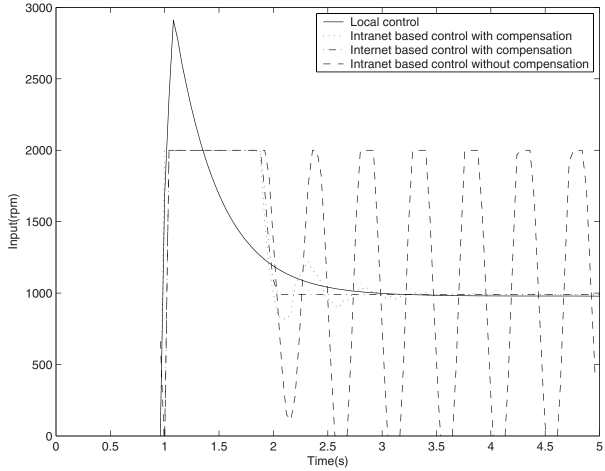

Fig. 11.18 Inputs of servo plant (Simulation)

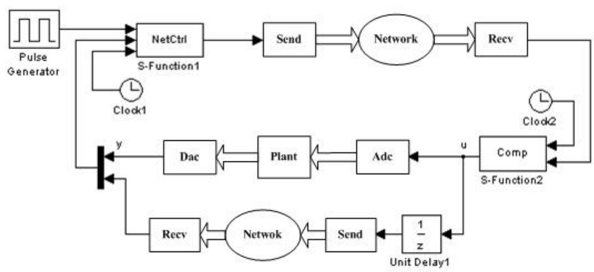

Fig. 11.19 The diagram of the real-time experiment

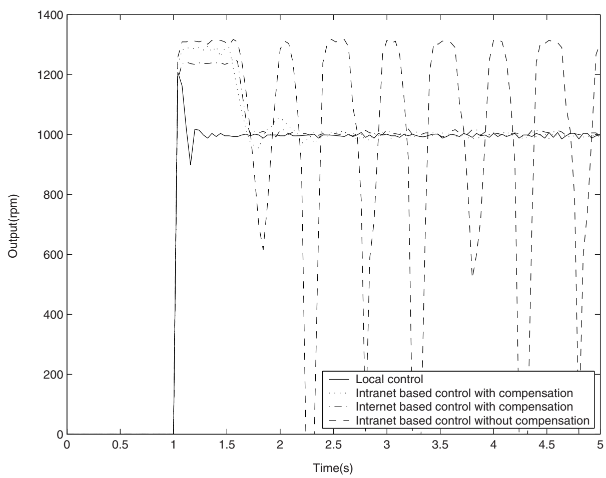

Fig. 11.20 Outputs of servo plant (Experiment).

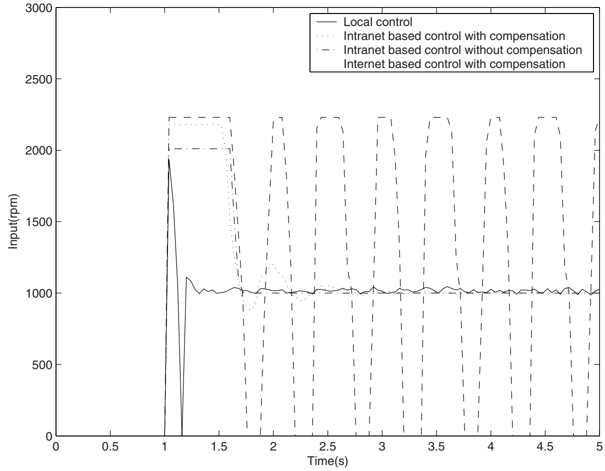

Fig. 11.21 Inputs of servo plant (Experiment)

and practical experiments were presented to show the effectiveness of the control scheme proposed in this chapter.

Chapter 12
Analysis and Synthesis of NCSs with Random Feedback Delay

12.1 Introduction

Recently, there are some preliminary results on designing networked controllers for compensating random time-delay and data dropout, see for example, [116, 120, 119], where the random network delay in the forward channel in NCS has been studied. But, the random network delay in the feedback channel makes the control design and stability analysis much more difficult. This chapter proposes a predictive control scheme for networked control systems with random time delay in the feedback channel and also provides analytical stability criteria of closed-loop networked predictive control systems. In order to compensate for the network transmission delay, a network delay compensator is proposed. The predicted sequences are sent to the actuator in a package. The networked predictive controller chooses the latest input value from the predicted sequences available on the actuator side. Under this control scheme, the closed-loop system will be a switched linear system. The theory of switched systems can be used to judge whether the closed-loop system with random time-delay is stable.

This chapter is organized as follows: Section 12.2 presents a networked predictive control scheme such that the closed-loop system is asymptotically stable. Section 12.3 details the stability analysis of closed-loop networked predictive control systems for system with both constant and random network delay. Real-time simulations and practical experiments are presented in Section 12.4. Some conclusion remarks are given in Section 12.5.

12.2 Design of Networked Predictive Controller

This chapter considers the case where the system sensor is far away from the plant but the controller is next to the plant. So, the network delay in the

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 177–197.
springerlink.com

forward channel is not considered. A networked predictive control scheme for NCS with random network delay in the feedback channel is proposed. The main part of the scheme is the networked predictive controller, which compensates for the network delay in the feedback channel and achieves the desired control performance.

Consider a MIMO (multi-input multi-output) discrete system described in the following state space form

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t \end{aligned} \tag{12.1}$$

where $x_t \in R^n$, $u_t \in R^m$, and $y_t \in R^l$ are the state, input, and output vectors of the system, respectively, $A \in R^{n\times n}$, $B \in R^{n\times m}$ and $C \in R^{l\times n}$ the system matrices. For the simplicity of stability analysis, it is assumed that the reference input of the system is zero. Also, the following assumptions are made.

Assumption 12.2.1 The pair, (A, B), is completely controllable, and the pair, (A, C) is completely observable.

Assumption 12.2.2 *The number of consecutive data dropouts is less than N_1, where N_1 is a positive integer.*

Assumption 12.2.3 *The upper bound of the network delay is not greater than N, where N is a positive integer.*

Remark 12.1. In a real NCS, if the data packet does not arrive at a destination in a certain transmission time, it means this data packet is lost, based on the commonly used network protocols. From the physical point of view, it is natural to assume that only a finite number of consecutive data dropouts can be tolerated in order to avoid that the NCS becomes open-loop. Thus, the number of consecutive data dropouts should be less than a finite number N_1. Similarly, the network delay should also be bounded by a finite number N. Clearly, the upper bound number of consecutive data dropouts should not be greater than the upper bound of the network delay (*i.e.* $N_1 \leq N$).

The state observer is designed as

$$\hat{x}_{t+1|t} = A\hat{x}_{t|t-1} + Bu_t + L(y_t - C\hat{x}_{t|t-1}) \tag{12.2}$$

where $\hat{x}_{t+1|t} \in R^n$ and $u_t \in R^m$ are the one-step ahead state prediction and the input of the observer at time t, respectively. The matrix $L \in R^{n\times l}$ can be designed using observer design approaches.

Following the state observer described by (12.2), based on the output data up to $t - k$, the state predictions from time $t - k + 1$ to t are constructed as

$$\begin{aligned}\hat{x}_{t-k+1|t-k} &= A\hat{x}_{t-k|t-k-1} + Bu_{t-k} + L(y_{t-k} - C\hat{x}_{t-k|t-k-1})\\ \hat{x}_{t-k+2|t-k} &= A\hat{x}_{t-k+1|t-k} + Bu_{t-k+1}\\ &\vdots\\ \hat{x}_{t|t-k} &= A\hat{x}_{t-1|t-k} + Bu_{t-1}\end{aligned} \tag{12.3}$$

which results in

$$\begin{aligned}\hat{x}_{t-k+i|t-k} = \ & A^{i-1}(A-LC)\hat{x}_{t-k|t-k-1} + \sum_{j=1}^{i} A^{i-j}Bu_{t-k+j-1}\\ & + A^{i-1}Ly_{t-k},\ i = 1, 2, 3, \cdots, k.\end{aligned} \tag{12.4}$$

In order to compensate for the network transmission delay, a network delay compensator is proposed. When the time-delay is random and data dropouts happen in the feedback channel, the observer will still use measurement output y_{t-1} received last time if measurement output y_t is lost or y_t is delayed, otherwise, y_{t-j} will be used if y_{t-i} arrives after y_{t-j}, and $j < i$. Thus, with the introduction of bounded random scalar k_t, measurement output y_{t-k_t}, denotes three types of measurement output transmitted in the feedback channel, i.e., random delay, data dropout and first sent later arrived. A very important characteristic of the network is that it can transmit a set of data at the same time. Thus, the predicted sequences are sent to the actuator in a package. The networked predictive controller chooses the latest input value from the predicted sequences available on the actuator side, that is, the controller is designed using the state feedback control strategy:

$$u_t = K\hat{x}_{t|t-k} \tag{12.5}$$

where the state feedback matrix $K \in R^{m\times n}$. This controller can compensate for the network delay caused in the feedback channel. Using (12.4), the output of the networked predictive control at time t is determined by

$$u_t = KA^{k-1}(A-LC)\hat{x}_{t-k|t-k-1} + \sum_{j=1}^{k} KA^{k-j}Bu_{t-k+j-1} + KA^{k-1}Ly_{t-k}. \tag{12.6}$$

From equation (12.6), it is clear that the future control predictions depend on the state estimation $\hat{x}_{t-k|t-k-1}$, the past control input up to u_{t-1} and the past output up to y_{t-k} of the system.

In particular, the augmented system without time-delay and with $u_t = K\hat{x}_{t|t-1}$ can be described as follows:

$$\begin{aligned}\hat{x}_{t+1|t} &= (A+BK-LC)\hat{x}_{t|t-1} + LCx_t\\ x_{t+1} &= Ax_t + BK\hat{x}_{t|t-1}.\end{aligned} \tag{12.7}$$

In the case of no network delay, it is assumed that the state-feedback controller is designed by a modern control method, for example, LQG, eigenstructure assignment, etc.

12.3 Stability of Networked Predictive Control Systems

This section considers the stability of networked control systems for two cases: one is the case of the constant network delay and the other is the case of the random network delay.

12.3.1 Constant Network Delay

It assumes that the network delay in the feedback channel is constant. For this case, there is the following theorem:

Theorem 12.2. *For networked predictive control systems with constant network delay k in the feedback channel, the closed-loop system is stable if and only if all eigenvalues of the following matrix are within the unit circle.*

$$\Psi = \left[\begin{array}{ccccc} A & 0 & 0 & \cdots & 0 \\ KA^{k-1}LC & KB & KAB & \cdots & KA^{k-2}B \\ 0 & I & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & I \\ LCA^{k} & AB & LCAB & \cdots & LCA^{k-2}B \\ 0 & B & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \end{array}\right.$$

$$\left.\begin{array}{cccc} B & 0 & 0 \cdots 0 & 0 \\ KA^{k-1}B & 0 & 0 \cdots 0 & KA^{k-1}(A-LC) \\ 0 & 0 & 0 \cdots 0 & 0 \\ \vdots & \vdots & \vdots \cdots \vdots & \vdots \\ 0 & 0 & 0 \cdots 0 & 0 \\ LCA^{k-1}B & A-LC & 0 \cdots 0 & 0 \\ 0 & I & 0 \cdots 0 & 0 \\ \vdots & \vdots & \vdots \cdots \vdots & \vdots \\ 0 & 0 & 0 \cdots 0 & 0 \\ 0 & 0 & 0 \cdots I & 0 \end{array}\right] \qquad (12.8)$$

where $\Psi \in R^{[km+(k+1)n]\times[km+(k+1)n]}$.

Proof. The necessary and sufficient condition of the closed-loop system stability is that the closed-loop poles are stable. Using the networked predictive controller

$$u_t = KA^{k-1}(A-LC)\hat{x}_{t-k|t-k-1} + \sum_{j=1}^{k} KA^{k-j}Bu_{t-k+j-1} + KA^{k-1}LCx_{t-k} \tag{12.9}$$

results in the following closed-loop system:

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ &= Ax_t + BKA^{k-1}LCx_{t-k} + \textstyle\sum_{j=1}^{k} BKA^{k-j}Bu_{t-k+j-1} \\ &\quad + BKA^{k-1}(A-LC)\hat{x}_{t-k|t-k-1}. \end{aligned} \tag{12.10}$$

The state observer gives

$$\hat{x}_{t+1|t} = (A-LC)\hat{x}_{t|t-1} + LCx_t + Bu_t. \tag{12.11}$$

Combination of equations (12.9), (12.10) and (12.11) gives

$$\bar{\xi}(t+1) = \Omega\bar{\xi}(t) \tag{12.12}$$

where

$$\begin{aligned} \overline{\xi}(t) = \ & \big[\, x_t^T \ x_{t-1}^T \cdots x_{t-k+1}^T \ x_{t-k}^T \ u_{t-1}^T \cdots u_{t-k+1}^T \\ & u_{t-k}^T \ \hat{x}_{t|t-1}^T \cdots \hat{x}_{t-k+1|t-k}^T \ \hat{x}_{t-k|t-k-1}^T \,\big]^T \end{aligned} \tag{12.13}$$

$$\Omega = \begin{bmatrix} A & 0 \cdots 0 & BKA^{k-1}LC & BKB & \cdots & BKA^{k-2}B \\ I & 0 \cdots 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots \ \vdots \ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 \cdots I & 0 & 0 & \cdots & 0 \\ 0 & 0 \cdots 0 & KA^{k-1}LC & KB & \cdots & KA^{k-2}B \\ 0 & 0 \cdots 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 \cdots 0 & 0 & 0 & \cdots & I \\ LC & 0 \cdots 0 & BKA^{k-1}LC & BKB & \cdots & BKA^{k-2}B \\ 0 & 0 \cdots 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 \cdots 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 \cdots 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

$$\left.\begin{matrix}
BKA^{k-1}B & 0 & \cdots 0 & BKA^{k-1}(A-LC) \\
0 & 0 & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
KA^{k-1}B & 0 & \cdots 0 & KA^{k-1}(A-LC) \\
0 & 0 & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
BKA^{k-1}B & A-LC & \cdots 0 & BKA^{k-1}(A-LC) \\
0 & I & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
0 & 0 & \cdots I & 0
\end{matrix}\right] . \tag{12.14}$$

Applying a state transformation such that the sub-matrix in the upper left corner of the above matrix becomes zero, system (12.12) is equivalent to the following system

$$\overline{\xi}'(t+1) = \Omega'\overline{\xi}'(t) \tag{12.15}$$

where

$$\overline{\xi}'(t) = \Big[x_t'^T \; x_{t-1}'^T \cdots x_{t-k+1}'^T \; x_{t-k}'^T \; u_{t-1}'^T \cdots u_{t-k+1}'^T \; u_{t-k}'^T \; \hat{x}_{t|t-1}'^T \\ \cdots \hat{x}_{t-k+1|t-k}'^T \; \hat{x}_{t-k|t-k-1}'^T \Big]^T$$

$$\Omega' = \begin{bmatrix}
0 & 0 & \cdots & 0 & BKA^{k-1}LC & BKB & \cdots & BKA^{k-2}B \\
I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
0 & 0 & \cdots & I & A & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & KA^{k-1}LC & KB & \cdots & KA^{k-2}B \\
0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I \\
LC & LCA & \cdots & LCA^{k-1} & BKA^{k-1}LC & BKB & \cdots & BKA^{k-2}B \\
 & & & & +LCA^k & & & \\
0 & 0 & \cdots & 0 & 0 & B & \cdots & 0 \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0
\end{bmatrix}$$

$$\left.\begin{matrix} BKA^{k-1}B & 0 & \cdots 0 & BKA^{k-1}(A-LC) \\ 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 & \cdots 0 & 0 \\ KA^{k-1}B & 0 & \cdots 0 & KA^{k-1}(A-LC) \\ 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 & \cdots 0 & 0 \\ BKA^{k-1}B & A-LC & \cdots 0 & BKA^{k-1}(A-LC) \\ 0 & I & \cdots 0 & 0 \\ \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 & \cdots 0 & 0 \\ 0 & 0 & \cdots I & 0 \end{matrix}\right] . \tag{12.16}$$

Application of a state transformation such that the first sub-matrix row in the above matrix is zero yields

$$\overline{\xi}''(t+1) = \Omega''\overline{\xi}''(t) \tag{12.17}$$

where

$$\begin{aligned} \overline{\xi}''(t) = & \left[x_t''^T \ x_{t-1}''^T \cdots x_{t-k+1}''^T \ x_{t-k}''^T \ u_{t-1}''^T \cdots u_{t-k+1}''^T \right. \\ & \left. u_{t-k}''^T \ \hat{x}_{t|t-1}''^T \cdots \hat{x}_{t-k+1|t-k}''^T \ \hat{x}_{t-k|t-k-1}''^T \right]^T \end{aligned}$$

$$\Omega'' = \left[\begin{matrix} 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ I & 0 & \cdots & 0 & 0 & B & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & I & A & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & KA^{k-1}LC & KB & \cdots & KA^{k-2}B \\ 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I \\ LC & LCA & \cdots & LCA^{k-1} & LCA^k & AB & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & B & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{matrix}\right.$$

$$
\left.\begin{array}{cccc}
0 & 0 & \cdots 0 & 0 \\
0 & 0 & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
KA^{k-1}B & 0 & \cdots 0 & KA^{k-1}(A-LC) \\
0 & 0 & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
0 & A-LC & \cdots 0 & 0 \\
0 & I & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
0 & 0 & \cdots I & 0
\end{array}\right]. \tag{12.18}
$$

From the structure of equation (12.18), it is clear that the system is stable if and only if the following system is stable

$$
Z(t+1) = \Xi Z(t) \tag{12.19}
$$

where

$$
Z(t) = \left[\, z_1^T(t)\ z_2^T(t)\ \cdots\ z_{3k+1}^T(t) \,\right]
$$

$$
\Xi = \left[\begin{array}{ccccccc}
0 & \cdots & 0 & 0 & B & 0 \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \cdots & \vdots \\
0 & \cdots & I & A & 0 & 0 \cdots & 0 \\
0 & \cdots & 0 & KA^kLC & KB & 0 \cdots & KA^{k-2}B \\
0 & \cdots & 0 & 0 & I & 0 \cdots & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots \cdots & \vdots \\
0 & \cdots & 0 & 0 & 0 & 0 \cdots & I \\
LCA & \cdots & LCA^k & LCA^{k+1} & AB & 0 \cdots & 0 \\
0 & \cdots & 0 & 0 & B & 0 \cdots & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots \cdots & \vdots \\
0 & \cdots & 0 & 0 & 0 & 0 \cdots & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 \cdots & 0
\end{array}\right]
$$

$$\left.\begin{matrix}
0 & 0 & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
KA^{k-1}B & 0 & \cdots 0 & KA^{k-1}(A-LC) \\
0 & 0 & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
0 & A-LC & \cdots 0 & 0 \\
0 & I & \cdots 0 & 0 \\
\vdots & \vdots & \cdots \vdots & \vdots \\
0 & 0 & \cdots 0 & 0 \\
0 & 0 & \cdots I & 0
\end{matrix}\right]. \tag{12.20}$$

Using the state transformation such that the first $k \times n$ rows in (12.20) are zeros, the following system is derived

$$Z'(t+1) = \Xi' Z'(t) \tag{12.21}$$

where

$$\begin{aligned} Z'^T(t) = & \left[z_1'^T(t) \cdots z_k'^T(t) \; z_{k+1}'^T(t) \; z_{k+2}'^T(t) \cdots z_{2k}'^T(t) \right. \\ & \left. z_{2k+1}'^T(t) \; z_{2k+2}'^T(t) \cdots z_{3k}'^T(t) \; z_{3k+1}'^T \right]^T \end{aligned} \tag{12.22}$$

$$\Xi' = \begin{bmatrix}
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
0 & \cdots & I & A & 0 & 0 & \cdots & 0 \\
0 & \cdots & 0 & KA^{k-1}LC & KB & 0 & \cdots & \\
0 & \cdots & 0 & 0 & I & 0 & \cdots & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & I0 \\
LCA & \cdots & LCA^{k-1} & LCA^k & AB & LCAB & \cdots & LCA^{k-2}B \\
0 & \cdots & 0 & 0 & B & 0 & \cdots & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0
\end{bmatrix}$$

$$\left.\begin{array}{cccc} 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots & \cdots \vdots & \vdots \\ B & 0 & \cdots 0 & 0 \\ KA^{k-1}B & 0 & \cdots 0 & KA^{k-1}(A-LC) \\ 0 & 0 & \cdots 0 & 0 \\ \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 & \cdots 0 & 0 \\ LCA^{k-1}B & A-LC & \cdots 0 & 0 \\ 0 & I & \cdots 0 & 0 \\ \vdots & \vdots & \cdots \vdots & \vdots \\ 0 & 0 & \cdots 0 & 0 \\ 0 & 0 & \cdots I & 0 \end{array}\right] . \tag{12.23}$$

From the structure of equation (12.21), it can be concluded that the system is stable if and only if the following system is stable

$$Z''(t+1) = \Psi Z''(t) \tag{12.24}$$

where Ψ is defined in (12.8),

$$\begin{aligned} Z''(t) = \; & \left[z_1''^T(t) \; z_2''^T(t) \; z_3''^T(t) \; \cdots \; z_k''^T(t) \right. \\ & \left. z_{k+1}''^T(t) \; z_{k+2}''^T(t) \; \cdots \; z_{2k+1}''^T(t) \; z_{2k+2}''^T(t) \right]^T . \end{aligned} \tag{12.25}$$

Therefore, the closed-loop system is stable if and only if all eigenvalues of matrix (12.8) are within the unit circle.

Remark 12.3. Since the matrix in (12.8) is related to gain matrices K and L, in general, these gain matrices should be designed to guarantee the closed-loop system with constant network delay is stable. If gain matrices K and L are given, it is clear that the stability of the closed-loop system with constant network delay is only related to matrices A, B, C and network delay k.

12.3.2 Random Network Delay

It is assumed that the network delay k_t in the feedback channel is random but bounded by N, that is, $k_t \in \{1, 2, \cdots, N\}$. The networked predictive controller is in the form of

$$u_t = K\hat{x}_{t|t-k_t}, \text{ subject to } \; k_t \leq k_{t-1} + 1 \tag{12.26}$$

where the network delay k_t is a random number but $k_t \in \{1, 2, \cdots, N\}$.

Under this control scheme, the closed-loop system will be a switched linear system. The theory of switched systems can be used to judge whether the closed-loop system with random time-delay is stable with the observer gain L and feedback gain K ([32]).

In order to present the results in this section , for the sake of simplicity, let

$$\Lambda(0) = \begin{bmatrix} \overbrace{A\ 0 \cdots 0\ 0}^{(N+1)n} & \overbrace{0 \cdots 0\ 0}^{Nm} & BK & 0 \cdots 0\ 0 \\ I\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots\ \vdots & \vdots \cdots \vdots\ \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots I\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & K & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & I \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots\ \vdots & \vdots \cdots \vdots\ \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0\ 0 & 0 \cdots I\ 0 & 0 & 0 \cdots 0\ 0 \\ LC\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & A+BK-LC & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & I & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots\ \vdots & \vdots \cdots \vdots\ \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0\ 0 & 0 \cdots 0\ 0 & 0 & 0 \cdots I\ 0 \end{bmatrix}$$

$$\in R^{(2nN+mN+2n)\times(2nN+mN+2n)} \tag{12.27}$$

$$\Lambda(k_t) = \begin{bmatrix} \Lambda_{11}(k_t) & \Lambda_{12}(k_t) & \Lambda_{13}(k_t) \\ \Lambda_{21}(k_t) & \Lambda_{22}(k_t) & \Lambda_{23}(k_t) \\ \Lambda_{31}(k_t) & \Lambda_{32}(k_t) & \Lambda_{33}(k_t) \end{bmatrix} \in R^{(2nN+mN+2n)\times(2nN+mN+2n)} \tag{12.28}$$

for $k_t = 1, 2, \cdots, N$, where

$$\Lambda_{11}(k_t) = \begin{bmatrix} A & \overbrace{0 \cdots 0}^{(k_t-1)n} & \Pi(1) & 0 \cdots 0\ 0 \\ I & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0 & 0 \cdots I & 0 & 0 \cdots 0\ 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots I\ 0 \end{bmatrix}, \quad \Lambda_{12}(k_t) = \begin{bmatrix} \Pi(2) & \Pi(3) & \cdots & \Pi(4) & \Pi(5) & 0 \cdots 0\ 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots 0\ 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots 0\ 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots 0\ 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots 0\ 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots 0\ 0 \end{bmatrix} \tag{12.29}$$

$$\Lambda_{13}(k_t) = \begin{bmatrix} \overbrace{0\ 0 \cdots 0}^{k_t n} & \Pi(6) & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \end{bmatrix}, \quad \Lambda_{21}(k_t) = \begin{bmatrix} \overbrace{0\ 0 \cdots 0}^{k_t n} & \Pi(7) & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \end{bmatrix} \tag{12.30}$$

$$\Lambda_{22}(k_t) = \begin{bmatrix} KB & \Pi(8) & \cdots & \Pi(9) & \Pi(10) & 0 \cdots & 0 & 0 \\ I & 0 & \cdots & 0 & 0 & 0 \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots & 0 & 0 \\ 0 & 0 & \cdots & I & 0 & 0 \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots\ \vdots & \cdots & \vdots\ \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 \cdots & I & 0 \end{bmatrix}, \quad \Lambda_{23}(k_t) = \begin{bmatrix} \overbrace{0\ 0 \cdots 0}^{k_t n} & \Pi(11) & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots\ \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0\ 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \end{bmatrix} \tag{12.31}$$

$$\Lambda_{31}(k_t) = \begin{bmatrix} LC & \overbrace{0 \cdots 0}^{(k_t-1)n} & \Pi(12) & 0 \cdots 0\ 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots\ \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0\ 0 \end{bmatrix}, \quad \Lambda_{32}(k_t) = \begin{bmatrix} BKB & \Pi(13) & \cdots & \Pi(14) & \Pi(15) & \cdots 0\ 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots 0\ 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots \vdots\ \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots 0\ 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots 0\ 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \cdots \vdots\ \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots 0\ 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots 0\ 0 \end{bmatrix} \tag{12.32}$$

$$\Lambda_{33}(k_t) = \begin{bmatrix} \Pi(16) & \overbrace{0 \cdots 0}^{(k_t-1)n} & \Pi(17) & 0 \cdots 0 & 0 \\ I & 0 \cdots 0 & 0 & 0 \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0 & 0 \\ 0 & 0 \cdots 0 & I & 0 \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 & I \cdots 0 & 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots I & 0 \end{bmatrix} \tag{12.33}$$

where

$\Lambda_{11}(k_t) \in R^{(nN+n)\times(nN+n)}$, $\Lambda_{12}(k_t) \in R^{(nN+n)\times(mN)}$
$\Lambda_{13}(k_t) \in R^{(nN+n)\times(nN+n)}$, $\Lambda_{21}(k_t) \in R^{(mN)\times(nN+n)}$
$\Lambda_{22}(k_t) \in R^{(mN)\times(mN)}$, $\Lambda_{23}(k_t) \in R^{(mN)\times(nN+n)}$
$\Lambda_{31}(k_t) \in R^{(nN+n)\times(nN+n)}$, $\Lambda_{32}(k_t) \in R^{(nN+n)\times(mN)}$
$\Lambda_{33}(k_t) \in R^{(nN+n)\times(nN+n)}$

$\Pi(1) = BKA^{k_t-1}LC$, $\Pi(2) = BKB$, $\Pi(3) = BKAB$, $\Pi(4) = BKA^{k_t-2}B$, $\Pi(5) = BKA^{k_t-1}B$, $\Pi(6) = BKA^{k_t-1}(A-LC)$, $\Pi(7) = KA^{k_t-1}LC$, $\Pi(8) = KAB$, $\Pi(9) = KA^{k_t-2}B$, $\Pi(10) = KA^{k_t-1}B$, $\Pi(11) = KA^{k_t-1}(A-LC)$, $\Pi(12) = BKA^{k_t-1}LC$, $\Pi(13) = BKAB$, $\Pi(14) = BK^{k_t-2}AB$, $\Pi(15) = BKA^{k_t-1}AB$, $\Pi(16) = A-LC$, $\Pi(17) = BKA^{k_t-1}(A-LC)$.

Then, the main results in this section can be stated as follows:

Theorem 12.4. *For the networked predictive control systems with random network delay in the feedback channel, the closed-loop system is stable if there exists a positive definite matrix* $P \in R^{(2nN+mN+2n)\times(2nN+mN+2n)}$ *such that*

$$\Lambda^T(k_t)P\Lambda(k_t) - P < 0 \tag{12.34}$$

for $k_t = 0, 1, 2, \cdots, N$.

Proof. Following the case of the constant network delay, if the random network delay is $k_t, k_t \neq 0$, the control input, state prediction, and plant state vectors are expressed by

$$u_t = KA^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} + \sum_{j=1}^{k_t} KA^{k_t-j}Bu_{t-k_t+j-1} + KA^{k_t-1}LCx_{t-k_t} \tag{12.35}$$

$$\begin{aligned} \hat{x}_{t+1|t} = {} & (A-LC)\hat{x}_{t|t-1} + LCx_t + BKA^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} \\ & + \sum_{j=1}^{k_t} BKA^{k_t-j}Bu_{t-k_t+j-1} + BKA^{k_t-1}LCx_{t-k_t} \end{aligned} \tag{12.36}$$

$$\begin{aligned} x_{t+1} = \; & Ax_t + BKA^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} + \sum_{j=1}^{k_t} BKA^{k_t-j}Bu_{t-k_t+j-1} \\ & + BKA^{k_t-1}LCx_{t-k_t}. \end{aligned} \tag{12.37}$$

Combination of equations (12.35), (12.36) and (12.37) gives the following augmented system:

$$X_{t+1} = \Lambda(k_t)X_t \tag{12.38}$$

where

$$\begin{aligned} X_t = \big[& x_t^T \; x_{t-1}^T \cdots x_{t-k_t+1}^T \; x_{t-k_t}^T \; x_{t-k_t-1}^T \cdots x_{t-N+1}^T \; x_{t-N}^T \\ & u_{t-1}^T \; u_{t-2}^T \cdots u_{t-k_t+1}^T \; u_{t-k_t}^T \; u_{t-k_t-1}^T \cdots u_{t-N+1}^T \; u_{t-N}^T \\ & \hat{x}_{t|t-1}^T \; x_{t-1|t-2}^T \cdots \hat{x}_{t-k_t+1|t-k_t}^T \; \hat{x}_{t-k_t|t-k_t-1}^T \\ & x_{t-k_t-1|t-k_t-2}^T \cdots \hat{x}_{t-N+1|t-N}^T \; \hat{x}_{t-N|t-N-1}^T \big]^T. \end{aligned} \tag{12.39}$$

While there is no time-delay in the feedback channel, i.e., $k_t = 0$, the augmented system (12.7) simply becomes

$$X_{t+1} = \Lambda(0)X_t. \tag{12.40}$$

As the time-delay is random, the closed-loop system is a switched system which is composed of $N+1$ discrete-time subsystems, i.e.,

$$X_{t+1} = \Lambda(k_t)X_t \tag{12.41}$$

where $k_t = 0, 1, \cdots, N$. The switched system can be described as

$$X_{t+1} = \Lambda_{\delta(k)}X_t \tag{12.42}$$

where $\delta(k) : \{0, 1, \cdots\} \to \{0, 1, 2, \cdots, N\}$, $\delta(k)$ is a switching signal.

Let $V_t = X_t^T P X_t$, then

$$\begin{aligned} V_{t+1} - V_t &= X_{t+1}^T P X_{t+1} - X_t^T P X_t \\ &= X_t^T [\Lambda_{\delta(k)}^T P \Lambda_{\delta(k)} - P] X_t^T. \end{aligned} \tag{12.43}$$

From (12.34), it follows that $V_{t+1} - V_t < 0$, for $\delta(k)$. Therefore, system (12.38) is stable for all switching sequences $\delta(k)$.

Remark 12.5. Since network delay k_t randomly varies from 0 to the upper bound N, it is necessary to design gain matrices K and L to ensure matrices $\Lambda(k_t)$ are stable (i.e., their eigenvalues are within the unit circle), for $k_t = 0, 1, ..., N$. But, it does not mean that the closed-loop system is stable because the closed-loop system with random network delay is a switched system. Theorem 12.4 shows that the closed-loop system is stable if (12.34) is satisfied. So, when gain matrices K and L are designed, then (12.34) is a set of LMIs, which are only related to matrices A, B, C and random network delay k_t. The LMI Toolbox can be used to find feasible solution P ([15]).

12.4 Simulation & Experiments

In this section, two illustrative examples are constructed to verify the design method developed in this chapter.

12.4.1 Numerical Simulation

Example 12.6. Simulation studies are presented for an open-loop unstable discrete system which is difficult to control and is in the form of (12.1) with the following system matrices:

$$A = \begin{bmatrix} -1.100 & 0.271 & -0.488 \\ 0.482 & 0.100 & 0.24 \\ 0.002 & 0.3681 & 0.7070 \end{bmatrix}, B = \begin{bmatrix} 5 & 5 \\ 3 & -2 \\ 5 & 4 \end{bmatrix}, C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 & 1 \end{bmatrix}.$$

The matrices K and L are designed by pole assignment to ensure the closed-loop system without network delay is stable and they are

$$K = \begin{bmatrix} 0.4471 & -0.1073 & -0.6597 \\ -0.7316 & 0.0594 & 0.8210 \end{bmatrix}, L = \begin{bmatrix} -0.2879 & 0.3113 \\ 0.0281 & 0.0582 \\ 0.2610 & -0.0572 \end{bmatrix}.$$

For $k_t = 0, 1, 2, 3$, all eigenvalues of matrices $\Lambda(k_t)$ are stable and a common positive definite matrix P satisfying inequalities (12.34) is found. Thus, system (12.1) is stable with random delay $k_t \leq 3$ based on the results of Theorem 12.4, For the initial conditions of the states, $x(0) = [1\ 1\ 1]^T$, $u(t) = [0\ 0]^T, t \leq 0$. The simulation results for $k_t = 4$ are shown in Fig. 12.1 and Fig. 12.2. However, it is found from simulation that if the standard state feedback controller $u_t = K\hat{x}_{t-k|t-k-1}$ is used rather than the network delay compensator $u_t = K\hat{x}_{t|t-k}$, the closed-loop system with the same K, L is unstable for $k_t = 1, 2, 3$.

12.4.2 Practical Experiments

Example 12.7. To implement networked control systems, a test rig is built, based on an ARM 9 embedded system. The feedback channel is through a network and the communication protocol between the controller and the sensor

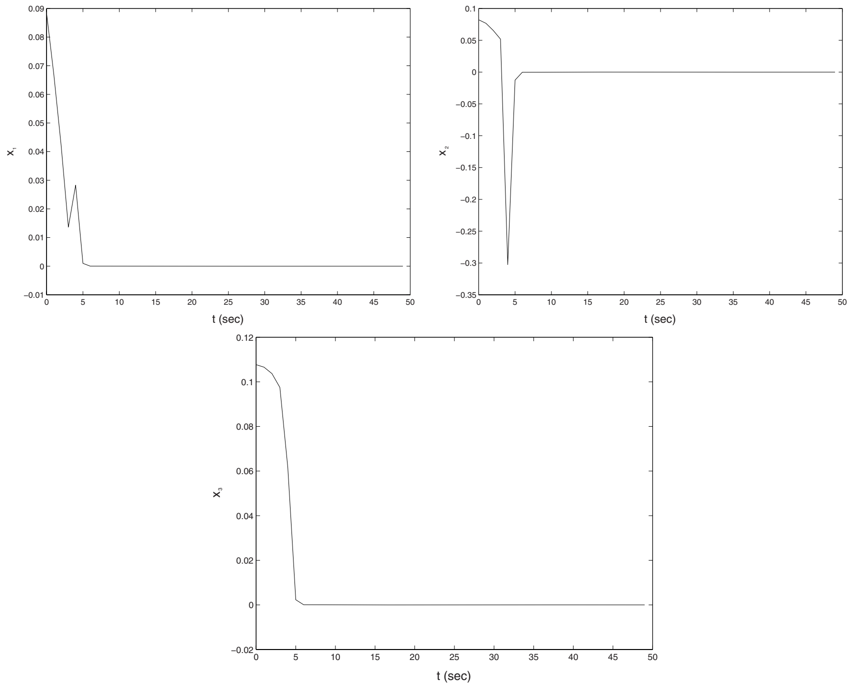

Fig. 12.1 States x_1, x_2 and x_3

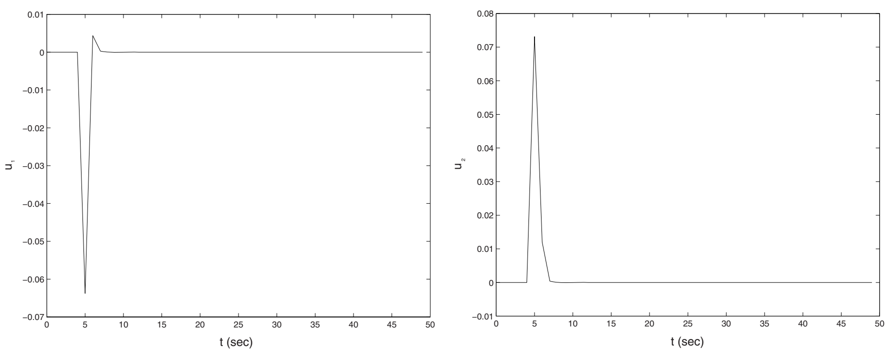

Fig. 12.2 Control inputs u_1 and u_2

is UDP. The kernel chip of the embedded board is ATMEL's AT91RM9200, which is a cost-effective, high-performance 32-bit RISC microcontroller for Ethernet- based embedded systems. A 10M/100M self-adaptive network controller is integrated in the chip and the chip also has a high computing performance and can work at speed up to 180 MHz. 2-channel 16-bit high speed digital-analog (D/A) converters and 8-channel 16-bit high speed analog-digital (A/D) converters in the controller board provide I/O interfaces for the controlled plant.

In order to validate the proposed method, a servo motor control system which consists of a DC motor, load plate, speed and angle sensors is considered. The model of the motor control plant at sampling period 0.04 second is identified to be

$$G(z^{-1}) = \frac{A(z^{-1})}{B(z^{-1})} = \frac{0.05409z^{-2} + 0.115z^{-3} + 0.0001z^{-4}}{1 - 1.12z^{-1} - 0.213z^{-2} + 0.335z^{-3}}.$$

The system can also be written as the state space form with the following system matrices

$$A = \begin{bmatrix} 1.12 & 0.213 & -0.335 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0.0541 & 0.1150 & 0.0001 \end{bmatrix}.$$

The matrices K and L are designed to be

$$K = \begin{bmatrix} -0.0270 & -0.575 & -0.0001 \end{bmatrix}, L = \begin{bmatrix} 6 \\ 6 \\ 6 \end{bmatrix}$$

which ensure the close-loop system without time delay is stable.

To illustrate the operation of the proposed networked predictive control scheme, three cases are considered:

a) Local control. There is no network in the closed-loop system, i.e., the output signal from the sensor is directly connected to the controller. So, the network delay is zero. The design of matrices K and L has ensured that the closed-loop system is stable.

b) Intranet based control. In this case, the output signal is physically transmitted between two Intranet IP addresses 193.63.131.217 and 193.63.131.219 which are both located on the campus network of the University of Glamorgan. It is measured that the maximum network delay is 0.12 second. As the sampling period is 0.04 second, the upper bound $N = 3$. For $k_t = 0, 1, 2, 3$, all eigenvalues of matrices $A(k_t)$ are stable and a common positive definite matrix P satisfying inequalities (12.34) is found using the LMI toolbox. That means the closed-loop system is stable with K and L given above.

c) Internet based control. In this case, the output signal is transmitted between two Internet IP addresses 193.63.131.219 and 81.106.241.34. The former is located at the University of Glamorgan, UK. The latter is located at the Pontypridd, Wales. The maximum network delay is measured to be 0.32 second and the sampling period is still 0.04 second. So, the upper bound $N = 8$. Similar to the Intranet control case, for $k_t = 0, 1, 2, ..., 8$, all eigenvalues of matrices $\Lambda(k_t)$ are stable and also a common positive definite matrix P satisfying inequalities (12.34) is derived using the LMI toolbox. That implies that the closed-loop system is stable for given K and L above.

To evaluate the performance of the networked predictive control scheme, one real-time simulation and one real-time experiment are carried out.

1) Real-time simulation. In this simulation, the servo motor plant to be controlled is represented by its model but the network is a real one. The simulations are performed using Matlab/simulink/Real time Workshop. The real-time simulation diagram is shown in Fig. 12.3. The reference input is a square wave generated by the pulse generator block, which changes between 0v to 7v with period 5s. The controller block Netctrl is the networked predictive controller. Blocks Recv9 and Send9 are the receiver and sender of the UDP communication protocol. All of them are designed using Matlab S-Functions. The simulated plant and the controller are executed in a ARM 9 embedded system. The real network (Intranet or Internet) is between UDP communication blocks Recv9 and Send9 in Fig. 12.3.

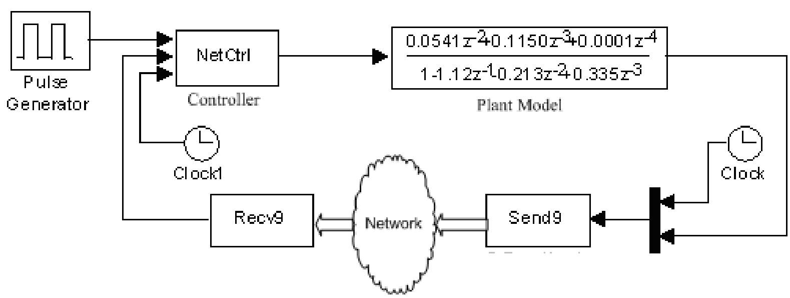

Fig. 12.3 Simulation Diagram

Four real time simulations are conducted: local control (i.e., no network), Intranet based control without delay compensation, Intranet based control with delay compensation and Internet based control with delay compensation. The real-time simulations results are shown in Fig. 12.4 and Fig. 12.5. The Internet based control without delay compensation is also conducted, but it is found that the system is no longer stable due to the large network delay, which is between 0.2-0.3s.

2) Real-time experiment. The difference between the real-time simulations and real-time experiments is that the plant model of the servo motor in the real-time simulations is replaced by D/A block Dac9 and A/D block

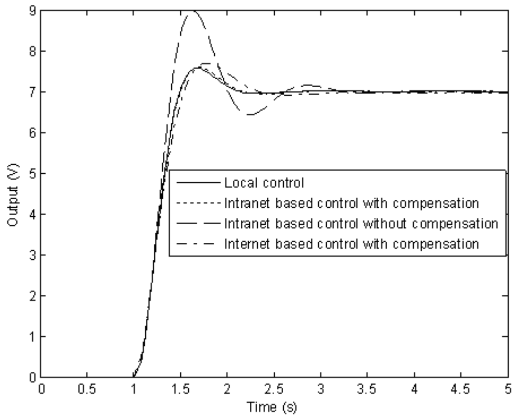

Fig. 12.4 Outputs of servo plant (Simulation)

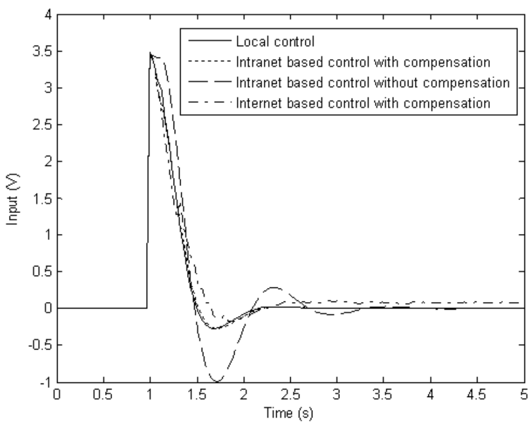

Fig. 12.5 Inputs of servo plant (Simulation)

Adc9 and the real servo motor. The diagram of the real-time experiments is shown in Fig. 12.6. The two blocks Dac9 and Adc9 are the driver of the A/D and D/A channels in the embedded system and are designed in Matlab S-Function. Similarly, four real-time experiments are made: local control (i.e., no network), Intranet based control without delay compensation, Intranet based control with delay compensation and Internet based control with delay compensation. The real-time experiments results are shown in Fig. 12.7 and Fig. 12.8. Also, it is found that the Internet based control without delay compensation is unstable.

From the results of real-time simulations , it is clear that the network transmission delay degrades the performance of NCS. But the networked

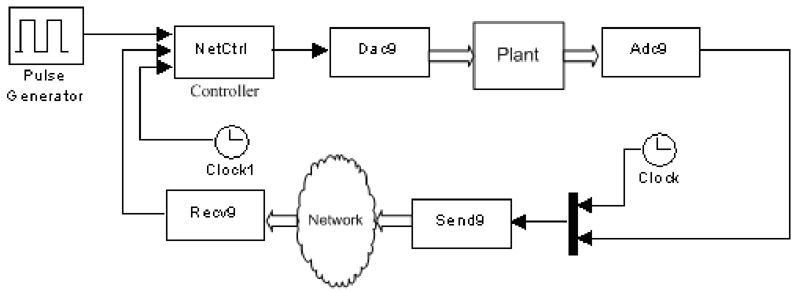

Fig. 12.6 Experiment Diagram

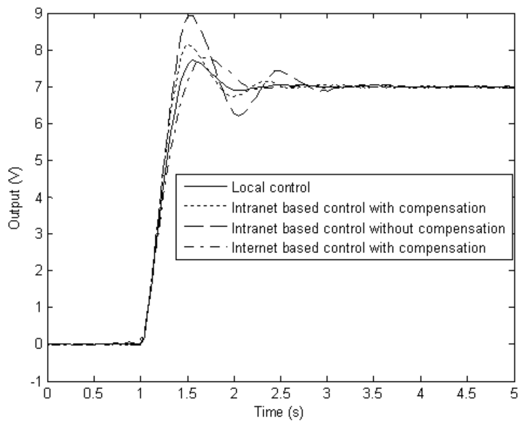

Fig. 12.7 Outputs of servo plant (Experiment)

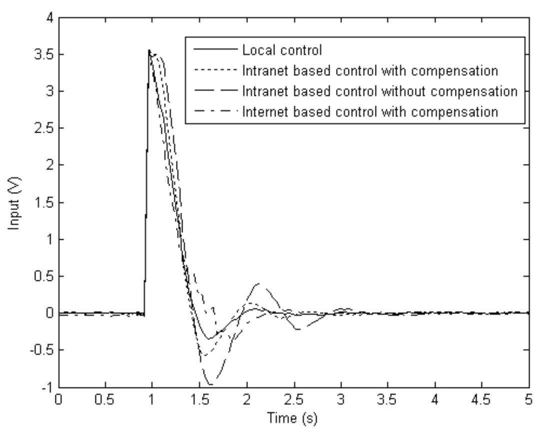

Fig. 12.8 Inputs of servo plant (Experiment)

predictive control scheme proposed in this chapter can compensate for the network delay actively. Its performance is very close to that of the local control scheme (i.e., no network). Although it is hard to make the model of the servo motor plant be exactly the same as the real practical one, the results of the real-time experiments are very similar to those of the real-time simulations.

12.5 Conclusion

The design and stability analysis of networked predictive control systems have been discussed. The network time delay in the feedback channel is considered in two cases: the fixed network delay and the random network delay. For both cases, the stability criteria have been obtained for networked predictive control. It has been concluded that the closed-loop networked predictive control system with bounded random network delay is stable if the corresponding switched system is stable. This provides a significant foundation for the design and analysis of networked control systems. Also, numerical simulations and practical experiments have successfully demonstrated the operation of the networked predictive control scheme proposed in this chapter.

Chapter 13
Analysis and Synthesis of NCSs with Random Forward and Feedback Delay

13.1 Introduction

The random network delays in the feedback channel and forward channel NCS have been studied, respectively. But, the random network delay in the forward and feedback channels makes the controller design and stability analysis much more difficult. This chapter proposes a predictive control scheme for networked control systems with random network delay in both the feedback and forward channels and also provides an analytical stability criterion for closed-loop networked predictive control systems. Some preliminary results have appeared in [120], however, the results obtained in this chapter are more integrated, including detailed analysis, simulations and practical experiments.

This chapter is organized as follows: Section 13.2 presents a networked predictive control scheme for systems with both forward and feedback network delays. Section 13.3 details the stability analysis of closed-loop networked predictive control systems for system with both constant and random network delay in both forward and feedback channels. Real-time simulations and practical experiments are presented in Section 13.4. Some conclusion remarks are given in Section 13.5.

13.2 Design of NPC Systems with both Forward and Feedback Network Delays

A networked predictive control scheme for NCS with random network delay in the forward and feedback channels is proposed. The main part of the scheme is the networked predictive controller, which compensates for the network delay and data dropout in the forward (from controller to actuator) and feedback (from sensor to controller) channels and achieves the desired

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 199–223.
springerlink.com

control performance. A very important characteristic of the network is that it can transmit a set of data at the same time.

In order to compensate for the network transmission delay, a network delay compensator in the channel from the controller to the actuator is proposed. It is assumed that control predictions at time t are packed and sent to the plant side through a network. On the actuator side, only the latest control prediction sequence is kept. The network delay compensator chooses the control value from the latest prediction control sequence. For example, if the latest predictive control sequence on the plant side is

$$\left[u^T_{t-i_t|t-i_t} \; u^T_{t-i_t+1|t-i_t} \; \cdots \; u^T_{t|t-i_t} \; \cdots \; u^T_{t+M-i_t|t-i_t} \right] \tag{13.1}$$

where i_t is a bounded random integer, the control value $u_{t|t-i_t}$ is available to be chosen as the control input of the plant at time t, the output of the network delay compensator will be

$$u_t = u_{t|t-i_t}. \tag{13.2}$$

From the above, it is shown that in the case of no network delay in the forward channel, the input to the plant actuator is the output of the controller. In the case of a delay iT, where T is the sampling period, the control input to the actuator is the ith-step ahead control prediction received in the current sampling period. If the data are lost within the current sampling period, the control input should be the ith-step ahead control prediction for the current time which is received in the previous sampling period. In the same way, when the time-delay is random and data dropouts happen in the feedback channel, the observer will use the measurement output y_{t-1} received at the last sampling instant if the measurement output y_t is lost or y_t is delayed, otherwise, y_{t-j} will be used if y_{t-i} arrives after y_{t-j}, where $j < i$. Thus, with the introduction of a bounded random integer k_t, measurement output y_{t-k_t}, denotes three types of measurement output transmitted in the feedback channel, i.e., random delay, data dropout and first sent late arrival. These methods play a very important role in compensating for time-delay and data dropout in the proposed networked predictive control implementation. When random network delays exist in both forward channel and feedback channel, the scheme proposed in this chapter can achieve the desired control performance, which is similar to that of the system without network time delay. Thus, on the sensor side, the measurement output is sent to the controller side through the feedback channel. On the controller side, the control prediction sequence at time t, which consists of the future control predictions, is packed and sent to the plant side through the forward channel. The network delay compensator chooses the latest control value from the control latest prediction sequence on the plant side. The networked predictive control system is shown in Fig. 13.1

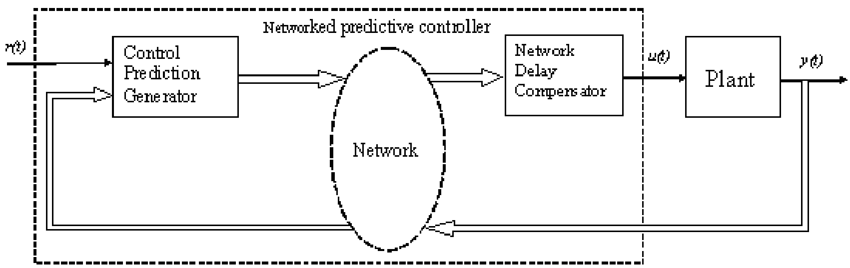

Fig. 13.1 The networked predictive control system

Consider a MIMO (multi-input multi-output) discrete system described in the following state space form

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t \end{aligned} \tag{13.3}$$

where $x_t \in R^n$, $u_t \in R^m$, and $y_t \in R^l$ are the state, input, and output vectors of the system, respectively, $A \in R^{n\times n}$, $B \in R^{n\times m}$ and $C \in R^{l\times n}$ the system matrices. For simplicity in carrying out the stability analysis, it is assumed that the reference input of the system is zero. Also, the following assumptions are made:

Assumption 13.2.1 *The pair (A, B) is completely controllable, and the pair (A, C) is completely observable.*

Assumption 13.2.2 *The upper bound of the network delay in the forward channel is not greater than M_1 multiples of the sampling period of system (M_1 is a positive integer). i.e.,* delay $\leq M_1T$. *For example, a delay in the network of 0.6 sec, and with a sampling period of 0.03 sec, then the multiple M_1 will be 20.*

Assumption 13.2.3 *The upper bound of the network delay in the feedback channel is not greater than N_1 multiples of the sampling period of system (N_1 is a positive integer).*

Assumption 13.2.4 *The number of consecutive data dropouts in the forward channel and feedback channel must be less than M_d and N_d , both of which are integer multiples of the sampling period of the system, respectively.*

Remark 13.1. In a practical NCS, if the data packet does not arrive at a destination in a certain transmission time, it means this data packet is lost, based on the commonly used network protocols. From the physical point of view, it is natural to assume that only a finite number of consecutive data dropouts can be tolerated in order to avoid the NCS becoming open-loop. Thus, the number of consecutive data dropouts in the channels from the controller to the actuator and from the sensor to the controller should be less than the finite numbers M_d and N_d, respectively. Similarly, the network delay from controller to actuator and from sensor to controller should also be bounded by finite numbers M_1 and N_1 multiple of the sampling period of the system, respectively. Clearly, the upper bound number of consecutive data dropouts and network delay should not be greater than the upper bound of the length of prediction (*i.e.* $N_1+N_d+M_1+M_d \leq M+N$, where $M+N$ is the length of the control prediction sequence, which is sent from the controller to the actuator.)

According to Assumption 13.2.1, the state observer is designed as

$$\hat{x}_{t+1|t} = A\hat{x}_{t|t-1} + Bu_t + L(y_t - C\hat{x}_{t|t-1}) \tag{13.4}$$

where $\hat{x}_{t+1|t} \in R^n$ and $u_t \in R^m$ are the one-step ahead state prediction and the input of the observer at time t and the matrix $L \in R^{n\times l}$, which can be designed using observer design approaches.

Following the state observer described by (13.4) , based on the output data up to $t-k$, where k is an integer multiple of the sampling period. The state predictions from time $t-k$ to t are constructed as

$$\begin{aligned}
\hat{x}_{t-k+1|t-k} &= A\hat{x}_{t-k|t-k-1} + Bu_{t-k} + L(y_{t-k} - C\hat{x}_{t-k|t-k-1}) \\
\hat{x}_{t-k+2|t-k} &= A\hat{x}_{t-k+1|t-k} + Bu_{t-k+1} \\
&\vdots \\
\hat{x}_{t|t-k} &= A\hat{x}_{t-1|t-k} + Bu_{t-1}
\end{aligned} \tag{13.5}$$

which results in

$$\begin{aligned}
\hat{x}_{t|t-k} = \ & A^{k-1}(A-LC)\hat{x}_{t-k|t-k-1} + \sum_{j=1}^{k} A^{k-j}Bu_{t-k+j-1} \\
& + A^{k-1}Ly_{t-k}, \ j = 1, 2, 3, \cdots, k.
\end{aligned} \tag{13.6}$$

When there is time-delay and data dropout in the forward channel, the state prediction from time t to $t+i$ are constructed by

$$\begin{aligned}
\hat{x}_{t+1|t-k} &= A\hat{x}_{t|t-k} + Bu_{t|t-k} \\
\hat{x}_{t+2|t-k} &= A\hat{x}_{t+1|t-k} + Bu_{t+1|t-k} \\
&\vdots \\
\hat{x}_{t+i|t-k} &= A\hat{x}_{t+i-1|t-k} + Bu_{t+i-1|t-k}
\end{aligned} \tag{13.7}$$

where i is an integer multiple of the sampling period. In particular, for the augmented system without time-delay, then, $u_t = K\hat{x}_{t|t-1}$, can be described as follows:

$$\begin{aligned}
\hat{x}_{t+1|t} &= (A + BK - LC)\hat{x}_{t|t-1} + LCx_t \\
x_{t+1} &= Ax_t + BK\hat{x}_{t|t-1}.
\end{aligned} \tag{13.8}$$

For the case of no network delay, the controller and observer can be designed separately based on Assumption 13.2.1. It is assumed that the state-feedback controller is designed by a modern control method, for example, LQG, pole assignment, H_2 and H_∞ in the presence of disturbance, etc. For the case where there are both the forward network delay i and feedback network delay k, the control predictions are calculated by

$$u_{t+i|t-k} = K\hat{x}_{t+i|t-k} \tag{13.9}$$

where the state feedback matrix is $K \in R^{m\times n}$. Thus,

$$\begin{aligned}\hat{x}_{t+i|t-k} &= (A+BK)^i \hat{x}_{t|t-k} \\ &= (A+BK)^i (A^{k-1}(A-LC)\hat{x}_{t-k|t-k-1} + \textstyle\sum_{j=1}^{k} A^{k-j}Bu_{t-k+j-1} \\ &\quad + A^{k-1}Ly_{t-k}).\end{aligned} \tag{13.10}$$

As a result, the output of the networked predictive control at time t is determined by

$$u_{t|t-k} = KA^{k-1}(A-LC)\hat{x}_{t-k|t-k-1} + \sum_{j=1}^{k} KA^{k-j}Bu_{t-k+j-1} + KA^{k-1}Ly_{t-k}. \tag{13.11}$$

From equation (13.11), it is clear that the future control predictions depend on the state estimation $\hat{x}_{t-k|t-k-1}$ and the past control input up to u_{t-1} and the past output up to y_{t-k} of the system. Since there exist the forward delay i and feedback delay k, the control input of the plant is designed as

$$u_t = u_{t|t-i-k}. \tag{13.12}$$

Combining this predictive controller with the networked delay compensator, both the forward and feedback network delays will be compensated within a certain range of time-delays. In the next section, stability analysis of the closed-loop system will be presented using this control scheme.

13.3 Stability Criteria of Closed-Loop NPC Systems

With the networked predictive control scheme proposed in this chapter, a very important problem is to study the stability of the closed-loop system. Firstly, the stability of the closed-loop system with constant network delay is investigated, and necessary and sufficient conditions for the closed-loop system to be stable is derived. Secondly, for the random time delay, the problem is more interesting because this case is closer to the time-delays experienced in a real network system. In this case, the stability problem of the closed-loop system is solved using the theory of switched systems.

13.3.1 Constant Delays in both Forward and Feedback Channels

In this case, it is assumed that the network delays i and k in the forward and feedback channels are constant. The first result is presented as follows.

Theorem 13.2. *For the networked predictive control systems with constant network delay i and k in the forward channel and feedback channel, respectively, the closed-loop system is stable if and only if all eigenvalues of the following matrix are within the unit circle.*

$$\Psi = \left[\begin{array}{ccccccccccccc}
A & 0 & 0 & \cdots & 0 & \cdots & 0 & B & 0 & 0 & \cdots & 0 & 0 \\
M_iA^{k-1}LC & 0 & 0 & \cdots & M_iB & \cdots & M_iA^{k-2} & M_iA^{k-1}B & 0 & 0 & \cdots & 0 & M_iA^{k-1}(A-LC) \\
0 & I & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & I & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & \cdots & I & 0 & 0 & 0 & \cdots & 0 & 0 \\
LCA^{k+i} & AB & LCAB & \cdots & LCA^{i+1}B & \cdots & LCA^{k+i-2}B & LCA^{k+i-1}B & A-LC & 0 & \cdots & 0 & 0 \\
0 & B & 0 & \cdots & 0 & \cdots & 0 & 0 & I & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & I & 0
\end{array}\right] \qquad (13.13)$$

where $\Psi \in R^{[(k+i)m+(k+i+2)n]\times[(k+i)m+(k+i+2)n]}$.

Proof. The necessary and sufficient condition for closed-loop system stability is that its closed-loop poles are stable. Using the networked predictive controller, from the predictive control obtained in the previous section, it is clear that

$$\begin{aligned} u_t = \; & K(A+BK)^i(A^{k-1}(A-LC)\hat{x}_{t-i-k|t-i-k-1} + \sum_{j=1}^{k} A^{k-j}Bu_{t-i-k+j-1} \\ & + A^{k-1}LCx_{t-i-k}) \end{aligned} \tag{13.14}$$

then,

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ &= Ax_t + Bu_{t|t-i-k} \\ &= Ax_t + BK(A+BK)^i(A^{k-1}(A-LC)\hat{x}_{t-i-k|t-i-k-1} \\ &\quad + \textstyle\sum_{j=1}^{k} A^{k-j}Bu_{t-i-k+j-1} + A^{k-1}LCx_{t-i-k}) \end{aligned} \tag{13.15}$$

and

$$\begin{aligned} \hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_t + L(y_t - C\hat{x}_{t|t-1}) \\ &= (A-LC)\hat{x}_{t|t-1} + LCx_t + BK(A+BK)^i(A^{k-1}(A-LC) \\ &\quad \times \hat{x}_{t-i-k|t-i-k-1} + \textstyle\sum_{j=1}^{k} A^{k-j}Bu_{t-i-k+j-1} + A^{k-1}LCx_{t-i-k}). \end{aligned} \tag{13.16}$$

Let $M_i = K(A+BK)^i$,

$$\begin{aligned} X(t+1) = \big[& x_{t+1}^T \; x_t^T \cdots x_{t-k+2}^T \; x_{t-k-i+1}^T \quad u_t^T \; u_{t-1}^T \cdots u_{t-i}^T \cdots u_{t-k-i+1}^T \\ & \hat{x}_{t+1|t}^T \; \hat{x}_{t|t-1}^T \cdots \hat{x}_{t-k-i+2|t-k-i+1} \; \hat{x}_{t-k-i+1|t-k-i} \big]^T \end{aligned}$$

$$\Lambda = \left[\begin{array}{cccccccccc} A & 0 & \cdots & 0 & BM_iA^{k-1}LC & 0 & \cdots & BM_iB & \cdots & BM_iA^{k-2}B \\ I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ 0 & 0 & \ddots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & M_iA^{k-1}LC & 0 & \cdots & M_iB & \cdots & M_iA^{k-2} \\ 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & I \\ LC & 0 & \cdots & 0 & BM_iA^{k-1}LC & 0 & \cdots & BM_iB & \cdots & BM_iA^{k-2}B \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 \end{array} \right.$$

$$
\left.\begin{array}{cccccc}
BM_iA^{k-1}B & 0 & 0 & \cdots & 0 & BM_iA^{k-1}(A-LC) \\
0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 \\
M_iA^{k-1}B & 0 & 0 & \cdots & 0 & M_iA^{k-1}(A-LC) \\
0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & 0 \\
BM_iA^{k-1}B & A-LC & 0 & \cdots & 0 & BM_iA^{k-1}(A-LC) \\
0 & I & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \ddots & 0 & 0 \\
0 & 0 & 0 & \cdots & I & 0
\end{array}\right] . \qquad (13.17)
$$

Then, the closed-loop system can be written as

$$
X(t+1) = \Lambda X(t). \qquad (13.18)
$$

Let

$$
\begin{aligned}
\bar{X}(t) = \ & \big[x_t \ x_{t-1} \ \cdots \ x_{t-k-i+1} \ x_{t-k-i} \ u'_{t-1} \ \cdots \ u_{t-k-i+1} \ u_{t-k-i} \\
& \hat{x}_{t|t-1} \ \cdots \ \hat{x}_{t-k-i+1|t-k-i} \ \hat{x}_{t-k-i|t-k-i-1} \big]
\end{aligned} \qquad (13.19)
$$

and a state transformation be

$$
\bar{X}(t+1) = \begin{bmatrix}
I & -A & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & I & -A & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & I & -A & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & I & 0 & \cdots & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & 0 \\
0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & I
\end{bmatrix} \bar{X}(t)
$$

then, by the state transformation

$$\begin{bmatrix} x''_t \\ x''_{t-1} \\ \vdots \\ x''_{t-k-i+1} \\ x''_{t-k-i} \\ u''_{t-1} \\ \vdots \\ u''_{t-k-i+1} \\ u''_{t-k-i} \\ \hat{x}''_{t|t-1} \\ \vdots \\ \hat{x}''_{t-k-i+1|t-k-i} \\ \hat{x}''_{t-k-i|t-k-i-1} \end{bmatrix} = \begin{bmatrix} I & 0 & \cdots & 0 & -B & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & I & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & -B & 0 & \cdots & 0 & I & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & I \end{bmatrix} \begin{bmatrix} x'_t \\ x'_{t-1} \\ \vdots \\ x'_{t-k-i+1} \\ x'_{t-k-i} \\ u'_{t-1} \\ \vdots \\ u'_{t-k-i+1} \\ u'_{t-k-i} \\ \hat{x}'_{t|t-1} \\ \vdots \\ \hat{x}'_{t-k-i+1|t-k-i} \\ \hat{x}'_{t-k-i|t-k-i-1} \end{bmatrix}$$

results in

$$\overline{\xi}''(t+1) = \Omega''\overline{\xi}''(t) \tag{13.20}$$

where

$$\begin{aligned} \overline{\xi}''(t) = \big[& x''^T_t \; x''^T_{t-1} \; \cdots \; x''^T_{t-k-i+1} \; x''^T_{t-k-i} \;\; u''^T_{t-1} \; \cdots \; u''^T_{t-k-i+1} \; u''^T_{t-k-i} \\ & \hat{x}''^T_{t|t-1} \; \cdots \; \hat{x}''^T_{t-k-i+1|t-k-i} \; \hat{x}''^T_{t-k-i|t-k-i-1} \big]^T \end{aligned}$$

$$
\Omega'' = \begin{bmatrix}
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \ddots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & 0 & \ddots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & I & A & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & M_iA^{k-1}LC & 0 & \cdots & M_iB & \cdots & M_iA^{k-2} & M_iA^{k-1}B & 0 & 0 & \cdots & 0 & M_iA^{k-1}(A-LC) \\
0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & I & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \ddots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
LC & LCA & \cdots & LCA^{k+i-1} & LCA^{k+i} & AB & \cdots & 0 & \cdots & 0 & 0 & A-LC & 0 & \cdots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 & B & \cdots & 0 & \cdots & 0 & 0 & I & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & I & 0
\end{bmatrix}. \tag{13.21}
$$

From the structure of equation (13.20), it can be concluded that the system is stable if and only if the following system is stable

$$Z(t+1) = \Xi Z(t) \tag{13.22}$$

where

$$Z(t) = [z_1^T(t)\ z_2^T(t)\ \cdots\ z_{3(k+i)+1}^T(t)]^T \tag{13.23}$$

$$
\Xi = \begin{bmatrix}
0 & \cdots & 0 & 0 & B & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\ddots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & \ddots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & I & A & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & M_iA^{k-1}LC & 0 & \cdots & M_iB & \cdots & M_iA^{k-2} & M_iA^{k-1}B & 0 & 0 & \cdots & 0 & M_iA^{k-1}(A-LC) \\
0 & \cdots & 0 & 0 & I & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & \cdots & I & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
\vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & I & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \ddots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
LCA & \cdots & LCA^{k+i-1} & LCA^{k+i} & AB & \cdots & 0 & \cdots & 0 & 0 & A-LC & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & 0 & B & \cdots & 0 & \cdots & 0 & 0 & I & 0 & \cdots & 0 & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & I & 0
\end{bmatrix}. \tag{13.24}
$$

By the state transformation

$$\begin{bmatrix} z'_1(t) \\ \vdots \\ z'_{k+i}(t) \\ z'_{k+i+1}(t) \\ z'_{k+i+2}(t) \\ \vdots \\ z'_{2(k+i)}(t) \\ z'_{2(k+i)+1}(t) \\ z'_{2(k+i)+2}(t) \\ \vdots \\ z'_{3(k+i)}(t) \\ z'_{3(k+i)+1} \end{bmatrix} = \begin{bmatrix} I & 0 & \cdots & 0 & 0 & -B & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 & -B & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & I & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & I & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & I & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & I \end{bmatrix} \begin{bmatrix} z_1(t) \\ \vdots \\ z_{(k+i)}(t) \\ z_{k+i+1}(t) \\ z_{k+i+2}(t) \\ \vdots \\ z_{2(k+i)}(t) \\ z_{2(k+i)+1}(t) \\ z_{2(k+i)+2}(t) \\ \vdots \\ z_{3(k+i)}(t) \\ z_{3(k+i)+1} \end{bmatrix}$$

the following system is derived

$$Z'(t+1) = \Xi' Z'(t) \tag{13.25}$$

where

$$\begin{aligned} Z'(t) = \Big[& z_1'^T(t) \cdots z_{(k+i)}'^T(t) \; z_{k+i+1}'^T(t) \; z_{k+i+2}'^T(t) \cdots z_{2(k+i)}'^T(t) \\ & z_{2(k+i)+1}'^T(t) \; z_{2(k+i)+2}'^T(t) \cdots z_{3(k+i)}'^T(t) \; z_{3(k+i)+1}'^T \Big]^T \end{aligned} \tag{13.26}$$

$$
\Xi' = \left[\begin{array}{cccccccccccccccc}
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\ddots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
0 & \ddots & 0 & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & I & A & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & M_iA^{k-1}LC & 0 & 0 & \cdots & M_iB & \cdots & M_iA^{k-2} & M_iA^{k-1}B & 0 & 0 & \cdots & 0 & M_iA^{k-1}(A-LC) \\
0 & \cdots & 0 & 0 & I & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & I & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & I & 0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & \ddots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\
LCA & \cdots & LCA^{k+i-1} & LCA^{k+i} & AB & LCAB & \cdots & LCA^{k+i-1}B & \cdots & LCA^{k+i-2}B & LCA^{k+i-1}B & A-LC & 0 & \cdots & 0 & 0 \\
0 & \cdots & 0 & 0 & B & 0 & \cdots & 0 & \cdots & 0 & 0 & I & 0 & \cdots & 0 & 0 \\
\vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots & \vdots & \ddots & \cdots & \vdots & \vdots \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\
0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & I & 0
\end{array}\right]. \tag{13.27}
$$

From the structure of equation (13.25), it can be concluded that the system is stable if and only if the following system is stable

$$
Z''(t+1) = \Psi Z''(t) \tag{13.28}
$$

where Ψ is defined in (13.13). Therefore, the closed-loop system is stable if and only if all eigenvalues of matrices (13.13) are within the unit circle.

13.3.2 Random Network Delay

From Assumptions 13.2.1-13.2.4, it is assumed that the network delay and consecutive data dropout in the forward and feedback channels are random but bounded by M and N respectively, that is, for integers i_t and k_t, $i_t \in \{0, 1, 2, \cdots, M\}$, $k_t \in \{0, 1, 2, \cdots, N\}$. Then, i and k in the constant case are replaced by i_t and k_t, respectively. Similar results to the constant case can be obtained. However, the stability of the closed-loop NPCS will be determined by all $(M + 1) \times (N + 1)$ matrices as $i_t \in \{0, 1, 2, \cdots, M\}$ and $k_t \in \{0, 1, 2, \cdots, N\}$.

Under this control scheme, the closed-loop system will be a switched linear system, the theory of switched system can be used to design the observer gain L and feedback gain K such that the closed-loop system with random time-delay is stable ([172]).

In order to present the results in this section, for simplicity, let

$$\Lambda(0,0) = \begin{bmatrix} \overbrace{A\; 0 \cdots 0\; 0}^{(M+N+1)n} & \overbrace{0 \cdots 0\; 0}^{(M+N)m} & BK & 0 \cdots 0\; 0 \\ I\; 0 \cdots 0\; 0 & 0 \cdots 0\; 0 & 0 & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots\; \vdots & \vdots \cdots \vdots\; \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots I\; 0 & 0 \cdots 0\; 0 & 0 & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0\; 0 & 0 \cdots 0\; 0 & K & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0\; 0 & I \cdots 0\; 0 & 0 & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots\; \vdots & \vdots \cdots \vdots\; \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0\; 0 & 0 \cdots I\; 0 & 0 & 0 \cdots 0\; 0 \\ LC\; 0 \cdots 0\; 0 & 0 \cdots 0\; 0 & A + BK - LC & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0\; 0 & 0 \cdots 0\; 0 & I & 0 \cdots 0\; 0 \\ \vdots\; \vdots \cdots \vdots\; \vdots & \vdots \cdots \vdots\; \vdots & \vdots & \vdots \cdots \vdots\; \vdots \\ 0\; 0 \cdots 0\; 0 & 0 \cdots 0\; 0 & 0 & 0 \cdots 0\; 0 \\ 0\; 0 \cdots 0\; 0 & 0 \cdots 0\; 0 & 0 & 0 \cdots I\; 0 \end{bmatrix} \tag{13.29}$$

$\Lambda(0,0) \in R^{[2n(M+N)+m(M+N)+2n] \times [2n(M+N)+m(M+N)+2n]}$

$$
\Lambda(i_t,0) =
\left[\begin{array}{ccccccccc}
A & \overbrace{0\cdots}^{i_t n} & M_1(i_t) & \cdots 0 & \overbrace{0\cdots 0\;0}^{(M+N)m} & 0 & \cdots & M_2(i_t) & \cdots 0\;0 \\
I & 0\cdots & 0 & \cdots 0 & 0\cdots 0\;0 & 0 & \cdots & 0 & \cdots 0\;0 \\
\vdots & \vdots\cdots & \vdots & \cdots\vdots & \vdots\cdots\vdots\;\vdots & \vdots & \cdots & \vdots & \cdots\vdots\;\vdots \\
0 & 0\cdots & I & \cdots 0 & 0\cdots 0\;0 & 0 & \cdots & 0 & \cdots 0\;0 \\
0 & 0\cdots & M_2(i_t) & \cdots 0 & 0\cdots 0\;0 & 0 & \cdots & M_4(i_t) & \cdots 0\;0 \\
0 & 0\cdots & 0 & \cdots 0 & I\cdots 0\;0 & 0\cdots & 0 & \cdots & 0\;0 \\
\vdots & \vdots\cdots & \vdots & \cdots\vdots & \vdots\cdots\vdots\;\vdots & \vdots & \cdots & \vdots & \cdots\vdots\;\vdots \\
0 & 0\cdots & 0 & \cdots 0 & 0\cdots I\;0 & 0 & \cdots & 0 & \cdots 0\;0 \\
LC & 0\cdots & 0 & \cdots 0 & 0\cdots 0\;0 & M_5(i_t) & \cdots & 0 & \cdots 0\;0 \\
0 & 0\cdots & 0 & \cdots 0 & 0\cdots 0\;0 & I & \cdots & 0 & \cdots 0\;0 \\
\vdots & \vdots\cdots & \vdots & \cdots\vdots & \vdots\cdots\vdots\;\vdots & \vdots & \cdots & \vdots & \cdots\vdots\;\vdots \\
0 & 0\cdots & 0 & \cdots 0 & 0\cdots 0\;0 & 0 & \cdots & 0 & \cdots 0\;0 \\
0 & 0\cdots & 0 & \cdots 0 & 0\cdots 0\;0 & 0 & \cdots & 0 & \cdots I\;0
\end{array}\right]
\in R^{[2n(M+N)+m(M+N)+2n]\times[2n(M+N)+m(M+N)+2n]}
$$

where $i_t = 1, 2, \cdots, N_{ca}$, $M_1(i_t) = BK(A+BK)^{i_t-1}LC$, $M_2(i_t) = BK(A+BK)^{i_t-1}(A+BK-LC)$, $M_3(i_t) = LC$, $M_4(i_t) = K(A+BK)^{i_t-1}(A+BK-LC)$, $M_5(i_t) = BK(A+BK)^{i_t-1}LC$.

$$
\Lambda(i_t,k_t) = \begin{bmatrix} \Lambda_{11}(i_t,k_t) & \Lambda_{12}(i_t,k_t) & \Lambda_{13}(i_t,k_t) \\ \Lambda_{21}(i_t,k_t) & \Lambda_{22}(i_t,k_t) & \Lambda_{23}(i_t,k_t) \\ \Lambda_{31}(i_t,k_t) & \Lambda_{32}(i_t,k_t) & \Lambda_{33}(i_t,k_t) \end{bmatrix}
\in R^{[2n(M+N)+m(M+N)+2n]\times[2n(M+N)+m(M+N)+2n]} \quad (13.30)
$$

for $i = 1, 2, \cdots, M$ and $k_t = 1, 2, \cdots, N$, where $M_{i_t} = K(A+BK)^{i_t}$

$$\Lambda_{11}(i_t,k_t) = \begin{bmatrix} A & \overbrace{0 \cdots 0}^{(i_t+k_t-1)n} & BM_{i_t}A^{k_t-1}LC & 0 \cdots 0 & 0 \\ I & 0 \cdots 0 & 0 & 0 \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0 & 0 \\ 0 & 0 \cdots I & 0 & 0 \cdots 0 & 0 \\ \vdots & \vdots \cdots \vdots & \vdots & \vdots \cdots \vdots & \vdots \\ 0 & 0 \cdots 0 & 0 & 0 \cdots 0 & 0 \\ 0 & 0 \cdots 0 & 0 & 0 \cdots I & 0 \end{bmatrix} \in R^{[n(M+N)+n]\times[n(M+N)+n]}$$

$$\Lambda_{12}(i_t,k_t) = \begin{bmatrix} \overbrace{0 \cdots 0}^{i_t m} \; BM_{i_t}B \cdots BM_{i_t}A^{k_t-2}B \quad BM_{i_t}A^{k_t-1}B \quad 0 \cdots 0 \; 0 \\ 0_{[n(M+N)]\times[m(M+N)]} \end{bmatrix} \in R^{[n(M+N)+n]\times[m(M+N)]}$$

$$\Lambda_{13}(i_t,k_t) = \begin{bmatrix} \overbrace{0\; 0 \cdots 0}^{(i_t+k_t)n} \quad BM_{i_t}A^{k_t-1}(A-LC) \; 0 \cdots 0 \; 0 \\ 0_{[n(M+N)]\times[n(M+N)+n]} \end{bmatrix} \in R^{[n(M+N)+n]\times[n(M+N)+n]}$$

$$\Lambda_{21}(i_t,k_t) = \begin{bmatrix} \overbrace{0\; 0 \cdots 0}^{(i_t+k_t)n} \quad M_{i_t}A^{k_t-1}LC \; 0 \cdots 0 \; 0 \\ 0_{[m(M+N-1)]\times[n(M+N)+n]} \end{bmatrix} \in R^{[m(M+N)]\times[n(M+N)+n]}$$

$$\Lambda_{22}(i_t,k_t) = \begin{bmatrix} \overbrace{0 \cdots 0}^{i_t m} & M_{i_t}B & M_{i_t}AB & \cdots & M_{i_t}A^{k_t-2}B & M_{i_t}A^{k_t-1}B & 0 \cdots & 0 & 0 \\ I \cdots 0 & 0 & 0 & \cdots & 0 & 0 & 0 \cdots & 0 & 0 \\ \vdots \cdots \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \cdots & \vdots & \vdots \\ 0 \cdots 0 & 0 & 0 & \cdots & 0 & 0 & 0 \cdots & 0 & 0 \\ 0 \cdots 0 & 0 & 0 & \cdots & I & 0 & 0 \cdots & 0 & 0 \\ \vdots \cdots \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \; \vdots \cdots & \vdots & \vdots \\ 0 \cdots 0 & 0 & 0 & \cdots & 0 & 0 & 0 \cdots & 0 & 0 \\ 0 \cdots 0 & 0 & 0 & \cdots & 0 & 0 & 0 \cdots & I & 0 \end{bmatrix} \in R^{[m(M+N)]\times[m(M+N)]}$$

$$\Lambda_{23}(i_t,k_t)=\begin{bmatrix} \overbrace{0\,0\cdots 0}^{(i_t+k_t)n} & M_{i_t}A^{k_t-1}(A-LC)\;0\cdots 0\;0 \\ 0_{[m(M+N-1)]\times[n(M+N)+n]} & \end{bmatrix}$$
$$\in R^{[m(M+N)]\times[n(M+N)+n]}$$

$$\Lambda_{31}(i_t,k_t)=\begin{bmatrix} LC & \overbrace{0\cdots 0}^{(i_t+k_t-1)n} & 0\,0\cdots 0\,0 \\ 0_{[n(M+N)]\times[n(M+N)+n]} & & \end{bmatrix}$$
$$\in R^{[n(M+N)+n]\times[n(M+N)+n]}$$

$$\Lambda_{32}(i_t,k_t)=0_{(n(M+N)+n)\times(m(M+N))}$$

$$\Lambda_{33}(i_t,k_t)=\begin{bmatrix} A+BK-LC & \overbrace{0\cdots 0}^{(i_t+k_t-1)n} & 0\,0\cdots 0\,0 \\ I & 0\cdots 0 & 0\,0\cdots 0\,0 \\ \vdots & \vdots\cdots\vdots & \vdots\;\vdots\cdots\vdots\;\vdots \\ 0 & 0\cdots I & 0\,0\cdots 0\,0 \\ 0 & 0\cdots 0 & I\,0\cdots 0\,0 \\ \vdots & \vdots\cdots\vdots & \vdots\;\vdots\cdots\vdots\;\vdots \\ 0 & 0\cdots 0 & 0\,0\cdots 0\,0 \\ 0 & 0\cdots 0 & 0\,0\cdots I\,0 \end{bmatrix}.$$
$$\in R^{(n(M+N)+n)\times(n(M+N)+n)}$$

Then, the main results in this section can be stated as follows:

Theorem 13.3. *For the networked predictive control system with random network delays i_t and k_t in both forward and feedback channels, respectively, the closed-loop system is stable if there exists a positive definite matrix*

$$P\in R^{[2n(M+N)+m(M+N)+2n]\times[2n(M+N)+m(M+N)+2n]}$$

such that

$$\Lambda^T(i_t,k_t)P\Lambda(i_t,k_t)-P<0 \tag{13.31}$$

for $i_t=0,1,2,\cdots,M$ and $k_t=0,1,2,\cdots,N$.

Proof. Following the case of the constant network delay, if the random network delay in the forward channel is $k_t, k_t\neq 0$, the control input, state prediction, and plant state vectors are expressed by

$$\begin{aligned} u_t = {} & K(A+BK)^{i_t}[A^{k_t-1}(A-LC)\hat{x}_{t-i_t-k_t|t-i_t-k_t-1} \\ & +\textstyle\sum_{j=1}^{k_t}A^{k_t-j}Bu_{t-i_t-k_t+j-1}+A^{k_t-1}LCx_{t-i_t-k_t}] \end{aligned} \tag{13.32}$$

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ &= Ax_t + Bu_{t|t-i_t-k_t} \\ &= Ax_t + BK(A+BK)^{i_t}[A^{k_t-1}(A-LC)\hat{x}_{t-i_t-k_t|t-i_t-k_t-1} \\ &\quad + \textstyle\sum_{j=1}^{k_t} A^{k_t-j}Bu_{t-i_t-k_t+j-1} + A^{k_t-1}LCx_{t-i_t-k_t}]. \end{aligned} \tag{13.33}$$

As the input and output of the closed system are sent through the network, control u_t can be used by the actuator with the implementation of the network delay compensator, but it can not be obtained by the observer previously due to the network-induced delay, therefore, the observer is designed based on (y_{t-k_t}, u_{t-k_t}) received on the side of the observer

$$\hat{x}_{t-k_t+1|t-k_t} = A\hat{x}_{t-k_t|t-k_t-1} + Bu_{t-k_t} + L(y_{t-k_t} - C\hat{x}_{t-k_t|t-k_t-1}). \tag{13.34}$$

When there is no delay in the feedback channel, i.e., $k_t = 0$, and $i_t \neq 0$, then,

$$\begin{aligned} \hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_t + L(y_t - C\hat{x}_{t|t-1}) \\ \hat{x}_{t+2|t} &= A\hat{x}_{t+1|t} + Bu_{t+1|t} \\ &\vdots \\ \hat{x}_{t+i_t|t} &= A\hat{x}_{t+i_t-1|t} + Bu_{t+i_t-1|t}. \end{aligned} \tag{13.35}$$

Let $u_t = u_{t|t-i_t} = K\hat{x}_{t|t-i_t}$, then,

$$\hat{x}_{t|t-i_t} = (A+BK)^{i_t-1}[(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} + LCx_{t-i_t}] \tag{13.36}$$

$$u_t = K(A+BK)^{i_t-1}[(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} + LCx_{t-i_t}]. \tag{13.37}$$

The closed-loop system is

$$\begin{aligned} x_{t+1} &= Ax_t + BK(A+BK)^{i_t-1}LCx_{t-i_t} + \\ &\quad BK(A+BK)^{i_t-1}(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} \\ \hat{x}_{t+1|t} &= (A+BK-LC)\hat{x}_{t|t-1} + LCx_t \\ u_t &= K(A+BK)^{i_t-1}[(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} + LCx_{t-i_t}] \end{aligned} \tag{13.38}$$

then, combination of equations (13.32), (13.33), (13.34) and (13.38) gives the following augmented system:

$$X_{t+1} = \Lambda(i_t, k_t)X_t \tag{13.39}$$

where

$$\begin{aligned} X_t = \big[& x_t^T \; x_{t-1}^T \cdots x_{t-i_t-k_t+1}^T \; x_{t-i_t-k_t}^T \; x^T(t-i_t-k_t-1) \cdots x_{t-M-N+1}^T \\ & x_{t-M-N}^T \; u_{t-1}^T \; u_{t-2}^T \cdots u_{t-i_t-k_t+1}^T \; u_{t-i_t-k_t}^T \; u^T(t-i_t-k_t-1) \cdots \\ & u_{t-M-N+1}^T \; u_{t-M-N}^T \; \hat{x}_{t|t-1}^T \; x_{t-1|t-2}^T \cdots \hat{x}_{t-i_t-k_t+1|t-i_t-k_t}^T \\ & \hat{x}_{t-i_t-k_t|t-i_t-k_t-1}^T \; x^T t-i_t-k_t-1|t-i_t-k_t-2 \cdots \\ & \hat{x}_{t-M-N+1|t-M-N}^T \; \hat{x}_{t-M-N|t-M-N-1}^T \big]^T \end{aligned} \tag{13.40}$$

$i_t = 1, 2, \cdots, M$, $k_t = 0, 1, 2, \cdots, N$, $\Delta(i_t, 0)$ for $i_t = 1, 2, \cdots, N_{ca}$ and $k_t = 0$ are defined in (13.30). While there is no time-delay in the forward and feedback channels, i.e., $i_t = 0$ and $k_t = 0$, the augmented system (13.8) is extended as

$$X_{t+1} = \Lambda(0, 0)X_t \tag{13.41}$$

where $\Delta(0, 0)$ for $i_t = 0, k_t = 0$ is described in (13.29). As the time-delay is random, the closed-loop system is a switched system which is composed of $(M + 1) \times (N + 1)$ discrete-time subsystems, i.e.,

$$X_{t+1} = \Lambda(i_t, k_t)X_t \tag{13.42}$$

where $i_t = 0, 1, \cdots, N$ and $k_t = 0, 1, \cdots, N$. The switched system can be described as

$$X_{t+1} = \Lambda_{\delta(k)}X_t \tag{13.43}$$

where $\delta(k) : \{0, 1, \cdots\} \to \{0, 1, 2, \cdots, M\} \times \{0, 1, 2, \cdots, N\}$, $\delta(k)$ is switching signal.

Let $V_{k_t} = X_t^T P X_t$, then

$$\begin{aligned} V_{t+1} - V_t &= X_{t+1}^T P X_{t+1} - X_t^T P X_t \\ &= X_t^T [\Lambda_{\delta(k)}^T P \Lambda_{\delta(k)} - P] X_t^T. \end{aligned} \tag{13.44}$$

From (13.31), it follows that $V_{t+1} - V_t < 0$, for $\delta(k)$. Therefore, system (13.39) is stable.

Remark 13.4. It can be deduced easily from (13.31) that each subsystem is stable. If K and L are designed previously, then (13.31) is a set of LMIs. LMI tool-box can be used to find a feasible solution P ([15]).

13.4 Simulation & Experiments

In this section, two illustrative examples are constructed to verify the design method developed in this chapter.

13.4.1 Numerical Simulation

Example 13.5. An illustrative example is constructed to verify the design method developed in this chapter, it is multi-input and multi-output system, this simulation not only contributes to the theory development, but also solves the practical problems which are modelled as the similar state-space equations. Simulation studies are presented for an open-loop unstable discrete system in the form of (13.3) with the following system matrices:

$$A = \begin{bmatrix} 1.010 & 0.271 & -0.488 \\ 0.482 & 0.100 & 0.24 \\ 0.002 & 0.3681 & 0.7070 \end{bmatrix}, B = \begin{bmatrix} 5 & 5 \\ 3 & -2 \\ 5 & 4 \end{bmatrix}, C = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 3 & 1 \end{bmatrix}.$$

Take

$$K = \begin{bmatrix} 0.5858 & -0.1347 & -0.4543 \\ -0.5550 & 0.0461 & 0.4721 \end{bmatrix}, L = \begin{bmatrix} -0.3614 & 0.3326 \\ 0.0332 & 0.0576 \\ 0.2481 & -0.0750 \end{bmatrix}$$

for nominal system (13.3) such that the closed-loop system is stable. It can be concluded from the simulation that the closed-loop system is not stable with $u_t = K\hat{x}_{t-i-k+1|t-i-k}$, however, it is stable with control $u_{t+i|t-k} = K\hat{x}_{t|t-k-i}$ for $i = 0, 1$ and $k = 0, 1, 2$. By Theorem 13.3, there exists a common positive definite matrix $P \in R^{30\times 30}$ such that six inequalities in (13.31) are satisfied, so the closed-loop system (13.3) is stable with random time-delay $i = 0, 1$ and $k = 0, 1, 2$. The following figures are the response of the close-loop system with time-delay $i = 1$ and $k = 2$. The initial conditions are

$$x_0 = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.1 \end{bmatrix}, u(t) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \text{ for } t = 0,\ 1,\ 2,\ 3.$$

13.4.2 Practical Experiments

Example 13.6. To implement networked control systems, a test rig is built, based on two ARM 9 embedded boards. The two boards are used, one on the controller side and the other on the plant side and are connected through the network. The communication protocol between them is UDP. The kernel chip of the embedded board is ATMEL's AT91RM9200, which is a cost-effective, high-performance 32-bit RISC microcontroller for Ethernet- based embedded systems. A 10M/100M self-adaptive network controller is integrated in the chip and the chip also has a high computing performance and can work at speeds up to 180 MHz. 2-channel 16-bit high speed analog-digital (D/A) converters and 8-channel 16-bit high speed analog-digital (A/D) converters in the controller board provide I/O interfaces for the controlled plant. In order to validate the proposed method, a servo motor control system which consists of a DC motor, load plate, speed and angle sensors is considered. The model of the motor control plant at sampling period 0.04 second is identified to be

$$G(z^{-1}) = \frac{A(z^{-1})}{B(z^{-1})} = \frac{0.05409z^{-2} + 0.115z^{-3} + 0.0001z^{-4}}{1 - 1.12z^{-1} - 0.213z^{-2} + 0.335z^{-3}}.$$

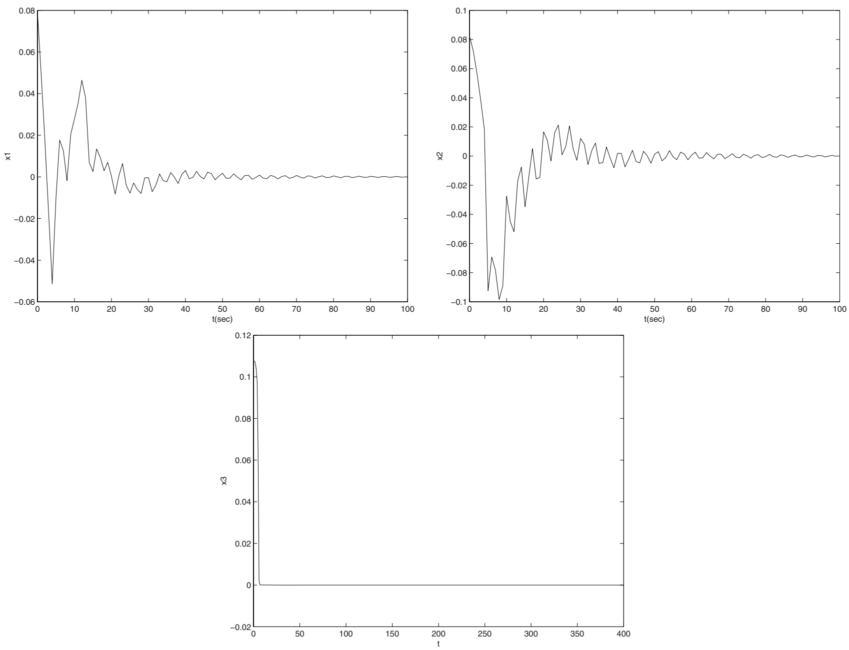

Fig. 13.2 States x_1, x_2 and x_3

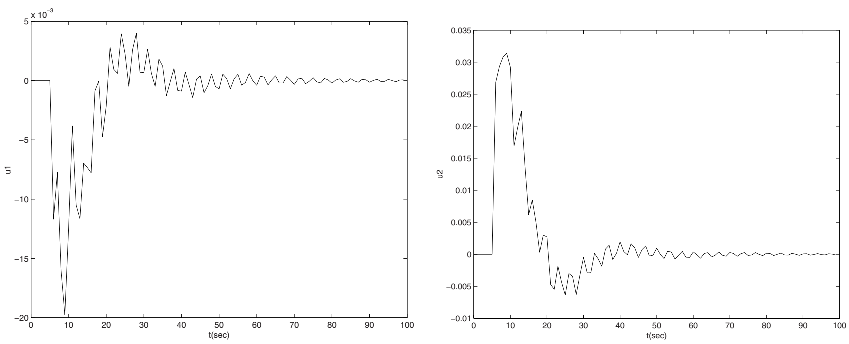

Fig. 13.3 Control inputs u_1 and u_2

The system can also be written in state space form with the following system matrices

$$A = \begin{bmatrix} 1.12 & 0.213 & -0.335 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 0.0541 & 0.1150 & 0.0001 \end{bmatrix}.$$

The matrices K and L are designed to be

$$K = \begin{bmatrix} -0.0270 & -0.575 & -0.0001 \end{bmatrix}, \; L = \begin{bmatrix} 6 \\ 6 \\ 6 \end{bmatrix}$$

which ensure the closed-loop system without time delay is stable.

To illustrate the operation of the proposed networked predictive control scheme, three cases are considered: a) Local control. There is no network in the closed-loop system, i.e., the output signal from the sensor is directly connected to the controller, so, the network delay is zero. The design of matrices K and L has ensured that the closed-loop system is stable. b) Intranet based control. In this case, the output and input signals are physically transmitted between two Intranet IP addresses 193.63.131.217 and 193.63.131.219 which are both located on the campus network of the University of Glamorgan. It is measured that the maximum network delay between the two embedded boards is 0.08 second, so the sum of the feedback and forward channel time delay is less than 0.16 seconds. As the sample time is 0.04s, the upper bound N=4. c) Internet based control. In this case, the output and input signals were transmitted between two Internet IP addresses 193.63.131.219 and 81.106.241.34. The former is located at the University of Glamorgan, UK., the latter is located off campus in Pontypridd, UK. The maximum network delay is measured to be 0.16 between the two embedded board, so the sum of the two channel time delay is less than 0.32s. The sampling period is still 0.04 second, so the upper bound N = 8. To evaluate the performance of the networked predictive control scheme, one real-time simulation and one real-time experiment are carried out.

1) Real-time simulation. In this simulation, the servo motor plant to be controlled is represented by its model but the network is a real one. The simulations are performed using Matlab/Simulink/Real time Workshop. The real-time simulation diagram is shown in Fig. 13.4, which is separated into two parts: the controller side and the plant side. The two parts are implemented in two ARM 9 embedded boards respectively. The reference input is a square wave generated by the pulse generator block, which changes between 0v to 7v with period 5s. The block Netctrl is the control prediction generator. The block Comp is the network delay compensator. The Blocks Recv9 and Send9 are the receiver and sender of the UDP communication protocol. All of them are designed using Matlab S-Functions. The simulated plant and the controller are executed in an ARM 9 embedded system. The real network (Intranet or Internet) is between UDP communication blocks Recv9 and Send9 in Fig. 13.4.

Four real time simulations are conducted: local control (i.e., no network), Intranet based control without delay compensation, Intranet based control with delay compensation and Internet based control with delay compensation. The real-time simulation results are shown in Fig. 13.5 and Fig. 13.6. The difference between the results in the case of local control and Intranet control without compensation is very small, which is hard to distinguish in Fig. 13.5

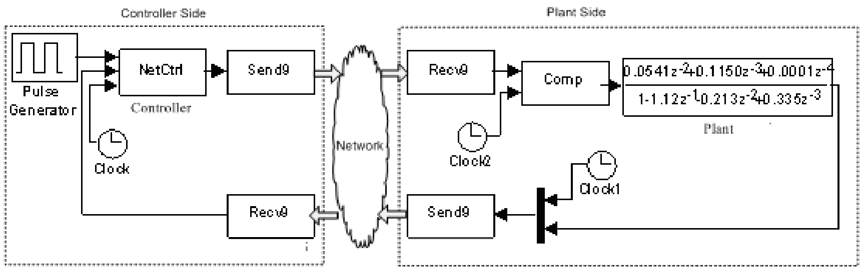

Fig. 13.4 Simulation Diagram

and Fig. 13.6. The Internet based control without delay compensation is also conducted, but it is found that the system is no longer stable due to the large network delay, which is between 0.16-0.32s.

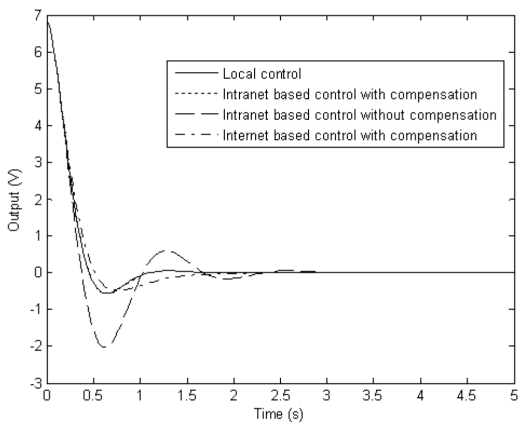

Fig. 13.5 Outputs of servo plant (Simulation)

2) Real-time experiment. The difference between the real-time simulation and real-time experiments is that the plant model of the servo motor in the real-time simulation is replaced by D/A block Dac9 and A/D block Adc9 and the real servo motor. The diagram of the real-time experiment is shown in Fig. 13.7. The two blocks Dac9 and Adc9 are the driver of the A/D and D/A channels in the embedded system and are designed in Matlab S-Function. Similarly, four real-time experiments are made: local control (i.e., no network), Intranet based control without delay compensation, Intranet based control with delay compensation and Internet based control with delay compensation. The real-time experiment results are shown in Fig. 13.8 and Fig. 13.9.

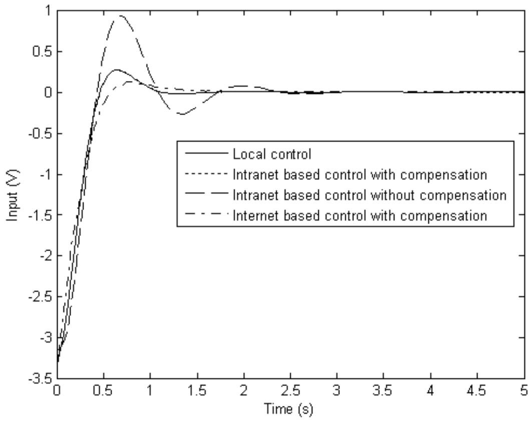

Fig. 13.6 Inputs of servo plant (Simulation)

Also, it is found that the Internet based control without delay compensation is unstable.

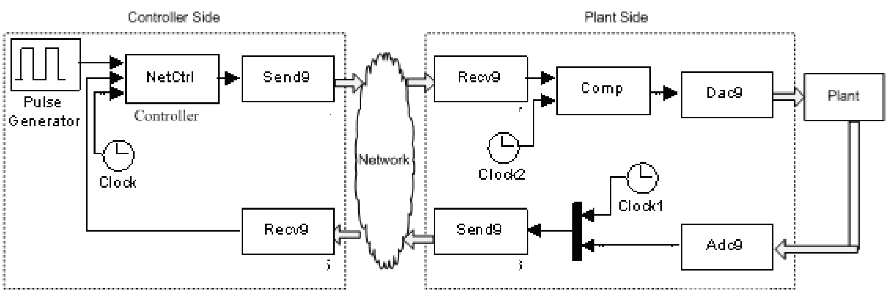

Fig. 13.7 Experiment Diagram

13.5 Conclusion

The design and stability analysis of networked predictive control systems have been discussed in this chapter. The networked predictive control compensates for the network delay and data dropout in the forward and feedback channels and achieves the desired control performance. The network delays in both forward and feedback channels have been considered in two cases: the fixed network delay and the random network delay. For both cases, the stability

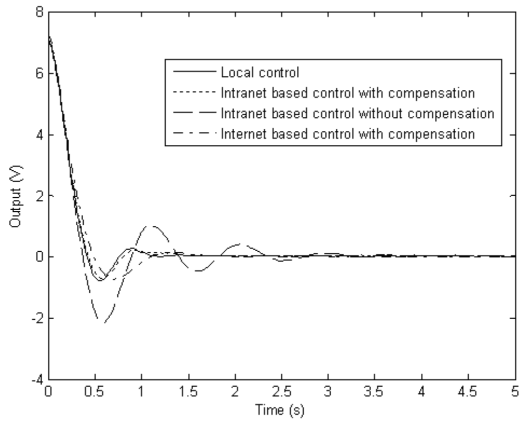

Fig. 13.8 Outputs of servo plant (Experiment)

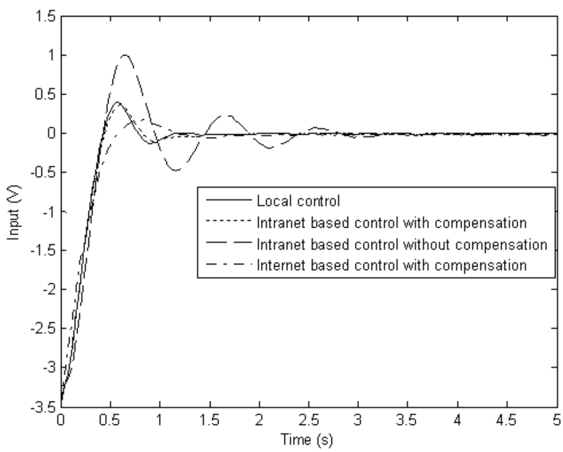

Fig. 13.9 Inputs of servo plant (Experiment)

criteria have been obtained for networked predictive control. Particularly, in the case of random network delay, it has been concluded that the closed-loop networked predictive control system is stable if the corresponding switched system is stable. This gives fundamental results for the design and analysis of networked control systems.

Chapter 14
Stochastic Analysis of NCSs with Random Delay and Data Dropout

14.1 Introduction

The stability problem of closed-loop NCS in the presence of network delays and data packet drops has been addressed in [214, 211, 210, 149]. In [142, 196, 182], the problem of stochastic stability of network worked control systems with random time-delay or data dropout has been discussed, in which the random time-delay is modelled as a Markov process. Furthermore, in [213], the sensor-to-controller and controller-actuator delays are modeled as two Markov chains, and the resulting closed-loop systems are jump linear systems with two modes, where state feedback is proposed, the last received state is used for feedback again if there are delays longer than one sampling period or if there is a package loss. In the work of [117], the networked predictive control with the modified MPC is proposed, but the time-delay in the forward channel is constant. In this chapter, networked predictive control will be proposed for control of NCSs with random network delay in both the forward channel and feedback channel, where the network induced-delay is in the form of a Markov chain.

This chapter is organized as follows. Section 14.2 gives the problem formulation and the scheme of predictive control and Section 14.3 presents the analysis of stochastic stability of the closed-loop system. Section 14.4 gives a numerical example and a practical experiment to demonstrate the effectiveness of the proposed method. Some conclusion remarks are given in Section 14.5.

14.2 Predictive Control of Networked Systems

To overcome unknown network transmission delay, a networked predictive control scheme is proposed. It mainly consists of a control prediction generator

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 225–242.
springerlink.com

and a network delay compensator. The control prediction generator is designed to generate a set of future control predictions. The network delay compensator is used to compensate for the unknown random network delay. Consider an MIMO discrete system described in the following state space form

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t \end{aligned} \tag{14.1}$$

where $x_t \in R^n$, $u_t \in R^m$, and $y_t \in R^l$ are the system state, input, and output vectors, respectively. $A,\ B,\ C$ are matrices of appropriate dimensions. The following assumptions can reasonably be made:

1 The pair, (A, B), is completely controllable, and the pair, (A, C) is completely observable.
2 The upper bound of the network delay from the sensor to the controller and from the controller to the actuator are not greater than N_{scd} and N_{cad}, respectively (positive integers).
3 The number of consecutive data dropouts in the channels from sensor to controller and from controller to actuator must be less than N_{dsc} and N_{dca} (positive integers).

Remark 14.1. In a practical NCS, if the data packet does not arrive at a destination in a certain transmission time, it means this data packet is lost, based on the commonly used network protocols. From the physical point of view, it is natural to assume that only a finite number of consecutive data dropouts can be tolerated in order to avoid the NCS becoming open-loop. Thus, the number of consecutive data dropouts in the channels from the sensor to the controller and from the controller to the actuator should be less than finite number N_{dsc} and N_{dca}, respectively. Similarly, the network delay from sensor to controller and from controller to actuator should also be bounded by finite numbers N_{scd} and N_{cad}, respectively. Clearly, the upper bound number of consecutive data dropouts and time delay should not be greater than the upper bounds N_{sc} and N_{ca} (*i.e.* $N_{dsc} + N_{scd} \leq N_{sc}$ and $N_{dca} + N_{cad} \leq N_{ca}$). $N_{ca} + N_{sc}$ is the length of prediction.

The state observer is designed as

$$\hat{x}_{t+1|t} = A\hat{x}_{t|t-1} + Bu_{t|t} + L(y_t - C\hat{x}_{t|t-1}) \tag{14.2}$$

where $\hat{x}_{t+1|t} \in R^n$ and $u_{t|t} \in R^m$ are the one-step ahead state prediction and the input of the observer at time t, respectively; and matrix $L \in R^{n\times l}$ which can be obtained using observer design approaches. The estimator of the state

$$\hat{x}_{t|t} = \hat{x}_{t|t-1} + M(y_t - C\hat{x}_{t|t-1}) \tag{14.3}$$

where $\hat{x}_{t|t-1} \in R^n$ is the state observer, $M \in R^{n\times l}$ is the matrix to be determined. Assume that the controller is of the following form:

$$u_{t|t} = K\hat{x}_{t|t} \tag{14.4}$$

where $K \in R^{m\times n}$ is the state feedback control matrix to be determined using modern control theory. In particular, the augmented system without time-delay, i.e., $u_t = u_{t|t}$, can be described as follows:

$$\begin{aligned} \hat{x}_{t+1|t} &= (A + BK - LC - BKMC)\hat{x}_{t|t-1} \\ &\quad +(BKMC + LC)x_t \\ x_{t+1} &= (A + BKMC)x_t + (BK - BKMC)\hat{x}_{t|t-1} \\ u_t &= K\hat{x}_{t|t} = K(I - MC)\hat{x}_{t|t-1} + KMCx_t. \end{aligned} \tag{14.5}$$

In order to compensate for the network transmission delay, a network delay compensator in the channel from the controller to the actuator is proposed. A very important characteristic of the network is that it can transmit a set of data at the same time. Thus, it is assumed that control predictions at time t are packed and sent to the plant side through a network. The network delay compensator chooses the latest control value from the control prediction sequences available on the plant side. For example, when there is no time delay in the channel from sensor to controller and the time-delay from controller to actuator is i_t, and the following predictive control sequences are received on the plant side:

$$[u^T_{t-i_1|t-i_1}, u^T_{t-i_1+1|t-i_1}, \cdots, u^T_{t|t-i_1}, \cdots, u^T_{t+N_{ca}-i_1|t-i_1}]^T \tag{14.6}$$

where the control values $u_{t|t-i_j}$ for $j = 1, 2, \cdots, t$, are available to be chosen as the control input of the plant at time t, the output of the network delay compensator will be

$$u_t = u_{t|t-min\{i_1,i_2,\cdots,i_t\}}. \tag{14.7}$$

When random network delays exist in both sensor-to-controller and controller-to-actuator channels, the scheme proposed in this chapter compensates for the network delays in the controller-to-actuator and sensor-to-controller channels and could achieve the desired control performance. Thus, on the sensor side, the output sequence is packed and sent to the controller side through the feedback channel. On the control side, the control prediction sequence at time t, is packed and sent to the plant side through the forward channel. The network delay compensator chooses the latest control value from the control prediction sequences available on the plant side.

In this chapter, the time-delays in both the controller-to-actuator channel and sensor-to-controller channel are modelled as an associated Markov chain process (random jump process). The usefulness of such a model representation is that it permits the decision-maker to properly treat the discrete-events, which significantly change the normal operation by exploiting the knowledge of their occurrence and the statistical patterns of the arrival information. Then, the transition probability from one time-delay to another will be described in the following special structure of the transition matrix.

In the following result, the assumption is made that the network delay in the controller to actuator channel is bounded by N_{ca}. The network delay compensator is in the form of

$$u_t = u_{t|t-i_t}, \text{ subject to } \ i_t \leq i_{t-1} + 1 \tag{14.8}$$

where the network delay i_t is modelled as a discrete state Markov chain taking values in a finite set $\mathcal{S} \overset{\Delta}{=} \{0, 1, 2, ..., N_{ca}\}$, with transition probability (α_{ij}) from mode i to mode j

$$\alpha_{ij} \ = \ Pr_{ca}(i_{t+1} = j \mid i_t = i) \tag{14.9}$$

where $0 \leq i, j \leq N_{ca}$, $\alpha_{ij} \geq 0$ and for any $i \in \mathcal{S}$, $\sum_{j=0}^{N_{ca}} \alpha_{ij} = 1$. From (14.6), a package containing $N_{ca} + 1$ control predictions will be sent through the network. If one package, such as

$$[u^T_{t-i_1|t-i_1}, u^T_{t-i_1+1|t-i_1}, \cdots, u^T_{t|t-i_1}, \cdots, u^T_{t+N_{ca}-i_1|t-i_1}]^T$$

is sent through the network, in which, $u^T_{t|t-i_1}$ will be used as the control input due to the network delay i_1. $u^T_{t+1|t-i_1}$ will be used as the next control input if there is no new coming package of future prediction sequences at time $t+1$ since the data may be lost or there is a longer delay. Then, from the constraint (14.8), it follows that

$$\alpha_{ij} = \ Pr_{ca}(i_{t+1} = j| \ i_t = i < j - 1) = 0 \tag{14.10}$$

i.e, $\alpha_{ij} = 0, \text{ for } i < j - 1$. On the other hand, the delay i_t can be reduced by up to the maximum number of steps. When the new package of the future prediction sequence arrives, the old package will be discarded. Then, a structured transition probability matrix is obtained as follows

$$Pr_{ca} = \begin{bmatrix} \alpha_{00} & \alpha_{01} & 0 & 0 & \cdots & 0 \\ \alpha_{10} & \alpha_{11} & 0 & \alpha_{12} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & 0 \\ \alpha_{(N_{ca}-1)0} & \alpha_{(N_{ca}-1)1} & \alpha_{(N_{ca}-1)2} & \alpha_{(N_{ca}-1)3} & \cdots & \alpha_{(N_{ca}-1)N_{ca}} \\ \alpha_{N_{ca}0} & \alpha_{N_{ca}1} & \alpha_{N_{ca}2} & \alpha_{N_{ca}3} & \cdots & \alpha_{N_{ca}N_{ca}} \end{bmatrix}. \tag{14.11}$$

In (14.11), each row represents the transition probabilities from one element in a fixed package to the element in all the package coming in sequence. The diagonal elements are the probabilities of package coming in sequence with equal delays. The elements above the diagonal are the probabilities of encountering longer delays, and the elements below the diagonal indicate package loss or discarded old data.

As the random time-delay also occurs in the sensor-to-controller channel, the same Markov chain model for this random time-delay k_t is adopted. It is assumed that the network delay k_t in the feedback channel is random but bounded by N_{sc}, that is, $k_t \in \{0, 1, 2, \cdots, N_{sc}\}$. The networked predictive controller is in the form of

$$u_t = K\hat{x}_{t|t-k_t}, \text{ subject to } \; k_t \leq k_{t-1} + 1 \tag{14.12}$$

where the network delay k_t is modelled as a discrete state Markov chain taking values in a finite set $\mathcal{S}_{sc} \overset{\Delta}{=} \{0, 1, 2, ..., N_{sc}\}$, and (β_{ij}) has a transition probability from mode i to mode j of

$$\beta_{ij} \;=\; Pr_{sc}(k_{t+1} = j \mid k_t = i) \tag{14.13}$$

where $0 \leq i, j \leq N_{sc}$, $\beta_{ij} \geq 0$ and for any $i \in \mathcal{S}_{sc}$, $\sum_{j=0}^{N_{sc}} \beta_{ij} = 1$. Then, from the constraint (14.12), it follows that

$$Pr_{sc}(k_{t+1} = j | k_t = i < j - 1) = 0. \tag{14.14}$$

Then, a structured transition probability matrix is obtained as follows

$$Pr_{sc} = \begin{bmatrix} \beta_{00} & \beta_{01} & 0 & 0 & \cdots & 0 \\ \beta_{10} & \beta_{11} & 0 & \beta_{12} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & 0 \\ \beta_{(N_{sc}-1)0} & \beta_{(N_{sc}-1)1} & \beta_{(N_{sc}-1)2} & \beta_{(N_{sc}-1)3} & \cdots & \beta_{(N_{sc}-1)N_{sc}} \\ \beta_{N_{sc}0} & \beta_{N_{sc}1} & \beta_{N_{sc}2} & \beta_{N_{sc}3} & \cdots & \beta_{N_{sc}N_{sc}} \end{bmatrix}. \tag{14.15}$$

In this case, the delay in the sensor-to-controller channel and the delay in the controller-to-actuator channel are modelled as Markov processes described in (14.9) and (14.13), respectively. Then, the Markovian jump parameter is

$$r_{sca} = (k_t, i_t) \tag{14.16}$$

$$Pr_{sca} = Pr_{sc} \otimes Pr_{ca}. \tag{14.17}$$

14.3 Stochastic Stability of Closed-Loop Networked Predictive Control Systems

Following the state observer described by (14.2), when $k_t = 0$, u_{t-k_t} will be $u_{t|t}$ which is defined as in (14.3) and (14.4). Based on the pair (y_{t-k_t}, u_{t-k_t})received on the side of the observer, and the input data applied by observer up to $t-1$, the state predictions from time $t - k_t + 1$ to t are constructed as

$$
\begin{aligned}
\hat{x}_{t-k_t+1|t-k_t} &= A\hat{x}_{t-k_t|t-k_t-1} + Bu_{t-k_t} + L(y_{t-k_t} - C\hat{x}_{t-k_t|t-k_t-1}) \\
\hat{x}_{t-k_t+2|t-k_t} &= A\hat{x}_{t-k_t+1|t-k_t} + Bu_{t-k_t+1} \\
&\vdots \\
\hat{x}_{t|t-k_t} &= A\hat{x}_{t-1|t-k_t} + Bu_{t-1}
\end{aligned}
\tag{14.18}
$$

which results in

$$
\begin{aligned}
\hat{x}_{t|t-k_t} = {} & A^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} + \textstyle\sum_{j=1}^{k_t} A^{k_t-j}Bu_{t-k_t+j-1} \\
& + A^{k_t-1}Ly_{t-k_t}
\end{aligned}
\tag{14.19}
$$

where $k_t = 1, 2, 3, \cdots, N_{sc}$. As a result, the output of the networked predictive control at time t is determined by

$$
\begin{aligned}
u_{t|t-k_t} = {} & KA^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} + \textstyle\sum_{j=1}^{k_t} KA^{k_t-j}Bu_{t-k_t+j-1} \\
& + KA^{k_t-1}Ly_{t-k_t}.
\end{aligned}
\tag{14.20}
$$

From equation (14.20), it is clear that the future control predictions depend on the state estimation $\hat{x}_{t-k_t|t-k_t-1}$ and the past control input up to u_{t-1} and the past output up to y_{t-k_t} of the system. When there is no delay in the feedback channel, i.e., $k_t = 0$, and $i_t \neq 0$, then,

$$
\begin{aligned}
\hat{x}_{t+1|t} &= A\hat{x}_{t|t-1} + Bu_t + L(y_t - C\hat{x}_{t|t-1}) \\
\hat{x}_{t+2|t} &= A\hat{x}_{t+1|t} + Bu_{t+1|t} \\
&\vdots \\
\hat{x}_{t+i_t|t} &= A\hat{x}_{t+i_t-1|t} + Bu_{t+i_t-1|t}.
\end{aligned}
\tag{14.21}
$$

Let $u_{t+i|t} = K\hat{x}_{t+i|t}, i = 1, 2, \cdots, i_t - 1$, and $u_t = u_{t|t-i_t} = K\hat{x}_{t|t-i_t}$, then,

$$
\hat{x}_{t|t-i_t} = (A+BK)^{i_t-1}[(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} + LCx_{t-i_t}] \tag{14.22}
$$

$$
u_t = K(A+BK)^{i_t-1}[(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} + LCx_{t-i_t}]. \tag{14.23}
$$

The closed-loop system is

$$
\begin{aligned}
x_{t+1} &= Ax_t + BK(A+BK)^{i_t-1}LCx_{t-i_t} + BK(A+BK)^{i_t-1} \\
&\quad \times(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} \\
\hat{x}_{t+1|t} &= (A+BK-LC-BKMC)\hat{x}_{t|t-1} + (BKMC+LC)x_t \\
u_t &= K(A+BK)^{i_t-1}[(A+BK-LC)\hat{x}_{t-i_t|t-i_t-1} + LCx_{t-i_t}].
\end{aligned}
\tag{14.24}
$$

For the case where there are both the network delay i_t in the controller-to-actuator channel and the network delay k_t in the sensor-to-controller channel, the state prediction from time $t+1$ to $t+i_t$ are constructed by

$$\begin{aligned}
\hat{x}_{t+1|t-k_t} &= A\hat{x}_{t|t-k_t} + Bu_{t|t-k_t} \\
\hat{x}_{t+2|t-k_t} &= A\hat{x}_{t+1|t-k_t} + Bu_{t+1|t-k_t} \\
&\vdots \\
\hat{x}_{t+i_t|t-k_t} &= A\hat{x}_{t+i_t-1|t-k_t} + Bu_{t+i_t-1|t-k_t}
\end{aligned} \tag{14.25}$$

the control predictions are calculated by

$$u_{t+i|t-k_t} = K\hat{x}_{t+i|t-k_t}, i = 0, 1, \cdots, i_t \tag{14.26}$$

where the state feedback matrix $K \in R^{m\times n}$. Thus,

$$\begin{aligned}
\hat{x}_{t+i_t|t-k_t} &= (A+BK)^{i_t}\hat{x}_{t|t-k_t} \\
&= (A+BK)^{i_t}(A^{k_t-1}(A-LC)\hat{x}_{t-k_t|t-k_t-1} \\
&\quad + \textstyle\sum_{j=1}^{k_t} A^{k_t-j}Bu_{t-k_t+j-1} + A^{k_t-1}Ly_{t-k_t}).
\end{aligned} \tag{14.27}$$

Since there exist the forward delay i_t and feedback delay k_t, the control input of the plant is given by

$$u_t = u_{t|t-i_t-k_t}. \tag{14.28}$$

Combining this predictive controller with the networked delay compensator, both the forward and feedback network delays will be compensated for a certain amount of time-delay. In this section, stability analysis of the closed-loop system will be presented using this control scheme.

When the random network delay in the controller-actuator channel is k_t ($k_t > 0$) the control input, state prediction, and plant state vectors are expressed by

$$\begin{aligned}
u_t &= K(A+BK)^{i_t}[A^{k_t-1}(A-LC)\hat{x}_{t-i_t-k_t|t-i_t-k_t-1} \\
&\quad + \textstyle\sum_{j=1}^{k_t} A^{k_t-j}Bu_{t-i_t-k_t+j-1} + A^{k_t-1}LCx_{t-i_t-k_t}]
\end{aligned} \tag{14.29}$$

$$\begin{aligned}
x_{t+1} &= Ax_t + Bu_t \\
&= Ax_t + Bu_{t|t-i_t-k_t} \\
&= Ax_t + BK(A+BK)^{i_t}[A^{k_t-1}(A-LC)\times \\
&\quad \hat{x}_{t-i_t-k_t|t-i_t-k_t-1} + \textstyle\sum_{j=1}^{k_t} A^{k_t-j}Bu_{t-i_t-k_t+j-1} + A^{k_t-1}LCx_{t-i_t-k_t}]
\end{aligned} \tag{14.30}$$

$$\hat{x}_{t-k_t+1|t-k_t} = A\hat{x}_{t-k_t|t-k_t-1} + Bu_{t-k_t} + L(y_{t-k_t} - C\hat{x}_{t-k_t|t-k_t-1}). \tag{14.31}$$

Note that $\Lambda(0,0)$ for $i_t = 0, k_t = 0$ is described in (14.34), and $\Lambda(i_t, 0)$ for $i_t = 1, 2, \cdots, N_{ca}$ and $k_t = 0$ are defined in (14.35), then, combination of equations (14.29), (14.30) and (14.31) gives the following augmented system:

$$X_{t+1} = \Lambda(i_t, k_t)X_t \tag{14.32}$$

where

$$X_t = \big[\, x_t^T \; x_{t-1}^T \cdots x_{t-i_t-k_t+1}^T \; x_{t-i_t-k_t}^T \; x_{t-i_t-k_t-1}^T \cdots x_{t-N_{sc}-N_{ca}+1}^T \; x_{t-N_{sc}-N_{ca}}^T \; u_{t-1}^T \; u_{t-2}^T \cdots u_{t-i_t-k_t+1}^T \; u_{t-i_t-k_t}^T \; u_{t-i_t-k_t-1}^T \cdots u_{t-N_{sc}-N_{ca}+1}^T \; u_{t-N_{sc}-N_{ca}}^T \; \hat{x}_{t|t-1}^T \; x_{t-1|t-2}^T \cdots \hat{x}_{t-i_t-k_t+1|t-i_t-k_t}^T \; \hat{x}_{t-i_t-k_t|t-i_t-k_t-1}^T \; x_{t-i_t-k_t-1|t-i_t-k_t-2}^T \cdots \hat{x}_{t-N_{sc}-N_{ca}+1|t-N_{sc}-N_{ca}}^T \; \hat{x}_{t-N_{sc}-N_{ca}|t-N_{sc}-N_{ca}-1}^T \,\big]^T \tag{14.33}$$

and $\Lambda(i_t,k_t)$, $i_t = 0,1,\cdots,N_{ca}$, $k_t = 0,1,2,\cdots,N_{sc}$ are expressed as follows:

$$\Lambda(0,0) = \begin{bmatrix} \Lambda_1(0,0) \\ \Lambda_{21}(0,0)\;\Lambda_{22}(0,0) \\ \Lambda_3(0,0) \\ \Lambda_{41}(0,0)\;\Lambda_{42}(0,0)\;\Lambda_{43}(0,0) \\ \Lambda_5(0,0) \\ \Lambda_{61}(0,0)\;\Lambda_{62}(0,0)\;\Lambda_{63}(0,0) \end{bmatrix} \tag{14.34}$$

where

$\Lambda(0,0) \in R^{[2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n][2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n]}$

$\Lambda_1(0,0) = \Big[\, M_1(0) \; \overbrace{0 \cdots 0\;0}^{(N_{ca}+N_{sc})n} \; \overbrace{0 \cdots 0\;0}^{(N_{ca}+N_{sc})m} \; M_2(0)\; 0 \cdots 0\; 0 \,\Big]$

$\Lambda_{21}(0,0) = I_{(N_{ca}+N_{sc})n\times(N_{ca}+N_{sc})n}$

$\Lambda_{22} = 0_{(N_{ca}+N_{sc})n\times[n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n]}$

$\Lambda_3(0,0) = \big[\, K(I-MC)\; 0\cdots 0\;0\;\; 0\cdots\; 0\;0\; KMC\; 0\cdots 0\;0 \,\big]$

$\Lambda_{41}(0,0) = 0_{(N_{ca}+N_{sc}-1)m\times(N_{ca}+N_{sc}+1)n}$

$\Lambda_{42}(0,0) = I_{(N_{ca}+N_{sc}-1)m\times(N_{ca}+N_{sc}-1)m}$

$\Lambda_{43}(0,0) = 0_{[(N_{ca}+N_{sc}-1)m\times(N_{ca}+N_{sc})n+2n]}$

$\Lambda_5(0,0) = \big[\, M_3(0)\; 0\cdots 0\;0\;\; 0\cdots 0\;0\; M_4(0)\; 0\cdots 0\;0 \,\big]$

$\Lambda_{61}(0,0) = 0_{[(N_{ca}+N_{sc})n+(N_{ca}+N_{sc})m+n]\times[(N_{ca}+N_{sc}+1]}$

$\Lambda_{62}(0,0) = 0_{(N_{ca}+N_{sc})n\times(N_{ca}+N_{sc})n}$

$\Lambda_{63}(0,0) = 0_{(N_{ca}+N_{sc})n\times n}$

$$\Lambda(i_t,0) = \begin{bmatrix} \Lambda_1(i_t,0) \\ \Lambda_{21}(i_t,0)\;\Lambda_{22}(i_t,0) \\ \Lambda_3(i_t,0) \\ \Lambda_{41}(i_t,0)\;\Lambda_{42}(i_t,0)\;\Lambda_{43}(i_t,0) \\ \Lambda_5(i_t,0) \\ \Lambda_{61}(i_t,0)\;\Lambda_{62}(i_t,0)\;\Lambda_{63}(i_t,0) \end{bmatrix} \tag{14.35}$$

where $\Lambda(i_t,0) \in R^{[2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n][2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n]}$

$\Lambda_1(i_t,0) = \Big[\, A\; \overbrace{0\cdots}^{i_t n}\; M_1(i_t)\cdots 0\;\; \overbrace{0\cdots 0\;0}^{(N_{ca}+N_{sc})m}\;\; 0\cdots M_2(i_t)\cdots 0\;0 \,\Big]$

$\Lambda_{21}(i_t,0) = I_{(i_t+2)n\times(i_t+2)n}$

$\Lambda_{22}(i_t,0) = 0_{(i_t+2)n\times[2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})-i_t n]}$

$\Lambda_3(i_t,0) = \big[\, 0\;0\cdots\; M_3(i_t)\cdots 0\;0\cdots 0\;0\;0\cdots M_4(i_t)\cdots 0\;0 \,\big]$

$\Lambda_{41}(i_t,0) = 0_{(N_{ca}+N_{sc}-1)m\times(N_{ca}+N_{sc}+1)n}$

$$\Lambda_{42}(i_t,0) = I_{(N_{ca}+N_{sc}-1)m\times(N_{ca}+N_{sc}-1)m}$$
$$\Lambda_{43}(i_t,0) = 0_{[(N_{ca}+N_{sc}-1)m\times(N_{ca}+N_{sc})n+2n]}$$
$$\Lambda_5(i_t,0) = \begin{bmatrix} M_3(0) & 0\cdots 0\ 0 & 0\cdots 0\ 0 & M_4(0)\ 0\cdots 0\ 0\end{bmatrix}$$
$$\Lambda_{61}(i_t,0) = 0_{[(N_{ca}+N_{sc})n+(N_{ca}+N_{sc})m+n]\times[(N_{ca}+N_{sc}+1]}$$
$$\Lambda_{62}(i_t,0) = 0_{(N_{ca}+N_{sc})n\times(N_{ca}+N_{sc})n}$$
$$\Lambda_{63}(i_t,0) = 0_{(N_{ca}+N_{sc})n\times n}$$

$$\Lambda(i_t,k_t) = \begin{bmatrix} \Lambda_{11}(i_t,k_t) & \Lambda_{12}(i_t,k_t) & \Lambda_{13}(i_t,k_t) \\ \Lambda_{21}(i_t,k_t) & \Lambda_{22}(i_t,k_t) & \Lambda_{23}(i_t,k_t) \\ \Lambda_{31}(i_t,k_t) & \Lambda_{32}(i_t,k_t) & \Lambda_{33}(i_t,k_t) \end{bmatrix}$$
$$\in R^{[2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n][2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n]}$$

for $i_t = 1,2,\cdots,N_{ca}$ and $k_t = 1,2,\cdots,N_{sc}$, where $\Phi_{i_t} = K(A+BK)^{i_t}$

$$\Lambda_{11} = \begin{bmatrix} \Lambda_{111} & \\ I_{n(M+N)\times(n(M+N)} & 0 \end{bmatrix}$$
$$\in R^{[n(M+N)+n]\times[n(M+N)+n]}$$
$$\Lambda_{111} = \begin{bmatrix} LC & \overbrace{0\cdots 0}^{(i_t+k_t-1)n} & BM_{i_t}A^{k_t-1}LC & 0\cdots 0 \end{bmatrix} \in R^{[n\times[n(M+N)+n]}$$
$$\Lambda_{12}(k_t,i_t) = \begin{bmatrix} \hat{\Lambda}_{121} & \\ \hat{\Lambda}_{122} & 0_{[(M+N+1)\times(M+N)m-(i_t+k_t)m]} \end{bmatrix}$$
$$\hat{\Lambda}_{121}(k_t,i_t) = \begin{bmatrix} \overbrace{0\cdots 0}^{i_t n} & \Pi(1) & \cdots & \Pi(2) & \Pi(3) \end{bmatrix}$$
$$\hat{\Lambda}_{122} = 0_{[n(M+N)+n]\times[m(M+N)]}$$
$$\Lambda_{13}(k_t,i_t) = \begin{bmatrix} \Lambda_{131} \\ 0_{(n(M+N)\times[n(M+N)+n]} \end{bmatrix}$$
$$\Lambda_{131} = \begin{bmatrix} \overbrace{0\ 0\cdots 0}^{k_t n} & \Pi(4) & 0\cdots 0\ 0 \end{bmatrix}$$
$$\Lambda_{21}(k_t,i_t) = \begin{bmatrix} \Lambda_{211} \\ 0_{[(m-1)(M+N)]\times[n(M+N)+n]} \end{bmatrix}$$
$$\Lambda_{211} = \begin{bmatrix} \overbrace{0\ 0\cdots 0}^{k_t n} & M_{i_t}A^{k_t-1}LC & 0\cdots 0\ 0 \end{bmatrix}$$
$$\Lambda_{22} = \begin{bmatrix} \Lambda_{221}(k_t,i_t) & \\ I_{(M+N-)m\times(M+N-1)m} & 0_{(M+N)m\times m} \end{bmatrix} \in R^{[m(M+N)]\times[m(M+N)]}$$
$$\Lambda_{221}(k_t,i_t) = \begin{bmatrix} \overbrace{0\cdots 0}^{i_t n} & \Pi(5) & \Pi(6) & \cdots & \Pi(7) & \Pi(8) & 0\cdots 0 \end{bmatrix} \in R^{[m]\times[m(M+N-1)]}$$
$$\Lambda_{23}(k_t,i_t) = \begin{bmatrix} \Lambda_{231} \\ \Lambda_{232} \end{bmatrix}$$
$$\Lambda_{231} = \begin{bmatrix} \overbrace{0\ 0\cdots 0}^{k_t n} & \Pi(9) & 0\cdots 0\ 0 \end{bmatrix}$$
$$\Lambda_{232} = 0_{[m(M+N-1)]\times[n(M+N)+n]}$$

$$\Lambda_{31} = \begin{bmatrix} \Lambda_{311} & \\ 0_{(M+N)n\times(i_t+k_t+1)n} & 0 \end{bmatrix} \in R^{[n(M+N)+n]\times[n(M+N)+n]}$$
$$\Lambda_{311} = \begin{bmatrix} LC & \overbrace{0 \cdots 0}^{(i_t+k_t-1)n} & 0 \end{bmatrix} \in R^{[n\times(i_t+k_t+1)n}$$
$$\Lambda_{32} = 0_{(n(M+N)+n)\times(mN)}$$
$$\Lambda_{33}(k_t, i_t) = \begin{bmatrix} \Lambda_{331} \\ \Lambda_{3311} \ \Lambda_{3312} \end{bmatrix}$$
$$\Lambda_{331} = \begin{bmatrix} \overbrace{\overline{A}\ 0 \cdots 0}^{k_t n}\ 0\ 0 \cdots 0\ 0 \end{bmatrix}$$
$$\Lambda_{3311} = I_{n(M+N)\times n(M+N)nm}$$
$$\Lambda_{3312} = 0_{n(M+N)\times n}$$

where $\overline{A} = A + BK - LC$, $M_3(0) = BKMC + LC$, $M_4(0) = A + BK - LC + BKMC$, $M_{i_t} = K(A + BK)^{i_t}$, $M_1(i_t) = BK(A + BK)^{i_t-1}LC$, $M_2(i_t) = BK(A + BK)^{i_t-1}(A + BK - LC)$, $M_3(i_t) = K(A + BK)^{i_t-1}LC$, $M_4(i_t) = K(A + BK)^{i_t-1}(A + BK - LC)$, $M_5(i_t) = BK(A + BK)^{i_t-1}LC$, $M_6(i_t) = BK(A+BK)^{i_t-1}(A+BK-LC)$, $M_7(i_t) = D_1K(A+BK)^{i_t-1}LC$, $M_8(i_t) = D_1K(A + BK)^{i_t-1}(A + BK - LC)$, $\Pi(1) = M_{i_t}B$, $\Pi(2) = BM_{i_t}A^{k_t-2}B$, $\Pi(3) = BM_{i_t}A^{k_t-1}B$,$\Pi(4) = BM_{i_t}A^{k_t-1}(A - LC)$, $\Pi(5) = M_{i_t}A^{k_t-1}LC$, $\Pi(6) = M_{i_t}AB$, $\Pi(7) = M_{i_t}A^{k_t-2}B$, $\Pi(8) = M_{i_t}A^{k_t-1}B$, $\Pi(9) = M_{i_t}A^{k_t-1}(A - LC)$.

Definition 14.2. [213] The jump system (14.32) with random time-delays is said to be stochastically stable if for the initial mode r_0, there exists a finite number $M(r_0) > 0$ such that

$$\mathscr{E}\left\{\sum_{t=0}^{\infty} X_{r_0}^T(t)X_{r_0}(t)|r_0\right\} < M(r_0). \tag{14.36}$$

It is noted that the closed-loop system in (14.32) is a delay-free discrete-time jump linear system with two modes. Then, we can utilize the existed results on the stochastic stability and stabilization to perform the stability analysis of system (14.32). Our main results in this section can be stated as follows:

Theorem 14.3. *System (14.32) with random time-delay k_t in the channel from the sensor-to-controller and random time-delay i_t in the channel from the controller-to-actuator, where k_t and i_t are Markov processes described by (14.10) and (14.13), respectively, is stochastically stable if and only if there exist symmetric positive definite matrices $S(i, j), i = 0, 1, \cdots, N_{ca}; j = 0, 1, \cdots, N_{sc}$ satisfying any one of the following four inequalities:*

$$\Lambda(i, j)(\sum_{k=0}^{N_{ca}}\sum_{h=0}^{N_{sc}} \alpha_{ki}\beta_{hj}S(k, h))\Lambda^T(i, j) < S(i, j) \tag{14.37}$$

$$\sum_{k=0}^{N_{ca}}\sum_{h=0}^{N_{sc}}\alpha_{ki}\beta_{hj}\Lambda(k,h)S(k,h)\Lambda^T(k,h) < S(i,j) \tag{14.38}$$

$$\Lambda^T(i,j)(\sum_{k=0}^{N_{ca}}\sum_{h=0}^{N_{sc}}\alpha_{ik}\beta_{jh}S(k,h))\Lambda(i,j) < S(i,j) \tag{14.39}$$

$$\sum_{k=0}^{N_{ca}}\sum_{h=0}^{N_{sc}}\alpha_{ik}\beta_{jh}\Lambda^T(k,h)S(k,h)\Lambda(k,h) < S(i,j) \tag{14.40}$$

where

$$\begin{aligned} S(i,j) \ &\in R^{[2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n]\times} \\ &[2n(N_{ca}+N_{sc})+m(N_{ca}+N_{sc})+2n] \\ &i=0,1,\cdots,N_{ca}; j=0,1,\cdots,N_{sc}. \end{aligned}$$

Proof. Sufficiency: Without loss of generality, it is assumed that (14.39) holds, then, based on the algorithm proposed in this chapter, the stochastic stability of the closed-loop systems is related to the time-delay in the feedback k_t channel and forward channel i_t. Then, a quadratic function can be taken as follows:

$$V(X_t,t) = X_t^T S(i_t,k_t)X_t. \tag{14.41}$$

It follows from (14.32) that

$$\begin{aligned} &\mathscr{E}\{\Delta V(X_t,t)\} \\ &= \mathscr{E}\{X_{t+1}^T S(i_{t+1},k_{t+1})X_{t+1}|X_t\} - X_t^T S(i_t,k_t)X_t \\ &= X_t^T \Lambda^T(i_t,k_t)(\sum_{k=0}^{N_{ca}}\sum_{h=0}^{N_{sc}}\alpha_{i_t k}\beta_{k_t h}S(k,h))\Lambda(i_t,k_t)X_t - X_t^T S(i_t,k_t)X_t. \end{aligned}$$

If (14.39) is satisfied for $i=0,1,\cdots,N_{ca}; j=0,1,\cdots,N_{sc}$, then, there exists a positive scalar σ such that the following inequality holds:

$$\mathscr{E}\{\Delta V(X_t,t)\} \le -\sigma||X_t||^2. \tag{14.42}$$

Add the above inequalities from $t=0$ to $t=T$, it results in

$$\begin{aligned} \mathscr{E}\{\sum_{i=0}^{T}||X_t||^2\} \ &\le \frac{1}{\sigma}(\mathscr{E}\{V(X_0,0)\} - \mathscr{E}\{V(X_{T+1},T+1)\} \\ &\le \frac{1}{\sigma}\mathscr{E}\{V(X_0,0)\}. \end{aligned} \tag{14.43}$$

From the above inequality, it can been easily seen that (14.36) is satisfied, then, the jump system (14.32) with random time-delays is stochastically stable. The other three conditions in Theorem 14.3 can be obtained by the same techniques used in the proof.

The necessity of the proof is similar to the proof of Theorem 1 in [213], it is omitted here.

Remark 14.4. At each sampling time, the current controller-to-actuator delay i_t can be known by the proposed network delay compensator on the of side of actuator. However, it is not known for the observer. The result obtained in the above theorem is mainly on the stability analysis of this kind of systems. The performance can also be improved with the method proposed in this chapter, which can also be refereed to our recent results, which appeared in [120].

Remark 14.5. Since the matrix inequalities in (14.37)-(14.40) are related to gain matrices K and L, in general, these gain matrices should be designed to guarantee that the closed-loop system with constant network delay is stable. If gain matrices K and L are designed previously, then (14.37)-(14.40) are a set of linear matrix inequalities (LMIs). Therefore, one can use LMI tool box to find the feasible solutions.

Remark 14.6. Another important issue in a communication channel is the problem of bandwidth. To solve such a problem, the quantization, as a popular way, has been considered, see [49, 47, 40, 114]. In a network environment, signals are usually quantized before being communicated, and the number of quantization levels is closely related to the information flow between the components of the system and thus to the capacity required to transmit the information. In the networked predictive control systems, as for how to solve bandwidth problem using quantization, it remains our future study.

14.4 Examples

14.4.1 Numerical Example

Example 14.7. In this section, an illustrative example is constructed to verify the design method developed in this chapter. Simulation studies are presented for an open-loop unstable discrete system which is difficult to control and is in the form of (14.1) with the following system matrices:

$$A = \begin{bmatrix} -1.100 & 0.271 \\ 0.188 & 0.482 \end{bmatrix}, B = \begin{bmatrix} 0.5 \\ 0.3 \end{bmatrix}, C = \begin{bmatrix} 0.1 & 0.2 \end{bmatrix}.$$

It is assumed that the upper bound of the time-delay in the sensor-to-controller channel and controller-to-actuator channel is 2, that is, $N_{sc} = 2$,

$N_{ca} = 2$, and the structured transition probability matrices for the random time-delays are

$$Pr_{ca} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.5 & 0.3 & 0.2 \\ 0.3 & 0.4 & 0.3 \end{bmatrix}, Pr_{sc} = \begin{bmatrix} 0.6 & 0.4 & 0 \\ 0.3 & 0.4 & 0.3 \\ 0.3 & 0.5 & 0.2 \end{bmatrix}$$

then,

$$Pr_{sca} = Pr_{ca} \otimes Pr_{sc} = \begin{bmatrix} 0.3 & 0.2 & 0 & 0.3 & 0.2 & 0 & 0 & 0 & 0 \\ 0.15 & 0.2 & 0.15 & 0.15 & 0.2 & 0.15 & 0 & 0 & 0 \\ 0.15 & 0.25 & 0.1 & 0.15 & 0.25 & 0.1 & 0 & 0 & 0 \\ 0.3 & 0.2 & 0 & 0.18 & 0.12 & 0 & 0.12 & 0.08 & 0 \\ 0.15 & 0.2 & 0.15 & 0.09 & 0.12 & 0.09 & 0.06 & 0.08 & 0.06 \\ 0.15 & 0.25 & 0.1 & 0.09 & 0.15 & 0.06 & 0.06 & 0.1 & 0.04 \\ 0.18 & 0.12 & 0 & 0.24 & 0.16 & 0 & 0.18 & 0.12 & 0 \\ 0.09 & 0.12 & 0.09 & 0.12 & 0.16 & 0.12 & 0.09 & 0.12 & 0.09 \\ 0.09 & 0.15 & 0.06 & 0.12 & 0.20 & 0.08 & 0.09 & 0.15 & 0.06 \end{bmatrix}.$$

The matrices $K = \begin{bmatrix} 0.1634 & -1.7594 \end{bmatrix}$, $L = \begin{bmatrix} -3.3422 \\ 3.4661 \end{bmatrix}$ and $M = \begin{bmatrix} -1.9745 & -0.3282 \end{bmatrix}$ are designed such that the nominal system (14.1) is stable. Then, using LMI Toolbox, positive matrices $S(i, j), i = 0, 1, 2; j = 0, 1, 2$ satisfying inequalities (14.37) have been found, which is not presented here because of its size. From Theorem 14.3, system (14.1) with random time-delay k_t in the channel from the sensor-to-controller and i_t in the channel from the controller-to-actuator , where k_t and i_t are Markov processes described by (14.9) and (14.13) respectively, is stochastically stable.

14.4.2 Practical Experiments

Example 14.8. To implement networked control systems, a test rig is built, based on an ARM 9 embedded boards. The figure of the board is shown in Fig. 14.1. The kernel chip of the embedded board is ATMEL's AT91RM9200, which is a cost-effective, high-performance 32-bit RISC microcontroller for Ethernet- based embedded systems. A 10M/100M self-adaptive network controller is integrated in the chip and the chip also has a high computing performance and can work at speeds up to 180 MHz. 2-channel 16-bit high speed analog-digital (D/A) converters and 8-channel 16-bit high speed analog-digital (A/D) converters in the controller board provide I/O interfaces for the controlled plant.

In order to validate the proposed method, a servo motor control system which consists of a DC motor, load plate, speed and angle sensors is considered. The figure of the plant is shown in Fig. 14.2.

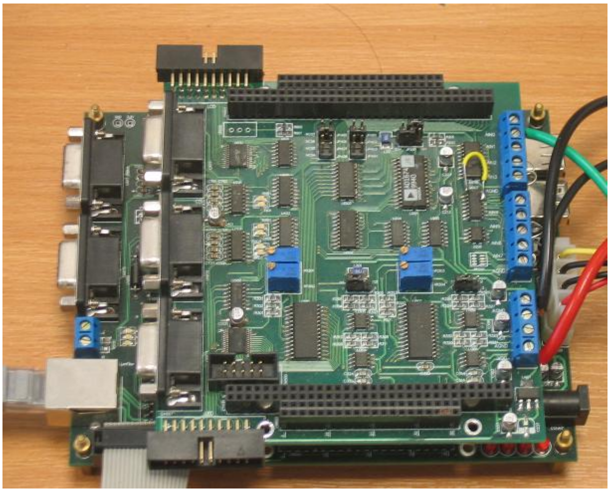

Fig. 14.1 ARM 9 Embedded Board

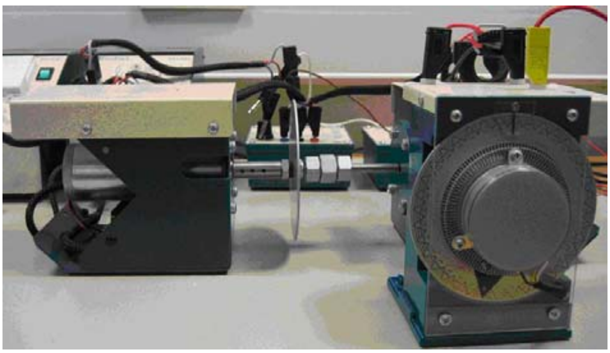

Fig. 14.2 Servo Plant

The model of the motor control plant at sampling period 0.04 second is identified to be

$$G(z^{-1}) = \frac{A(z^{-1})}{B(z^{-1})} = \frac{0.05409z^{-2} + 0.115z^{-3} + 0.0001z^{-4}}{1 - 1.12z^{-1} - 0.213z^{-2} + 0.335z^{-3}}.$$

The system can also be written as the state space form with the following system matrices

$$A = \begin{bmatrix} 1.12 & 0.213 & -0.335 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$
$$C = \begin{bmatrix} 0.0541 & 0.1150 & 0.0001 \end{bmatrix}.$$

The matrices K and L are designed to be

$$K = \begin{bmatrix} -0.0270 & -0.575 & -0.0001 \end{bmatrix}, \; L = \begin{bmatrix} 6 \\ 6 \\ 6 \end{bmatrix}$$

which ensure the close-loop system without time delay is stable.

14.4.2.1 Simulation

To verify the proposed networked predictive control scheme, three simulations are considered: local control (i.e., no network), networked control with compensation, networked control without compensation. It is assumed that the Markov matrices for the forward and feedback channel time delay could be described as

$$Pr_{ca} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.5 & 0.3 & 0.2 \\ 0.3 & 0.4 & 0.3 \end{bmatrix}, \; Pr_{sc} = \begin{bmatrix} 0.2 & 0.8 & 0 & 0 \\ 0.2 & 0.2 & 0.6 & 0 \\ 0.2 & 0.2 & 0.3 & 0.3 \\ 0.1 & 0.2 & 0.3 & 0.4 \end{bmatrix}.$$

These simulations are carried out by Matlab Simulink. The diagram of networked simulation is shown in Fig. 14.3. To simulate the network delay, block Markov delay is developed. The function of this block is to make the artificial random time delay but subject to Markov matrix. The block Netctrl is the control prediction generator and the block Comp is the network delay compensator. All these blocks are designed using Matlab S-Functions. The simulation results are shown in Fig. 14.4 and Fig. 14.5.

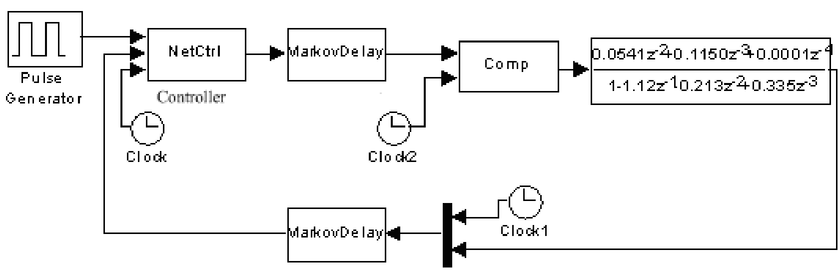

Fig. 14.3 Simulation Diagram

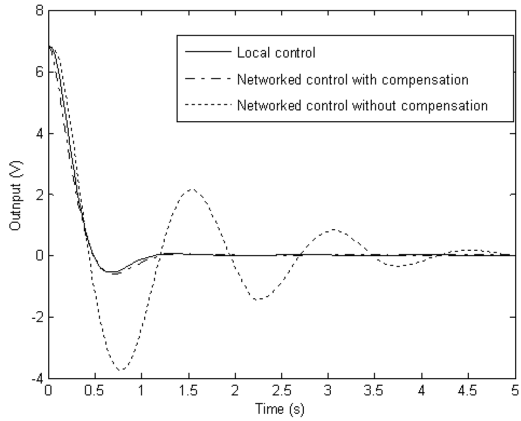

Fig. 14.4 Output of servo plant (Simulation)

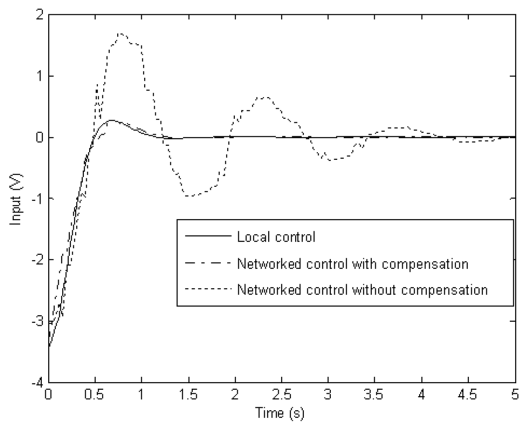

Fig. 14.5 Input of servo plant (Simulation)

14.4.2.2 Real Time Experiment

The difference between the real-time simulations and real-time experiments is that the plant model of the servo motor in the real-time simulations is replaced by D/A block Dac9 and A/D block Adc9 and the real servo motor. The diagram of the real-time experiments is shown in Fig. 14.6. The two blocks Dac9 and Adc9 were the driver of the A/D and D/A channels in the embedded system and were designed in Matlab S-Function. Similarly, three real-time experiments were made: local control (i.e., no network), networked control with compensation, networked control without compensation. The real-time experiments results are shown in Fig. 14.7 and Fig. 14.8.

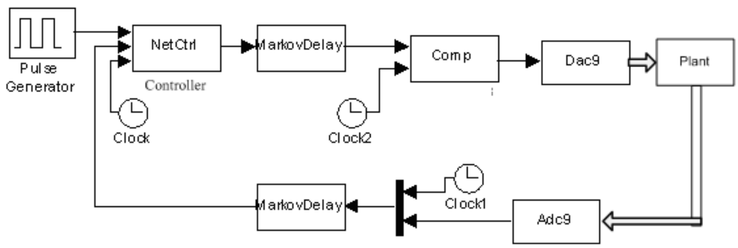

Fig. 14.6 Diagram of real time experiment

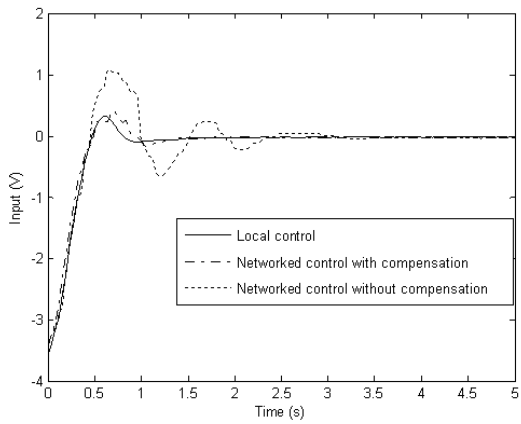

Fig. 14.7 Input of servo plant (Experiment)

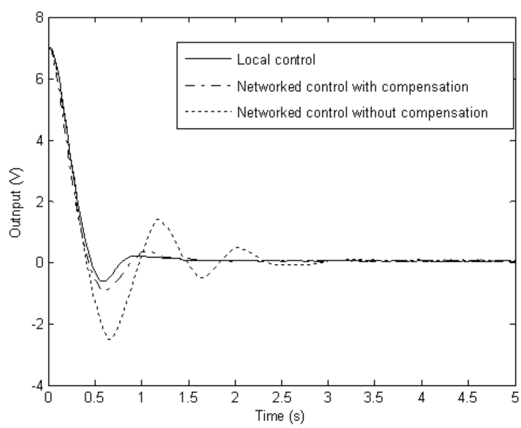

Fig. 14.8 Input of servo plant (Experiment)

From the results of real-time simulations and real-time experiments, it is clear that the network transmission delay degrades the performance of NCS. But the networked predictive control scheme proposed in this chapter can compensate for the network delay actively. Its performance is very close to that of the local control scheme (i.e., no network). Although it is hard to make the model of the servo motor plant be exactly the same as the real practical one, the results of the real-time experiments are very similar to those of the real-time simulations.

14.5 Conclusion

A new networked predictive control scheme has been proposed for MIMO networked distributed control systems with random network delay and the stochastic stability of the closed-loop networked predictive control systems has also been discussed in this chapter. The criteria for the stochastic stability of the closed-loop networked predictive control system are given for random network delays in both the controller-to-actuator channel and sensor-to-controller channel.

Chapter 15
Networked Data Fusion with Packet Losses and Variable Delays

15.1 Introduction

Multi-senor data fusion has received significant attentions for military and nonmilitary applications in the recent years. Data fusion technique combines data from multiple sensors, and relates information to achieve improved accuracies and more specific inferences than those achieved by a single sensor alone [105, 29, 130, 129, 171]. Applications of multisensor data fusion are widespread. Military applications include: automated target recognition, guidance for autonomous vehicles, remote sensing, battlefield surveillance, and automated threat recognition systems, such as identification-friend-foe-neutral(IFFN) systems. Nonmilitary applications include monitoring of manufacturing processes, condition-based maintenance of complex machinery, robotics, and medical applications.

Now networked data fusion is a new area in the signal processing area. By network, the Data Processing Center (DPC) possesses the ability to process the remote data measured with multi-senors. Particularly, Internet based data fusion allows remote monitoring and fault detection over it, which makes data fusion benefit from the ways of retrieving and processing data form anywhere around the world at any time. The networked data fusion has also opened up a complete new range of real-world applications, namely tele-manufacturing, tele-surgery, museum guidance, space exploration, disaster rescue, and health care. However, the insertion of communication network brings about two issues which do not happen in conventional signal processing scenarios. The one is that network-induced delays occur while exchanging data among devices connected to the shared medium. The other is that some packets not only suffer transmission delays but, even worse, may be lost during transmission in that network itself can be viewed as a web of unreliable paths [214]. Both of the two issues attract a myriad of attentions in the signal processing and control areas where network is frequently employed as an useful tool. [180, 190, 186, 118, 133, 178, 219].

Y. Xia et al.: Analysis and Synthesis of Dynamical Systems, LNCIS 387, pp. 243–270.
springerlink.com

An important and practical problem in data fusion systems is to find an optimal state estimator, based on successive or disconnected measurements [11]. Kalman filter, as it is, is the best known recursive least mean squares (LMS) algorithm to optimally estimate the unknown states of a dynamic system. As long as remote measurement data in time arrive at DPC, the conventional Kalman filtering is carried on, and the resulting state estimations are optimal in the LMS sense. Unfortunately, due to the unreliability of the network, the remote data transmitted over network may fail to arrive in sequence or be completely lost. Especially when serious data losses occur, stability of the networked Kalman filter will be open to question. And also, how to process the delayed data is another issue to be addressed when those data in DPC are fused.

In recent researches on network models, paper [42] considered the state estimation with lost measurements resulting from time-varying channel conditions and introduced a more general multiple state Markovian chain to model the loss and nonloss channel states, and the asymptotic mean squares estimation error for a suboptimal linear estimator was analyzed and optimized by a linear matrix inequality (LMI) approach. In particular, paper [165] developed a suboptimal jump linear estimator for complexity reduction in computing the corrector gain using finite loss history where the loss process is modelled by a two-state Markovian chain. Paper [163] investigated the filtering with independent and identical distributed packet losses by identifying a threshold condition. Paper [81] also modeled the transmission channel with a two-state Markov chain, and provided a sufficient condition for the Kalman filter with the observation losses. In particular, the sufficient condition turns to be a sufficient and necessary one for scalar models. The contribution in paper [81] is mainly cited as a criterion for deciding the stability of the networked Kalman filter.

Many data processing methods have been proposed in recent years in the presence of networks [131, 4, 121]. Paper [4] presented an optimal algorithm for updating with out-of-sequence measurements, however, the method was referring to one-step-lag measurements. A related problem has been studied by paper [174], in which the transmission time from each radar to the fusion center is assumed to be known only statistically but not exactly, that is, no time tag is appended to the measurement. However, this assumption is not realistic for today's wide-band communication links. Paper [131] proposed an algorithm for state estimate with the multiple step lags, however, the algorithm is suboptimal for omitting the white noise estimate.

In this chapter, we present a practical architecture and some algorithms for the networked data fusion system with packet losses and variable delays. The data fusion system proposed in this chapter is based on the well-known Federated Filter. The stability of the networked filter is judged by the criteria proposed in paper [81]. One of the contributions of this chapter is the algorithm presented in the section 15.3.3. Compared with the algorithm in paper [131], the algorithm in this chapter is optimal. The optimal solution for multistep out-of-sequence measurements (OOSM) problem was presented in paper [217]. However, the optimality of the algorithm for updating with OOSMs is lost in the direct discrete-time model of the process noise, and it remains effective only in the discretized continuous-time model and simple OOSM scenario [218]. Besides, unlike contributions in paper [212], the system considered in this chapter is equi-interval sampled and there is completely no prior information about when packet delays occur. And also, the advantage of the proposed algorithm is presented in Theorem 15.8.

This chapter is organized as follows: Section 15.2 gives the problem formulation. Section 15.3 presents the method of data fusion. Subsection 15.3.1 is the recall of the Federated Filter. subsection 15.3.2 shows the results of the stability of networked Kalman Filter. Subsection 15.3.3 details the method to process delay data. Subsection 15.3.4 presents the principle for data fusion. A numerical example is designed in Section 15.4. Section 15.5 presents conclusion.

15.2 Problem Formulation

M sensors provide measurements which are used for data fusion. It is assumed that the network is inserted to transmit the remote data to the Data Processing Center (DPC). DPC is constructed on the basis of the Federated Filter, which comprises the Local Filters (LFs) and Master Filter (MF). Here, the Local Filters receive specific measurements over network. The *ith* Local Filter, for instance, corresponds to the measurements obtained with the *ith* sensor.

Consider the discrete time-variant linear stochastic control system with M sensors:

$$x(k) = \Phi(k, k-1)x(k-1) + \omega(k, k-1) \tag{15.1}$$

$$y_i(k) = H_i(k)x(k) + v_i(k) \tag{15.2}$$

where $x(k) \in R^n$ is the state, $y_i(k) \in R^m$, $i = 1, 2, \ldots, M$ are the measurements, $\omega(k, k-1) \in R^n$ is the input noise, $v_i(k) \in R^m$ are the measurement noises and $\Phi(k, k-1)$, $H_i(k)$ are the time-variant matrices with compatible dimensions.

Assumption 15.2.1 $\omega(k, k-1)$, $v(k)=\{v_i(k), i = 1, 2, \ldots, M\}$ *are the uncorrelated white noises with zero mean and*

$$\mathcal{E}\{\omega(k, k-1)\omega^T(j, j-1)\} = Q(k, k-1)\delta_{kj}$$

$$\mathcal{E}\{v(k)v^T(j)\} = R(k)\delta_{kj}$$

$$\mathcal{E}\{\omega(k,k-1)v^T(j)\} = 0.$$

Assumption 15.2.2 *The initial state $x(0)$ is independent of $\omega(k,k-1)$ and*

$$\mathcal{E}\{x(0)\} = \mu_0$$

$$\mathcal{E}\{[x(0)-\mu_0][x(0)-\mu_0]^T\} = P_0.$$

We will make full use of the measurements from all places to obtain state estimations with high accuracy, and therefore, the well-known Federated Filter is a desirable choice here. However, for packet losses and delays, an investigation to the networked Kalman Filer is necessary, then delayed data processing should also be considered in the course of data fusion. Note that the cross-correlated sensor noises are independent here. Optimal Kalman Filtering fusion formula for sensors network with dependent cross-correlated noises are presented in paper [168]. The method and architecture developed in this chapter hold even if FF is replaced with the proposed distributed filters in paper [168].

15.3 Method of Data Fusion

15.3.1 Recall of Federated Filters

A decentralized filter has an advantage over a centralized filter in the amount of computation and guarantee of stability of the filter. Among the decentralized filters, the Federated filter [20, 21] is a well-known one and employed here to work as 'calculator' in the Data Processing Center (DPC).

The architecture of the Federated Filter chosen in this chapter is shown as Fig. 15.1. The Federated Filter consists of M Local Filters (LFs) which

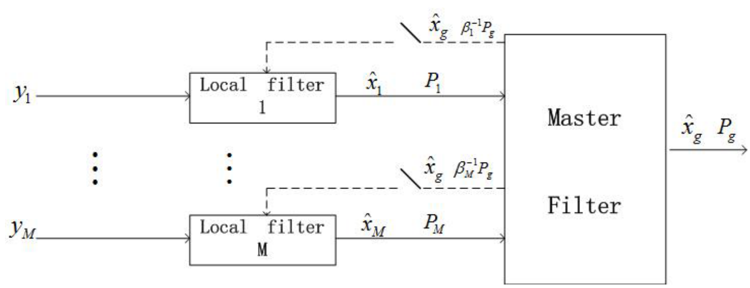

Fig. 15.1 The architecture of Federated Filter

estimate the states with M local measurements by the means of conventional Kalman filtering, and a Master Filter (MF) whose task is only to fuse the data from the M Local Filters. Since the Federated Filter is a mature technology for distributed filtering, we only make a simple recall here.

To eliminate the correlation between the states in different Local filters, information distribution technology is introduced. Assume β_i is the information distribution coefficient for the ith Local Filter. Since the global covariance P can be viewed as a representation of the total amount of the information involved by the global estimation. That is, the state estimation of the ith Local filter $\hat{x}_i$ contributes $\beta_i^{-1} \times P$ to the global estimation, so we have $\sum\limits_{i=1}^{M} \beta_i^{-1} \times P = P$, furthermore,

$$\sum_{i=1}^{M} \beta_i^{-1} = 1. \tag{15.3}$$

From paper [20, 21], when the Kalman filtering is carried on in the Local Filters, to keep the states among Local Filters independent by means of information distributed technology, we must guarantee that

I. initial local covariances are set to be $\beta_i \times$ the common system value, and
II. local filters use $\beta_i \times$ the common process noise covariance value.

Then the Master Filter fuses the data from M Local Filters as the following criteria to derive the global state estimate $\hat{x}_g$.

$$P_g = [\sum_{i=1}^{M} P_i^{-1}]^{-1} \tag{15.4}$$

$$\hat{x}_g = P_g \sum_{i=1}^{M} P_i^{-1} \hat{x}_i \tag{15.5}$$

where $\hat{x_i}$ and P_i are the state estimation and corresponding error covariance obtained in the Local Filter i, and $\hat{x_g}$ and P_g, are the global optimal state estimation and its covariance, shown in Fig. 15.1.

However, it does not suffice to keep the global optimality just with the technology above to fuse the remote data. To resolve the problem, we introduce 'reset' from the Master Filter to the Local Filters. As Fig. 15.1 displays, $\hat{x}_g$ and $\beta_i^{-1} P_g$ are introduced to the ith Local Filter to substitute for the original state estimation and covariance. So the Federated Filter is global optimal with local filter being suboptimal, but if the 'reset' is not introduced, the Federated Filter will be a suboptimal filter.

15.3.2 Stability of Networked Kalman Filtering

The system model is given as follows [81]

$$x(k) = \Phi x(k-1) + \omega(k, k-1) \tag{15.6}$$

where $x(k) \in R^n$ is the state, $\Phi \in R^{n \times n}$ is the system transition matrix, $\omega(k, k-1)$ is a zero-mean Gaussian white noise sequence and the initial state is x_0. The sensor measurements are obtained starting from $k \geq 1$ in the form

$$y(k) = Hx(k) + v(k) \tag{15.7}$$

where $H \in R^{m \times n}$, $y(k) \in R^m$ and then $y(k)$ is transmitted by a channel. Here $\omega(k, k-1)$ and $v(k)$ are two mutually independent sequences of independent and identical distribution (i.i.d) Gaussian white noise with covariance matrices $Q(k, k-1)$ and $R(k) > 0$.

Remark 15.1. Here we omit the subscript of the measurement and let it be $y(k)$ for a concise derivation in this section.

In the conventional Kalman filtering, the covariance $P(k+1|k)$ is defined as $P(k+1|k) = \mathcal{E}\{[x(k+1) - \hat{x}(k+1|k)][x(k+1) - \hat{x}(k+1|k)]^T\}$. Now let $P(k+1|k) = P_{k+1}$. It is easy to derive the following Riccati equation:

$$P_{k+1} = \Phi P_k \Phi^T + Q - \Phi P_k H^T [H P_k H^T + R]^{-1} H P_k \Phi^T, k \geq 1 \tag{15.8}$$

$\gamma_k \in \{0, 1\}$ is used to indicate the arrival (with value 1) of loss (with value 0) of a packet. To capture the temporal correlation of channel variation, γ_k is modeled by a two-state Markovian chain with transition matrix

$$\alpha = \begin{bmatrix} 1-q & q \\ p & 1-p \end{bmatrix} \tag{15.9}$$

where p and q are called the failure rate and recovery rate, respectively , with p, $q > 0$. $1-p$, for instance, denotes the probability of the channel remaining at the normal state 1 after one step if it starts with state 1. Without loss of generality, we adopt the same assumption that $\gamma_1 = 1$ as that in paper [81]. Due to the insertion of the network, Riccati equation (15.8) is modified as follows.

$$P_{k+1} = \Phi P_k \Phi^T + Q - \gamma_k \Phi P_k H^T [H P_k H^T + R]^{-1} H P_k \Phi^T, k \geq 1. \tag{15.10}$$

Assumption 15.3.1 *The failure and recovery rate p, q are both in $(0, 1)$.*

Assumption 15.3.2 *System $[\Phi, H]$ is observable.*

Two sequences $\{\tau_k\}$, $\{\beta_k\}$ are introduced:

$$\tau_1 = \inf\{k : k > 1, \gamma_k = 0\}.$$

It is assumed that the infimum of an empty set is $+\infty$, therefore, τ_1 denotes at the first time the channel state γ_k switches from normal state 1. Similarly, we can define

$$\beta_1 = \inf\{k : k > \tau_1, \gamma_k = 1\}.$$

It is evident that β_k denotes at the kth time the channel state γ_k equals 1, that is, recovering from the abnormal state 0.

It is clear that

$$1 < \tau_1 < \beta_1 < \tau_2 < \cdots < \tau_k < \beta_k < \cdots.$$

Lemma 15.2. [81] Under Assumption 15.3.2, with probability one the two sequences $\{\tau_k, k \geq 1\}$ and $\{\beta_k, k \geq 1\}$, have finite values for each of their entries.

Now we define

$$\beta_k^- = \beta_k - 1$$

and note that β_k^- is the last instant in a period of successive packet losses. The time β_k^- is useful for analyzing the filtering performance in that it provides a basis for estimating to what extent the covariance process may deteriorate due to successive packet losses.

Labeling a subsequence of the covariance P_k by the sequence of times β_k, we denote

$$M_k = P_{\beta_k}.$$

In fact, from the assumption of prediction covariance for Riccati equation in (15.8) M_k denotes the value of the covariance $P(\beta_k|\beta_k^-)$. For an unstable scalar model, starting from $\tau + 1$, P_t monotonically increases to reach a maximum M_k at time β_k before turning downward, therefore, the sequence $M_k,\ k \geq 1$ gives the upper envelope of the covariance process. For this reason, we shall call M_k the peak covariance process [81].

Definition 15.3. [81] The sequence M_k is stable if $\sup_{k>1} \mathcal{E}\|M_k\| < \infty$. Accordingly, it is to say that the filtering system satisfies peak covariance stability.

Definition 15.4. [81] For the observable linear system $[\Phi, H]$, the observability index is the smallest integer I_0 such that $[H^T \quad \Phi H^T \cdots (\Phi^{I_0-1})^T H^T]$ has rank n.

Let S^n denote the set of all $n \times n$ nonnegative definite real matrices, then we define

$$S_0^n = \{P : 0 \leq P \leq \Phi \tilde{P} \Phi^T + Q, for\ some\ \tilde{P} \geq 0\}$$

which is a convex subset of S^n.

We introduce some positive constants. For $1 \leq i \leq (I_0 - 1) \bigvee 1$, and ${d_i}^0$ and ${d_i}^1$ satisfy the following inequality

$$\|F(P)\| \leq {d_i}^1 \|P\| + {d_i}^0 \qquad \forall P \in S_0^n \tag{15.11}$$

where $\bigvee$ denotes choosing a greater number from its left side and right side. By the fact $F(p) \leq \Phi P \Phi^T + Q$, it is clear that the pair (d_i^0, d_i^1) always exists.

Lemma 15.5. [81] The peak covariance process is stable if the following two conditions hold

(1) $|\lambda_\Phi|(1 - q) < 1$

(2) $[1 + \sum_{i=1}^{\infty} d_i^1 (1 - p)^i] \sum_{j=1}^{\infty} \|\Phi^j\|^2 (1 - q)^{j-1} < 1$

where λ_Φ is an eigenvalue of the largest magnitude for matrix Φ.

For the scalar model with *i.i.d* packet losses, the peak covariance stability is equivalent to the usual stability ($i.e., \sup_{t \geq 1} E|P_t| < \infty$). Although for the vector model we can't assert the foregoing equivalence, criteria (15.4), (15.5) are still of usefulness since they cover some practical models with packet loss as high as several percents. This will be used in analyzing the stability of data fusion in this chapter.

15.3.3 Delay Data Processing

As Section 15.1 mentioned, the network can be viewed as a web of unreliable transmission paths, therefore, network-induced delays might exist during packets transmission. In the scope of discussion, the network-induced delay denotes the delayed arrival of the measurements. If such a case occurs, for every measurement containing the information of the system, there is a necessity to make full use of it to assure the integrity of information in the course of data fusion. But the problem is how to deal with the delay data. A method introduced in the following portion is to solve this problem.

Before the proposed algorithm is introduced, we have the following assumption to simplify the analysis without losses of generalization for practical application.

Assumption 15.3.3 *We assume all the sensors are faultless, and all the measurements for the different Local Filters are compressed in the same packets, so the packet losses and delay caused by the unreliability of the transmission by network will be identical to all the Local Filters.*

The system state equation and observation equations are shown in (15.1) and (15.2). We consider the following case as illustrated in Fig. 15.2, where $y_i(k)$ denotes the measurement for the ith Local Filter at time k.

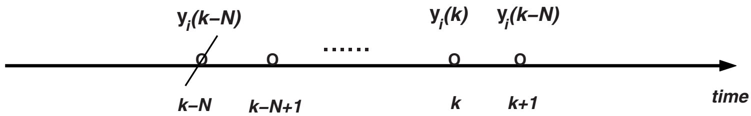

Fig. 15.2 Out of sequence measurement $y_i(k-N)$

As Fig. 15.2 displays, measurement $y_i(k-N)$ which should arrive at the Data Processing Center (DPC) at time $k-N$ arrives at time $k+1$ after time k. When such a case occurs, the DPC will think that measurement $y_i(k-N)$ has lost at time $k-N$, and if criteria (15.4), (15.5) in Section 15.3.2 can be satisfied with the conditions of the system and network, Kalman predication technology can be used at time $k-N$ to have a data fusion. But when time $k+1$ comes, measurement $y_i(k-N)$ arrives, and to keep the optimality of the estimation, utilizing the delayed measurement $y_i(k-N)$ to modify the obtained state estimation $\hat{x}_i(k|k)$ and its error covariance $P_i(k|k)$ is beneficial so that some compensations can be given to the packet losses at time $k-N$.

Defining $\Phi(k,k-j) = \Phi(k,k-1)\cdots\Phi(k-j+1,k-j), j \in Z_+$ and using (15.1), we obtain the relationship between $x(k)$ and $x(k-N)$

$$x(k) = \Phi(k,k-N)x(k-N) + \omega(k,k-N) \tag{15.12}$$

where

$$w(k,k-N) = \sum_{j=1}^{N} \Phi(k,k-j+1)\omega(k-j+1,k-j)$$

and $\Phi(k,k) = I$.

We can derive

$$x(k-N) = \Phi(k-N,k)x(k) + \omega(k-N,k) \tag{15.13}$$

where $\Phi(k-N,k) = \Phi(k,k-N)^{-1}$, $\omega(k-N,k) = -\Phi(k-N,k)\omega(k,k-N)$. It is clear that

$$\mathcal{E}\{\omega(k-N,k)\} = 0. \tag{15.14}$$

Note that $\Phi(k,k-N)$ should always be invertible if it comes from a real system, because $\Phi(k,k-N)$ comes from an exponential that is always invertible.

To utilize the delay measurement $y_i(k-N)$ arriving at time $k+1$, we first construct an estimator for system states at time $k-N$ with the obtained state estimation $\hat{x}_i(k|k)$. Based on (15.13), a new estimator is constructed in the following form

$$\hat{x}_i(k-N|k) = \Phi(k-N,k)\hat{x}_i(k|k) + \hat{\omega}_i(k-N,k|k) \tag{15.15}$$

where $\hat{x}_i(k|k)$ denotes the state estimation for the *ith* Local Filter and $\hat{\omega}_i(k-N,k|k)$ denotes the white noise estimation for the *ith* Local Filter.

For the measurement equation at time $k-N$, we have

$$\begin{aligned} y_i(k-N) = & \ H_i(k-N)\Phi(k-N,k)x(k) \\ & +H_i(k-N)\omega(k-N,k) + v_i(k-N). \end{aligned} \tag{15.16}$$

Then the estimator for $y_i(k-N)$ based on $\hat{x}_i(k|k)$, when $y_i(k-N)$ arrives at time $k+1$, is

$$\begin{aligned} \hat{y}_i(k-N|k) = & \ H_i(k-N)\Phi(k-N,k)\hat{x}_i(k|k) \\ & +H_i(k-N)\hat{\omega}_i(k-N,k|k). \end{aligned} \tag{15.17}$$

Defining $\tilde{y}_i(k|j) = y_i(k) - \hat{y}_i(k|j)$, $\tilde{x}_i(k|j) = x(k) - \hat{x}_i(k|j)$, $\tilde{\omega}_i(k-N,k|k) = \omega(k-N,k) - \hat{\omega}_i(k-N,k)$, we obtain

$$\begin{aligned} \tilde{y}_i(k-N|k) = & \ H_i(k-N)\Phi(k-N,k)\tilde{x}_i(k|k) \\ & +H_i(k-N)\tilde{\omega}_i(k-N,k|k) + v_i(k-N). \end{aligned} \tag{15.18}$$

Due to error information $\tilde{x}_i(k)$ contained in the expression (15.18), $\tilde{y}_i(k-N|k)$ can be used to modify the obtained estimation $\hat{x}_i(k|k)$.

A modified unbiased estimator for the state $x(k)$ in the *ith* Local Filter is

$$\hat{x}_i^*(k|k) = \hat{x}_i(k|k) + M_i\tilde{y}_i(k-N,k). \tag{15.19}$$

From (15.14), (15.15), (15.16), (15.17), (15.18), it is clear that the estimation for the white noise in the *ith* Local Filter must be computed first in order to modify the obtained state estimation $x_i(k|k)$, so we have the following theorem.

Theorem 15.6. *Let the observation error be $\epsilon_i(k-n) = y_i(k-n) - y_i(k-n|k-n-1)$, the covariance for noise $\omega(k-j+1,k-j)$ be $Q(k-j+1,k-j) = \mathcal{E}\{\omega(k-j+1,k-j)\omega^T(k-j+1,k-j)\}$, the covariance for $\omega(k-N,k)$ be $Q(k-N,k) = \mathcal{E}\{\omega(k-N,k)\omega^T(k-N,k)\}$, the error for the estimated white noise $\omega_i(k-N,k|k)$ be $\tilde{\omega}_i(k-N,k|k) = \omega(k-N,k) - \hat{\omega}_i(k-N,k|k)$ and the covariance for estimated white noise $\omega_i(k-N,k|k)$ be $Q_i^*(k-N,k) = \mathcal{E}\{\tilde{\omega}_i(k-N,k|k)\tilde{\omega}_i^T(k-N,k|k)\}$, then the optimal estimator for the white noise $\omega_i(k-N,k)$ is given by*

$$\hat{\omega}_i(k-N,k|k) = -\Phi(k-N,k)\sum_{n=0}^{N-1} C_i(n) \times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) +R_i(k-n)]^{-1}\epsilon_i(k-n) \quad (15.20)$$

where

$$C_i(n) = \{\Phi(k,k-n)Q(k-n,k-n-1) + \sum_{j=n+2}^{N}\Phi(k,k-j+1)Q(k-j+1,k-j)\times \left[\prod_{m=n+1}^{j-1}\Phi(k-m+1,k-m)[I-K_i(k-m)H_i(k-m)]\right]^T\} \times H_i^T(k-n). \quad (15.21)$$

The covariance for the estimated white noise $\omega_i(k-N,k)$ is computed by

$$Q_i^*(k-N,k) = Q(k-N,k) - \Phi(k-N,k)\sum_{n=0}^{N-1} C_i(n) \times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) + R_i(k-n)]^{-1} \times C_i^T(n)\Phi^T(k-N,k) \quad (15.22)$$

where

$$Q(k-N,k) = \Phi(k-N,k)\left[\sum_{j=1}^{N}\Phi(k,k-j+1)Q(k-j+1,k-j)\Phi^T(k,k-j+1)\right] \times\Phi^T(k-N,k). \quad (15.23)$$

Proof. According to the Projection Theorem, the white noises can be estimated in the following form:

$$\hat{\omega}_i(k-N,k|k) = \hat{\omega}_i(k-N,k|k-1) + \mathcal{E}\{\omega(k-N,k)\epsilon_i^T(k)\}\mathcal{E}\{\epsilon_i(k)\epsilon_i^T(k)\}^{-1}\epsilon_i(k)$$

and

$$\mathcal{E}\{\omega(k-N,k)\epsilon_i^T(k)\} = -\Phi(k-N,k) \times\mathcal{E}\{\sum_{j=1}^{N}\Phi(k,k-j+1)\omega(k-j+1,k-j)\,[y_i(k)-\hat{y}_i(k|k-1)]^T\}$$

$$
\begin{aligned}
&= -\Phi(k-N,k)\{Q(k,k-1)H_i^T(k) + \mathcal{E}\{\sum_{j=2}^{N}\Phi(k,k-j+1) \\
&\quad \times\omega(k-j+1,k-j)\tilde{x}_i^T(k-1|k-1))\Phi^T(k,k-1)H_i^T(k)\}\} \\
&= -\Phi(k-N,k)\{Q(k,k-1)H_i^T(k) + \Phi(k,k-1)Q(k-1,k-2) \\
&\quad \times [I-K_i(k-1)H_i(k-1)]^T\,\Phi^T(k,k-1)H_i^T(k) + \mathcal{E}\{\sum_{j=3}^{N}\Phi(k,k-j+1) \\
&\quad \times\omega(k-j+1,k-j)\tilde{x}_i^T(k-2|k-2)\Phi^T(k-1,k-2) \\
&\quad \times [I-K_i(k-1)H_i(k-1)]^T\,\Phi^T(k,k-1)H_i^T(k)\}\} \\
&= -\Phi(k-N,k)\{Q(k,k-1)H_i^T(k) + \Phi(k,k-1)Q(k-1,k-2) \\
&\quad \times [I-K_i(k-1)H_i(k-1)]^T\,\Phi^T(k,k-1)H_i(k) \\
&\quad +\Phi(k,k-2)Q(k-2,k-3)[I-K_i(k-2)H_i(k-2)]^T\Phi^T(k-1,k-2) \\
&\quad \times[I-K_i(k-1)H_i(k-1)]^T\Phi^T(k,k-1)H_i^T(k) + \mathcal{E}\{\sum_{j=4}^{N}\Phi(k,k-j+1) \\
&\quad \times\omega(k-j+1,k-j)\times\tilde{x}_i^T(k-3|k-3)\Phi^T(k-2,k-3) \\
&\quad [I-K_i(k-2)H_i(k-2)] \\
&\quad \times\Phi^T(k-1,k-2)[I-K_i(k-1)H_i(k-1)]^T\Phi^T(k,k-1)H_i^T(k)\}\} \\
&= \vdots \\
&= -\Phi(k-N,k)\{Q(k,k-1) + \sum_{j=2}^{N}\Phi(k,k-j+1)Q(k-j+1,k-j) \\
&\quad \times[\prod_{m=1}^{j-1}\Phi(k-m+1,k-m)[I-K_i(k-m)H_i(k-m)]]^T\}H_i^T(k)
\end{aligned}
$$

and

$$
\mathcal{E}\{\epsilon_i(k)\epsilon_i^T(k)\} = H_i(k)P_i(k|k-1)H_i^T(k) + R_i(k).
$$

By induction, we can obtain

$$
\begin{aligned}
&\hat{\omega}_i(k-N,k|k-n) \\
&= \hat{\omega}_i(k-N,k|k-n-1) + \mathcal{E}\{\omega(k-N,k)\epsilon_i^T(k-n)\} \\
&\quad \times\mathcal{E}\{\epsilon_i(k-n)\epsilon_i^T(k-n)\}^{-1}\epsilon_i(k-n)
\end{aligned}
$$

$$
\mathcal{E}\{\omega(k-N,k)\epsilon_i^T(k-n)\} = -\Phi(k-N,k)C_i(n)
$$

$$
\begin{aligned}
\mathcal{E}\{\epsilon_i(k-n)\epsilon_i^T(k-n)\} = {} & H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) \\
& +R_i(k-n) \qquad (15.24)
\end{aligned}
$$

where

$$C_i(n) = \{\Phi(k, k-n)Q(k-n, k-n-1) + \sum_{j=n+2}^{N} \Phi(k, k-j+1)Q(k-j+1, k-j) \times \left[\prod_{m=n+1}^{j-1} \Phi(k-m+1, k-m)[I - K_i(k-m)H_i(k-m)]\right]^T \} H_i^T(k-n).$$

In view of the fact that $\hat{\omega}_i(k-N+1, k-N|k-N) = 0$, we have

$$\begin{aligned} &\hat{\omega}_i(k-N, k|k) \\ &= -\Phi(k-N, k)\sum_{n=0}^{N-1} C_i(n)[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) \\ &\quad +R_i(k-n)]^{-1}\epsilon_i(k-n). \end{aligned} \tag{15.25}$$

The covariance is

$$\begin{aligned} Q_i^*(k-N, k) &= \mathcal{E}\{\tilde{\omega}_i(k-N, k|k)\tilde{\omega}_i^T(k-N, k|k)\} \\ &= \mathcal{E}\{\tilde{\omega}_i(k-N, k|k)\,[\omega(k-N, k) - \hat{\omega}_i(k-N, k|k)]^T\} \end{aligned}$$

from the Projection Theorem, it is clear that

$$\tilde{\omega}(k-N, k|k) \perp \hat{\omega}(k-N, k|k)$$

then the above expression is rewritten as follows:

$$\begin{aligned} Q_i^*(k-N, k) &= \mathcal{E}\{[\omega(k-N, k) - \hat{\omega}_i(k-N, k|k)]\,\omega^T(k-N, k)\} \\ &= Q(k-N, k) - \mathcal{E}\{\hat{\omega}_i(k-N, k|k)\omega^T(k-N, k)\} \\ &= Q(k-N, k) - \mathcal{E}\{-\Phi(k-N, k)\sum_{n=0}^{N-1} C_i(n) \\ &\quad\times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) \\ &\quad+R_i(k-n)]^{-1}\epsilon_i(k-n)\omega^T(k-N, k)\}. \end{aligned}$$

From the above derivation, we can have the following result immediately,

$$\begin{aligned} Q_i^*(k-N, k) &= Q(k-N, k) - \Phi(k-N, k)\sum_{n=0}^{N-1} C_i(n) \\ &\quad\times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) \\ &\quad+R_i(k-n)]^{-1}C_i^T(n)\Phi^T(k-N, k). \end{aligned}$$

Covariance matrix $Q(k-N,k)$ is easily obtained as follows:

$$
\begin{aligned}
Q(k-N,k) &= \mathcal{E}\{\omega(k-N,k)\omega^T(k-N,k)\} \\
&= \Phi(k-N,k)[\sum_{j=1}^{N}\Phi(k,k-j+1)Q(k-j+1,k-j) \\
&\quad \times\Phi^T(k,k-j+1)]\Phi^T(k-N,k).
\end{aligned}
$$

This completes the proof.

Consequently, the modified state estimation is given in the following theorem.

Theorem 15.7. *Let the modified state estimation error be* $\tilde{x}_i^*(k|k) = x(k) - \hat{x}_i^*(k|k)$*, the covariance for the modified state estimate be*

$$P_i^*(k|k) = \mathcal{E}\{\tilde{x}_i^*(k|k)\tilde{x}_i^{*T}(k|k)\}$$

and the covariance between $\tilde{x}_i(k|k)$ *and* $\tilde{\omega}_i(k,k-N|k)$ *be*

$$P_i^{\tilde{x}\tilde{\omega}} = \mathcal{E}\{\tilde{x}_i(k|k)\tilde{\omega}_i^T(k,k-N|k)\}$$

then the optimal filter for delayed data processing in minimum variance sense is given by

$$\hat{x}_i^*(k|k) = \hat{x}_i(k|k) + M_i[y_i(k-N) - \hat{y}_i(k-N|k)] \tag{15.26}$$

$$M_i = [P_i(k|k)\Phi^T(k-N,k) + P_i^{\tilde{x}\tilde{\omega}}]H_i^T(k-N)W_i^{-1}. \tag{15.27}$$

The covariance is computed by

$$
\begin{aligned}
P_i^*(k|k) = P_i(k|k) - [P_i^{\tilde{x}\tilde{\omega}} + P_i(k|k)\Phi^T(k-N,k)]H_i^T(k-N)W_i^{-1}H_i(k-N) \\
\times[P_i^{\tilde{x}\tilde{\omega}} + P_i(k|k)\Phi^T(k-N,k)]^T
\end{aligned} \tag{15.28}
$$

where

$$
\begin{aligned}
W_i = H_i(k-N)\{\Phi(k-N,k)P_i(k|k)\Phi^T(k-N,k) + \Phi(k-N,k)P_i^{\tilde{x}\tilde{\omega}} \\
+[\Phi(k-N,k)P_i^{\tilde{x}\tilde{\omega}}]^T + Q_i^*(k-N,k)\}H_i^T(k-N^T + R_i(k-N)
\end{aligned}
$$

and

$$
\begin{aligned}
P_i^{\tilde{x}\tilde{\omega}} &= \Phi(k,k-N)\sum_{n=0}^{N-1}P_i(k-N|k-N)D_i^T(n) \\
&\times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) + R_i(k-n)]^{-1} \\
&\times C_i^T(n)\Phi^T(k-N,k) - \Phi(k,k-N)Q_i^*(k-N,k)
\end{aligned}
$$

where

$$D_i(n) = \begin{cases} H_i(k-n)\Phi(k-n,k-n-1) & N=1 \\ H_i(k-n)\Phi(k-n,k-n-1) & \\ \times \prod_{j=n}^{N-2} [I - K_i(k-j-1)H_i(k-j-1)] & \\ \times\Phi(k-j-1,k-j-2) & N>1 \end{cases}.$$

Proof. Using (15.19), we get

$$\begin{aligned} \tilde{x}_i^*(k|k) &= x(k) - \hat{x}_i^*(k|k) \\ &= \tilde{x}_i(k|k) - M_i\tilde{y}_i(k-N|k) \\ &= [I - M_iH_i(k-N)\Phi(k-N,k)]\tilde{x}_i(k|k) \\ &\quad -M_iH_i(k-N)\tilde{\omega}_i(k-N,k|k) - M_iv_i(k-N). \end{aligned} \tag{15.29}$$

Using (15.29), we get

$$\begin{aligned} &P_i^*(k|k) \\ &= \mathcal{E}\{\tilde{x}_i^*(k|k)\tilde{x}_i^{*T}(k|k)\} \\ &= [I - M_iH_i(k-N)\Phi(k-N,k)]P_i(k|k)[I - M_iH_i(k-N)\Phi(k-N,k)]^T \\ &\quad -[I - M_iH_i(k-N)\Phi(k-N,k)]P_i^{\tilde{x}\tilde{\omega}}[M_iH_i(k-N)]^T \\ &\quad -M_iH_i(k-N)[P_i^{\tilde{x}\tilde{\omega}}]^T[I - M_iH_i(k-N)\Phi(k-N,k)]^T \\ &\quad +M_iH_i(k-N)Q_i^*(k-N,k)[M_iH_i(k-N)]^T \\ &\quad +M_iR_i(k-N)M_i^T. \end{aligned} \tag{15.30}$$

The following formula are of great use in our proof.

$$\frac{\partial \mathrm{tr}(XA^T)}{\partial X} = A \tag{15.31}$$

$$\frac{\partial \mathrm{tr}(AX^T)}{\partial X} = A \tag{15.32}$$

$$\frac{\partial \mathrm{tr}(XAX^T)}{\partial X} = X(A + A^T). \tag{15.33}$$

Using the above formula to differentiate $\mathrm{tr}(P_i^*(k|k))$ with respect to M_i for a minimum variance estimate, we can derive

$$M_i = [P_i(k|k)\Phi^T(k-N,k) + P_i^{\tilde{x}\tilde{\omega}}]H_i^T(k-N)W_i^{-1} \tag{15.34}$$

where

$$W_i = H_i(k-N)\{\Phi(k-N,k)P_i(k|k)\Phi^T(k-N,k) + \Phi(k-N,k)P_i^{\tilde{x}\tilde{\omega}} \\ +[\Phi(k-N,k)P_i^{\tilde{x}\tilde{\omega}}]^T + Q_i^*(k-N,k)\}H_i^T(k-N) + R_i(k-N)$$

$P_i^{\tilde{x}\tilde{\omega}}$ is derived in the following way,

$$\begin{aligned} &P_i^{\tilde{x}\tilde{\omega}} = \mathcal{E}\{\tilde{x}_i(k|k)\tilde{\omega}_i^T(k-N,k|k)\} \\ &= \mathcal{E}\{[\Phi(k,k-1)\tilde{x}_i(k-1|k-1) + \omega(k,k-1) + K_i(k)\epsilon_i(k)]\tilde{\omega}_i^T(k-N,k|k)\} \\ &= \mathcal{E}\{[\Phi(k,k-1)\Phi(k-1,k-2)\tilde{x}_i(k-2|k-2) + \Phi(k,k-1)\omega(k-1,k-2) \\ &+ \omega(k,k-1) + \Phi(k,k-1)K_i(k-1)\epsilon_i(k-1) + K_i(k)\epsilon_i(k)]\tilde{\omega}_i^T(k-N,k|k)\}. \end{aligned}$$

From Projection Theorem, we know $\tilde{\omega}_i(k-N,k|k) \perp \epsilon_i(k-n), n = 0,1,\dots,k-1$, so the above expression can be rewritten by induction in the following form.

$$\begin{aligned} P_i^{\tilde{x}\tilde{\omega}} &= \mathcal{E}\{[\Phi(k,k-N)\tilde{x}_i(k-N|k-N) \\ &\quad + \sum_{j=1}^{N}\Phi(k,k-j+1)\omega(k-j+1,k-j)]\tilde{\omega}_i^T(k-N,k|k)\} \\ &= \mathcal{E}\{\Phi(k,k-N)\tilde{x}_i(k-N|k-N)\tilde{\omega}_i^T(k-N,k|k)\} \\ &\quad + \mathcal{E}\{\omega(k,k-N)\tilde{\omega}_i^T(k-N,k|k)\} \\ &= \mathcal{E}\{\Phi(k,k-N)\tilde{x}_i(k-N|k-N)\tilde{\omega}_i^T(k-N,k|k)\} \\ &\quad - \Phi(k,k-N)\mathcal{E}\{\omega(k-N,k)\tilde{\omega}_i^T(k-N,k|k)\} \\ &= \mathcal{E}\{\Phi(k,k-N)\tilde{x}_i(k-N|k-N)\tilde{\omega}_i^T(k-N,k|k)\} \\ &\quad - \Phi(k,k-N)Q_i^*(k-N,k). \end{aligned} \tag{15.35}$$

The last term follows from Projection Theorem.
$\mathcal{E}\{\Phi(k,k-N)\tilde{x}_i(k-N|k-N)\tilde{\omega}_i^T(k-N,k|k)\}$ can be obtained as follows,

$$\begin{aligned} &\mathcal{E}\{\Phi(k,k-N)\tilde{x}_i(k-N|k-N)\tilde{\omega}_i^T(k-N,k|k)\} \\ &= \Phi(k,k-N)\mathcal{E}\{\tilde{x}_i(k-N|k-N)[\omega(k-N,k) - \hat{\omega}_i(k-N,k|k)]^T\} \\ &= -\Phi(k,k-N)\mathcal{E}\{\tilde{x}_i(k-N|k-N)\hat{\omega}_i^T(k-N,k|k)\} \\ &= \Phi(k,k-N)\sum_{n=0}^{N-1}\mathcal{E}\{\tilde{x}_i(k-N|k-N)\epsilon_i^T(k-n)\} \\ &\quad \times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n) + R_i(k-n)]^{-1} \\ &\quad \times C_i^T(n)\Phi^T(k-N,k) \end{aligned}$$

$\mathcal{E}\{\tilde{x}_i(k-N|k-N)\epsilon_i^T(k-n)\}$ is derived as follow,

$$\begin{aligned} &\mathcal{E}\{\tilde{x}_i(k-N|k-N)\epsilon_i^T(k-n)\} \\ &= \mathcal{E}\{\tilde{x}_i(k-N|k-N)[H_i(k-n)\Phi(k-n,k-n-1)\tilde{x}_i(k-n-1|k-n-1) \\ &\quad + H_i(k-n)\omega(k-n,k-n-1) + v_i(k-n)]^T\} \end{aligned}$$

$$
\begin{aligned}
&= \mathcal{E}\{\tilde{x}_i(k-N|k-N)[H_i(k-n)\Phi(k-n,k-n-1)\tilde{x}_i(k-n-1|k-n-1)]^T\} \\
&= \mathcal{E}\{\tilde{x}_i(k-N|k-N)[H_i(k-n)\Phi(k-n,k-n-1)[I-K_i(k-n-1) \\
&\quad H_i(k-n-1)]\Phi(k-n-1,k-n-2)\tilde{x}_i(k-n-2|k-n-2)]^T\} \\
&= \vdots \\
&= P_i(k-N|k-N)D_i^T(n)
\end{aligned}
$$

where

$$
D_i(n) = \begin{cases} H_i(k-n)\Phi(k-n,k-n-1) & N=1 \\ H_i(k-n)\Phi(k-n,k-n-1) & \\ \times \prod_{j=n}^{N-2}[I-K_i(k-j-1)H_i(k-j-1)] & \\ \times\Phi(k-j-1,k-j-2) & N>1 \end{cases} .
$$

Then we know that

$$
\begin{aligned}
P_i^{\tilde{x}\tilde{\omega}} &= \Phi(k,k-N)P_i(k-N|k-N)\sum_{n=0}^{N-1} D_i^T(n) \\
&\times[H_i(k-n)P_i(k-n|k-n-1)H_i^T(k-n)+R_i(k-n)]^{-1} \\
&\times C_i^T(n)\Phi^T(k-N,k)-\Phi(k,k-N)Q_i^*(k-N,k).
\end{aligned} \tag{15.36}
$$

Substituting (15.34) into (15.30) yields

$$
\begin{aligned}
P_i^*(k|k) &= P_i(k|k) - P_i(k|k)\Phi^T(k-N,k)H_i^T(k-N)W_i^{-1}H_i(k-N)\times \\
&\quad \Phi(k-N,k)P_i(k|k) - P_i^{\tilde{x}\tilde{\omega}}H_i^T(k-N)W_i^{-1}H_i(k-N)[P_i^{\tilde{x}\tilde{\omega}}]^T \\
&\quad -P_i(k|k)\Phi^T(k-N,k)H_i^T(k-N)W_i^{-1}H_i(k-N)[P_i^{\tilde{x}\tilde{\omega}}]^T \\
&\quad -P_i^{\tilde{x}\tilde{\omega}}H_i^T(k-N)W_i^{-1}H_i(k-N)\Phi(k-N,k)P_i(k|k) \\
&= P_i(k|k) - [P_i^{\tilde{x}\tilde{\omega}} + P_i(k|k)\Phi^T(k-N,k)]H_i^T(k-N)W_i^{-1}H_i(k-N) \\
&\quad [P_i^{\tilde{x}\tilde{\omega}} + P_i(k|k)\Phi^T(k-N,k)]^T.
\end{aligned} \tag{15.37}
$$

This completes the proof.

If the delay time is equal to 1, it is clear that the main results in paper [4] is a special case of Theorem 15.7. To show the advantage of the proposed algorithm, we have the following theorem.

Theorem 15.8. *The relation between $P_i(k|k)$ and $P_i^*(k|k)$ is that*

$$0 \le P_i^*(k|k) \le P_i(k|k).$$

Proof. From the definition of $P_i^*(k|k)$, it is clear that $0 \le P_i^*(k|k)$.

From (15.28), we can obtain

$$P_i(k|k) - P^*(k|k) = [P_i^{\tilde{x}\tilde{\omega}} + P_i(k|k)\Phi^T(k-N,k)]H_i^T(k-N)W_i^{-1}H_i(k-N) \times [P_i^{\tilde{x}\tilde{\omega}} + P_i(k|k)\Phi^T(k-N,k)]^T.$$

It is clear that $P_i(k|k) - P^*(k|k) \geq 0$ if $W_i \geq 0$.
From (15.18), we can have

$$\begin{aligned} &\mathcal{E}\{\tilde{y}_i(k-N|k)\tilde{y}_i^T(k-N|k)\} \\ &= \mathcal{E}\{[H_i(k-N)\Phi(k-N,k)\tilde{x}_i(k|k) + H_i(k-N)\tilde{\omega}_i(k-N,k|k) + v_i(k-N)] \\ &\quad \times [H_i(k-N)\Phi(k-N,k)\tilde{x}_i(k|k) + H_i(k-N)\tilde{\omega}_i(k-N,k|k) + v_i(k-N)]^T\} \\ &= W_i. \end{aligned}$$

Now we can know that

$$W_i \geq 0.$$

Furthermore,

$$P_i^*(k|k) \leq P_i(k|k).$$

This completes the proof.

Remark 15.9. In this section, symbols $x(k)$, and $\omega(k-N,k)$ represent the state and noise of the original system, respectively. However, the outputs of the two estimators, $\hat{x}_i(k|k)(\hat{x}_i^*(k|k))$, and $\hat{\omega}_i(k-N,k|k)$ are the estimations for the state $x(k)$ and the noise $\omega(k-N,k)$ of the original system from the *ith* Local Filter as different Local Filters receives different measurements obtained with different sensors. Therefore, two errors for the *ith* Local Filter are defined as $\tilde{x}_i(k|k) = x(k) - \hat{x}_i(k|k)(\tilde{x}_i^*(k|k) = x(k) - \hat{x}_i^*(k|k))$, $\tilde{\omega}_i(k-N,k|k) = \omega(k-N,k) - \hat{\omega}_i(k-N,k|k)$.

One may ask if the stability can be guaranteed if the delayed data are processed by the proposed algorithm in Theorems 15.6 and 15.7, as for this problem, we have the following theorem.

Theorem 15.10. *Local Filters are peak covariance stability if the delayed data are processed by the proposed algorithm in Theorems 15.6 and 15.7.*

Proof. The conclusion follows from the definition of the peak covariance stability and Theorem 15.8 directly.

The essential of our proposed algorithm can be interpreted in the way of projection theory. Assume set $I = \{y_i(t) : t \neq k-N\}$, which indicates that the measurement $y_i(k-N)$ may be lost or delayed. Assume at time $t = k$, state estimation $\hat{x}_i(k|k)$ and covariance $P_i(k|k)$ have been obtained by the conventional kalman filtering technology. According to the projection theory, $[x(k) - \hat{x}_i(k|k)] \perp I$, however, when time $k+1$ comes, measurement $y_i(k-N)$ arrives too, and measurement set becomes $I \bigcup y_i(k-N)$. Due to the change of measurement set, we cannot assert that $[x(k) - \hat{x}_i(k|k)] \perp [I \bigcup y_i(k-N)]$, that is, $\hat{x}_i(k|k)$ is no longer the optimal state estimate for $x(k)$ in the ith Local Filter, therefore, as presented in this section, a new algorithm is designed to modify $\hat{x}_i(k|k)$ to $x_i^*(k|k)$ so that $[x(k) - \hat{x}_i^*(k|k)] \perp [I \bigcup y_i(k-N)]$.

15.3.4 Principle for Data Fusion

The architecture for networked data fusion is shown as Fig. 15.3.

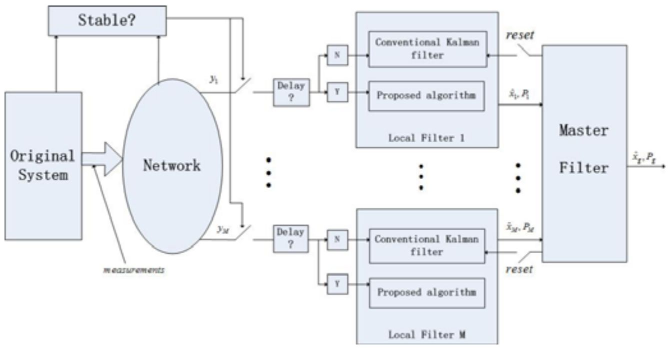

Fig. 15.3 Architecture for networked data fusion

Under Assumption 15.3.3, all the sensors are faultless, and the measurements are compressed in one packet for transmission by network, however, before the transmission, to guarantee the stability of the networked Kalman filter, we must make a judgement on the network by criteria (15.4), (15.5). If these two criteria can be met, the next work can be done, otherwise, some measures must be taken to improve the quality of the network used for transmission.

For the measurements are compressed in one packet, the scenarios of packet losses and delay are identical to all the Local Filters, in the view of this, there are two cases that should be considered in the Local Filters.

1. The packet is not delayed on transmission. There are also two scenarios in this case.
   a. The packet isn't lost. For this normal case, conventional Kalman filtering is applied in the Local Filters.
   b. The packet is lost on transmission. When such a case occurs, due to the lack of the measurements, conventional Kalman predication technology has to be applied to predicate the state in the Local Filters.

2. The packet is delayed on transmission. For the delayed packet, the proposed algorithm in Section 15.3.3 can be applied to compensate the delay.

No matter which case as the above described occurs, (15.4) and (15.5) in Section 15.3.1 are always the algorithm expressions for the Master Filter.

As Fig. 15.3 displays, when the packet is not delayed, the 'reset' is introduced to the Local Filter, so in such case, the estimation in the Local Filter is suboptimal and the global estimation is optimal, however, in another case, the 'reset' is not introduced. From Theorem 15.7, the estimation in the Local Filter is optimal, but the global estimation is suboptimal.

15.4 Numerical Examples

From criteria (15.4), (15.5), stability of networked Kalman filter based on a unstable model can be identified. When it comes to a stable model, the networked Kalman filter is stable under any network condition. What's more, it is stable in a sense of usual stability. In view of this, we design two numerical examples, an unstable and a stable one, respectively, to show the effectiveness of our method of data fusion.

15.4.1 An Unstable Example

We consider such an example as illustrated in Fig. 15.4.

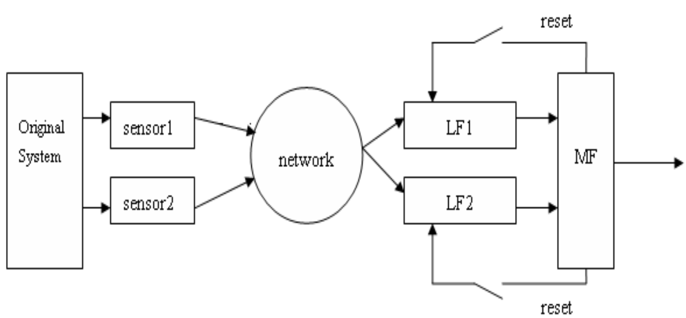

Fig. 15.4 Structure for data fusion

Measurements obtained from the original system located in the distant place are transmitted to the DPC over network. Due to the existence of effect of noises while obtaining measurements with sensor 1 and sensor 2, there is a necessity to utilize the Federated Filter to remove the effect of noise on measurements and improve the accuracy of state estimation.

To make the model nontrivial, we choose the vector model example in paper [81] directly. Let the system be specified by

$$\Phi(k,k-1)=\begin{bmatrix}1.3 & 0.3\\ 0 & 1.2\end{bmatrix}, H_1(k)=\begin{bmatrix}1 & 1\end{bmatrix}, H_i=\begin{bmatrix}1 & 1.5\end{bmatrix}, Q=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}.$$

Assume $R_1 = 1.0$, $R_2 = 1.5$, $\beta_1 = \beta_2 = 2$ and take $q = 0.65$, $p = 0.03$. In paper [81], the Networked Kalman Filter under such assumptions has proven to be peak covariance stable, which can also be seen in the following simulation.

Then a γ_k sequence corresponding to the chosen parameters p and q can be chosen by analyzing the transition matrix (15.9). Let X denote the total simulation steps and N_1 denote how many times measurements are successfully accepted in the DPC, then in view of (15.9), the times channel state γ_k switches from normal state 1 to abnormal state 0 are about $N_1 p$ and the average times channel state stays at state 0 are about $\sum_{j=1}^{\infty} j(1-q)^{j-1}$, so we have

$$N_1 + N_1 p \sum_{j=1}^{\infty} j(1-q)^{j-1} \approx X. \tag{15.38}$$

According to expression (15.38), we derive

$$N_1 \approx \frac{q}{p+q} X. \tag{15.39}$$

If we take $X = 1000$, then $N_1 \approx 956$, $N_1 p \approx 28$ and we can have Table 15.1, where k denotes the times channel state γ_k stays at abnormal state

Table 15.1 Transmission failure times

k	P	T
1	0.0195	19
2	0.006825	7
3	0.002389	2
4	0.000836	1
5	0.000293	0

0 if it starts with state 1 and P denotes the probability of that foregoing case. T denotes how many times that case may emerge in the γ_k sequence for simulation. Note that T is a round number, so it permits slightly modification.

Two cases are considered in our simulation. The one is that there only exist data losses in the network, and the other is that there exist not only

data losses but also network-induced delay.
Case 1:

As described above, we choose the following γ_k sequence as illustrated in Fig. 15.5.

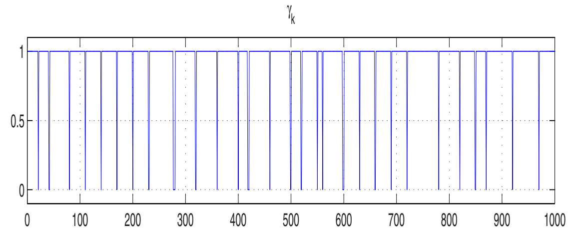

Fig. 15.5 γ_k sequence for stability judgement

The result of simulation can be shown by observing the variation in the length of the trace of covariance. Fig. 15.6 is the traces of error covariances for Local Filters 1 and 2 and Master Filter with the γ_k sequence as shown in Fig. 15.5. It is clear that filters are stable in the sense of peak covariance process with parameter $q = 0.65$, $p = 0.03$. Besides, it can clearly be seen that each filter's accuracy is improved much.

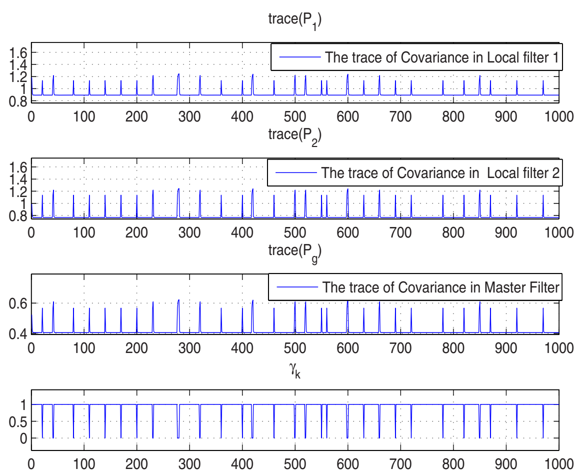

Fig. 15.6 Trace for covariance

Case 2:

Fig. 15.7 is the γ_k sequence chosen to denote the network with packet losses and delays. Note that here $\gamma_k = N(N \geq 1)$ indicates that packet lags $N-1$ times behind the time at which it should arrive.

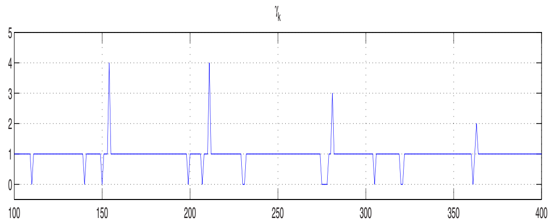

Fig. 15.7 γ_k sequence for network with data losses and delay

Table 15.2 Comparison of covariance for Local Filter 1 with delayed data being processed

N	$P_1(k\|k)$	$P_1^*(k\|k)$
4	$\begin{bmatrix} 0.5526 & -0.1906 \\ -0.1906 & 0.5202 \end{bmatrix}$	$\begin{bmatrix} 0.5457 & -0.1106 \\ -0.1106 & 0.4020 \end{bmatrix}$
5	$\begin{bmatrix} 0.5483 & -0.1918 \\ -0.1918 & 0.5197 \end{bmatrix}$	$\begin{bmatrix} 0.5471 & -0.1827 \\ -0.1827 & 0.0483 \end{bmatrix}$
3	$\begin{bmatrix} 0.5745 & -0.1693 \\ -0.1693 & 0.5421 \end{bmatrix}$	$\begin{bmatrix} 0.5519 & -0.1879 \\ -0.1879 & 0.5104 \end{bmatrix}$
2	$\begin{bmatrix} 0.5483 & -0.1918 \\ -0.1918 & 0.5197 \end{bmatrix}$	$\begin{bmatrix} 0.5469 & -0.1918 \\ -0.1918 & 0.5194 \end{bmatrix}$

Table 15.3 Comparison of covariance for Local Filter 2 with delayed data being processed

N	$P_2(k\|k)$	$P_2^*(k\|k)$
4	$\begin{bmatrix} 0.5765 & -0.2426 \\ -0.2426 & 0.3664 \end{bmatrix}$	$\begin{bmatrix} 0.5962 & -0.2347 \\ -0.2347 & 0.3475 \end{bmatrix}$
5	$\begin{bmatrix} 0.5723 & -0.2423 \\ -0.2423 & 0.3603 \end{bmatrix}$	$\begin{bmatrix} 0.5516 & -0.2001 \\ -0.2001 & 0.2769 \end{bmatrix}$
3	$\begin{bmatrix} 0.5941 & -0.2354 \\ -0.2354 & 0.3699 \end{bmatrix}$	$\begin{bmatrix} 0.5759 & -0.2432 \\ -0.2432 & 0.3657 \end{bmatrix}$
2	$\begin{bmatrix} 0.5723 & -0.2423 \\ -0.2423 & 0.3663 \end{bmatrix}$	$\begin{bmatrix} 0.5706 & -0.2424 \\ -0.2424 & 0.3659 \end{bmatrix}$

Table 15.4 Comparison of covariance for the Master Filter with delayed data being processed

N	$P_g(k\|k)$	$P_g^*(k\|k)$
4	$\begin{bmatrix} 0.2815 & -0.1106 \\ -0.1106 & 0.2106 \end{bmatrix}$	$\begin{bmatrix} 0.2775 & -0.1036 \\ -0.1036 & 0.1841 \end{bmatrix}$
5	$\begin{bmatrix} 0.2794 & -0.1107 \\ -0.1107 & 0.2105 \end{bmatrix}$	$\begin{bmatrix} 0.2774 & -0.0911 \\ -0.0911 & 0.0167 \end{bmatrix}$
3	$\begin{bmatrix} 0.2909 & -0.1046 \\ -0.1046 & 0.2147 \end{bmatrix}$	$\begin{bmatrix} 0.2811 & -0.1102 \\ -0.1102 & 0.2087 \end{bmatrix}$
2	$\begin{bmatrix} 0.2794 & -0.1107 \\ -0.1107 & 0.2105 \end{bmatrix}$	$\begin{bmatrix} 0.2787 & -0.1108 \\ -0.1108 & 0.2103 \end{bmatrix}$

Three tables, as shown in Tables 15.2, 15.3 and 15.4 can be obtained by the principle proposed in Section 15.3.4, where N in the first column denotes the value of γ_k greater than 1 with corresponding sequence in Fig. 15.7, $P_i(k|k)$ and $P_i^*(k|k)$ are covariance terms in the (15.28), respectively, and $P_g(k|k)$ and $P_g^*(k|k)$ are the covariance at the time before the delayed data are processed and the one with delayed data being processed in Master Filter.

These three tables can show the effectiveness of the algorithm for processing the delayed data and the rationality of the architecture for data fusion in Section 15.3.4. First, the stability of the networked filter in the peak covariance sense can be guaranteed by criteria (15.4), and (15.5), and Theorem 15.10. Then from these three tables, the covariances for the Local Filters and Master Filter reduce as shown in Theorem 15.8.

15.4.2 A Stable Example

A stable example is modeled by $\Phi(k,k-1) = \begin{bmatrix} 0.8 & 0.3 \\ 0 & 0.9 \end{bmatrix}$, $H_1 = \begin{bmatrix} 1 & 1 \end{bmatrix}$, $H_2 = \begin{bmatrix} 1 & 1.5 \end{bmatrix}$, $R_1 = 1$, $R_2 = 1.5$, $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Since the networked Kalman filter based on a stable model is stable under any condition, we still take $q = 0.65$, $p = 0.03$, $\beta_1 = 2$, $\beta_2 = 2$ for the numerical example.

First of all, let's compare the difference between the Federated Filter with reset and that without reset, and the following γ_k sequence as illustrated in Fig. 15.8, which still comply with the requirements of Table 15.1, is chosen here. Note that Fig. 15.8 only displays the part needed for showing the results of our comparison in the γ_k sequence, and in the other part , we have $\gamma_k = 1$.

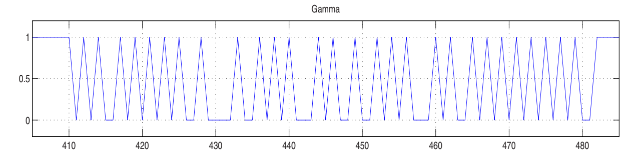

Fig. 15.8 γ_k sequence for comparison

As Fig. 15.8 displays, the network becomes bad suddenly in a certain part of γ_k sequence, which can be used for comparing the difference between the Federate Filter with reset and that without reset, and the simulation results are illustrated in Fig. 15.9 and Fig. 15.10:

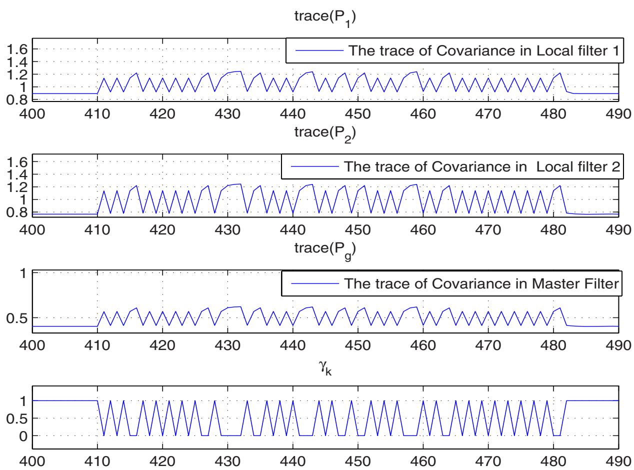

Fig. 15.9 Trace for the covariance of Federated Filter with reset

It is apparent that the error covariance for the Federate Filter with reset has much higher accuracy, however, the Federated Filter without reset can recover from the abnormal state in the length of a larger number more quickly.

The γ_k sequence, used for Case 2 in the last numerical example is shown in Fig. 15.7.

The process noises for the simulation are shown in Fig. 15.11.

The result of simulation for the first component of the system state is shown in Fig. 15.12.

The result of simulation for the second component of the system state is shown in Fig. 15.13 .

Like the simulation in the last subsection, three tables as shown in Tables 15.5, 15.6 and 15.7 can be obtained to show the effectiveness of the algorithm

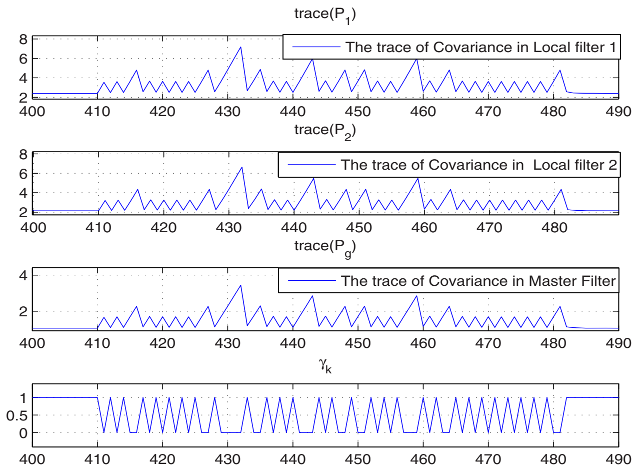

Fig. 15.10 Trace for the covariance of Federated Filter without reset

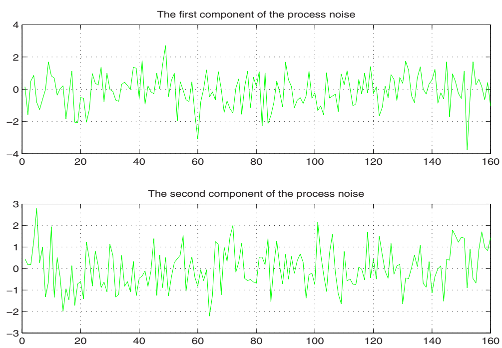

Fig. 15.11 The process noise in the system

for processing the delayed data and the rationality of the architecture for data fusion in Section 15.3.4.

15.5 Conclusion

In this chapter, a method of data fusion over network with packet losses and delays has been considered. The Federated Filter is used for multi-sensor data fusion, and the stability of networked filter can be guaranteed by known criteria. The method to process delay data using the Federated Filter has

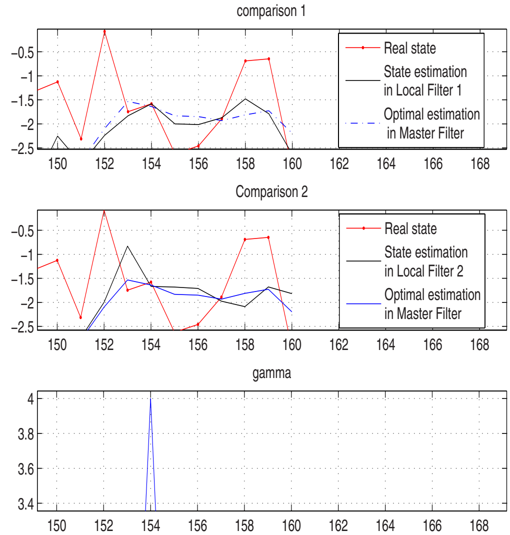

Fig. 15.12 Comparisons of estimation for the first component of the the system state based on the input process noise shown in Fig. 15.11

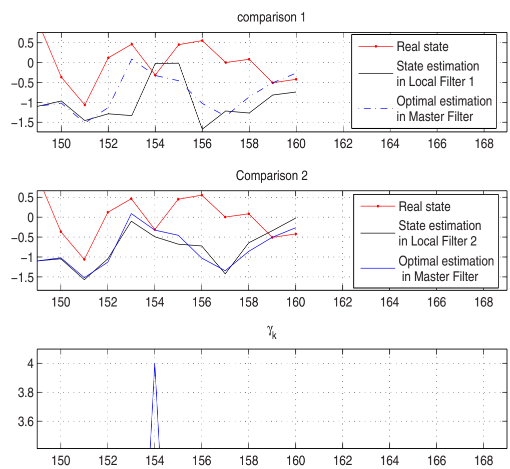

Fig. 15.13 Comparisons of estimation for the second component of the the system state based on the input process noise shown in Fig. 15.11

Table 15.5 Comparison of covariance for Local Filter 1 with delayed data being processed

N	$P_1(k\|k)$	$P_1^*(k\|k)$
4	$\begin{bmatrix} 0.4428 & -0.1241 \\ -0.1241 & 0.4498 \end{bmatrix}$	$\begin{bmatrix} 0.0666 & -0.0103 \\ -0.0103 & 0.4018 \end{bmatrix}$
3	$\begin{bmatrix} 0.4597 & -0.1090 \\ -0.1090 & 0.4650 \end{bmatrix}$	$\begin{bmatrix} 0.3467 & -0.1352 \\ -0.1352 & 0.4487 \end{bmatrix}$
2	$\begin{bmatrix} 0.4428 & -0.1241 \\ -0.1241 & 0.4498 \end{bmatrix}$	$\begin{bmatrix} 0.4364 & -0.1328 \\ -0.1328 & 0.4376 \end{bmatrix}$

Table 15.6 Comparison of covariance for Local Filter 2 with delayed data being processed

N	$P_2(k\|k)$	$P_2^*(k\|k)$
4	$\begin{bmatrix} 0.4523 & -0.1709 \\ -0.1709 & 0.3150 \end{bmatrix}$	$\begin{bmatrix} 0.2560 & -0.2332 \\ -0.2332 & 0.2952 \end{bmatrix}$
3	$\begin{bmatrix} 0.4655 & -0.1656 \\ -0.1656 & 0.3179 \end{bmatrix}$	$\begin{bmatrix} 0.4431 & -0.2047 \\ -0.2047 & 0.2066 \end{bmatrix}$
2	$\begin{bmatrix} 0.4523 & -0.1709 \\ -0.1709 & 0.3150 \end{bmatrix}$	$\begin{bmatrix} 0.4509 & -0.1598 \\ -0.1598 & 0.2195 \end{bmatrix}$

Table 15.7 Comparison of covariance for the Master Filter with delayed data being processed

N	$P_g(k\|k)$	$P_g^*(k\|k)$
4	$\begin{bmatrix} 0.2230 & -0.0761 \\ -0.0761 & 0.1821 \end{bmatrix}$	$\begin{bmatrix} 0.0470 & -0.0343 \\ -0.0343 & 0.0935 \end{bmatrix}$
3	$\begin{bmatrix} 0.2302 & -0.0718 \\ -0.0718 & 0.1853 \end{bmatrix}$	$\begin{bmatrix} 0.1941 & -0.0866 \\ -0.0866 & 0.1259 \end{bmatrix}$
2	$\begin{bmatrix} 0.2230 & -0.0761 \\ -0.0761 & 0.1821 \end{bmatrix}$	$\begin{bmatrix} 0.2215 & -0.0753 \\ -0.0753 & 0.1411 \end{bmatrix}$

been proposed. Finally, two examples are given to illustrate the attainments in this chapter, and the simulation results have shown the consistency of the analysis in Section 15.3.

Index

Lecture Notes in Control and Information Sciences

Edited by M. Thoma, F. Allgöwer, M. Morari

Further volumes of this series can be found on our homepage:
springer.com

Vol. 387: Xia, Y.;
Fu, M.; Shi, P.:
Analysis and Synthesis of Dynamical Systems
with Time-Delays
283 p. 2009 [978-3-642-02695-9]

Vol. 386: Huang, D.;
Nguang, S.K.:
Robust Control for Uncertain Networked Control
Systems with Random Delays
159 p. 2009 [978-1-84882-677-9]

Vol. 385: Jungers, R.:
The Joint Spectral Radius
144 p. 2009 [978-3-540-95979-3]

Vol. 384: Magni, L.; Raimondo, D.M.;
Allgöwer, F. (Eds.):
Nonlinear Model Predictive Control
572 p. 2009 [978-3-642-01093-4]

Vol. 383: Sobhani-Tehrani E.;
Khorasani K.;
Fault Diagnosis of Nonlinear Systems
Using a Hybrid Approach
360 p. 2009 [978-0-387-92906-4]

Vol. 382: Bartoszewicz A.;
Nowacka-Leverton A.;
Time-Varying Sliding Modes for Second
and Third Order Systems
192 p. 2009 [978-3-540-92216-2]

Vol. 381: Hirsch M.J.; Commander C.W.;
Pardalos P.M.; Murphey R. (Eds.)
Optimization and Cooperative Control Strategies:
Proceedings of the 8th International Conference
on Cooperative Control and Optimization
459 p. 2009 [978-3-540-88062-2]

Vol. 380: Basin M.
New Trends in Optimal Filtering and Control for
Polynomial and Time-Delay Systems
206 p. 2008 [978-3-540-70802-5]

Vol. 379: Mellodge P.; Kachroo P.;
Model Abstraction in Dynamical Systems:
Application to Mobile Robot Control
116 p. 2008 [978-3-540-70792-9]

Vol. 378: Femat R.; Solis-Perales G.;
Robust Synchronization of Chaotic Systems
Via Feedback
199 p. 2008 [978-3-540-69306-2]

Vol. 377: Patan K.
Artificial Neural Networks for
the Modelling and Fault
Diagnosis of Technical Processes
206 p. 2008 [978-3-540-79871-2]

Vol. 376: Hasegawa Y.
Approximate and Noisy Realization of
Discrete-Time Dynamical Systems
245 p. 2008 [978-3-540-79433-2]

Vol. 375: Bartolini G.; Fridman L.; Pisano A.;
Usai E. (Eds.)
Modern Sliding Mode Control Theory
465 p. 2008 [978-3-540-79015-0]

Vol. 374: Huang B.; Kadali R.
Dynamic Modeling, Predictive Control
and Performance Monitoring
240 p. 2008 [978-1-84800-232-6]

Vol. 373: Wang Q.-G.; Ye Z.; Cai W.-J.;
Hang C.-C.
PID Control for Multivariable Processes
264 p. 2008 [978-3-540-78481-4]

Vol. 372: Zhou J.; Wen C.
Adaptive Backstepping Control of Uncertain
Systems
241 p. 2008 [978-3-540-77806-6]

Vol. 371: Blondel V.D.; Boyd S.P.;
Kimura H. (Eds.)
Recent Advances in Learning and Control
279 p. 2008 [978-1-84800-154-1]

Vol. 370: Lee S.; Suh I.H.;
Kim M.S. (Eds.)
Recent Progress in Robotics:
Viable Robotic Service to Human
410 p. 2008 [978-3-540-76728-2]

Vol. 369: Hirsch M.J.; Pardalos P.M.;
Murphey R.; Grundel D.
Advances in Cooperative Control and
Optimization
423 p. 2007 [978-3-540-74354-5]

Vol. 368: Chee F.; Fernando T.
Closed-Loop Control of Blood Glucose
157 p. 2007 [978-3-540-74030-8]

Vol. 367: Turner M.C.; Bates D.G. (Eds.)
Mathematical Methods for Robust and Nonlinear
Control
444 p. 2007 [978-1-84800-024-7]

Vol. 366: Bullo F.; Fujimoto K. (Eds.)
Lagrangian and Hamiltonian Methods for Nonlinear Control 2006
398 p. 2007 [978-3-540-73889-3]

Vol. 365: Bates D.; Hagström M. (Eds.)
Nonlinear Analysis and Synthesis Techniques for Aircraft Control
360 p. 2007 [978-3-540-73718-6]

Vol. 364: Chiuso A.; Ferrante A.; Pinzoni S. (Eds.)
Modeling, Estimation and Control
356 p. 2007 [978-3-540-73569-4]

Vol. 363: Besançon G. (Ed.)
Nonlinear Observers and Applications
224 p. 2007 [978-3-540-73502-1]

Vol. 362: Tarn T.-J.; Chen S.-B.; Zhou C. (Eds.)
Robotic Welding, Intelligence and Automation
562 p. 2007 [978-3-540-73373-7]

Vol. 361: Méndez-Acosta H.O.; Femat R.; González-Álvarez V. (Eds.):
Selected Topics in Dynamics and Control of Chemical and Biological Processes
320 p. 2007 [978-3-540-73187-0]

Vol. 360: Kozlowski K. (Ed.)
Robot Motion and Control 2007
452 p. 2007 [978-1-84628-973-6]

Vol. 359: Christophersen F.J.
Optimal Control of Constrained Piecewise Affine Systems
190 p. 2007 [978-3-540-72700-2]

Vol. 358: Findeisen R.; Allgöwer F.; Biegler L.T. (Eds.): Assessment and Future Directions of Nonlinear Model Predictive Control
642 p. 2007 [978-3-540-72698-2]

Vol. 357: Queinnec I.; Tarbouriech S.; Garcia G.; Niculescu S.-I. (Eds.):
Biology and Control Theory: Current Challenges
589 p. 2007 [978-3-540-71987-8]

Vol. 356: Karatkevich A.:
Dynamic Analysis of Petri Net-Based Discrete Systems
166 p. 2007 [978-3-540-71464-4]

Vol. 355: Zhang H.; Xie L.:
Control and Estimation of Systems with Input/Output Delays
213 p. 2007 [978-3-540-71118-6]

Vol. 354: Witczak M.:
Modelling and Estimation Strategies for Fault Diagnosis of Non-Linear Systems
215 p. 2007 [978-3-540-71114-8]

Vol. 353: Bonivento C.; Isidori A.; Marconi L.; Rossi C. (Eds.)
Advances in Control Theory and Applications
305 p. 2007 [978-3-540-70700-4]

Vol. 352: Chiasson, J.; Loiseau, J.J. (Eds.)
Applications of Time Delay Systems
358 p. 2007 [978-3-540-49555-0]

Vol. 351: Lin, C.; Wang, Q.-G.; Lee, T.H., He, Y.
LMI Approach to Analysis and Control of Takagi-Sugeno Fuzzy Systems with Time Delay
204 p. 2007 [978-3-540-49552-9]

Vol. 350: Bandyopadhyay, B.; Manjunath, T.C.; Umapathy, M.
Modeling, Control and Implementation of Smart Structures 250 p. 2007 [978-3-540-48393-9]

Vol. 349: Rogers, E.T.A.; Galkowski, K.; Owens, D.H.
Control Systems Theory and Applications for Linear Repetitive Processes
482 p. 2007 [978-3-540-42663-9]

Vol. 347: Assawinchaichote, W.; Nguang, K.S.; Shi P.
Fuzzy Control and Filter Design for Uncertain Fuzzy Systems
188 p. 2006 [978-3-540-37011-6]

Vol. 346: Tarbouriech, S.; Garcia, G.; Glattfelder, A.H. (Eds.)
Advanced Strategies in Control Systems with Input and Output Constraints
480 p. 2006 [978-3-540-37009-3]

Vol. 345: Huang, D.-S.; Li, K.; Irwin, G.W. (Eds.)
Intelligent Computing in Signal Processing and Pattern Recognition
1179 p. 2006 [978-3-540-37257-8]

Vol. 344: Huang, D.-S.; Li, K.; Irwin, G.W. (Eds.)
Intelligent Control and Automation
1121 p. 2006 [978-3-540-37255-4]

Vol. 341: Commault, C.; Marchand, N. (Eds.)
Positive Systems
448 p. 2006 [978-3-540-34771-2]

Vol. 340: Diehl, M.; Mombaur, K. (Eds.)
Fast Motions in Biomechanics and Robotics
500 p. 2006 [978-3-540-36118-3]

Vol. 339: Alamir, M.
Stabilization of Nonlinear Systems Using Receding-horizon Control Schemes
325 p. 2006 [978-1-84628-470-0]

Vol. 338: Tokarzewski, J.
Finite Zeros in Discrete Time Control Systems
325 p. 2006 [978-3-540-33464-4]

Vol. 337: Blom, H.; Lygeros, J. (Eds.)
Stochastic Hybrid Systems
395 p. 2006 [978-3-540-33466-8]

Vol. 336: Pettersen, K.Y.; Gravdahl, J.T.; Nijmeijer, H. (Eds.)
Group Coordination and Cooperative Control
310 p. 2006 [978-3-540-33468-2]