

國立交通大學

電控工程研究所

博士論文

網路控制系統之智慧型資料遺失補償器設計

The Dropout Compensation in Networked Control
Systems with the Intelligent Message Estimator

研究生：謝鎮洲

指導教授：徐保羅 博士

中華民國一百零二年一月

網路控制系統之智慧型資料遺失補償器設計

研究生：謝鎮洲

指導教授：徐保羅 博士

國立交通大學

電控工程研究所

摘要

網路化控制在工業上已經成為一種趨勢。然而，網路頻寬的限制導致時間的延遲和資料的遺失等，輕則明顯降低系統效能，重則使整個系統產生不穩定。本論文提出兩種即時網路控制系統(NCS)之資料遺失補償的方法來改善系統的效能。第一個方法是針對不同的資料遺失率做補償，提出泰勒訊息估測器；第二個方法是針對不同分佈的遺失資料，提出即時轉移機率估測器和智慧型訊息補償器。即時轉移機率估測器利用即時短時間內訊息遺失的機率快速地計算出轉移機率，並且成功且有效率的監控資料遺失的分佈情形。本論文分別從追蹤效能和穩定度上分析出不同階數的最小方差估測器可適用的轉移機率範圍，以即時估測出的轉移機率來切換不同的補償器，進而建構出智慧型訊息補償器(IME)來有效的補償不同分佈的資料遺失效應。但是當系統出網路時間延遲時，單獨IME是無法處理的，本論文整合了IME和perfect delay compensator (PDC)達成了同時處理資料遺失和網路時間延遲的效應。經模擬及實驗證明本論文所提出不需模式之方法，皆能有效且簡單應用在實際運動控制系統上，達成網路時間延遲及訊息遺失的補償效果，達成系統的精密度。

關鍵詞：網路化控制系統、資料遺失、訊息估測、轉移機率、時間延遲

The Dropout Compensation in Networked Control Systems with the Intelligent Message Estimator

Student : Chen-Chou Hsieh

Advisor : Dr. Pau-Lo Hsu

Institute of Electrical and Control Engineering
National Chiao Tung University

ABSTRACT

The time delay and the dropout in network control systems (NCS) are unavoidable mainly because of their limited transmission bandwidth. In this study, the Taylor estimator scheme is developed to effectively improve tracking accuracy of motion NCS under a higher data dropout rate in uniform distributions. Furthermore, as the data dropout in real applications is in stochastic nature with time-varying properties, the missing messages occur consecutively to cause the message compensator based on uniform distribution of data dropout to be invalid. A real-time transition probability estimator is thus proposed to monitor the data dropout distribution in real time. With the least square algorithm, estimators with real-time orders according to different network conditions are then studied. In this dissertation, the proposed model-free intelligent message estimator (IME) is developed to compensate for different data dropout distributions in a switching mechanism based on the estimated transition probability and stability analysis. However, when both the time delay and the data dropout are simultaneously induced in NCS, the IME only itself cannot provide appropriate compensation for their mutual effects. The integrated compensator combining both the IME and the perfect delay compensator (PDC), which is designed to compensate for the time delay, leads to satisfactory compensation results for both the time delay and the data dropout. Both simulation and experimental results indicate that the integration of the IME and the PDC compensator has effectively improved control performance even with varied

networked time delays and data dropout distributions.

Keywords: motion NCS, dropout, message estimation, transition probability, time delay



Acknowledgment

本論文得以順利完成，首先要感謝指導教授 徐保羅博士在課業與研究上孜孜不倦的教誨，以及為人處事和靈性的啟迪，使我獲益良多，在此由衷地表達我最誠摯的敬意與感謝；同時，還要感謝 王伯群教授的協同指導，於研究上適時的建議與鼓勵，其溫文儒雅的學者風範，更是我學習的標竿。此外，也要感謝口試委員：洪哲文教授、胡竹生教授、周至宏教授、蔡孟勳教授及蕭得聖教授等師長於百忙之中撥冗審閱，斧正本論文，使其更加周延與完整。

論文進行期間，還要感謝實驗室的學長、學弟們和建良兄，在生活及研究上的相互幫助及砥礪。在這段博士修業時光中，學業的壓力、研究的壓力和家庭的壓力，真的讓我很累很累，但是因為有保羅老師的鼓勵和教導，家人與老婆蓉蓉的支持，雖然過程跌跌撞撞，但是卻讓我的人生的經歷更加豐富，相信如果沒有經過這段時間的磨練，我也不會成長至此。由於曾經一起同甘共苦生活過的學長學弟們，人數眾多無法一一列舉，只能由衷地感激，感謝你們在這段期間帶給我最珍貴的回憶。此外，還要謝謝慧霖於事務上的協助。

最後，謹將此論文獻給我最敬愛的父親 謝文賜先生與母親 張彩絹女士及家人，感謝你們使我得以在無虞的環境中專心求學，同時還要感謝老婆蓉蓉及其家人的支持與關心。因為有了大家的支持與關懷，我才能夠無後顧之憂，完成學位。再次地感謝求學過程中所有曾經幫助過我和默默祝福我的師長及朋友們，謝謝您們。

Table of Contents

Abstract (Chinese).....	i
Abstract (English).....	ii
Acknowledgment	iv
Table of Contents	v
List of Figures	vii
List of Tables.....	xii
Chapter 1 Introduction	1
1.1 Network-induced delays	1
1.2 Packet dropout	3
1.3 Problem statement	5
1.4 Proposed approach.....	6
1.5 Contents overview	7
Chapter 2 The Taylor message estimator for motion NCS	8
2.1 Data dropout effects.....	9
2.2 The Taylor message estimator	13
2.2.1 The order of the estimator.....	13
2.2.2 Coefficients of Taylor message estimator	16
2.3 Simulation results	17
2.3.1 Noise command signals	17
2.3.2 Circular motion command	18
2.3.3 NURBS motion commands	19
2.4 Experimental results	21
2.4.1 Experimental setup with the CAN bus.....	21
2.4.2 CNC applications	22
2.4.3 Feedforward control on NCS	23
2.5 Summary.....	27
Chapter 3 The Real-time transition probability estimator.....	30
3.1 The Distribution effect of data dropout	30
3.2 Two-state Markov chain network model.....	31
3.3 The transition probability estimator	33
3.4 Illustrative example	34

3.5	Summary	37
Chapter 4	The Intelligent message estimator	38
4.1	The structure of the multi-axis motion NCS	38
4.2	The Least-square estimator	41
4.3	Analysis of LSE with different orders	43
4.4	The IME architecture	47
4.5	Simulation results	49
4.6	Experimental results	53
4.7	Summary	59
Chapter 5	Stability analysis for NCS with the message estimator	61
5.1	Modeling NCS dynamics as a switched system	63
5.2	NCS with one-delay compensator	64
5.3	NCS with the Least square estimator	65
5.4	SMS Stability analysis of NCS	68
5.4.1	Probability regions of NCS stability	69
5.4.2	Example	70
5.5	Discussion	73
5.6	Summary	76
Chapter 6	Integration of delay and dropout compensation for NCS	78
6.1	The perfect delay compensation	78
6.2	Case of invariant dropout distribution and the time delay	80
6.3	Case of varied dropout distribution and the time delay	87
6.4	Summary	95
Chapter 7	Conclusions and Future Work	97
7.1	Conclusions	97
7.2	Future work	98
	References	99
	Vita	108
	Publication List	109

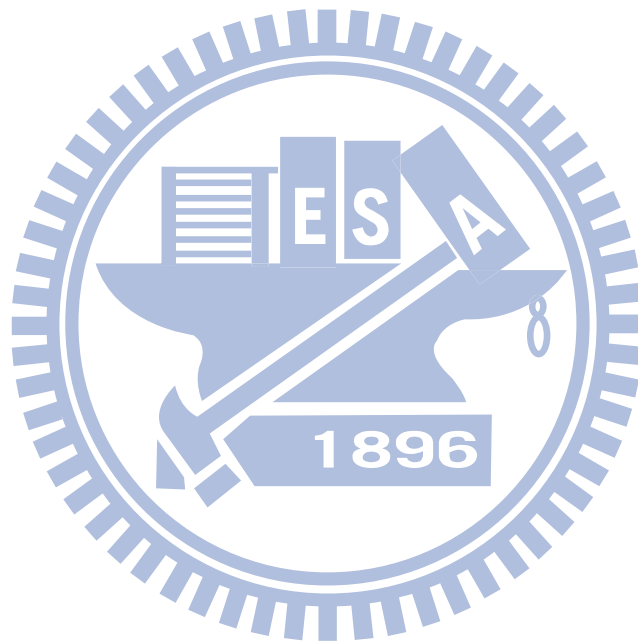
List of Figures

Fig. 2.1	Networked motion control systems.....	9
Fig. 2.2	Modeled NCS with data dropout.....	11
Fig. 2.3	Experimental result with data dropout rate = 19.97%.....	11
Fig. 2.4	NCS with the dropout compensator (Ling and Lemmon, 2004).....	12
Fig. 2.5	Equivalent LTI systems (Ling and Lemmon, 2004).....	13
Fig. 2.6	Motion commands with (a) smooth variation, (b) Significant variation.....	14
Fig. 2.7	Simulation result with the Taylor message estimator.....	15
Fig. 2.8	Analysis of compensation effects with different orders.....	16
Fig. 2.9	Output PSD with white noise input.....	18
Fig. 2.10	Tracking accuracy of different message estimators.....	19
Fig. 2.11	NURBS simulation result as $\varepsilon = 19.97\%$	20
Fig. 2.12	NURBS simulation result as $\varepsilon = 19.97\%$	20
Fig. 2.13	The transmission error without message estimator.....	21
Fig. 2.14	The transmission error with the Taylor message estimator.....	22
Fig. 2.15	Experimental setup.....	22
Fig. 2.16	Command message.....	23
Fig. 2.17	Experimental results with/without the message estimator.....	23
Fig. 2.18	The motion NCS with the message estimator and the feedforward controller.....	24
Fig. 2.19	The basic structure of ZPETC.....	24
Fig. 2.20	Tracking errors of ZPETC and 1-delay estimator with dropout rate (a) 0.49% , (b) 19.97% , (c) 42.14%	26
Fig. 2.21	Tracking errors of ZPETC and the optimal dropout compensator with dropout rate (a) 0.49% , (b) 19.97% , (c) 42.14%	26
Fig. 2.22	Tracking errors of ZPETC and the 3rd –order Taylor estimator with dropout rate (a) 0.49% , (b) 19.97% , (c) 42.14%	27

Fig. 2.23	Experimental result with the ZPETC and the Taylor message estimator	27
Fig. 3.1	(a) Distributed and (b) centralized dropout signals with the same dropout rate 20%	30
Fig. 3.2	NURBS position command.....	31
Fig. 3.3	Y-axis racking errors of (a) distributed and (b) centralized missing data (20% dropout rate)	31
Fig. 3.4	Two-state Markov chain network model.....	32
Fig. 3.5	Real-time calculation of local transition probability with the same overall transition probability $\rho_{D,D} = 0.5$	35
Fig. 3.6	Transition probabilities of (a) average of $\hat{\rho}_{D,D}(k)$ vs $\rho_{D,D}$ and (b) average of $\hat{\rho}_{R,R}(k)$ vs $\rho_{R,R}$	36
Fig. 3.7	Measuring time of the real-time transition probability estimator.....	37
Fig. 4.1	(a) General NCS architecture and (b) practical motion NCS architecture	39
Fig. 4.2	Multi-axis motion NCS.....	41
Fig. 4.3	Timeline analysis in a control period.....	41
Fig. 4.4	Transmission errors with (a) LSE(5,3) and (b) LSE(2,1) as $P_{D,D} = 0.2$	45
Fig. 4.5	Transmission errors with (a) LSE(5,3) and (b) LSE(2,1) with $\rho_{D,D} = 0.4$	46
Fig. 4.6	Analysis of compensation effects with different true transition probability $P_{D,D}$	46
Fig. 4.7	Simulation results with different estimators.....	49
Fig. 4.8	The system architecture of the proposed IME.....	49
Fig. 4.9	Contours of motion NCS without/with the Taylor estimator ($P_{D,D} = 0.2$).....	50
Fig. 4.10	Contouring errors with (a) the Taylor estimator (b) IME ($P_{D,D} = 0.2$).....	51
Fig. 4.11	Contouring errors with (a) the Taylor estimator and (b) IME ($P_{D,D} = 0.5$).....	52
Fig. 4.12	Contouring errors with (a) the Taylor estimator and (b) IME ($P_{D,D} = 0.6$)	53
Fig. 4.13	Experimental setup.....	54

Fig. 4.14 Contouring errors with the sampling period = 0.5 ms	
(a) Taylor estimator (b) IME ($P_{D,D} = 0.32$).....	55
Fig. 4.15 First-order differential of contouring errors of	
(a) the Taylor estimator (b) IME ($P_{D,D} = 0.32$).....	56
Fig. 4.16 Contouring errors with the sampling period = 0.4 ms	
(a) Taylor estimator (b) IME ($P_{D,D} = 0.54$).....	57
Fig. 4.17 Circle responses of (a) the Taylor estimator and (b) IME	
with the sampling period = 0.5 ms ($P_{D,D} = 0.32$).....	58
Fig. 4.18 Circle responses of (a) Taylor estimator and (b) IME	
with the sampling period = 0.4 ms ($P_{D,D} = 0.54$).....	59
Fig. 5.1 Structure of the motion network control systems.....	61
Fig. 5.2 Timing diagram of the network and system states.....	63
Fig. 5.3 Theoretical SMS stability region for one-delay compensator.....	71
Fig. 5.4 Theoretical SMS stability region for LSE(2,1).....	72
Fig. 5.5 Theoretical SMS stability region for LSE(3,2).....	72
Fig. 5.6 Theoretical SMS stability region for LSE(3,2).....	73
Fig. 5.7 Theoretical SMS stability regions for different LSE.....	73
Fig. 5.8 Theoretical SMS stability regions for different LSE	
(a) the rotating base inverted pendulum, (b) the CNC machine	75
Fig. 5.9 Tracking errors by applying (a) Taylor estimator (b) IME.....	76
Fig. 6.1 System states of real-time motion NCS vs networked time delay	79
Fig. 6.2 The control structure with PDC in the proposed NCS	79
Fig. 6.3 The tracking error with without delay compensation as $\tau_d = 10$ ms	80
Fig. 6.4 The tracking error without compensation as both dropout and delay induced	
($\rho_{D,R} = 0.8, \tau_d = 10$ ms).....	81

Fig. 6.5 The tracking error by applying IME with $\rho_{D,R} = 0.8$ and $\tau_d = 10\text{ ms}$	81
Fig. 6.6 The tracking error by applying PDC with $\rho_{D,R} = 0.8$ and $\tau_d = 10\text{ ms}$	82
Fig. 6.7 Applying both IME and PDC with $\rho_{D,R} = 0.8$ and $\tau_d = 10\text{ ms}$	
(a) the system response (b) the tracking error	83
Fig. 6.8 The tracking error for NCS without compensation as $\tau_d = 50\text{ ms}$	84
Fig. 6.9 The tracking error without compensation as both dropout and delay induced	
($\rho_{D,R} = 0.8, \tau_d = 50\text{ ms}$)	84
Fig. 6.10 The tracking error by applying IME only ($\rho_{D,R} = 0.8, \tau_d = 50\text{ ms}$)	85
Fig. 6.11 The tracking error by applying PDC only ($\rho_{D,R} = 0.8, \tau_d = 50\text{ ms}$)	85
Fig. 6.12 Applying both IME and PDC with $\rho_{D,R} = 0.8$ and $\tau_d = 50\text{ ms}$	
(a) the system response (b) the tracking error	86
Fig. 6.13 The system architecture of NCS with integration of PDC and IME.	87
Fig. 6.14 (a) The dropout signal with different distributions (b) the varied networked time	
delay.	88
Fig. 6.15 The tracking error of the Ideal Case without time delay and data dropout	89
Fig. 6.16 Tracking errors with different distributions (a) without compensation (b) applying	
IME	90
Fig. 6.17 Tracking errors with varied network delay (a) without compensation (b) applying	
PDC	91
Fig. 6.18 Tracking errors with different dropout distributions and $\tau_d = 50\text{ ms}$ by	
(a) applying IME (b) applying both IME and PDC	92
Fig. 6.19 Tracking errors with varied network time delay and $\rho_{D,R} = 0.8$ by	
(a) applying PDC (b) applying both IME and PDC	93
Fig. 6.20 The tracking error with varied network time delay and different dropout	



List of Tables

Table 1.1 Classification of time delay control design in NCS (Lai et al., 2011)	2
Table 1.2 Classification of dropout compensation in NCS.....	5
Table 2.1 The data dropout rate of CAN bus transmission rate.....	9
Table 5.1 Evaluation for SMS stability regions of different LSE.....	75
Table 6.1 Evaluation for compensated effects of PDC and IME.....	95



Chapter 1

Introduction

Because techniques of network communication have been rapidly developed in modern industries, there is a trend to integrate network protocol into traditional control systems as the networked control system (NCS) (Zhang et al., 2001; Walsh et al., 1999). Due to the scalability, flexibility, cost effectiveness, and easy maintenance of the NCS technology, many industrial applications such as factory automation, remote diagnostics and troubleshooting, remote mobile robots, aircraft, automobiles, nursing homes, and tele-operations become popular research topics. While the control technology over data communication networks is attractive, there still have challenges in design, analysis, and implementation of practical NCSs due to the characteristics of asynchronous communications and limited reliability of the NCS networks (Lian et al., 2002). These challenges are particularly evident when network infrastructures are employed in the networked control applications (Meyer and Conant, 2005; Schenato, 2006; Akyildiz et al., 2002). The first challenging problem is the time-varying network-induced delay between the controller and other devices such as sensors and actuators. This time-varying delay will affect the accuracy of timing dependent computations in the control system. The second challenging problem is the possible data packet dropout resulting from network traffic congestions and limited network reliability. In this case, the controller and/or actuator have to make decisions with incomplete information on how to control the system. Both challenging problems, i.e., the network induced delay and the packet dropout, can significantly degrade control performance of the overall real-time NCS. For the problem of the packet dropout, this paper will develop compensation strategies for packet dropout compensation in real-time NCS.

1.1 Network-induced delays

During the past decades, special issues in IEEE Transactions highlighted the significance of the development of NCS technologies (Antsakls and Baillieul, 2004; Bushnell, 2001). Among those technologies, NCS stability-based methods have been rapidly developed (Peng and Tian, 2006; Peng and Tian, 2007; Peng et al., 2007; Yue

et al., 2004; Zhang et al., 2001). These methods have focused on NCS stability analysis and controller design mainly with considerations of upper and lower bounds of the network-induced delay. Their main goal is to find the stability conditions for specific plants and then design the controller to meet the conditions. Most available approaches are model-based designs to alleviate the network time delay effect as categorized in Table 1.1. These NCS control design methods have been proposed mainly to eliminate the time-delay effect from its closed control loop, and their design procedures are usually based on the plant model with known or bounded delay information. In real networks usually with varied time delays, those developed model-based methodologies are usually not applicable. Thus, the model-free methodology is desired for real concern in NCS implementation. The model-free perfect delay compensation (PDC) is proposed for NCS (Lai et al., 2011), and the network-induced delay in NCS can thus become a simple pure delay added to the original loop. However, the stability of PDC is not yet discussed.

Table 1.1 Classification of time delay control design in NCS (Lai et al., 2011)

	Time Delay	Constraint	Methods
Model-based NCS design	Known delay	Constant delay time	1. Smith predictor (Peng et al., 2004)
		Time-varying delay < sampling time	1. State feedback controller (Tang et al., 2008)
			2. H_∞ robust controller (Gao and Chen, 2008)
	Bounded time-varying delay > sampling time	1. Model predictor control (Zhao et al., 2009)	
		2. Gain scheduler middleware (Tipsuwan and Chow, 2004)	
		3. Switched system approach (Xie et al., 2008)	
Unknown delay	Constant	4. Fuzzy controller (Lee et al., 2003)	
		5. Optimal controller (Li et al., 2009)	
	Time-varying delay	1. CDOB (Natori and Ohnishi, 2008)	
		2. Scattering transformation (Matiakis et al., 2009)	
		1. CDOB (Natori et al., 2008)	

1.2 Packet dropout

The packet dropout mainly occurs due to the packet collision. Moreover, for hard real-time network control systems, the packet dropout is also defined when the network-induced time is more than the system sampling period. Therefore, packet dropout which is unavoidable in high-speed-high-precision NCS is another important issue in motion NCS design. The packet dropout has been discussed mainly in stability analysis.. Therefore, conservative control design is usually adopted in order to guarantee the NCS stability. Among various NCS control methodologies, the queuing methodology was proposed to develop NCSs with deterministic communication behavior (Chan and Özgüner, 1995). They tried to compensate for packet dropout through sophisticated computation by the controller. However, their algorithms may not be feasible for implementation because of the requirement of accurate process models.

Addressing the simultaneous compensation of network induced delay and packet dropout, Soglo and Yang (2006) designed an agent-based networked control estimator for the controller to improve performance of NCSs. They modeled the NCS as an asynchronous dynamical system (ADS) with a rate constraint, and then used the bilinear matrix inequality method to solve the compensation problem. However, the assumption that the network-induced delays were less than one sampling period also limits their work in real applications and the network time delay is usually much longer compared with the sampling time.

Schenato studied optimal state estimation in NCS with random delay and packet loss for the NCS control design (Schenato, 2006). The mathematical analysis in his work was elegant and however, his work was strongly relied on an accurate process model with intensive computation of which is difficult in practical NCS.

Ling and Lemmon (2002, 2003, 2004) considered an optimal dropout compensation based on the power spectral density of NCS output signals. Although the optimal dropout compensator results in effective noise reduction to suppress the noise contamination, its tracking accuracy in control performance in real-time NCS is not concerned in their work.

Recently, Tian and Levy (2008) proposed the simple and model-free strategies to compensate for control packet dropout at actuators. The packet dropout compensator based on past control signals is similar to dynamic voltage scheduling from past voltage settings (Varma et al., 2003). Their work provided useful packet dropout compensation strategies for real-time NCS. However, their work was limited in uniform distribution of packet dropouts which are too simple to model the distribution of packet dropouts.

By applying the two-state Markov chain network model, various distributions of packet dropout can be effectively distinguished (Kawka and Alleyne, 2009), and networked control has emerged as a new example of a system class with abrupt changes in dynamics that can be suitably modeled as a Markovian jump linear system (MJLS).

Initial research focused on stability and performance analysis of MJLSs with certain Markov transition probabilities with exact model parameters (Chizeck et al., 1986; Oliveira et al., 2002). Later efforts have focused on the control of MJLSs with uncertainty in the model parameters (Costa et al., 1999) or the transition probability (Costa and Marques, 2000). In recent work, the linear matrix inequality (LMI) methods utilized extended parameters to reduce conservativeness due to plant and Markov transition probabilities uncertainties (Park et al., 2001; Park and Kwon, 2002; Val et al., 2002; Souza, 2006). These available approaches for alleviating the data dropout effect are categorized in Table 1.2. However, these works are usually more complicated and stabilization is mainly concerned in real-time NCS. In fact, Markov transition probabilities are in stochastic and time-varying natures which are difficult to accurately estimate. Therefore, on-line estimation of transition probability becomes crucial for real-time NCS applications, particularly for motion NCS to achieve high-speed-high-precision manufacturing processes.

Table 1.2 Classification of dropout compensation in NCS

	Data Dropout	Constraint	Methods
Dropout compensator design	Uniform distribution	constant dropout rate	1. Optimal dropout compensator (Ling and Lemmon, 2004) 2. Agent-based estimator (Soglo and Yang, 2006) 3. Optimal state estimation (Schenato,2006)
		time-varying dropout rate	1. Packet dropout compensator (Varma et al., 2003) 2. Proportional derivative predictor (Tian and Levy, 2008)
	Varied distribution	certain transition probability	1. Optimal control (Chizeck et al., 1986) 2. Quadratic state feedback control (Costa et al., 1999) 3. H_2 and H_∞ control (Oliveira et al., 2002)
		time-varying transition probability	1. Linear matrix inequality method (Park and Kwon, 2002) (Val et al., 2002) (Souza, 2006) 2. Optimal dropout compensator (Kawka and Alleyne, 2009)

1.3 Problem statement

Although many methods have been proposed to design controllers for NCS by concerning the data dropout effect in the past two decades, some critical problems are still to be solved as follows:

- (1) In motion NCS, many dropout compensators successfully compensate the data dropout effect as the data dropout rate is relatively low, such as the one-delay estimator (Ling and Lemmon, 2002) and the optimal compensator(Ling and Lemmon, 2002; Ling and Lemmon, 2004). However, when the data dropout rate increases, those available message estimators become invalid and a more effective dropout compensator for a higher dropout rate is still desired designed.
- (2) When data dropouts consecutively occur as the communication becomes worse, the compensators based on the assumption of uniform distribution of the data

dropout are not suitable for real applications. The most fundamental issue for NCS under the condition of dropout distribution in stochastic nature is to obtain an efficient estimation of the data dropout distribution. The processing time and algorithms of existent dropout distribution estimators are still time consuming and a real-time dropout distribution estimator is thus desired.

- (3) Although some sophisticated dropout compensation approaches were proposed, their processing time and modeling are very complicated to be realized. Under various different level dropout distributions, an intelligent and effective dropout compensator without complicated implementation is required for NCS.
- (4) To achieve stability analysis for motion NCS including the dropout compensator is a challenging task, because the dropout occurrence in a network infrastructure is in a random nature. A systematic analysis approach to guarantee stability of motion NCS is desired.
- (5) In addition to the dropout effect, the network-induced time delay is another crucial issue in NCS design. Since both the network introduced time delay and the data dropout are unavoidable for motion NCS, the developed message estimator should concern both the delay and dropout effects simultaneously in a feasible motion NCS.

1.4 Proposed approach

In this dissertation, five major approaches corresponding to the mentioned problems for motion NCS design are proposed as follows:

- (1) The Taylor estimator scheme is proposed to compensate for the data dropout. It maintains system dynamics for the motion NCS even under a higher data dropout rate with uniform distribution.
- (2) The real-time transition probability estimator is proposed to estimate the dropout distribution between two nodes in real network environments. Furthermore, the network loads and conditions can be monitored by applying the estimator in real time.
- (3) The intelligent message estimator (IME) scheme with the switching

least-square estimator is proposed based on the estimated data dropout condition under different levels of dropout distributions.

(4) By modeling the NCS with a switching system dynamics and applying the second moment stability (SMS) condition, stability regions of motion NCS with various dropout compensator can be determined to guarantee the stability of NCS with the switching law for the IME.

(5) By applying the integration of PDC and IME, the network-induced time delay in different levels, and the unavoidable data dropout under different distributions can be simultaneously compensated to restore the control design without being affected seriously by the network in NCS realization.

1.5 Contents overview

This dissertation is organized as follows: the Taylor estimator for different data dropout rates is presented in Chapter 2. Chapter 3 introduces the real-time transition probability estimator and Chapter 4 presents the model-free intelligent message estimator (IME) based on the real-time estimated transition probability. In Chapter 5, a switching system dynamics modeling and its stability analysis are introduced for motion NCS. In Chapter 6, the PDC compensation for the delay in NCS is introduced in motion NCS and its integration with the IME together solve the two major problems in NCS. Finally, conclusions and recommendations for further research are provided in Chapter 7.

Chapter 2

The Taylor message estimator for motion NCS

Although NCS possesses some advantages such as low cost, extensibility, flexibility, and easy maintenance, the unavoidable time-delay effect induced in the network seriously degrades its control performance and also reduces system reliability and stability. Recently, different approaches were proposed for NCS to mainly compensate for the time-delay effect like the queuing methodology (Luck and Ray, 1994), the sampling time scheduling (Kim et al., 1996), the gain scheduling PI control (Tipsuwan and Chow, 2004), scheduling and control co-design (Lu et al., 2004), and robust control (Chen et al., 2006; Jiang and Han, 2006).

A general network system is basically an event-triggered system and the time delay is mainly concerned. However, the real-time motion NCS, which includes the fixed sampling time and an interpolator to conduct provided motion contouring commands, is a typical event/time-triggered system (Hsieh and Hsu, 2005). When the network communication becomes heavy, some network nodes may not properly receive/transmit messages on time and the data dropout may thus occur. In general, the dropout rate of the network is closely related to both the network transmission rate and the specified sampling period. The data dropout not only increases system uncertainty of the NCS, but also it degrades motion accuracy in tracking and contouring (Hsieh et al., 2006). A Markov chain with two states treated as the vacant sampling can be applied to model the data dropout in a stochastic nature (Nilsson, 1998). Moreover, to handle the data dropout, there are two approaches: (1) using the past control signals to estimate the lost data (Ling and Lemmon, 2002) and (2) including the estimator which based on the power spectral density of NCS output signals (Ling and Lemmon, 2002; Ling and Lemmon, 2004).

A message estimator is proposed for the motion NCS to compensate for its dropout effect. Both simulation and experimental results have shown that a message estimator with a 3rd-order Taylor expansion is effective for estimating missing motion signals. Moreover, the motion control performance on the NCS is satisfactory by including the estimator in the controller. Since the proposed structure leads to a more reliable motion NCS with less uncertainty, it has been successfully integrated with the

feedforward control design to achieve high-precision motion accuracy (Tomizuka, 1978; Yeh and Hsu, 1999). The proposed network control structure has been successfully realized on a DYNA MTYE 1007 CNC machine tool to prove the feasibility of the present motion NCS.

2.1 Data dropout effects

Motion systems with synchronized control on multiple axes are designed mainly to meet specifications of precision accuracy in tracking or contouring. When motion control systems are realized on the NCS, the data bus containing either the command messages or the feedback measurements are transmitted through the network protocol, as shown in Fig. 2.1. The induced time delay in the NCS is unavoidable and the transmitted message may miss the hard real-time deadline, the sampling time T , and it always leads to erroneous motions in precise systems. Thus, the caused data dropout is crucial to motion performance in the real-time NCS. For the controller area network (CAN) bus, Table 2.1 shows all experimental measurements of the dropout rate with different transmission rates and sampling periods. Experimental results indicate that the dropout rate significantly decreases as the sampling period increases. Note that the control performance of the system also decreases as the sampling period increases in NCS (Lian et al., 2002). Therefore, to select a proper sampling time for the NCS, it is a trade-off by concerning between the network transmission and the control performance.

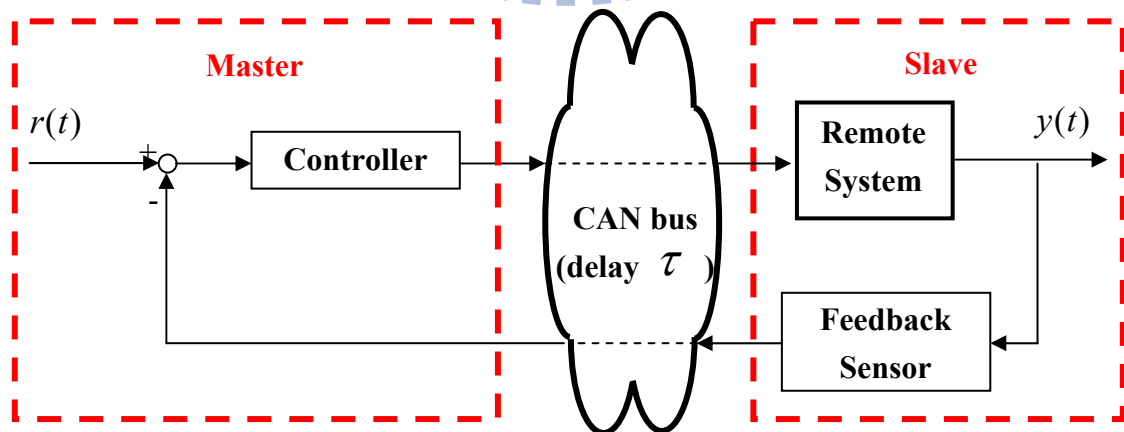


Fig. 2.1 Networked motion control systems

Table 2.1 The data dropout rate of CAN bus transmission rate

Transmission rate	Dropout rate (ε) $T = 2 \text{ ms}$	Dropout rate (ε) $T = 1 \text{ ms}$
1M bit/s	0.48%	0.49%
500 bit/s	0.51%	19.97%
250 bit/s	20.21%	42.14%

The data dropout occurs randomly on the network transmission either in command or feedback measurement signals. Actually, the dropout commands are properly estimated since most commands are relatively smooth compared with the measurements (Hsieh and Hsu, 2005; Hsieh et al., 2006). Therefore, this paper focuses on compensating the effect of the measurement data dropout. To model the data dropout in transmitting the feedback message data, Fig. 2.2 shows that d is a binary process with probability distribution of $P(d[k]=1) = \varepsilon$, $P(d[k]=0) = 1 - \varepsilon$, and the data dropout occurs when $d[n]=1$ [11-12]. The transmitted feedback signal $\bar{y}[n]$ is modeled as

$$\begin{cases} \bar{y}[k] = y[k], & \text{if } d[k] = 0 \\ \bar{y}[k] = 0, & \text{if } d[k] = 1 \end{cases} \quad \Leftarrow \text{dropout} \quad (2.1)$$

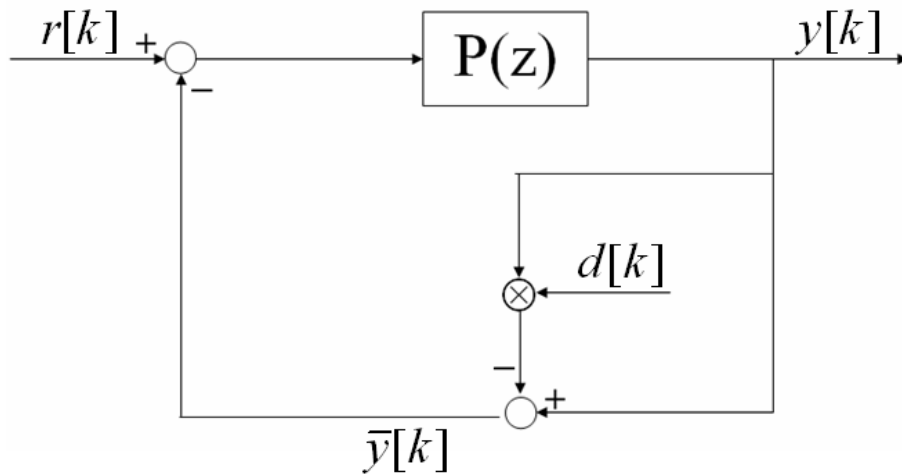


Fig. 2.2 Modeled NCS with data dropout.

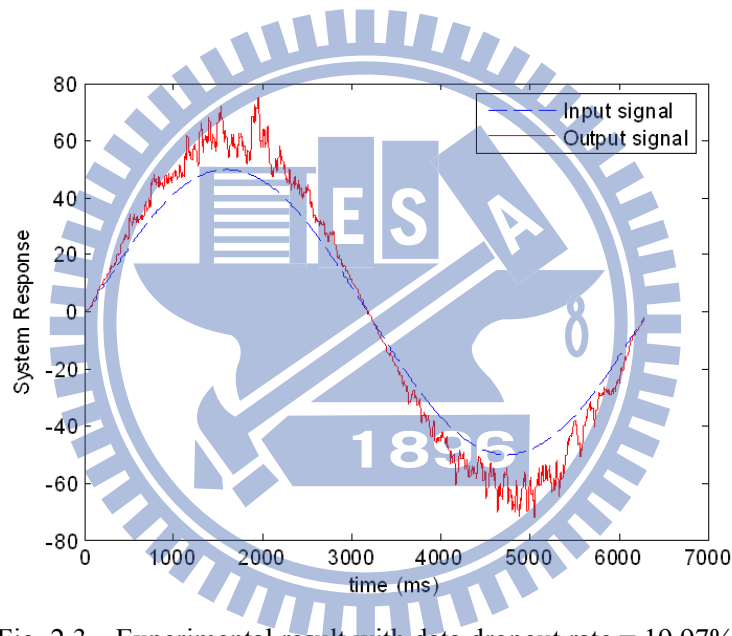


Fig. 2.3 Experimental result with data dropout rate = 19.97%

Experimental results with an 1 *ms* sampling period and a 500K *bit/s* transmission rate are shown in Fig. 2.3. Results indicate that the data dropout occurred in the feedback data directly affects the system performance. In the present experiments, the missing feedback messages are all treated as 0 values and it makes the designed controller to loss efficacy. To compensate for the dropout data, the designed message estimator $F(z)$ is shown in Fig. 2.4 and the NCS can be expressed as in the following:

$$\begin{cases} \bar{y}[k] = y[k], & \text{if } d[k] = 0 \\ \bar{y}[k] = \hat{y}[k], & \text{if } d[k] = 1 \end{cases} \Leftarrow \text{compensated dropout} \quad (2.2)$$

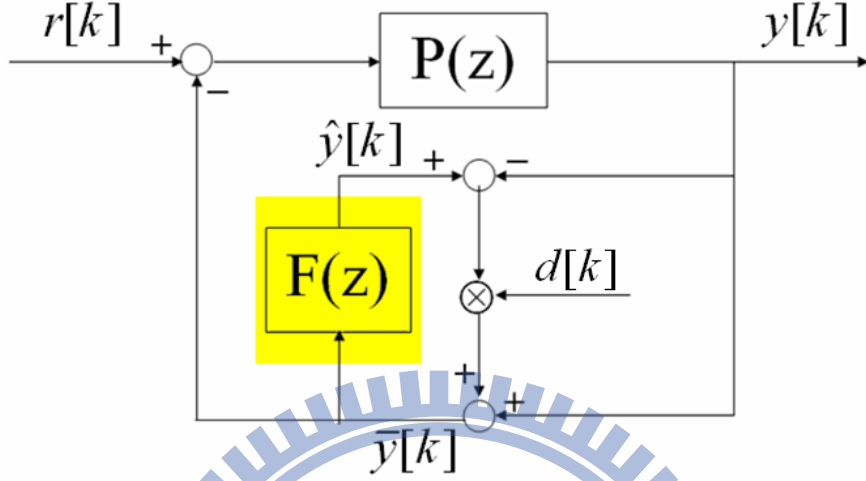


Fig. 2.4 NCS with the dropout compensator (Ling and Lemmon, 2004)

The power spectral density of the system output response are as in the following (Ling and Lemmon, 2004):

$$S_{yy}(z) = \left| \frac{P(z)}{1 - D(z)P(z)} \right|^2 S_{ww}(z) + \left| \frac{D(z)P(z)}{1 - D(z)P(z)} \right|^2 \frac{1}{1 - \varepsilon} \Delta \quad (2.3)$$

$$S_{\bar{y}\bar{y}}(z) = \left| \frac{P(z)(D(z) - 1)}{1 - D(z)P(z)} \right|^2 S_{ww}(z) + \left| \frac{D(z)(1 - P(z))}{1 - D(z)P(z)} \right|^2 \frac{1}{1 - \varepsilon} \Delta \quad (2.4)$$

where $|\cdot|$ means magnitude and $D(z) = \frac{1 - \varepsilon}{1 - \varepsilon \cdot F(z)}$. For $\varepsilon = 0$, $\Delta = 0$; for $\varepsilon > 0$, Δ is the unique positive solution to the following equation

$$\begin{aligned} \Delta = & \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \frac{P(e^{jw})D(e^{jw}) - 1}{1 - D(e^{jw})P(e^{jw})} \right|^2 S_{ww}(e^{jw}) dw \\ & + \frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \frac{D(e^{jw})(1 - P(e^{jw}))}{1 - D(e^{jw})P(e^{jw})} \right|^2 dw \cdot \frac{1}{1 - \varepsilon} \cdot \Delta \end{aligned} \quad (2.5)$$

where w is the frequency. Moreover, the networked control system shown in Fig. 2.4 can be transformed to an LTI system as shown in Fig. 2.5. The optimal dropout compensator $F(z)$ can be thus designed by minimizing the power spectral density of the output response $y[k]$ under the noise contamination $n[k]$ (Ling and Lemmon, 2002; Ling and Lemmon, 2004).

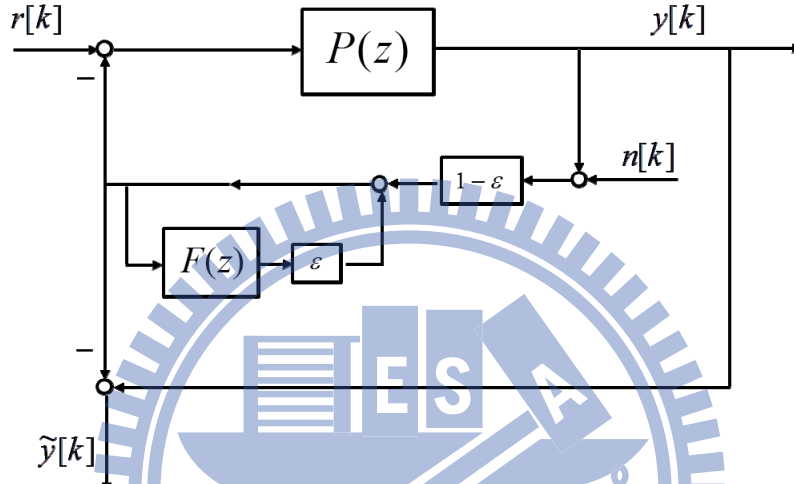


Fig. 2.5 Equivalent LTI systems (Ling and Lemmon, 2004)

2.2 The Taylor message estimator

Based on the structure shown in Fig. 2.4, the output of a message estimator will estimate the missing message when the dropout happens as in Eq. (2.2). The missing messages can be thus recovered to some extents to improve performance of the motion NCS. For the messages in a relatively low frequency, the improvement of control performance with a simple 1-delay message estimator is acceptable (Ling and Lemmon, 2002). However, as the frequency of the transmitted/received signals increases, the motion NCS owns faster dynamics and the improvement of the 1-delay message estimator is thus limited.

2.2.1 The order of the estimator

In the present paper, a Taylor message estimator is proposed for the motion NCS, because most dynamics of motion commands or motion measurements can be represented by a Taylor expansion with a suitable order except the motion commands

containing significant variation, as shown in Fig. 2.6 with different dynamic natures. Fig. 2.7 shows that the transmission error decreases when the order of the Taylor estimator increases for smooth commands. However, it also shows that the transmission error increases when the order of the Taylor estimator increases for commands with significant variation. Therefore, the selection of orders of the Taylor estimator is very important in motion NCS. So this paper used the integrated absolute errors (IAE) of transmission errors as a performance index to determine the orders of the Taylor estimator. Results shown in Fig. 2.8 indicate that the 3rd-order Taylor message estimator is more suitable in real applications by concerning different motion commands.

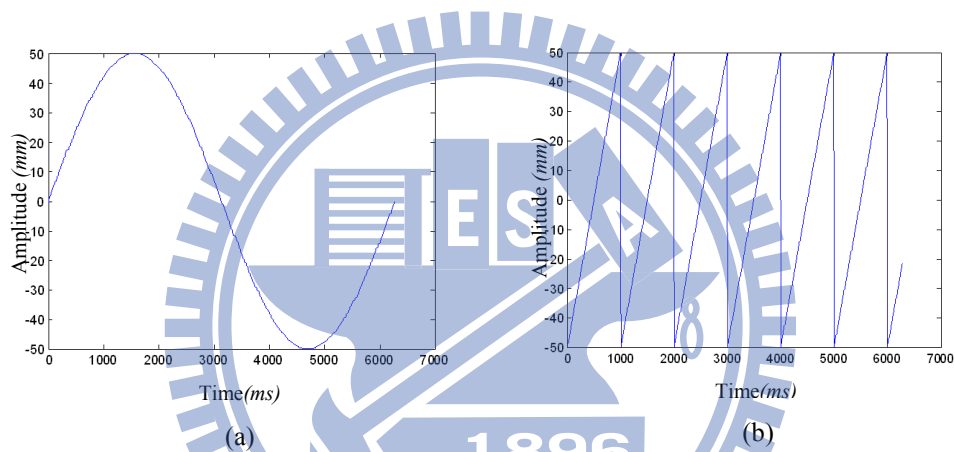
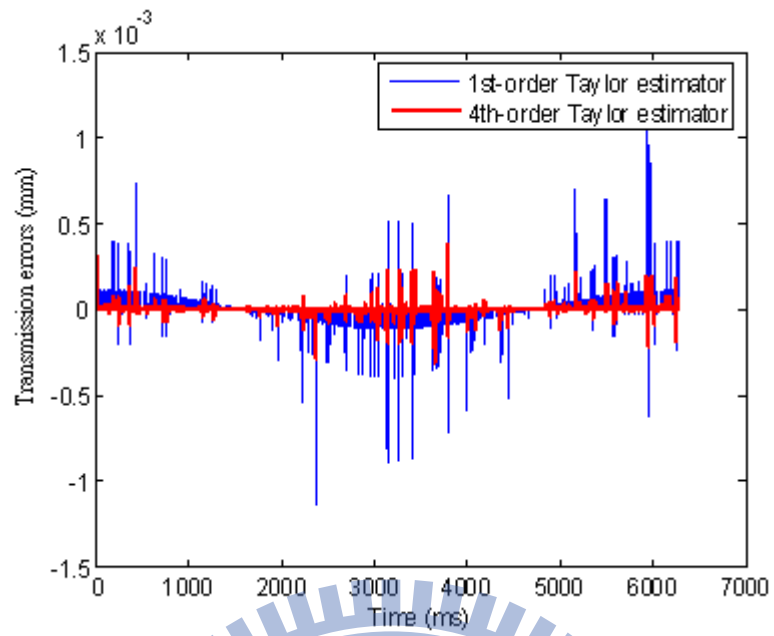
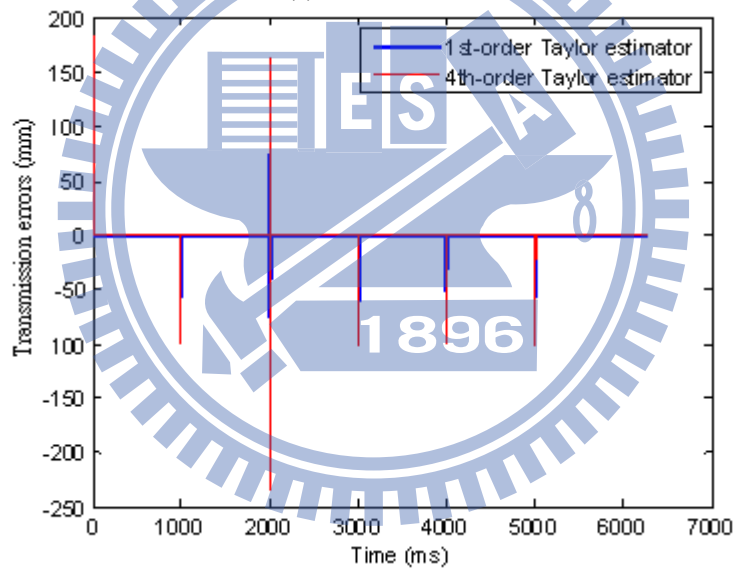


Fig. 2.6 Motion commands with (a) smooth variation, (b) Significant variation



(a) smooth variation



(b) significant variation

Fig. 2.7 Simulation result with the Taylor message estimator

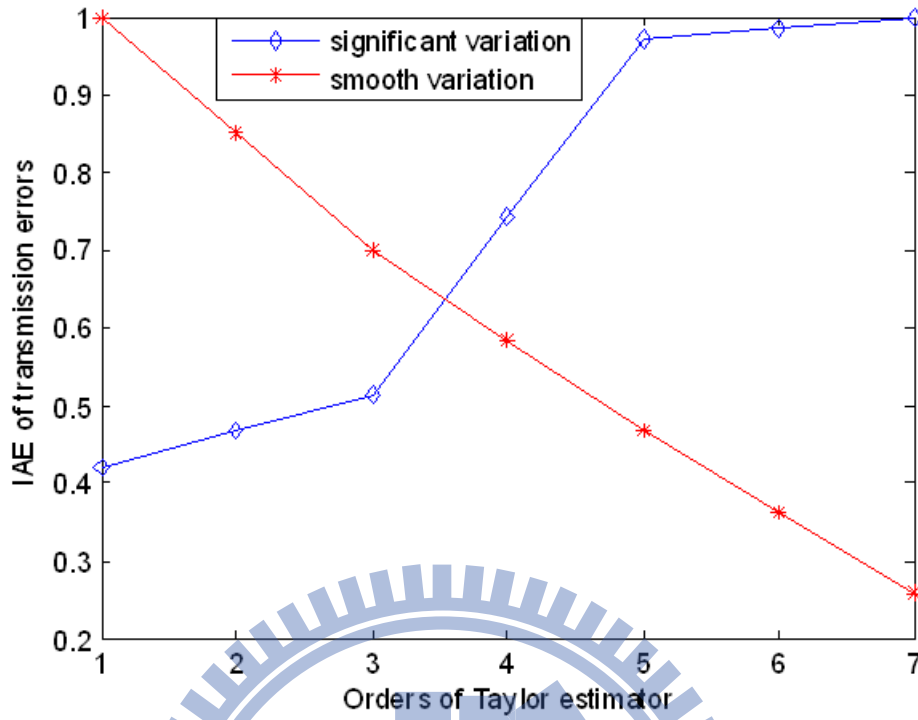


Fig. 2.8 Analysis of compensation effects with different orders

2.2.2 Coefficients of Taylor message estimator

If the current k_{th} position data $P(k)$ is lost, the Taylor expansion is processed to estimate the velocity, \hat{v}_{k-1} , from the past data

$$\hat{v}_{k-1} = \Delta P_{k-1} + \frac{1}{2} \cdot (\Delta P_{k-1} - \Delta P_{k-2}) + \frac{1}{8} \cdot (\Delta P_{k-1} - 2\Delta P_{k-2} + \Delta P_{k-3}) + \dots \quad (2.6)$$

where

$$\Delta P_{k-1} = P(k-1) - P(k-2), \quad \Delta P_{k-2} = P(k-2) - P(k-3), \quad \Delta P_{k-3} = P(k-3) - P(k-4)$$

The estimated value of the current position command can be expressed as $\hat{P}(k) = P(k-1) + \hat{v}_{k-1}$. Therefore, the estimated current result from the past messages is obtained. The different order Taylor message estimators $F(z)$ can be simply expressed in the z-transform as

- 1st-order Taylor estimator

$$F(z) = 2z^{-1} - z^{-2} \quad (2.7)$$

- 2nd-message Taylor estimator

$$F(z) = \frac{3}{2}z^{-1} - 2z^{-2} + \frac{1}{2}z^{-3} \quad (2.8)$$

- 3rd-order Taylor estimator

$$F(z) = \frac{21}{8}z^{-1} - \frac{19}{8}z^{-2} + \frac{7}{8}z^{-3} - \frac{1}{8}z^{-4} \quad (2.9)$$

2.3 Simulation results

2.3.1 Noise command signals

In the simulation analysis, the NCS structure shown in Fig. 2.4 was built on Matlab. The dynamic model of the DYNA CNC machine tool obtained from the system identification procedure was adopted as

$$P(z) = \frac{0.30554z^{-2} - 0.023766z^{-3} + 0.11104z^{-4} + 0.028834z^{-5} - 0.012243z^{-6} + 0.020811z^{-7} - 0.089113z^{-8}}{1 - 0.70669z^{-1} + 0.1934z^{-2} - 0.15112z^{-3} - 0.02566z^{-4} + 0.028011z^{-5}}$$

Moreover, three different message estimators were implemented for verifying the noise reduction and control performance as: (1) the 1-delay estimator, $F(z) = z^{-1}$, (2) the optimal estimator (Ling and Lemmon, 2004), and (3) the proposed 3rd-Taylor estimator. The dropout rate is chosen as $\varepsilon \in [0, 0.6]$. For the cases where the input command $r[k]$ is the white noise with zero mean, Fig. 2.9 indicate that based on the index of the auto-correlation value R_{yy} , the optimal dropout compensator results in the best noise reduction to suppress the noise contamination effect up to a 20% data dropout rate. On the other hand, the 3rd-order Taylor message estimator performs the worst for noise reduction. Note that the Taylor message estimator mainly estimates the missing message from the past data but the noise signals are unpredictable. Therefore, the obtained results also imply that the Taylor message estimator is not suitable for the highly noise-contaminated NCS.

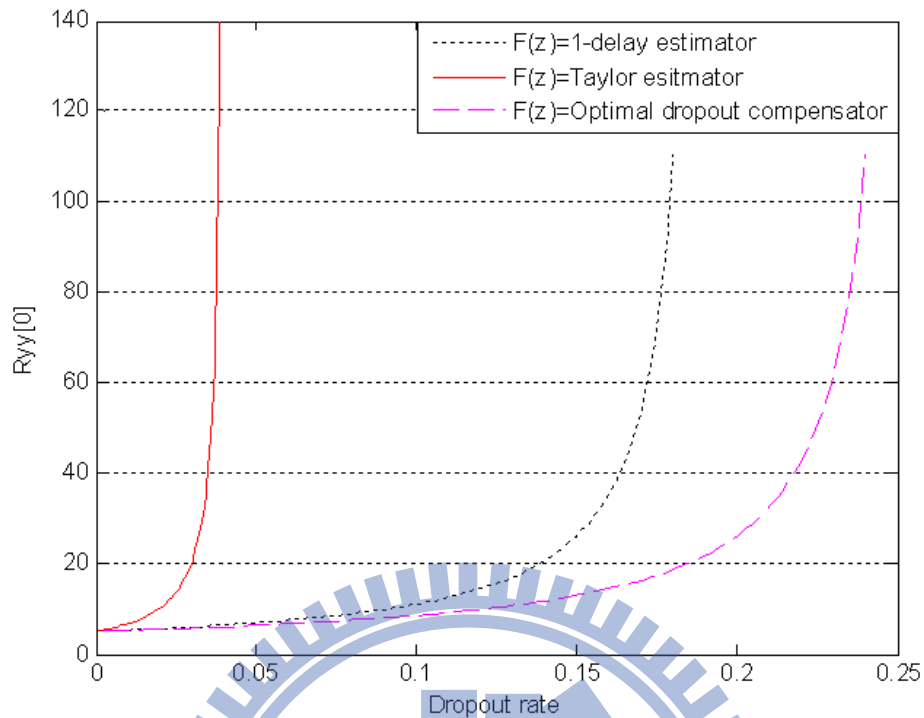


Fig. 2.9 Output PSD with white noise input.

2.3.2 Circular motion command

In real applications, motion commands in general are simple signals like G01, G02 and G03 in CNC codes as linear, clockwise and counter clockwise circular motions, respectively. Basically, a third-order Taylor expansion is suitable to represent most the basic CNC motion commands. Here, a sinusoidal wave in a single axis with the magnitude 50 mm under the feedrate 3000 mm/min as input $r[k]$ is adopted to verify the circular motion performance of NCS. Results of three different message estimators under different dropout rates are shown in Fig. 2.10. Simulation results indicate that by applying the optimal dropout compensator, it leads to the worst control accuracy and its dropout rate is limited to 20% only. Theoretically, the optimal dropout compensator is designed to minimize the power spectral density of the output signals due to the noise input and it is not suitable for the cases with contouring commands. On the other hand, the proposed 3rd-order Taylor message estimator results in the best control performance when the dropout rate is as high as to 50%.

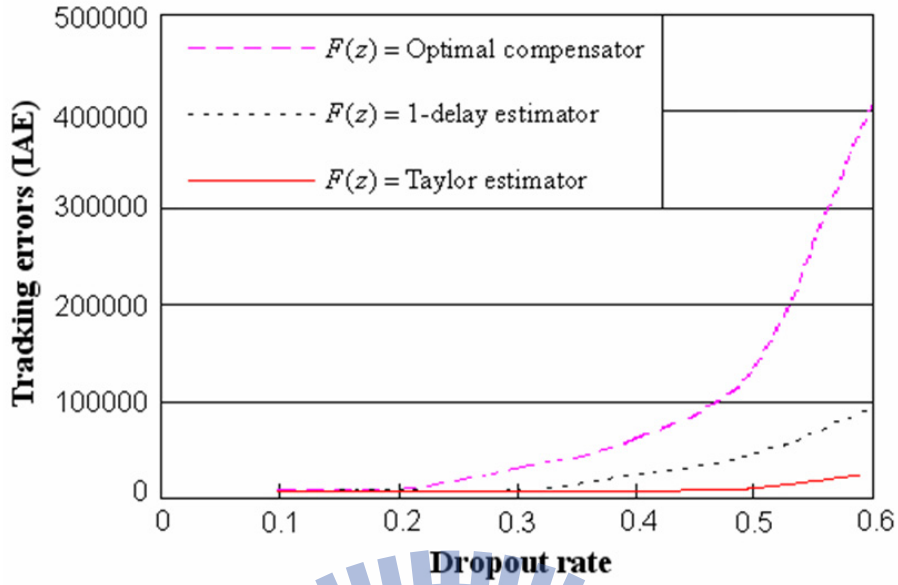


Fig. 2.10 Tracking accuracy of different message estimators

2.3.3 NURBS motion commands

The circle and butterfly contours are selected as the test control commands produced by applying the non-uniform rational B-Spline (NURBS) curve interpolator (Piegl and Tiller, 1995; Gopi and Manohar, 1997). The NURBS interpolator can create free-form curves easily by manipulating the values of control points, weight and knot vectors. The mathematical formulation of NURBS curve can be described as follows:

$$C(p) = \frac{\sum_{i=0}^n Z_{i,k}(p)\psi_i V_i}{\sum_{i=0}^n Z_{i,k}(p)W_i} = \sum_{i=0}^n R_{i,k}(p)V_i \quad (2.10)$$

and

$$R_{i,k}(p) = \frac{Z_{i,k}(p)\psi_i}{\sum_{i=0}^n Z_{i,k}(p)\psi_i} \quad (2.11)$$

where V_i is the control points; ψ_i is the corresponding weights of V_i ; $n+1$ is the number of control points; k is the order of the NURBS curve; $Z_{i,k}(p)$ is the k th order B-spline basis function; $R_{i,k}(p)$ is the rational basis function. With the circular

commands and the butterfly commands (Yeh and Hsu, 1999), Fig. 2.11 and Fig. 2.12 show that the proposed motion estimator can significantly improve control performance as data dropout occurs.

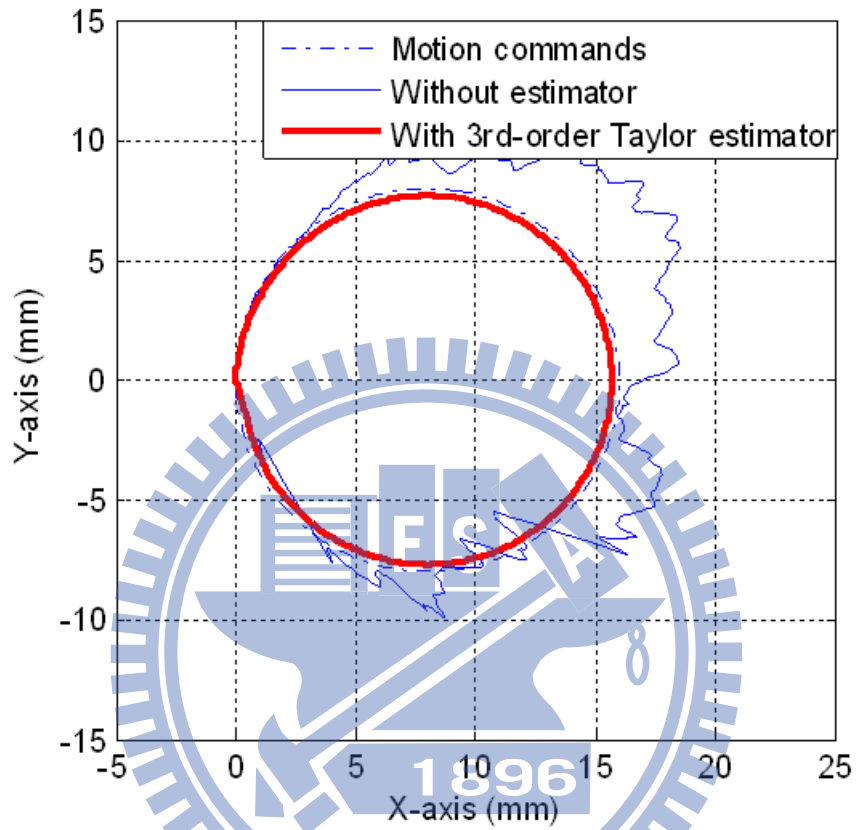


Fig. 2.11 NURBS simulation result as $\varepsilon = 19.97\%$.

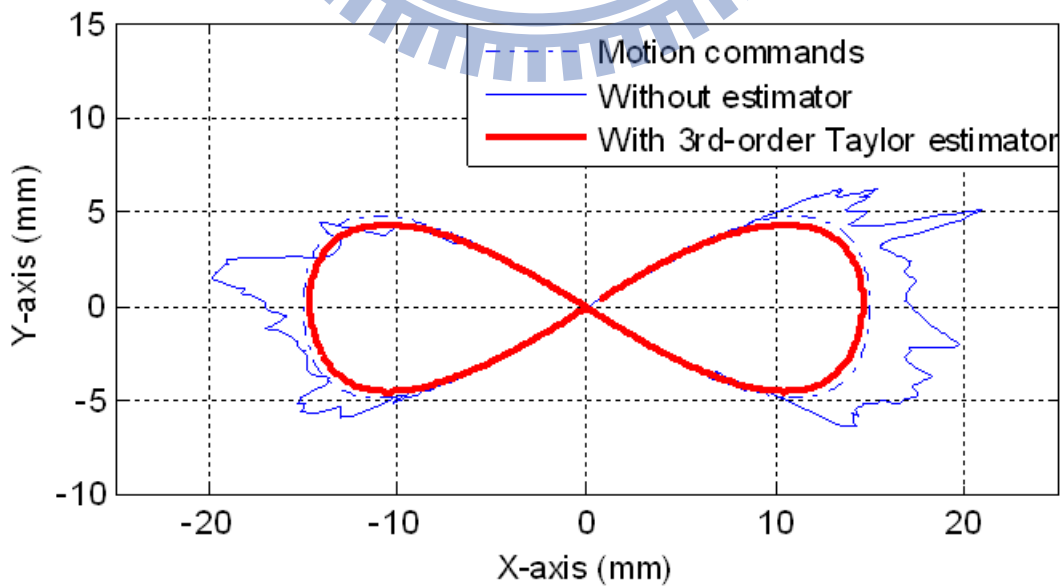


Fig. 2.12 NURBS simulation result as $\varepsilon = 19.97\%$.

2.4 Experimental results

2.4.1 Experimental setup with the CAN bus

The proposed approach was also verified on a CNC machine tool driven by the AC servo motor. The message estimator together with the controller were implemented on the TI TMS320F2812 DSP microcontroller and its internal CAN protocol was used to transmit/receive messages of the position commands and feedback measurements. The transmitted messages missing the deadline of the fixed sampling time were counted as the data dropout in a time base. Without applying the message estimator, a sinusoidal command was provided with a CAN transmission rate at 250K *bit/s*. Its missing message transmission error at every sampling period 1 *ms* was recorded as shown in Fig. 2.13. Experimental results shown in Fig. 2.14 indicate that the proposed 3rd-order Taylor message estimator effectively reduces the network transmission errors around $\frac{1}{100}$.

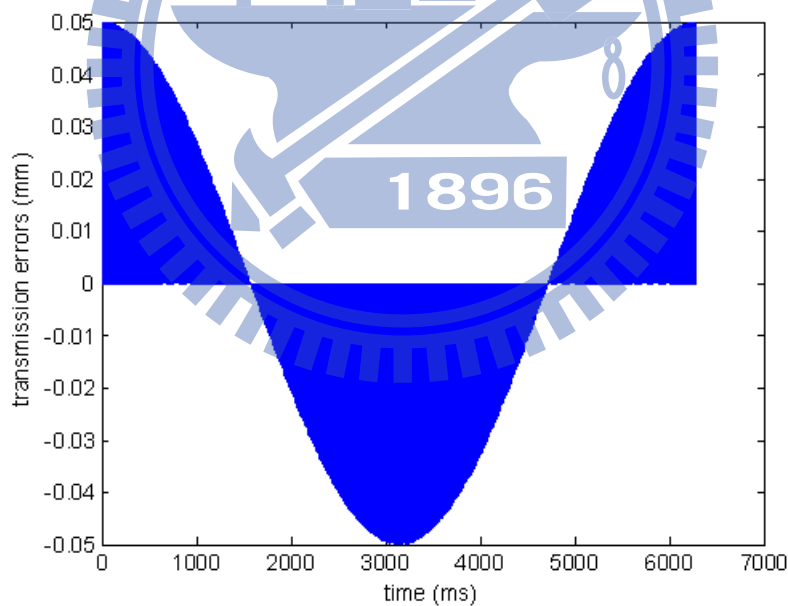


Fig. 2.13 The transmission error without message estimator

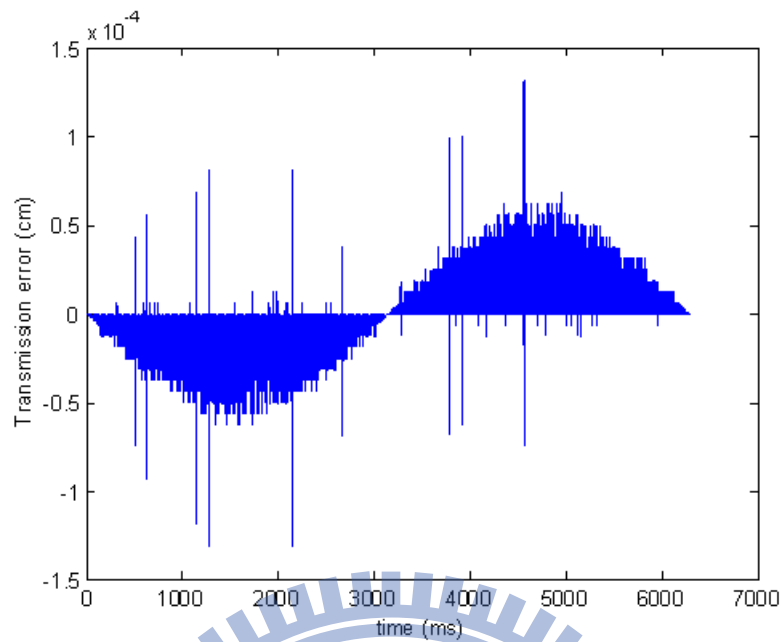


Fig. 2.14 The transmission error with the Taylor message estimator

2.4.2 CNC applications

Furthermore, the proposed 3rd-order Taylor message estimator and the controller were applied to the DYNA MTYE 1007 CNC machine in a NCS structure, as shown in Fig. 2.15. The sinusoidal command message with the position amplitude 50 mm under the feedrate 3000 mm/min are shown on Fig. 2.16. Experimental results indicate that without the message estimator, the significant tracking error of NCS on CNC leads to a relatively unstable system, as shown in Fig. 2.17. By applying the proposed Taylor message estimator, the motion NCS not only becomes more stable but also greatly reduces the tracking error.

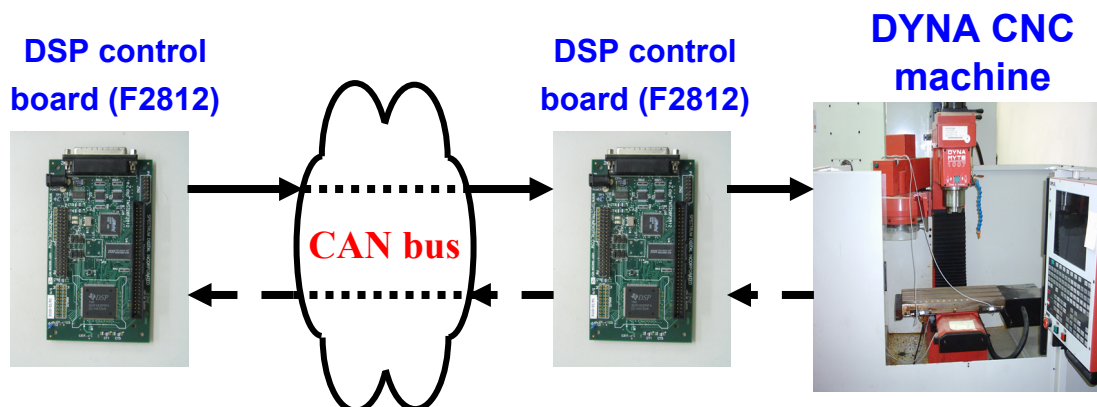


Fig. 2.15 Experimental setup.

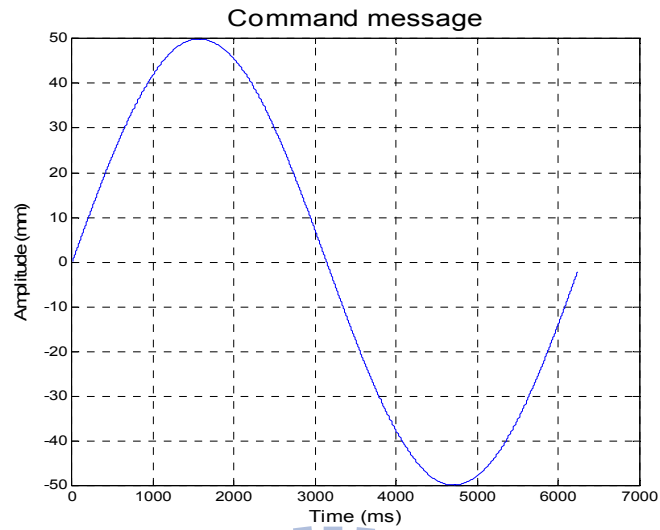


Fig. 2.16 Command message

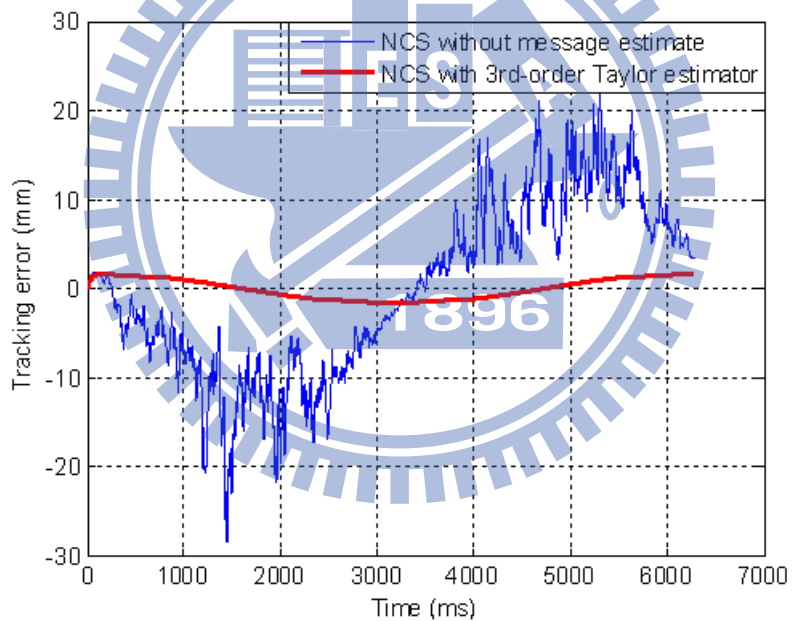


Fig. 2.17 Experimental results with/without the message estimator.

2.4.3 Feedforward control on NCS

The feedforward control has been successfully applied to motion systems by canceling poles and zeros of the plant model to improve tracking accuracy. Apparently, the model-based feedforward design is not suitable for general NCS because the dynamic model of a general NCS is usually uncertain due to both the time-delay effect and the data dropout. Since the proposed message estimator may

recover the missing message in the NCS to render a more reliable NCS model, the feedforward control structure as shown in the Fig. 2.18 integrates with the Taylor message estimator becomes feasible.

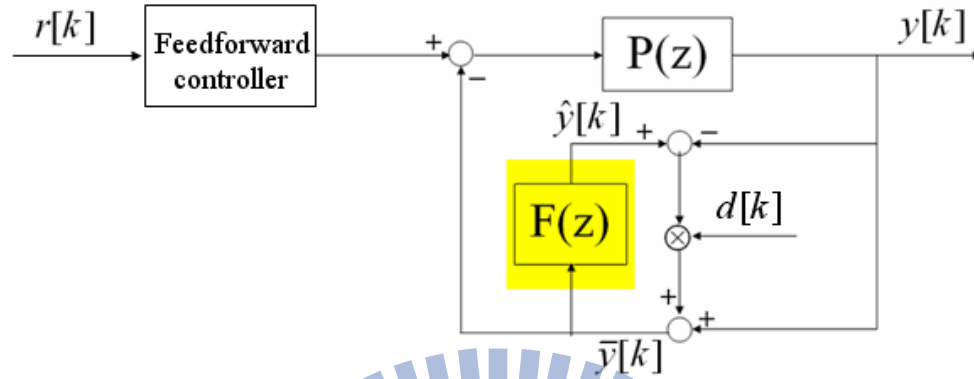


Fig. 2.18 The motion NCS with the message estimator and the feedforward controller.

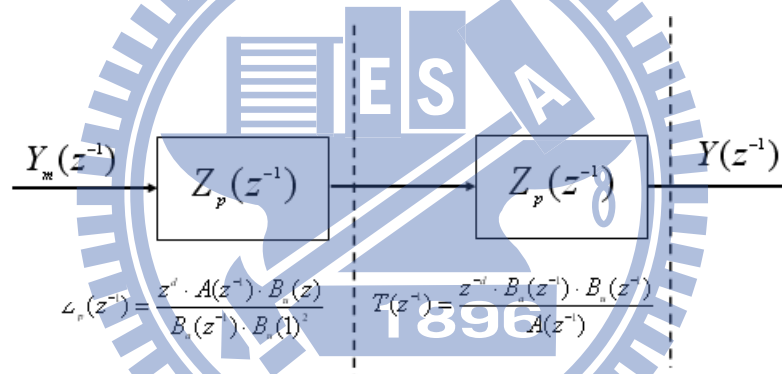


Fig. 2.19 The basic structure of ZPETC.

The basic feedforward structure of the zero phase error tracking control (ZPETC) shown in the Fig. 2.19 cancels all removable poles and zeros in the position control loop (Tomizuka, 1978). For those unstable zeros, their conjugate zeros are added to compensate for their phase error through the entire frequency range. If the transfer function of the original position loop is

$$T(z^{-1}) = \frac{z^{-d} \cdot B(z^{-1})}{A(z^{-1})} = \frac{z^{-d} \cdot B_a(z^{-1})B_u(z^{-1})}{A(z^{-1})} \quad (2.14)$$

the transfer function of the ZPETC can be expressed as

$$Z_p(z^{-1}) = \frac{z^d \cdot A(z^{-1}) \cdot B_u(z)}{B_a(z^{-1}) \cdot B_u(1)^2} \quad (2.15)$$

Then, the total transfer function $P(z^{-1})$ becomes

$$P(z^{-1}) = Z_p(z^{-1}) \cdot T(z^{-1}) = \frac{B_u(z)B_u(z^{-1})}{B_u(1)^2} = \frac{|B_u(z^{-1})|^2}{B_u(1)^2} \quad (2.16)$$

Thus, ZPETC leads to zero phase error in all frequency range. Besides, the DC gain is unity at the zero frequency, as in Eq. (2.16).

Based on the measured results of the CAN bus shown in Table 1, simulation results shown in Figs. 2.20-2.22 indicate that both the 1-delay message estimator and the optimal dropout compensator present unsatisfactory performance as the dropout rate increases. The tracking accuracy of the controller combining the Taylor estimator and the ZPETC leads to significant improvement in motion accuracy as shown in Fig. 2.22. Experimental results on the NCS of CNC shown in Fig. 2.23 also indicate that the tracking error is significantly reduced by applying the proposed message estimator.

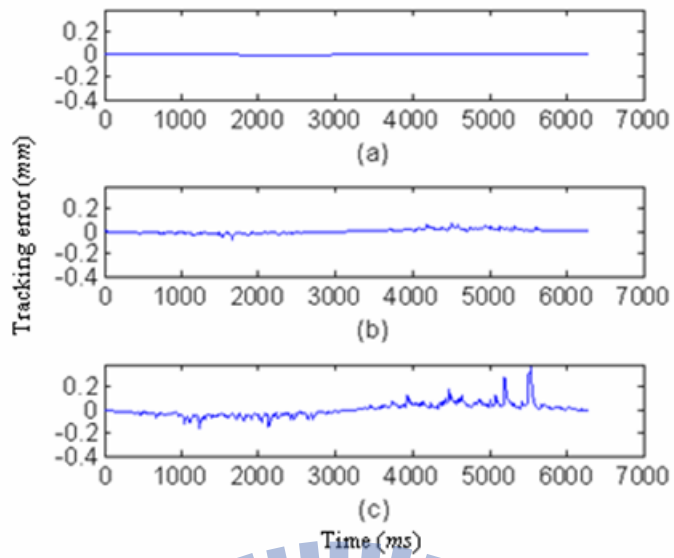


Fig. 2.20 Tracking errors of ZPETC and 1-delay estimator with dropout rate (a) 0.49% , (b) 19.97% , (c) 42.14% .

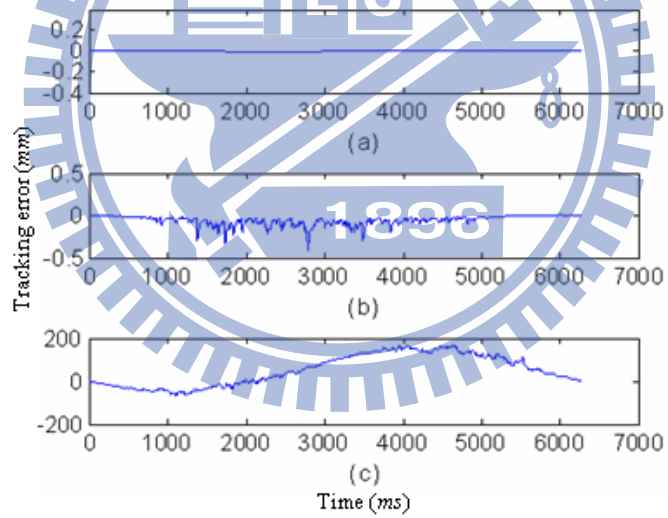


Fig. 2.21 Tracking errors of ZPETC and the optimal dropout compensator with dropout rate (a) 0.49% , (b) 19.97% , (c) 42.14% .

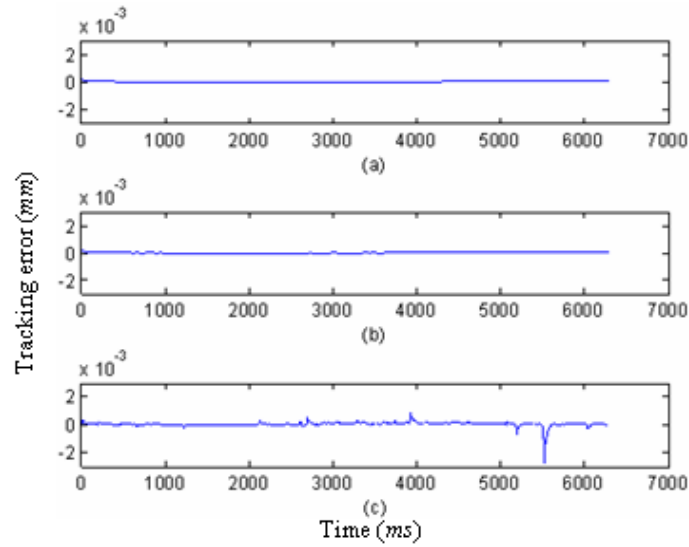


Fig. 2.22 Tracking errors of ZPETC and the 3rd-order Taylor estimator with dropout rate (a) 0.49%, (b) 19.97%, (c) 42.14%.

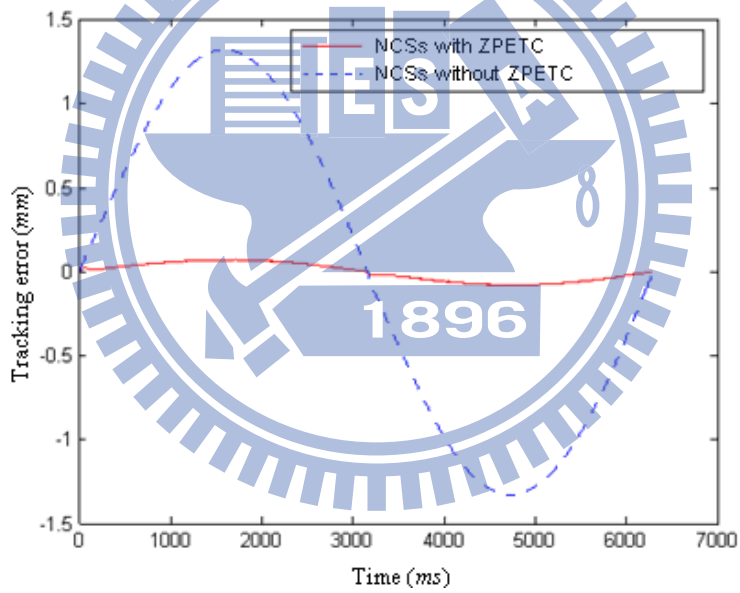


Fig. 2.23 Experimental result with the ZPETC and the Taylor message estimator

2.5 Summary

The dropout rate of the CAN bus increases rapidly even when the transmission rate slightly decreases, as shown in the Table 2.1. Therefore, the dropout effect of the NCS causes the serious motion error in precise motion systems. Basic motion control commands (CNC G-code RS-273-A, RS-274-B) can be properly described in both the position and the velocity. From both analytical and experimental results, the proposed

3rd-order message estimator can be suitably applied to the motion NCS to satisfactorily compensate for the missing commands or measurements. The novel control structure containing a 3rd-order Taylor message estimator is successfully applied to the motion NCS to improve the control performance significantly. Both simulation and experimental results are summarized as in the following:

- (1) Fig. 2.8 indicates that the 3rd-order Taylor message estimator is more suitable in real applications by considering different motion commands (both smooth and abruptly-changed). Moreover, simulation results indicate that the proposed 3rd-order Taylor estimator is effective in both the high and the low noise-contaminated signals.
- (2) In real applications of the present message estimator, the dropout data must be smooth and predictable, like position commands or velocity commands. Because the velocity, the acceleration and the jerk which are the first, second, and third derivatives of the position, the present 3rd-order Taylor message estimator is applicable to motion NCS to cover all information of the missing messages of the commands or measurements.
- (3) In practice, all motion commands and paths of CNC or robots are predictable and the proposed message estimator in NCS is suitable. Experimental results of the CNC machine tool indicate that by applying the proposed 3rd-order Taylor message estimator, the maximum tracking error is reduced from 12 *mm* to 2.4 *mm*.
- (4) The present 3rd-order Taylor message estimator not only reduces the tracking error, but also degrades the NCS model uncertainty to achieve reliable motion. By integrating the feedforward control together with the message estimator, the present NCS model uncertainty is reduced and results shown in Fig. 2.23 indicate that the tracking error further decreases from 2.4 *mm* to 0.08 *mm*.
- (5) The communication delay basically is in a stochastic nature. If it is less than one sampling period T , its delay effect on the degradation of NCS performance is negligible. However, as the delay becomes more serious, say several times than the sampling interval T like in the Ethernet, the data dropout will also become more serious and special design should be considered, like applying the Smith predictor.

By applying the proposed message estimator, advanced control design can be further employed for the motion NCS design to render satisfactory precision and responses. However, the proposed approach is feasible only as the data dropout rate is small and the missing message can be estimated in a deterministic approach. As the dropout rate increases to 50%, other statistical approaches to determine the stochastic model of the missing messages and to take proper actions may thus be required (Hansen and Yu, 2001).



Chapter 3

The Real-time transition probability estimator

3.1 The Distribution effect of data dropout

Traditionally, the data dropout rate ε is recognized as the quality of service (QoS) for NCS. However, in motion NCS, compared with evenly distributed missing data, continuous missing data will cause even more serious motion error. Fig. 3.1 (a)-(b) show two signals with the same data dropout rate 20% applied to the butterfly profile for fifth-order NURBS commands shown in Fig. 3.2 (Hsieh et al., 2006). By applying the same 3rd-order Taylor estimator for compensating the missing motion commands, simulation results show that the transmission error become more significant when the data dropout is more centralized, as shown in Fig. 3.3(b). Therefore, these results indicate that both data dropout and its distribution play crucial roles in NCS motion accuracy.

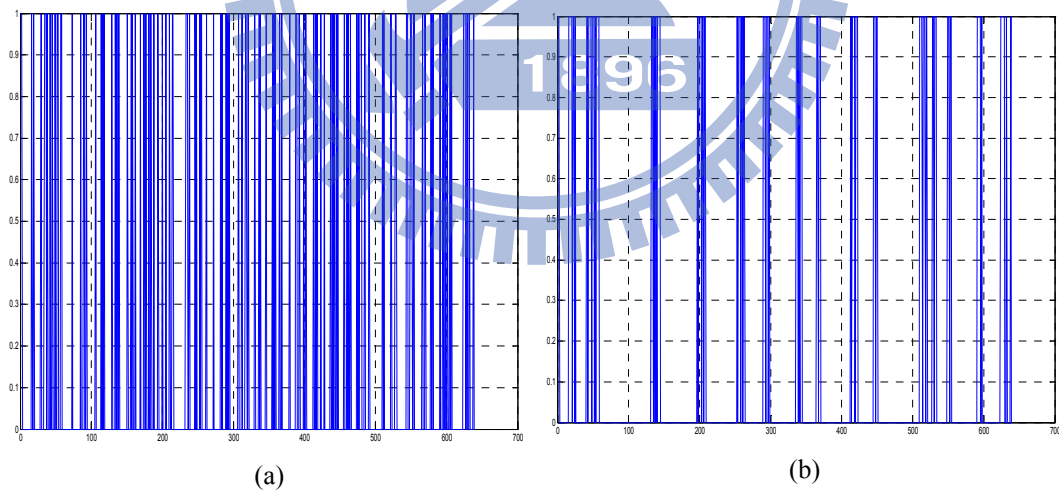


Fig. 3.1 (a) Distributed and (b) centralized dropout signals with the same dropout rate 20%

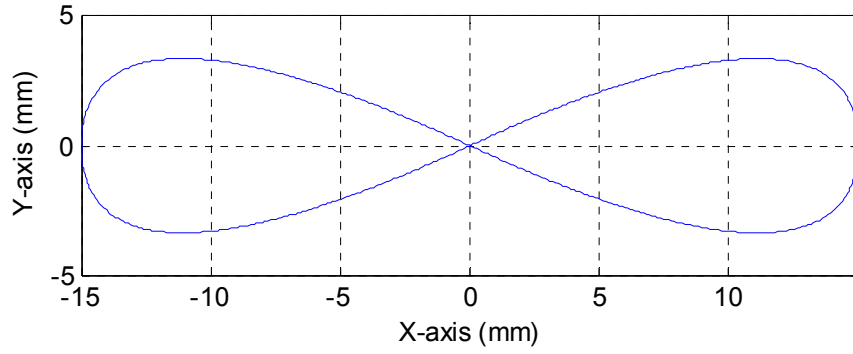


Fig. 3.2 NURBS position command

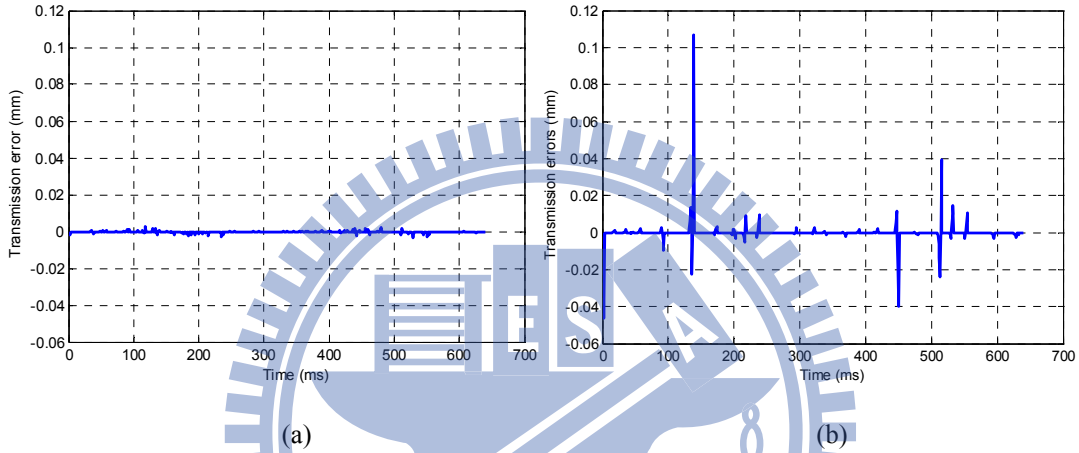


Fig. 3.3 Y-axis racking errors of (a) distributed and (b) centralized missing data (20% dropout rate)

3.2 Two-state Markov chain network model

In the Bernoulli model, the data dropout rate ρ describes the probability of both the received and dropout status (Adas, 1997). The Bernoulli model cannot capture the bursty behavior because the received signal or dropout signal at any instant is independent of all other outcomes. However, transmissions in NCS commonly exhibit bursty behavior. To capture bursty network losses, a two-state Markov chain was used (Gilbert, 1960; Wang and Moayeri, 1995) with both theoretical and practical complexity in modeling. In a Markov chain, the outcome for the future state that the system will transit to depends only on the present state and not any previous ones; the system is also memoryless. In this work, two parameters describe the distribution of packet dropouts, the dropout state D and the received R state. In our notation for success or failure in message received of the network at each

sampling period, the parameter $\rho_{i,j}$ of the probabilities is

$$\rho_{ij} = \Pr[\alpha(k+1) = j | \alpha(k) = i] \text{ for } i, j \in \{R, D\} \quad (3.1)$$

where $\rho_{D,R}$ is the probability of transitioning from a D (dropout) state to a R (received) state. Likewise, $\rho_{R,D}$ is the probability of transitioning from a R state to a D state. The probabilities of all transitions can be conveniently represented pictorially as shown in Fig. 3.4, or in a Markov transition matrix given by Eq. (3.2)

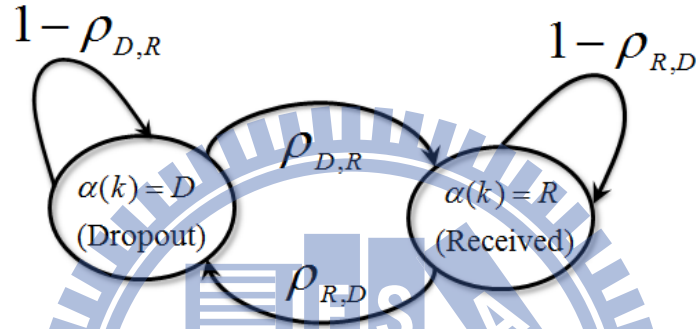


Fig. 3.4 Two-state Markov chain network model

$$P = [\rho_{ij}] = \begin{bmatrix} \rho_{D,D} & \rho_{D,R} \\ \rho_{R,D} & \rho_{R,R} \end{bmatrix} = \begin{bmatrix} 1 - \rho_{D,R} & \rho_{D,R} \\ \rho_{R,D} & 1 - \rho_{R,D} \end{bmatrix} \quad (3.2)$$

While in the Markov transition matrix, P itself describes the probabilities of all 1 step transitions, powers of the transition matrix, i.e. P^m , describe the likelihoods of transitions m steps in the future. That is $P^m = [\rho_{i,j}^m]$,

where

$$\rho_{i,j}^m = \Pr[\alpha(k+m) = j | \alpha(k) = i] \text{ for } i, j \in \{R, D\} \quad (3.3)$$

In this study, m is chosen as 1. The Markov transition matrix also provides information about the average behavior of the system. If all of the transition matrix entries are positive, it is said to be regular. And if the transition matrix P is regular, there is a unique steady-state vector μ such that $(P^T)^k \mu_0 \rightarrow \mu$ as $k \rightarrow \infty$, where the transpose of P is adopted. This steady-state vector implies both an eigenvector for the eigenvalue $\lambda=1$, and a probability vector whose entries sum to one. The existence of this steady state vector is guaranteed and for the network model

framework presented here, it is describes the average received and dropout of the network (Leon, 1998). For example, the network condition $(\rho_{D,R}, \rho_{R,D}) = (0.7, 0.2)$ corresponds to the transition matrix,

$$P = \begin{bmatrix} 0.3 & 0.7 \\ 0.2 & 0.8 \end{bmatrix}$$

where

$$\begin{aligned} \text{eig}(P^T) &= 0.1, 1 \\ \text{eigenvectors} &\begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \begin{bmatrix} 0.222 \\ 0.778 \end{bmatrix} \end{aligned}$$

(3.4)

The eigenvector corresponding to the eigenvalue 1 indicates that on average, the system in the R state will occupy 77.8% of the total time and in the D state will be 22.2% of the time under that network condition. The two-parameter model description is very convenient for graphically visualizing the effect of different network condition parameters, and it will be used throughout Chapter 5 to examine stability of motion NCS.

To further investigate the physical significance of the given network parameters, the amount of burstiness can be quantified as a function of those network parameters [Kawka 2006]. The expected number of consecutive D states and the expected number of consecutive R states are given by

$$\begin{aligned} E(\text{cons } D \text{ states}) &= \frac{1}{\rho_{D,R}} = \frac{1}{1 - \rho_{D,D}} \\ E(\text{cons } R \text{ states}) &= \frac{1}{\rho_{R,D}} = \frac{1}{1 - \rho_{R,R}} \end{aligned} \quad (3.5)$$

3.3 The transition probability estimator

Now that two-state Markov chain network model can clearly show the nature characteristic of network signals, measurement of transition probabilities is very

important and worthy issue. However, transition probabilities are caused by unpredictable collision, network traffic load and system sampling period. Therefore, Measurement of transition probabilities is time-consuming and difficult. Previous researches can successfully measure transition probabilities to construct two-state Markov chain network model (Bertuccelli and How, 2008; Azimi et al., 2005; Azimi et al., 2003). Nevertheless, wasting time and complication are still limitations to realize in real applications.

Motion NCS requires rapid and precise control. However, traditional transition probabilities are aimed at overall network signal, they must be not suitable for motion NCS. The real-time transition probability estimator is proposed to rapidly and precisely estimate local transition probabilities in motion NCS.

In real-time transition probability estimator, there are two opinions different to traditional transition probabilities. In the first place, an observation window is adopted as the local network signals. Observation windows are defined as a finite length of previous network signals. For example, if the length of observation windows is five and time index $k=10$, then $OW(10)=[\alpha(5) \alpha(6) \alpha(7) \alpha(8) \alpha(9)]$ is as an Observation Windows, where $\alpha(k)$ is network signals. From the observation window, network characters can be rapidly analyzed. In the second place, $\hat{\rho}_{ij}(k)$ is re-defined as

$$\hat{\rho}_{i,j}(k) = \Pr[\Pr(OW(k) = j) | \alpha(k) = i] \text{ for } i, j \in \{R, D\} \quad (3.2)$$

Under the definition, the local transition probability, $\hat{\rho}_{i,j}(k)$ can be efficiently estimated because all previous network signals are known. Moreover, the average of a $\hat{\rho}_{i,j}(k)$ with a relatively long duration can represent the transition probability, $\rho_{i,j}$.

3.4 Illustrative example

By applying the proposed real-time estimator of the local transition probability to the two signals with the same overall transition probability $\rho_{D,D} = 0.5$, as shown in Fig. 3.5 (a) and (b), respectively, their distributions are significantly different and

the estimation results for two signals at the 15th instance are $\hat{\rho}_{D,D}(15) = 0$ and $\hat{\rho}_{D,D}(15) = 60\%$, respectively. Therefore, the index of local transition probability $\hat{\rho}_{i,j}(k)$ is more appropriate to imply the distribution of the dropout data in motion NCS than the dropout rate only.

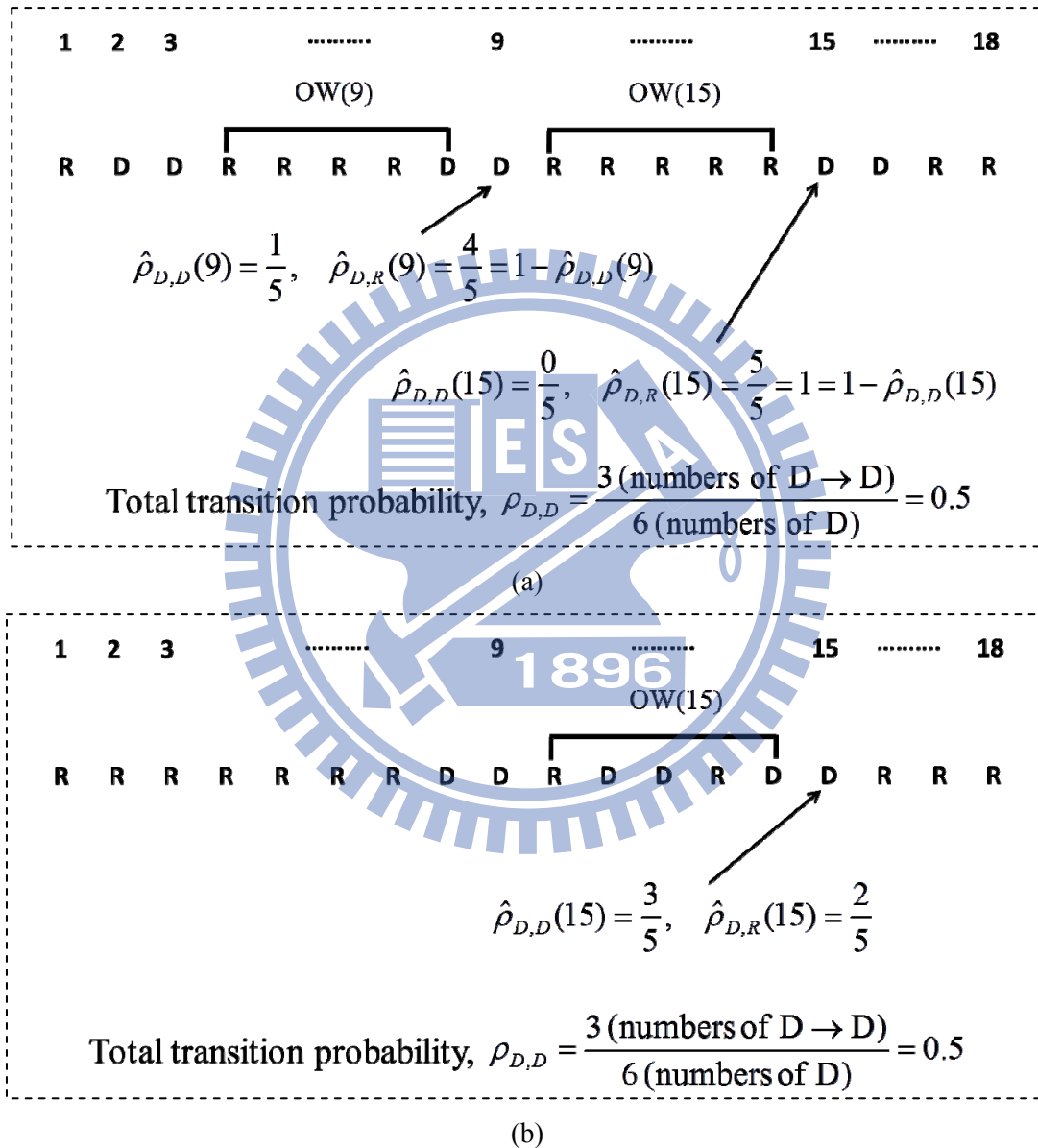


Fig. 3.5 Real-time calculation of local transition probability with the same overall transition probability $\rho_{D,D} = 0.5$.

Furthermore, the proposed real-time transition probability estimator is verified and simulation results are presented in Fig. 3.6 (a)-(b). In Fig. 3.6(a), the real-time transition probability estimator can efficiently measure different transition probabilities in network signals and its measuring time is less than 0.1 sec in Fig. 3.7.

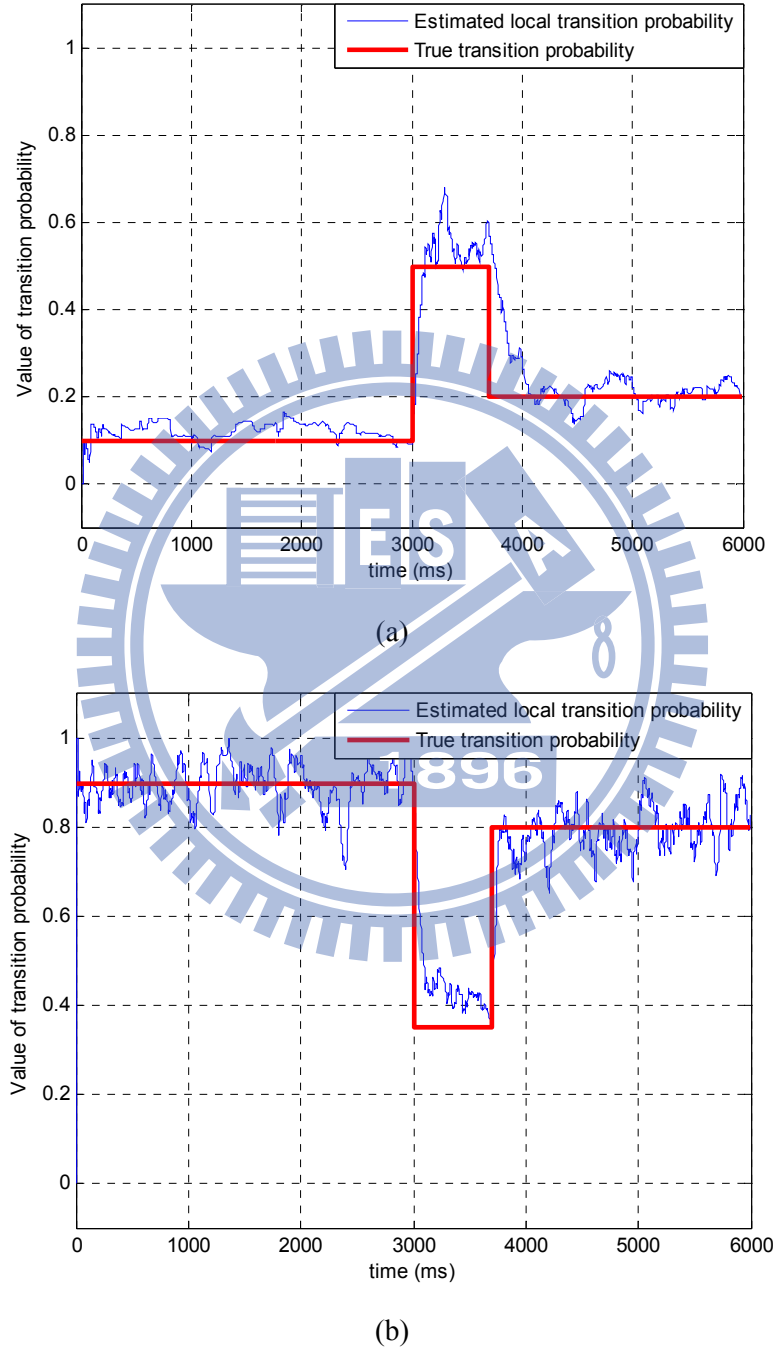


Fig. 3.6 Transition probabilities of (a) average of $\hat{\rho}_{D,D}(k)$ vs $\rho_{D,D}$ and (b) average of $\hat{\rho}_{R,R}(k)$ vs $\rho_{R,R}$

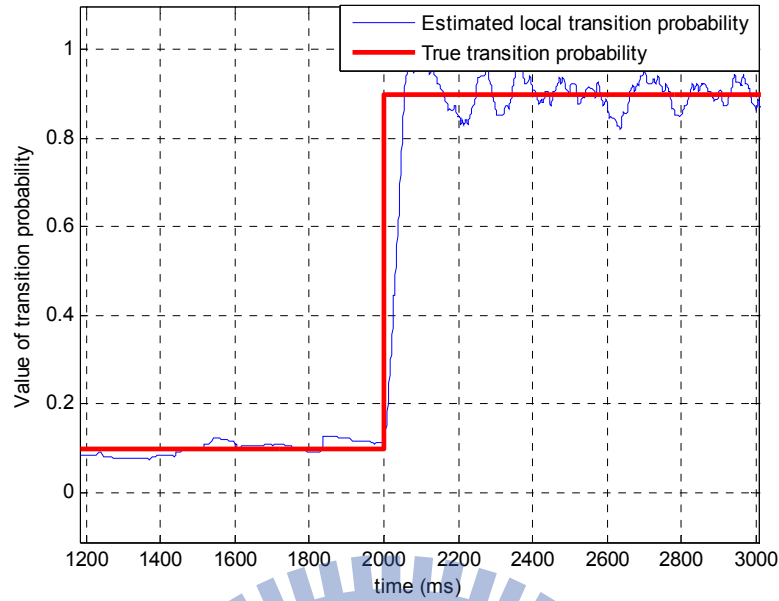


Fig. 3.7 Measuring time of the real-time transition probability estimator.

3.5 Summary

The proposed real-time transition probability estimator is designed with a real-time view of the communication quality because of the short-window signals are applied. The estimated results can be applied to on-line monitor QoS of network and results are summarized as follows:

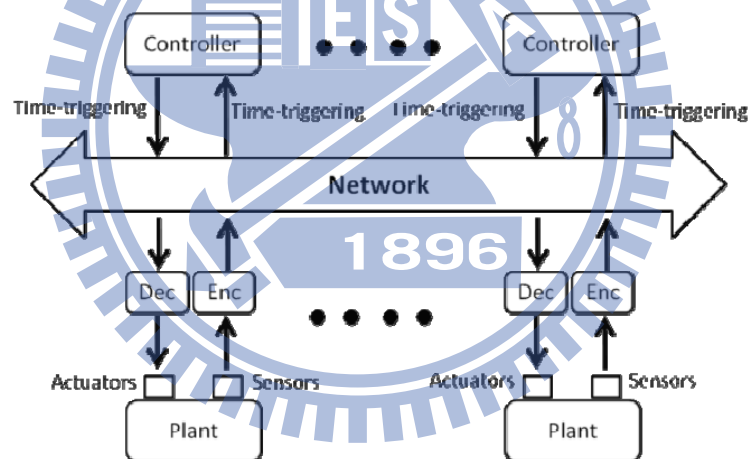
- (1) The transition probability matrix can be used to analyze the data dropout rate by its eigenvector and the expected number of consecutive dropout states can be estimated by obtaining the transition probability.
- (2) The transition probability ($\hat{\rho}_{D,D}$) can efficiently and rapidly monitor the real-time data dropout distribution of motion NCS with a straightforward algorithm by triggering those sampled points as the singles are missing.
- (3) The 50-points average of the local transition probability, $\hat{\rho}_{D,D}$ and $\hat{\rho}_{R,R}$, can suitably represent the transition probability of overall network communication. In simulation, the variation of network traffic load can be rapidly measured by applying the real-time transition probability estimator.

Chapter 4

The Intelligent message estimator

4.1 The structure of the multi-axis motion NCS

The general NCS architecture Fig. 4.1(a) shows that when the number of motion axes increases, the network traffic of the architecture also increases more seriously due to the commands and feedback messages that must be transmitted or received on time within a system sampling period. Therefore, it leads to the development of a faster network infrastructure to meet the requirement of synchronization such as Ethernet and EtherCAD. However, this also leads to higher-cost implementation given the modifications.



(a)

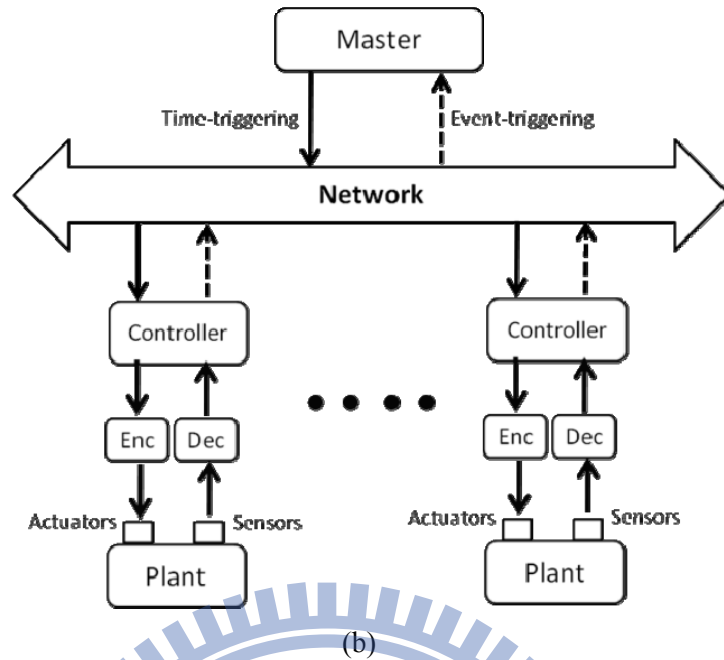


Fig. 4.1 (a) General NCS architecture and (b) practical motion NCS architecture

In real control applications, practical motion NCS architecture is generally modified as shown in Fig. 1(b), and only the command messages are transmitted from the master to the controller. Thus, the transmission can meet the hard real-time requirement within a sampling period to avoid possible heavy traffic over networks. Moreover, the feedback messages are transmitted according to certain monitoring functions by the event-triggering approach without occupying the network. Nevertheless, in motion NCS with multiple axes, when the network delay is longer than the sampling period, the missing message of motion NCS becomes unavoidable. Under such circumstances, a message estimator is thus required to estimate the missing commands and to compensate for their effect. This research is crucial for NCS and is still pursued by industries to promote NCS applications. Therefore, various message estimators with different advantages have been proposed to cope with the dropout effect for motion NCS under different conditions. For example, the one-delay message estimator is easily implemented because of its simple algorithm of replacing the missing message with previously received data (Ling and Lemmon, 2002). However, it cannot predict the variation of messages properly. On the other hand, the non-linear NCS was modeled as a Markovian jump linear system, and the finite loss history estimator (FLHE) was proposed to improve data dropout effects

when the dropout rate is accurately known (Smith and Seiler, 2004). Nevertheless, these methods generally require the accurate plant/network model. On the other hand, model-free strategies for control packet dropout compensators, such as the proportional plus derivative (PD) predictors with a different order of derivatives, are proposed to estimate dropout data and to compensate for their effect (Tian and Levy, 2008). Recently, the Taylor estimator was proposed to significantly improve the control performance in motion NCS (Hsieh et al., 2006; Hsieh and Hsu, 2008). All these reported methods are effective due to their design structures, which are commonly based on the assumption that the dropout data over the network are evenly distributed. Once the missing data occur in a continuous format, this will generally lead to a more serious maximum contouring error. Therefore, intelligent message estimator (IME) is proposed to compensate effects of data dropout based on real-time transition probability estimator.

Moreover, in multi-axis motion NCS, data dropout will lead to the problem of asynchronization among different axes. By applying the proposed IME, synchronization among different axes is also greatly improved. Both simulation and experimental results with the non-uniform rational B-spline (NURBS) motion commands have been verified. With different motion message estimators, the results indicate that the present IME maintains the lowest transmission error as well as the least motion contouring error when the transition probability ($\rho_{D,D}$) increases. The CAN-based two-axis AC servo motor control system was also successfully implemented with the proposed IME.

In motion NCS, the control messages for each motion axis must be transmitted on time through the network protocol to meet the control design specifications, as shown in Fig. 4.2. Since the time delay exists in stochastic and time-varying natures, the transmitted messages may miss the hard real-time deadline because the network bandwidth is saturated. Generally, it causes data dropout, as the network-induced time delay is longer than the system sampling time T_D , as shown in the timing diagram in Fig. 4.3.

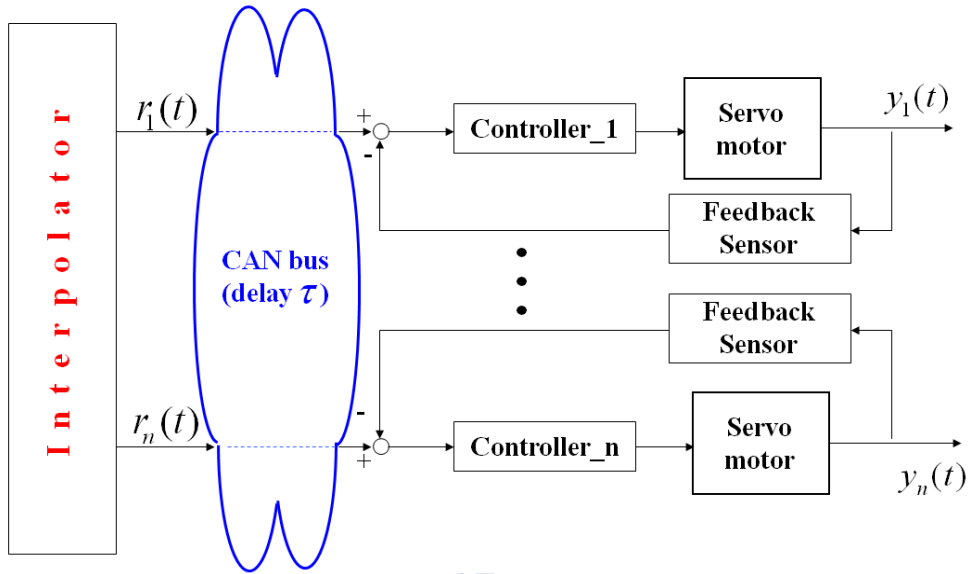


Fig. 4.2 Multi-axis motion NCS

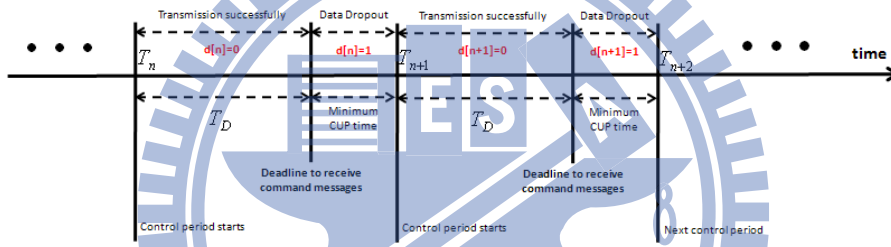


Fig. 4.3 Timeline analysis in a control period

4.2 The Least-square estimator

In motion NCS, the real position commands are smooth curves in most practical cases, but their curvatures may vary significantly along the contour. Practically, missing data with higher transition probability ($\rho_{D,D}$) will cause a more serious contouring error around the higher curvature. To estimate the missing messages in NCS, the one-delay estimator simply adopts the last received message as the current missing message, and the Taylor estimator estimates the current missing message from past received signals. If the past signal is also missing, the message obtained from the estimators also becomes unreliable. In this dissertation, the IME is proposed based on the integration of the least-square estimators with different orders based on the online measured local transition probability ($\hat{\rho}_{D,D}$). As the messages are serious dropouts, estimation based on the previous data is no longer reliable, and the

one-delay estimator is then included in the proposed IME.

Since the online estimation is time consuming, all parameters of the real-time least-square estimation (LSE) can be obtained in advance. Thus, to achieve an online estimation and compensation algorithm for the missing command in motion NCS, the IME is proposed based on past messages within a short window by applying the least-square approach. For a general time sequence $x[0], x[1], \dots, x[M]$, a polynomial sequence can be suitably described as

$$x[k] = c_0 + c_1 k + c_2 k^2 + \dots + c_N k^N \quad (4.1)$$

Thus,

$$\begin{aligned} x[1] &= c_0 + c_1 + c_2 + \dots + c_N \\ x[2] &= c_0 + c_1 2 + c_2 2^2 + \dots + c_N 2^N \\ &\vdots \\ x[M] &= c_0 + c_1 M + c_2 M^2 + \dots + c_N M^N \end{aligned} \quad (4.2)$$

By rearranging Eq. (4.2) as

$$\begin{bmatrix} x[1] \\ x[2] \\ \vdots \\ x[M] \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 2^0 & 2 & \dots & 2^N \\ \vdots & \vdots & \dots & \vdots \\ M^0 & M & \dots & M^N \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_N \end{bmatrix} \equiv x = A \cdot c \quad (4.3)$$

The normal equation from the least-square approach can be applied to the data to obtain coefficient vector c as

$$c = (A^T A)^{-1} A^T x \quad (4.4)$$

Thus, the missing value for the current missing message can be predicted as

$$\begin{aligned}
x[M+1] &= c_0 + c_1(M+1) + c_2(M+1)^2 + \dots + c_N(M+1)^N \\
&= \begin{bmatrix} (M+1)^0 & (M+1)^1 & \dots & (M+1)^N \end{bmatrix} \cdot c \\
&= \begin{bmatrix} (M+1)^0 & (M+1)^1 & \dots & (M+1)^N \end{bmatrix} \cdot (A^T A)^{-1} A^T x \\
&\equiv LSE(M, N) \cdot x
\end{aligned} \tag{4.5}$$

and the estimator matrix $LSE(M, N)$ can thus be pre-calculated for real-time implementation. M indicates the data number to be counted, and N is the order of polynomial functions.

To achieve an online estimation for NCS, parameters should be determined in advance. Therefore, the order and the data number of the least-square estimator should be determined with practical concerns. For example, the NURBS signal can be approximated by a third-order polynomial equation obtained from the LSE (Sorenson, 1970). Therefore, the length of OW can be properly chosen to as large as five to suitably estimate the NURBS and other curves. Three useful $LSE(M, N)$ are pre-calculated for real-time applications as follows:

- $LSE(5,3) = 3.2z^{-1} - 2.8z^{-2} - 0.8z^{-3} + 2.2z^{-4} - 0.8z^{-5}$ (4.6)

- $LSE(3,2) = 3z^{-1} - 3z^{-2} + z^{-3}$ (4.7)

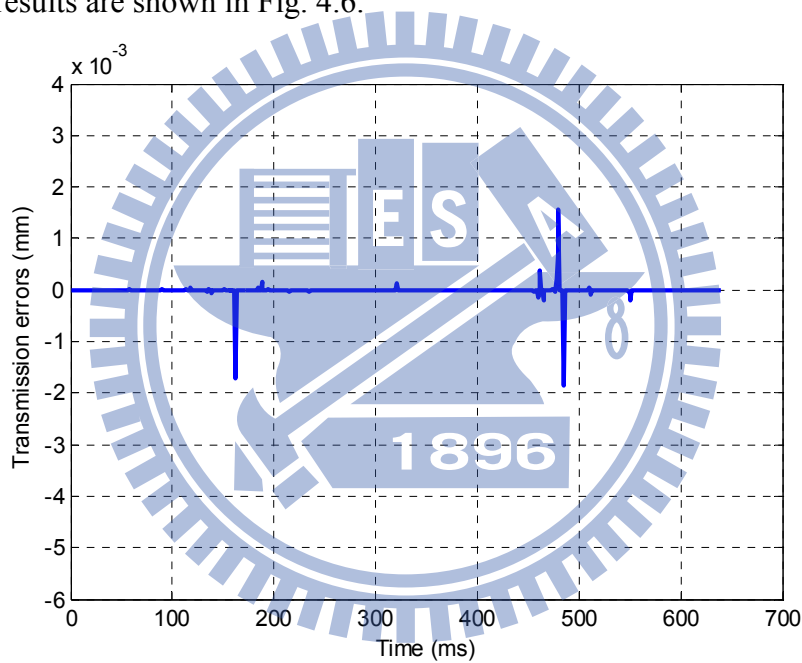
- $LSE(2,1) = 2z^{-1} - z^{-2}$ (4.8)

4.3 Analysis of LSE with different orders

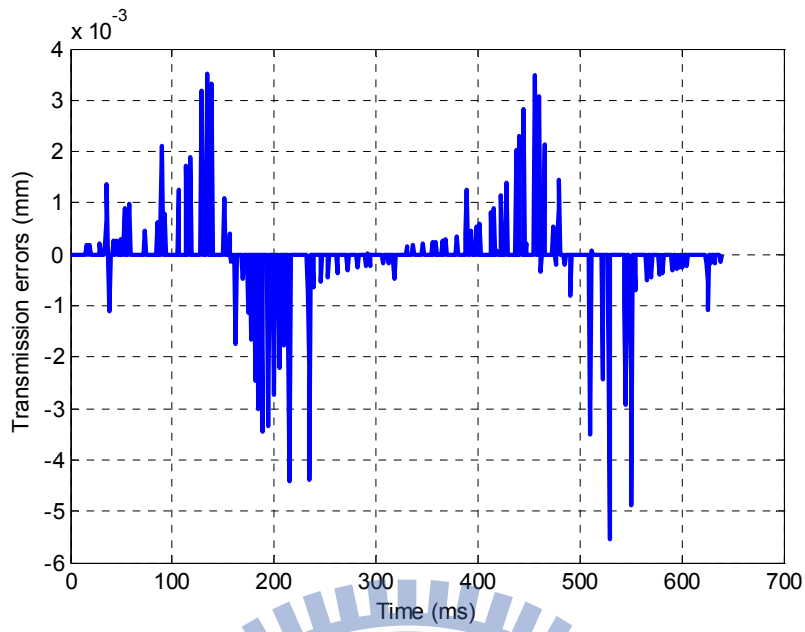
Fig. 4.4 shows the transmission errors obtained by separately applying the estimators of $LSE(5,3)$ and $LSE(2,1)$ to motion NCS in true transition probability $P_{D,D} = 0.2$. Simulation results show that $LSE(5,3)$ renders a better compensation effect as compared to $LSE(2,1)$, which should be applied in a more serious data dropout case. However, as $P_{D,D} = 0.4$, which implies that there are about two missing messages among the five transmitted messages, the transmission error

increases, and $LSE(5,3)$ is not suitable anymore. Fig. 4.5 shows that the compensation results applying $LSE(2,1)$ render better performance.

Furthermore, the least-square approach with a different M applied to a different $\hat{P}_{D,D}$ shows that applying $LSE(5,3)$ to compensate the missing data obtains the best motion accuracy as $0 \leq \hat{P}_{D,D} \leq 0.2$, but it becomes the worst as $\hat{P}_{D,D} > 0.2$. Moreover, $LSE(3,2)$ is more suitable for the situation, as $0.2 < \hat{P}_{D,D} \leq 0.4$. Moreover, $LSE(2,1)$ is most suitable for the situation as $0.4 < \hat{P}_{D,D} \leq 0.6$. In addition, the one-delay estimator possesses the best compensation effect as $0.6 < \hat{P}_{D,D} \leq 1$. These simulation results are shown in Fig. 4.6.

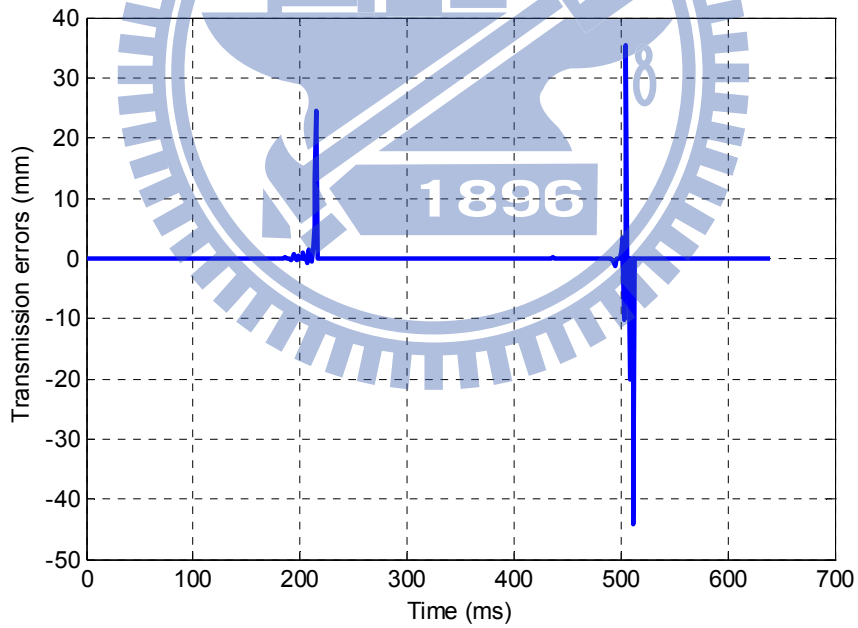


(a)

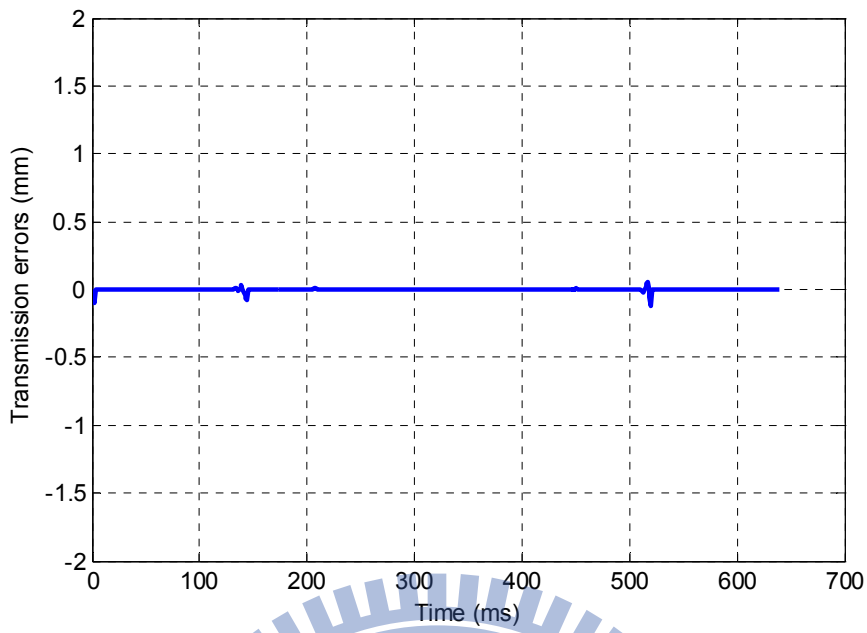


(b)

Fig. 4.4 Transmission errors with (a) $LSE(5,3)$ and (b) $LSE(2,1)$ as $P_{D,D} = 0.2$.



(a)



(b)

Fig. 4.5 Transmission errors with (a) $LSE(5,3)$ and (b) $LSE(2,1)$ with $\rho_{D,D} = 0.4$.

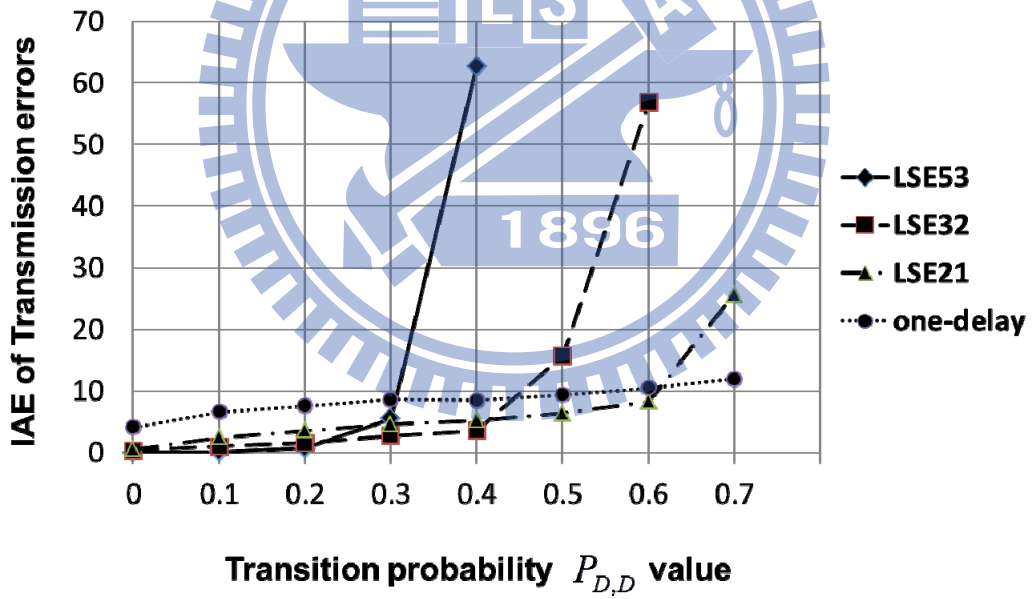


Fig. 4.6 Analysis of compensation effects with different true transition probability $P_{D,D}$

4.4 The IME architecture

It was discussed in the preceding section that intelligent Message Estimator (IME) adopts four useful message estimators for real-time applications as follows:

- $LSE(5,3)$ for low-data dropout cases

In this transmission case, all data within observation window length 5 are properly received, or at most, only one missing data is estimated among the four received data within the window. $LSE(5,3)$ is chosen to estimate a cubic-curve motion command with the order of 3 by using all five previous data, which may include estimated data at most. In other words, the third-order $LSE(5,3)$ can properly estimate the motion trajectory concerning its velocity, acceleration, and even the change of acceleration as the jerk, and the parameters are obtained from Eq. (4.6).

- $LSE(3,2)$ for medium-data dropout cases

In this case, the medium data dropout condition occurs, and the missing data within $0.2 < \hat{P}_{D,D}(k) \leq 0.4$. In other words, only three reliable data are accountable within the window to correctly estimate the missing data. Therefore, $LSE(3,2)$ is chosen to suitably estimate the quadric-curve trajectory with the order of 2 by using three previous data through considering both its velocity and acceleration from Eq. (4.7)

- $LSE(2,1)$ for heavy-dropout case cases

In this situation, the missing data within $0.4 < \hat{P}_{D,D}(k) \leq 0.6$, and $LSE(2,1)$ is chosen to estimate the motion trajectory concerning its velocity only by applying previous two data, either received or estimated.

- The one-delay estimator is adopted for serious-data dropout cases.

In this situation, network communication presents such a heavy data dropout rate; the missing data within $0.6 < \hat{P}_{D,D}(k) \leq 1$. Therefore, the estimation results

based on the mentioned least-square approach is not reliable anymore, and the one-delay estimator is chosen to estimate the position only by directly adopting the previous data as

$$1\text{-delay estimator} = z^{-1} \quad (4.9)$$

All switching laws according to Eq. (4.10a or 4.10b) based on the estimated $\hat{P}_{D,D}$ thus agree with both the simulation results and the theoretical analysis, as shown in Fig. 4.7. The proposed IME switching law based on the index of $\hat{P}_{D,D}$ can thus be applied suitably to estimate and recover the missing data for both centralized and distributed missing messages in motion NCS. Although the Taylor estimator has been proven to render more accurate results than the one-delay estimator, Fig. 4.7 further indicates that the proposed IME presents a much better performance under a different transition probability $P_{D,D}$, especially as the missing data becomes more serious. In summary, different LSEs are applied to different real-time $\hat{P}_{D,D}$, as shown in Fig. 4.8.

$$\begin{cases} 0 \leq \hat{P}_{D,D}(K) \leq 0.2, & LSE(5,3) \text{ is adopted} \\ 0.2 < \hat{P}_{D,D}(K) \leq 0.4, & LSE(3,2) \text{ is adopted} \\ 0.4 < \hat{P}_{D,D}(K) \leq 0.6, & LSE(2,1) \text{ is adopted} \\ 0.6 < \hat{P}_{D,D}(K) \leq 1, & 1\text{-delay estimator is adopted} \end{cases} \quad (4.10a)$$

$$\begin{cases} 0.8 \leq \hat{P}_{D,r}(K) \leq 1, & LSE(5,3) \text{ is adopted} \\ 0.6 < \hat{P}_{D,r}(K) \leq 0.8, & LSE(3,2) \text{ is adopted} \\ 0.4 < \hat{P}_{D,r}(K) \leq 0.6, & LSE(2,1) \text{ is adopted} \\ 0 < \hat{P}_{D,r}(K) \leq 0.4, & 1\text{-delay estimator is adopted} \end{cases} \quad (4.10b)$$

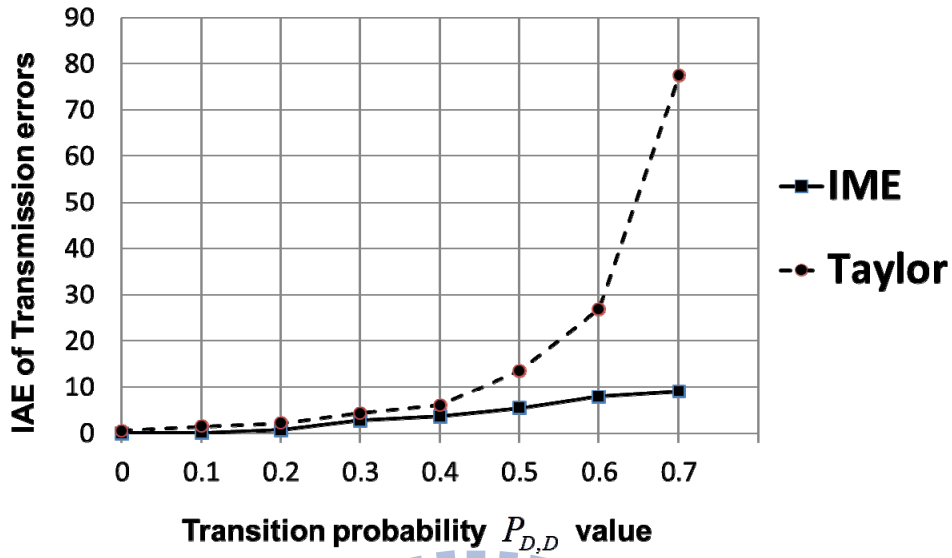


Fig. 4.7 Simulation results with different estimators

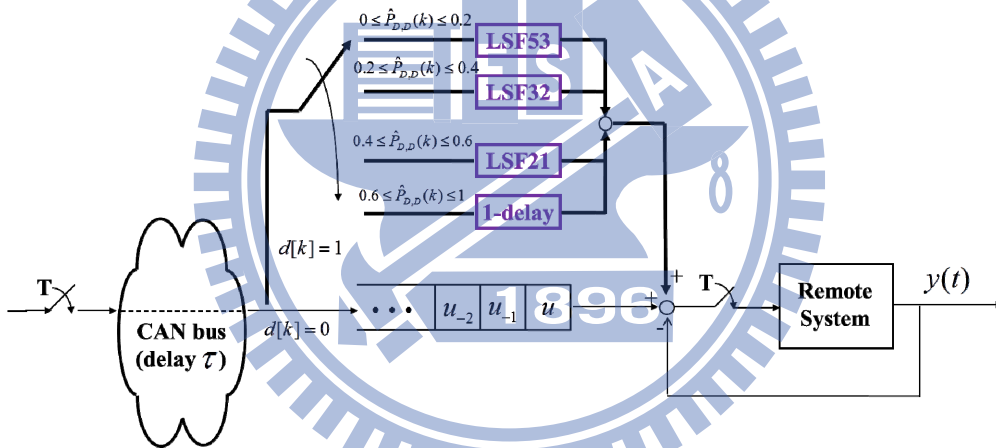


Fig. 4.8 The system architecture of the proposed IME

4.5 Simulation results

Applications of the present IME based on real-time transition probability $\hat{P}_{D,D}$ have been applied to the two-axis motion NCS, as shown in Fig. 4.1. The NURBS commands and the system response with $P_{D,D} = 0.2$ are shown in Fig. 4.9. The results show that the Taylor estimator can reduce the effects of data dropout at a lower $P_{D,D}$. However, Fig. 4.10 also shows that the contouring error obtained by applying IME is significantly reduced to achieve better contouring accuracy. Fig. 4.11

shows that the contouring accuracy of the present IME even renders a much better contouring accuracy when $P_{D,D}$ increases to 0.5. Furthermore, when the value of $P_{D,D}$ increases to 0.6, the Taylor estimator will lead to an unstable motion as shown in Fig. 4.12. Nevertheless, the proposed IME still results in a stable motion and maintains the contouring error as the least from the simulation results.

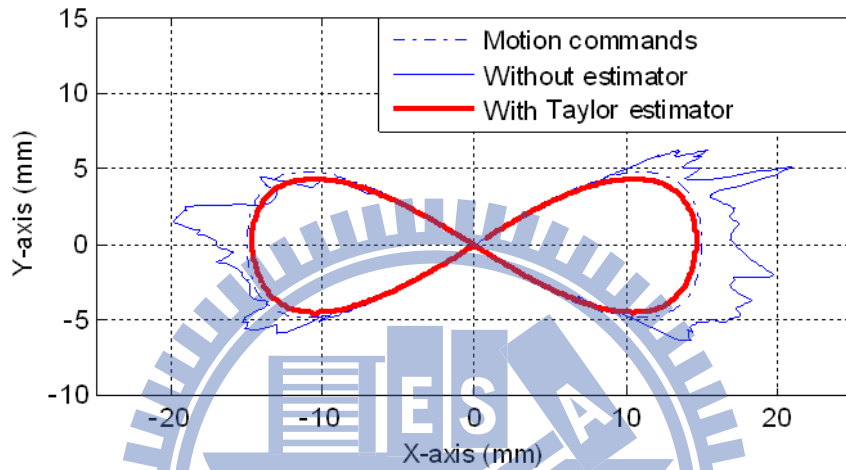
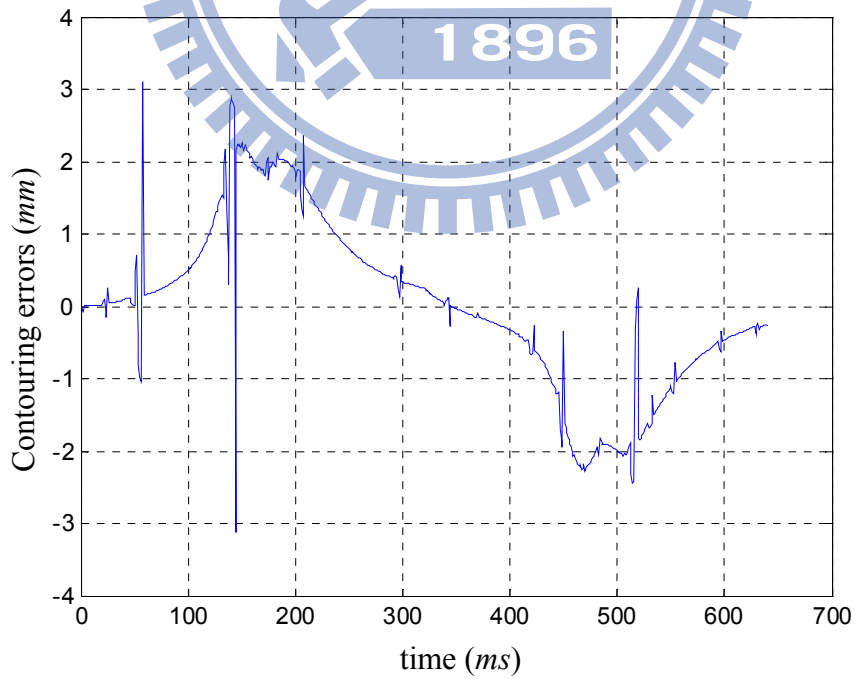
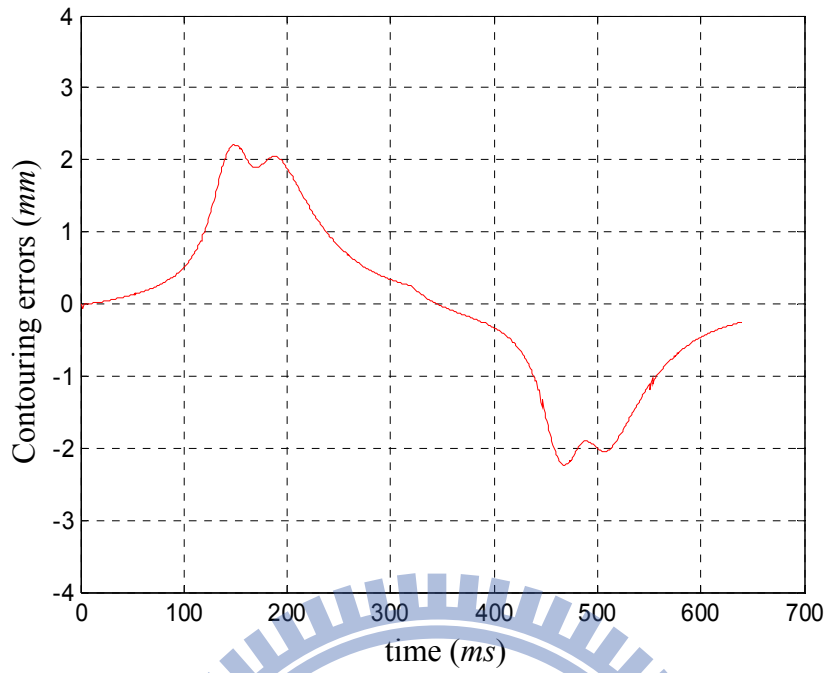


Fig. 4.9 Contours of motion NCS without/with the Taylor estimator ($P_{D,D} = 0.2$)

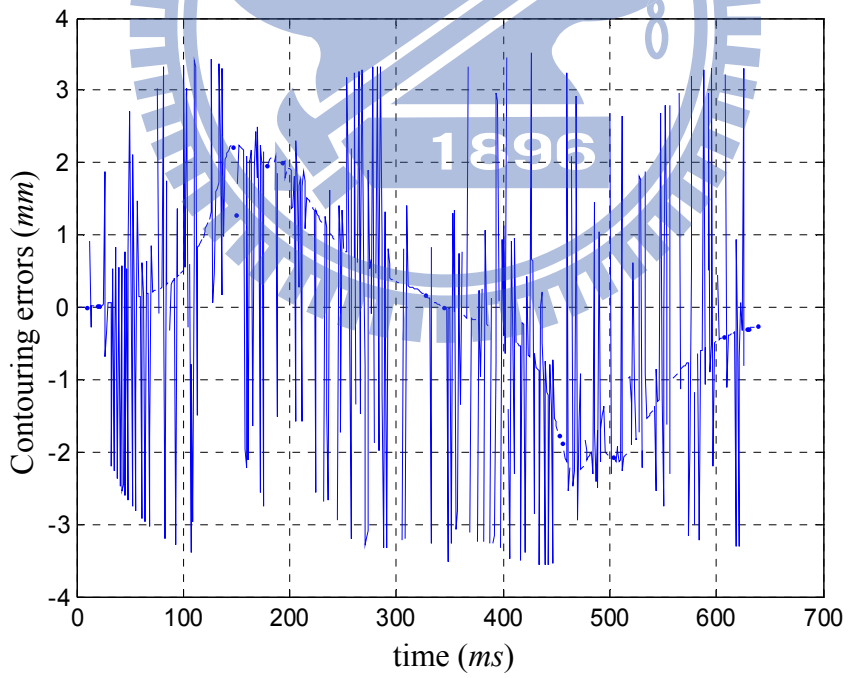


(a)

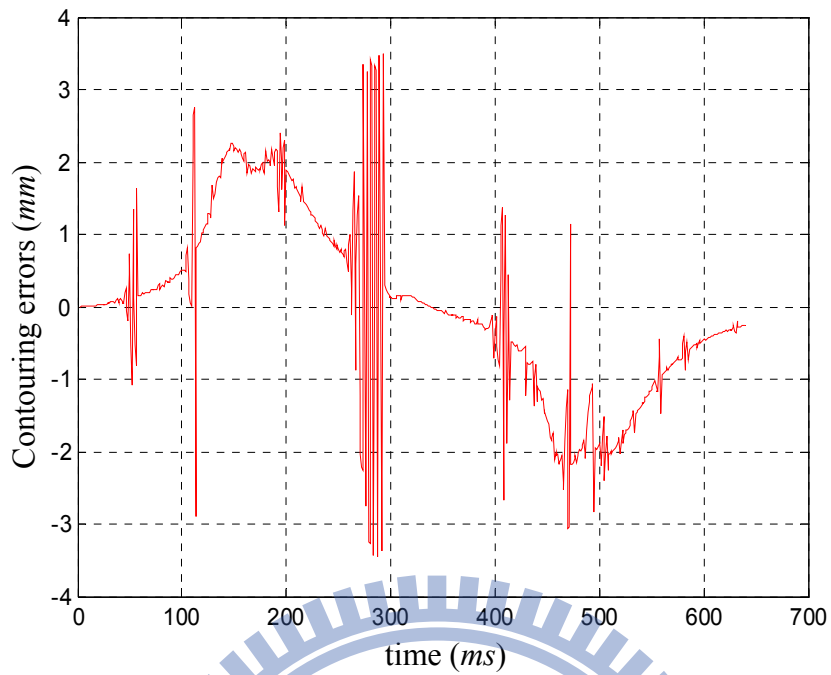


(b)

Fig. 4.10 Contouring errors with (a) the Taylor estimator (b) IME ($P_{D,D} = 0.2$)

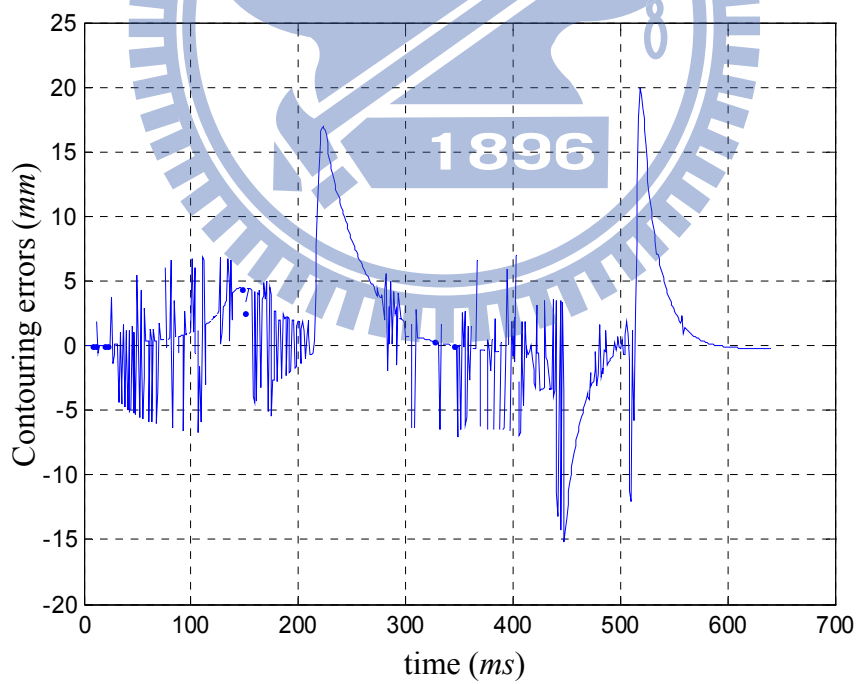


(a)



(b)

Fig. 4.11 Contouring errors with (a) the Taylor estimator and (b) IME ($P_{D,D} = 0.5$)



(a)

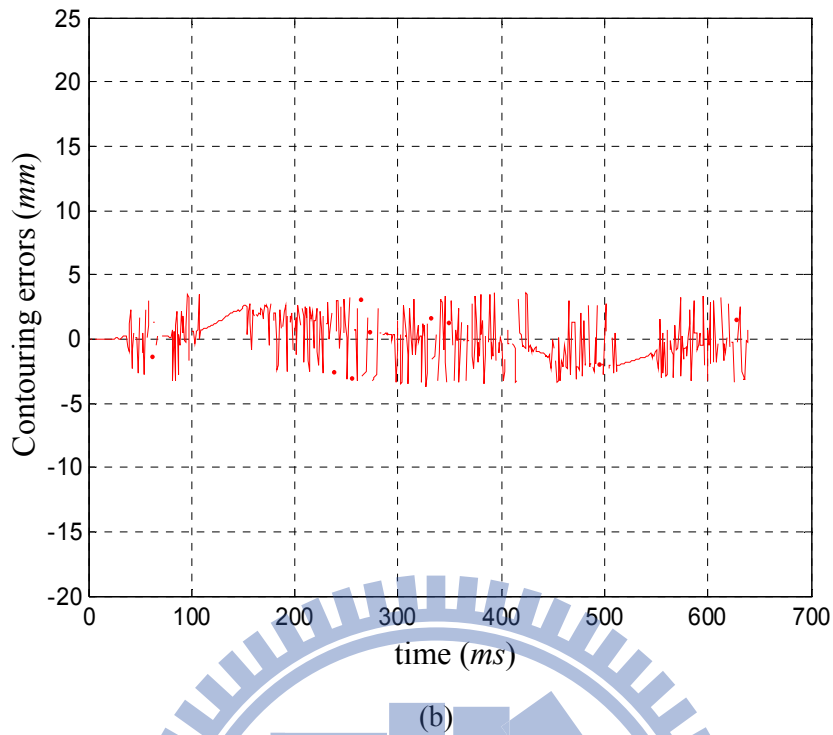


Fig. 4.12 Contouring errors with (a) the Taylor estimator and (b) IME ($P_{D,D} = 0.6$)

4.6 Experimental results

The proposed IME was applied to the CAN based two-axis AC servo motor control systems, as shown in Fig. 4.13. The butterfly NURBS profile for both the X-axis and Y-axis position amplitudes is 30 mm under the feed rate 3,000 mm/min, which is the same as in. Furthermore, $P_{D,D}$ is measured as 0.32 and 0.54, respectively, for the present CAN-bus implementation with the baud rate 1 M bit/sec under different sampling periods as 0.5 ms and 0.4 ms, respectively. The results indicate that increasing the sampling rate will result in more serious missing data due to the saturation of network bandwidth. Fig. 4.14 shows the contouring error when $P_{D,D} = 0.32$. The first-order differential results of the measured contouring error with less oscillation are also shown in Fig. 4.15. All results indicate that the proposed IME renders a more stable and reliable motion than the Taylor estimator. By observing the contouring error as shown in Fig. 4.16 with a more serious data dropout, the results also show that the proposed IME is more effective in reducing the asynchronization effect than the Taylor estimator in rendering a more accurate motion. Similar results

provided by the circular NURBS profile for the motion NCS obtained as shown in Figs. 4.17-4.18 also indicate the applicability of the proposed IME to different motion profiles.

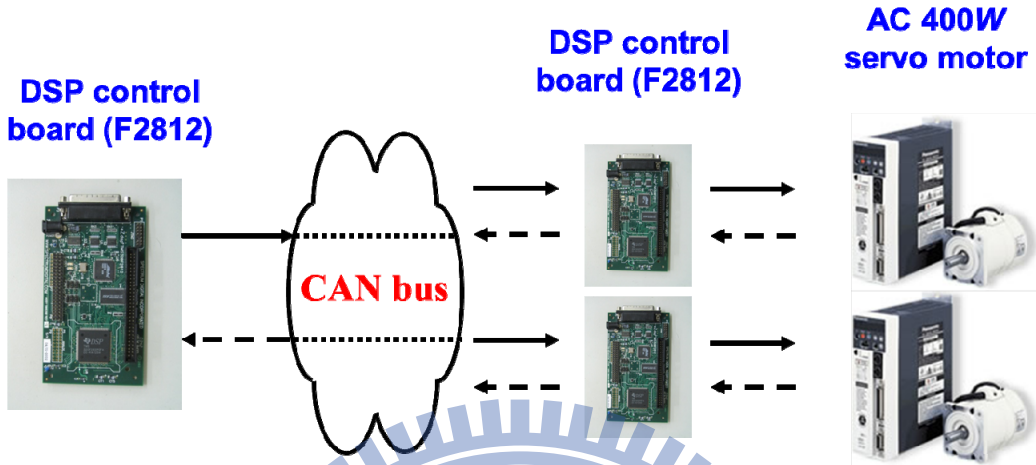
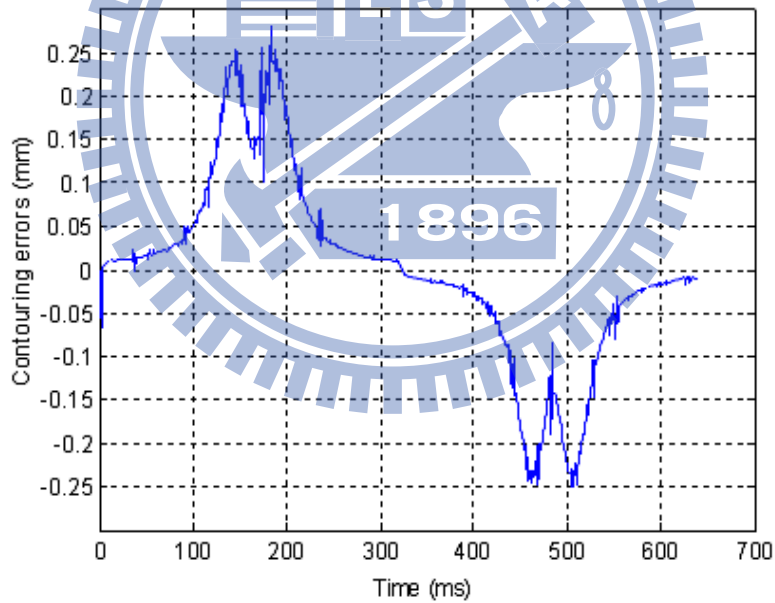
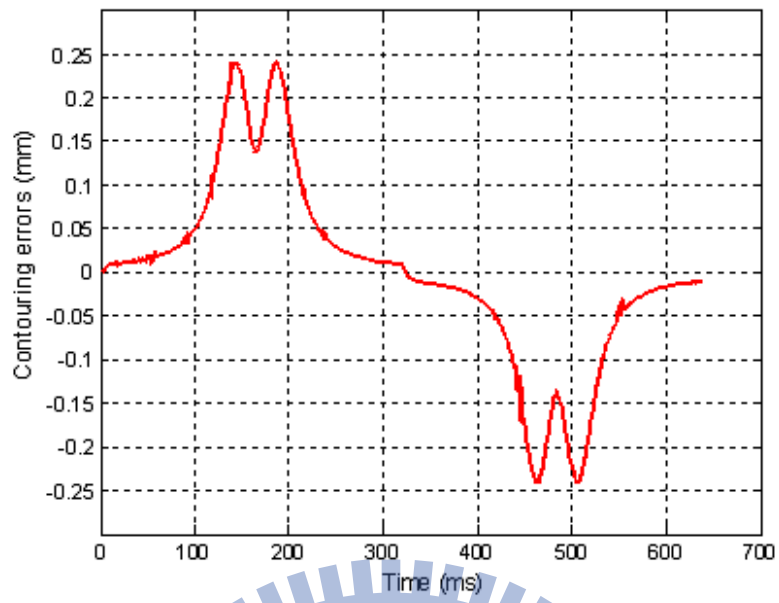


Fig. 4.13 Experimental setup



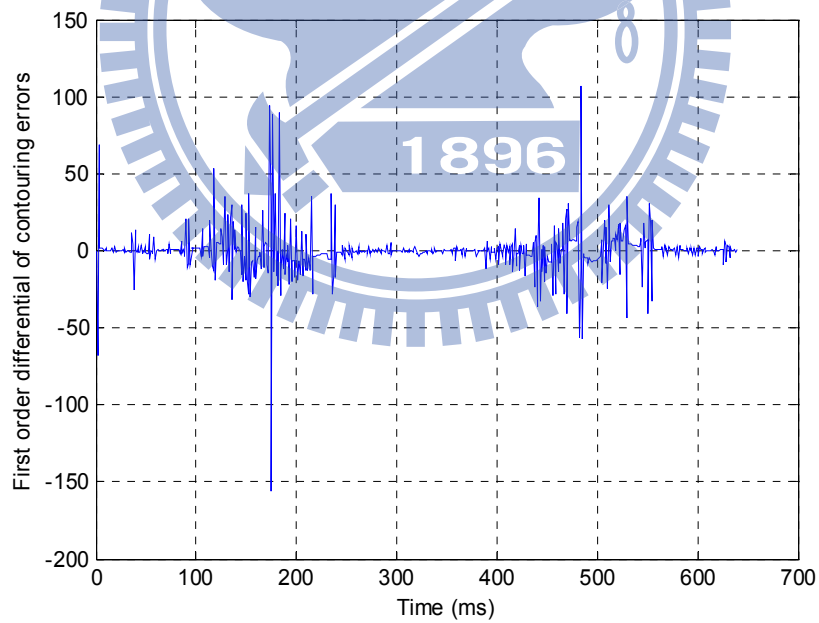
(a)



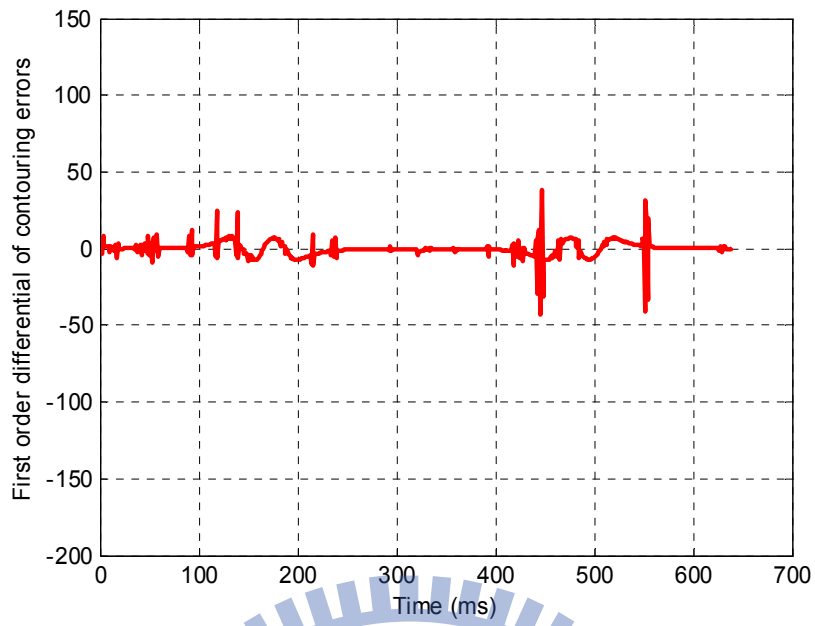
(b)

Fig. 4.14 Contouring errors with the sampling period = 0.5 ms

(a) Taylor estimator (b) IME ($P_{D,D} = 0.32$)

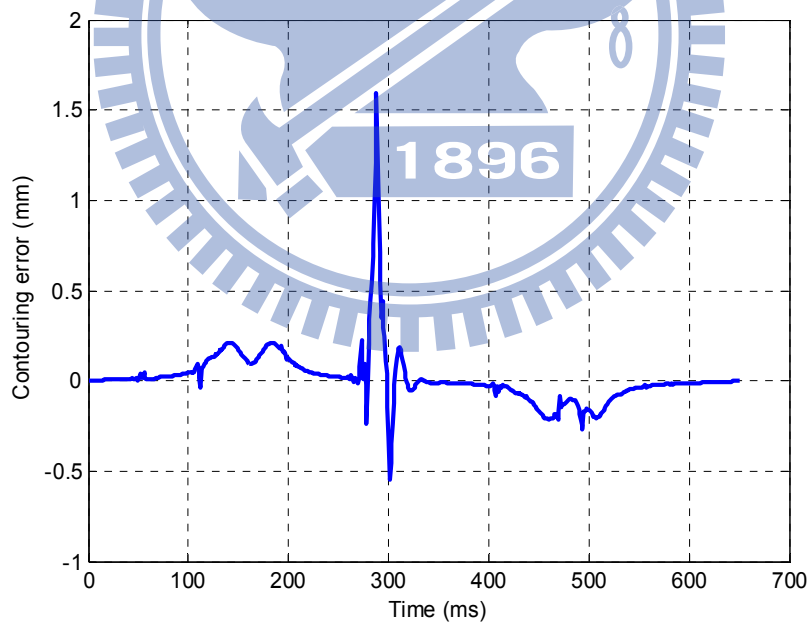


(a)

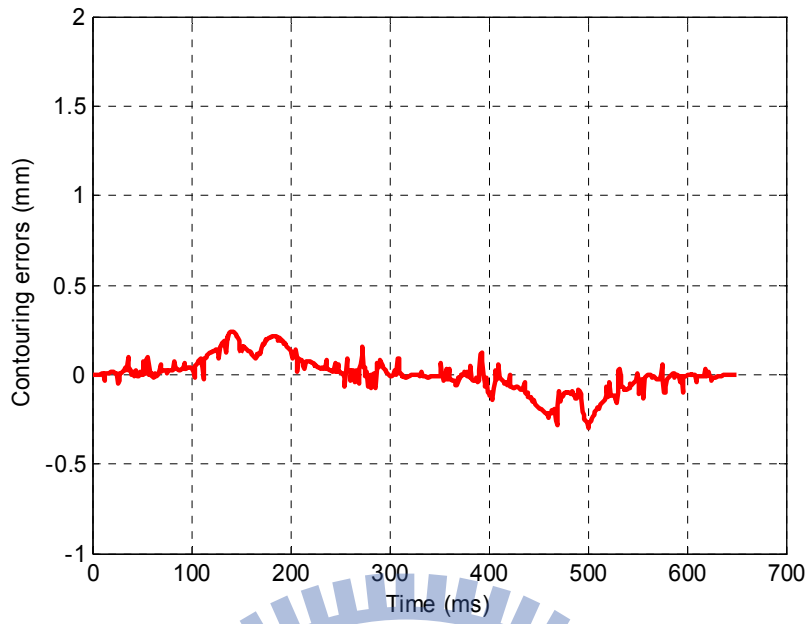


(b)

Fig. 4.15 First-order differential of contouring errors of
 (a) the Taylor estimator (b) IME ($P_{D,D} = 0.32$)



(a)

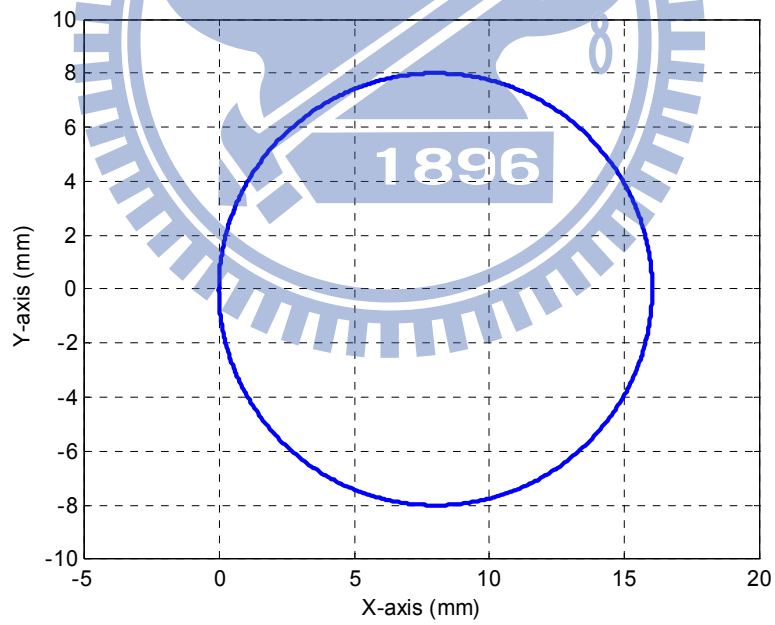


(b)

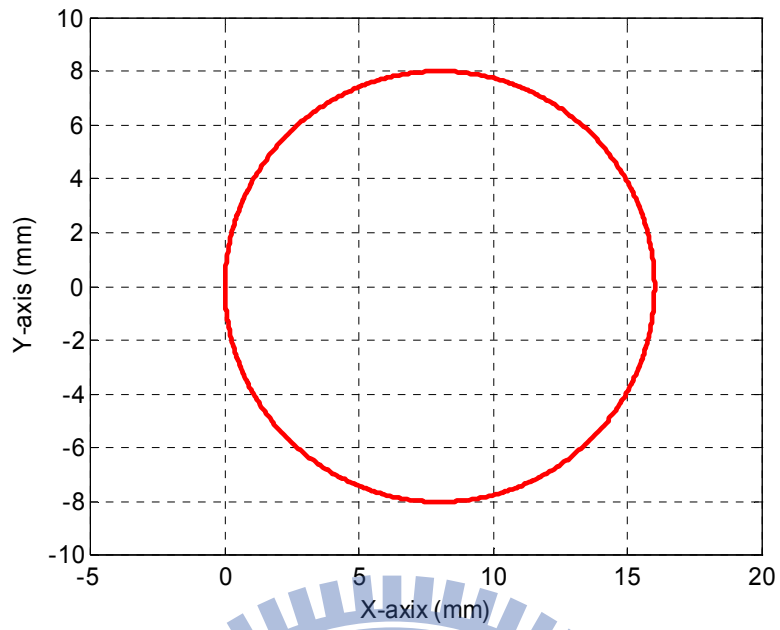
Fig. 4.16 Contouring errors with the sampling period = 0.4 ms

(a)

Taylor estimator (b) IME ($P_{D,D} = 0.54$)

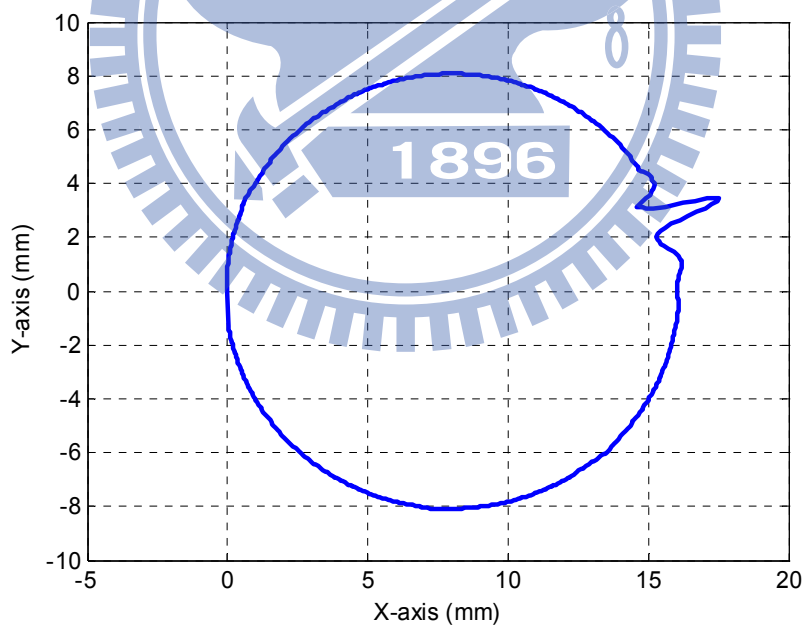


(a)

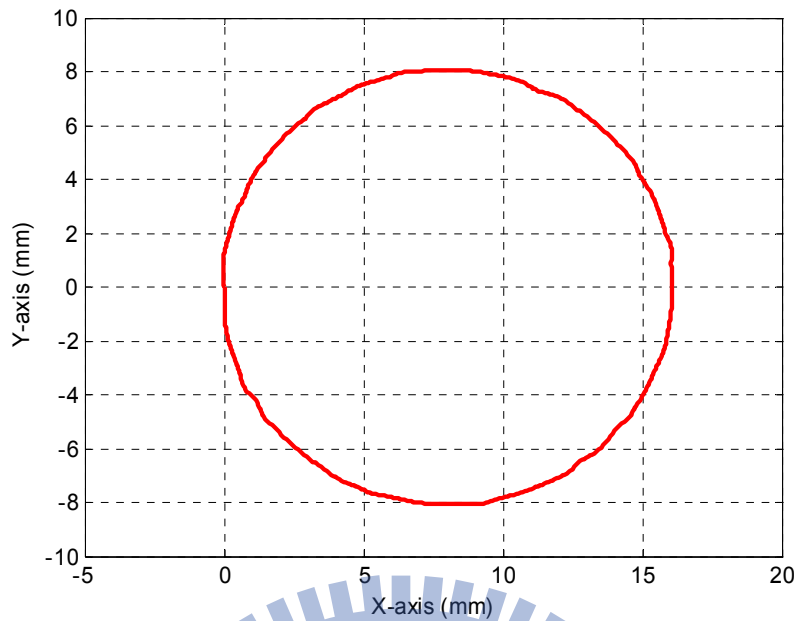


(b)

Fig. 4.17 Circle responses of (a) the Taylor estimator and (b) IME with the sampling period = 0.5 ms ($P_{D,D} = 0.32$)



(a)



(b)

Fig. 4.18 Circle responses of (a) Taylor estimator and (b) IME with the sampling period = 0.4 ms ($P_{D,D} = 0.54$)

4.7 Summary

Based on the on-line estimated transition probability $\hat{P}_{D,D}$ from Chapter 3, a suitable order of the least square estimators under different network communication conditions can be thus determined. Both simulation and experimental results have verified estimation performance by applying different orders of the least square estimators based on different communication quality $P_{D,D}$. Results indicate that applying the proposed IME, the missing commands can be properly estimated and the data dropout effect can be thus effectively reduced to improve contouring accuracy of motion NCS. Results are summarized as follows:

- (1) By applying tracking performance analysis, the switching laws of the proposed IME can be adopted based on estimated transition probability. Furthermore, the IME leads to the lowest contouring error under conditions of different data dropout distribution.
- (2) By applying the proposed IME to the CAN based two-axis AC servo motor control systems, contouring accuracy can be maintained well even under severe

missing commands. Moreover, the proposed IME renders the best performance as compared to the one-delay or the Taylor estimators in motion NCS.



Chapter 5

Stability analysis for NCS with the message estimator

Although physical dynamic systems operate in the continuous-time domain, the plant models in NCS are suitably represented in discrete-time domain with the sampling time over the network. This modeling paradigm is well suited for representing motion NCS as shown in Fig. 5.1. With digital computation and communication, digital control design applies the feedback sampled periodically for controller to provide actuator actions periodically. In addition, the network communication is constructed naturally with discrete-time analysis in our framework where transmissions are set to occur periodically. By modeling the continuous plant dynamics in discrete time through integration of the states over the sample period, the model of the NCS is shown in Fig. 5.1.

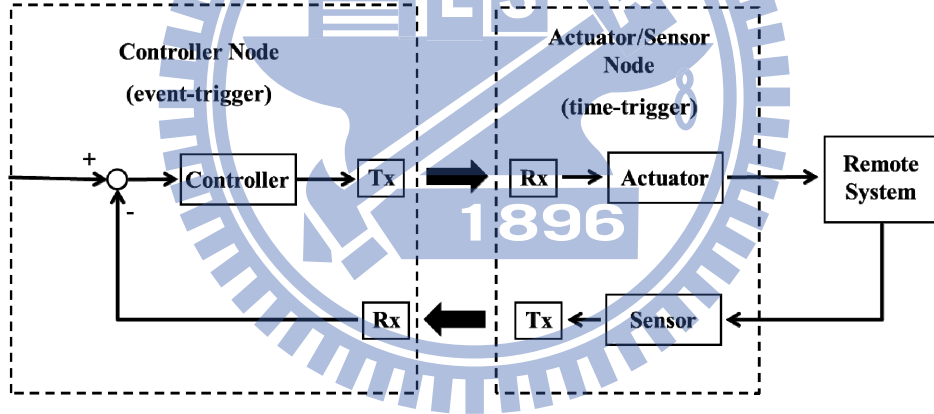


Fig. 5.1 Structure of the motion network control systems

Traditional state space representations of a continuous time plant given by

$$\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \quad (5.1)$$

Eq. (5.1) can be represented in discrete time with index $k \in Z = \{1, 2, 3, \dots\}$ as

$$\begin{aligned} x((k+1)T_s) &= Ax(kT_s) + Bu(kT_s) \\ y(kT_s) &= Cx(kT_s) + Du(kT_s) \end{aligned} \quad (5.2)$$

where

$$\begin{aligned} A &= e^{A_c T_s} \\ B &= \left(\int_0^{T_s} e^{A_c \lambda} d\lambda \right) B_c \end{aligned} \quad (5.3)$$

and $x(k) \in R^{n_x}$ is the state vector, $u(k) \in R^{n_u}$ is the control vector, and $y(k) \in R^{n_x}$ is the output vector. This conversion assumes that the continuous time input $u(t)$ is held constant for the duration of the sample period T_s .

Through this work, we usually assume the availability of full-state feedback from plants, and full-state feedback controllers are then obtained. The controller frequently used in this work is the discrete-time infinite time horizon linear quadratic (LQ) controller. The steady state LQ control problem can be posed (Gupta et al., 2005; Sinopoli et al., 2005; Ogata, 1995): For the linear discrete plant described by

$$x(k+1) = Ax(k) + Bu(k) \quad (5.4)$$

Determine the control

$$u(k) = -Kx(k) \quad (5.5)$$

to minimize the quadratic cost function

$$J(K) = \sum_{k=0}^{\infty} x(k)^T Q x(k) + u(k)^T R u(k) \quad (5.6)$$

where Q is positive semi-definite and R is positive definite. The solution to this problem is well-known (Ji et al., 1991) and it is computed by solving the steady-state Riccati equation for P as.

$$P = Q + A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A \quad (5.7)$$

The control gain K is then obtained as

$$K = (R + B^T P B)^{-1} B^T P A \quad (5.8)$$

The system models and the controller design for this approach are adopted throughout the remainder of this chapter.

5.1 Modeling NCS dynamics as a switched system

By using the hybrid control system framework with the time-triggered sensing/actuation, and the event-triggered communications shown in Fig. 5.1, the behavior of motion NCS can be naturally fitted into the format of a switched system with two discrete dynamic modes of operation (Kawka and Alleyne, 2009). The closed loop mode (*CL*) describes the system dynamics evolution over a sample period when the entire round trip communication of feedback and control was successful in the previous sample period. The open loop mode (*OL*) describes the system dynamics over a period when a new control packet was not received in the preceding sample period. This flow of information can be represented schematically by the diagrams as in Fig. 5.2, and the dropout messages occur mainly due to the increase of the time delay.

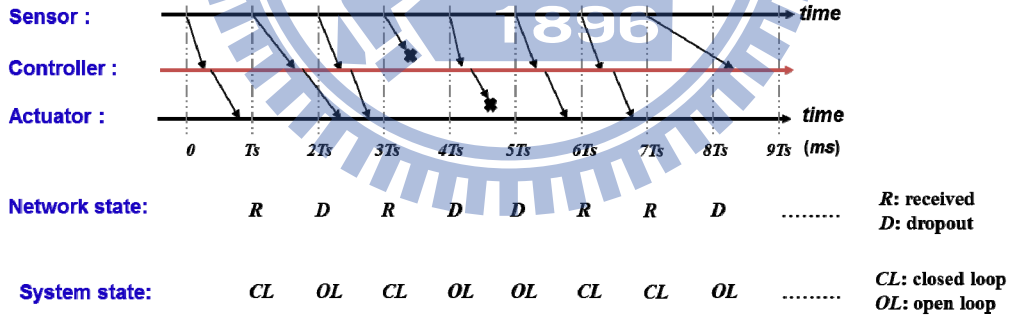


Fig. 5.2 Timing diagram of the network and system states

These two dynamics modes of operation can be described mathematically by

$$\begin{aligned}
 \text{Closed Loop mode} & \begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \\ u(k) = f(y(k-1), y(k-2), \dots, u(k-1), u(k-2), \dots) \end{cases} \quad (5.9) \\
 \text{Open Loop mode} & \begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \\ u(k) = f(y(k-m), y(k-m-1), \dots, u(k-m), u(k-m-1), \dots) \end{cases}
 \end{aligned}$$

(5.10)

where m characterizes the time instant of the last successfully received message.

In the *OL* mode, although the most recent feedback information $(y(k-1), u(k-1))$ is not available, information from the last successfully completed round trip transmission $(y(k-m), y(k-m-1), \dots, u(k-m), u(k-m-1), \dots)$ can be still used in the NCS applied by the actuator. That is why with this modeling strategy, the NCS naturally fits into the framework of a switched system with two discrete dynamics modes of operation, one for open loop and one for closed loop. For the switched system, three effective data dropout compensators are presented in this study as: (1) the one-delay compensator, (2) the Taylor estimator, and (3) the least square estimator.

5.2 NCS with one-delay compensator

The most basic strategy for handling the transient network dropout applies the one-delay compensator when no message is received by the actuator. For regulation stable systems, this choice would be safe during a consecutive communication loss period. However, applying incorrect control input to stable systems continuously during data dropout period still drives them into undesirable states. Above all, the one-delay compensator is relatively simple and stable for motion NCS. A mathematical description of the one-delay compensator at data dropout strategy for motion NCS is given by:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ e(k) &= r(k) - x(k) \end{aligned} \quad (5.11)$$

$$u(k) = \begin{cases} -Ke(k-1) & \text{if success (Closed loop)} \\ 0 & \text{if loss (Open loop)} \end{cases} \quad (5.12)$$

where $e(k) \in R^{n_x}$ is the error vector and $r(k) \in R^{n_u}$ is the reference vector. Eq. (5.11) and (5.12) can be expressed equivalently using the switched dynamics as

$$\tilde{x}(k+1) = A_{\sigma(k)}\tilde{x}(k) + B_{\sigma(k)}r(k), \quad \text{where} \quad (5.13)$$

$$\tilde{x}(k+1) = \begin{bmatrix} x(k+1) \\ e(k) \end{bmatrix}, \quad A_{OL} = \begin{bmatrix} A & 0 \\ -I & 0 \end{bmatrix}, \quad B_{OL} = \begin{bmatrix} 0 \\ I \end{bmatrix} \quad (5.14)$$

$$A_{CL} = \begin{bmatrix} A & BK \\ -I & 0 \end{bmatrix}, \quad B_{CL} = \begin{bmatrix} 0 \\ I \end{bmatrix}$$

and where \tilde{x} is the augmented plant state vector and account for the delay, r is the reference to be tracked, and $\sigma(k) \in \{OL, CL\}$ is the switched system index signal. If there is no reference command input and the objective of the closed-loop system is only regulation of the states to zero, the switched system can be represented in a simpler and more compact form given by

$$\tilde{x}(k+1) = A_{\sigma(k)}x(k) \quad \text{where} \quad (5.15)$$

$$\tilde{x}(k+1) = \begin{bmatrix} x(k+1) \\ u(k) \end{bmatrix}, \quad A_{OL} = \begin{bmatrix} A & 0 \\ -K & 0 \end{bmatrix} \quad (5.16)$$

$$A_{CL} = \begin{bmatrix} A & B \\ -K & 0 \end{bmatrix}$$

Although implementing the one-delay compensation results in more stable NCS than other strategies, the choice of a one-delay compensator also leads to poor performance of the motion NCS even during a short duration of dropout occurrences. Therefore, least square estimators with three different orders are adopted to improve performance during the short bursts of data dropouts as below.

5.3 NCS with the Least square estimator

In motion NCS, all the reference commands will not change dramatically and the actuating signals are thus in the relatively low frequency range. Therefore, the least square estimator can be adopted by using knowledge of past successful received control inputs to estimate the control actions during data dropouts. This strategy is simple to implement and does not require information of the plant model. Three least square estimators with different orders are adopted in this dissertation for different

data dropout situations. A mathematical description of these least square estimators can be pre-calculated in Chapter 4 for motion NCS as

$$LSE(5,3) = 3.2u(k-1) - 2.8u(k-2) - 0.8u(k-3) + 2.2u(k-4) - 0.8u(k-5) \quad (5.17)$$

$$LSE(3,2) = 3u(k-1) - 3u(k-2) + u(k-3) \quad (5.18)$$

$$LSE(2,1) = 2u(k-1) - u(k-2) \quad (5.19)$$

- NCS with $LSE(2,1)$

The state-feedback control law with $LSE(2,1)$ strategy without reference tracking as

$$u(k) = \begin{cases} -Kx(k-1) & \text{if success (Closed loop)} \\ 2u(k-1) - u(k-2) & \text{if loss (Open loop)} \end{cases} \quad (5.20)$$

Eq. (5.15) and (5.20) can be expressed equivalently using switched dynamics as

$$\tilde{x}(k+1) = A_{\sigma(k)} \tilde{x}(k), \quad \text{where}$$

$$\tilde{x}(k) = \begin{bmatrix} x(k) \\ x(k-1) \\ u(k) \\ u(k-1) \\ u(k-2) \end{bmatrix}, \quad A_{OL} = \begin{bmatrix} A & 0 & 0 & 2B & -B \\ I & 0 & 0 & 0 & 0 \\ -K & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & I & 0 \end{bmatrix} \quad (5.21)$$

$$A_{CL} = \begin{bmatrix} A & 0 & B & 0 & 0 \\ I & 0 & 0 & 0 & 0 \\ -K & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & I & 0 \end{bmatrix}$$

- NCS with $LSE(3,2)$

The $LSE(3,2)$ strategy is described with no reference tracking by Eq. (5.15) with control law as

$$u(k) = \begin{cases} -Kx(k-1) & \text{if success (Closed loop)} \\ 3u(k-1) - 3u(k-2) + u(k-3) & \text{if loss (Open loop)} \end{cases} \quad (5.22)$$

Eq. (5.15) and (5.22) can be expressed equivalently using switched dynamics as

$$\tilde{x}(k+1) = A_{\sigma(k)} \tilde{x}(k), \quad \text{where}$$

$$\tilde{x}(k) = \begin{bmatrix} x(k) \\ x(k-1) \\ u(k) \\ u(k-1) \\ u(k-2) \\ u(k-3) \end{bmatrix}, \quad A_{OL} = \begin{bmatrix} A & 0 & 0 & 3B & -3B & B \\ I & 0 & 0 & 0 & 0 & 0 \\ -K & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 \end{bmatrix} \quad (5.23)$$

$$A_{CL} = \begin{bmatrix} A & 0 & B & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 & 0 \\ -K & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 \end{bmatrix}$$

- NCS with *LSE(5,3)*

The *LSE(5,3)* strategy is described for regulation by Eq. (5.15) with the state-feedback control law as

$$u(k) = \begin{cases} -Kx(k-1) & \text{if success (Closed loop)} \\ 3.2u(k-1) - 2.8u(k-2) - 0.8u(k-3) + 2.2u(k-4) - 0.8u(k-5) & \text{if loss (Open loop)} \end{cases} \quad (5.24)$$

Eq. (5.15) and (5.24) can be expressed equivalently using the switched dynamics as

$$\tilde{x}(k+1) = A_{\sigma(k)}\tilde{x}(k), \quad \text{where}$$

$$\tilde{x}(k) = \begin{bmatrix} x(k) \\ x(k-1) \\ u(k) \\ u(k-1) \\ u(k-2) \\ u(k-3) \\ u(k-4) \\ u(k-5) \end{bmatrix}, \quad A_{ol} = \begin{bmatrix} A & 0 & 0 & 3.2B & -2.8B & -0.8 & 2.2B & -0.8B \\ I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -K & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I & 0 \end{bmatrix} \quad (5.25)$$

$$A_{cl} = \begin{bmatrix} A & 0 & B & 0 & 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -K & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & I & 0 \end{bmatrix}$$

5.4 SMS Stability analysis of NCS

When motion NCS is modeled using a switched system whose transitions are described by a Markov chain, existing stability results for stochastically switched systems can be thus applied. Several probabilistic types of stability may be deduced for different network conditions described by transition probabilities $\rho_{D,R}$ and $\rho_{R,D}$, as in Chapter 3. Consider a general discrete-time Markovian Jump Linear Systems (MJLS) given by

$$x(k+1) = A_{\sigma(k)}x(k) + B_{\sigma(k)}u(k) \quad (5.26)$$

where $x(k) \in R^{n_x}$ is the state vector and $u(k) \in R^{n_u}$ is the control vector, A and B are real valued matrices of appropriate dimensions that are function of $\sigma(k)$, a time-homogeneous Markov chain takes values in a finites set $\{1,2,\dots,N\}$. Several types of stability test are examined in (Ji et al., 1991) to show equivalence between mean-square stability, stochastic stability, and exponential mean square stability for MJLS.

- Mean-square stability: $\forall(x_0, \sigma_0), \lim_{k \rightarrow \infty} E\|x(k)\|^2 | x_0, \sigma_0] = 0$ (5.27)

- Stochastic stability: $\forall(x_0, \sigma_0), \sum_{k=0}^{\infty} E\|x(k)\|^2 | x_0, \sigma_0] < \infty$ (5.28)

- Exponential mean square stability:

$$\forall(x_0, \sigma_0) \exists \text{ constant } 0 < \alpha < 1, \beta > 0$$

$$\text{s.t. } \forall(x_0, \sigma_0), E\|x(k)\|^2 | x_0, \sigma_0] < \beta \alpha^k \|x_0\|^2 \quad (5.29)$$

where α and β are independent of x_0 and σ_0 . These equivalent notions are referred as second moment stability (SMS) (Ji et al., 1991).

- Almost sure stability:

A fourth type of stochastic stability called almost sure stability is also considered frequently in the literature (Bolzern et al., 2006).

$$\forall(x_0, \sigma_0), \lim_{k \rightarrow \infty} E\|x(k)\|^2 | x_0, \sigma_0] = 0 \quad (5.30)$$

Almost sure stability is a weaker definition of stability that is implied by the second moment stability of MJLS. However, SMS stability is not necessary for almost sure stability. To check SMS stability in general for a system with N different switched Markov dynamics, several necessary and sufficient conditions have been developed including checking if there exist $\{G_i > 0\}$ such that (Ji et al., 1991).

$$A_i^T \left[\sum_{j=1}^N \rho_{ij} G_j \right] A_i - G_i < 0, \quad i = 1, 2, \dots, N \quad (5.31)$$

5.4.1 Probability regions of NCS stability

One approach to obtain a solution to Eq. (5.31) is to use the Kronecker product (Graham, 1981). Horn and Johnson (1991) proposes matrix analysis for the Kronecker products. Let $A = (a_{ij})_{m \times n}$ be a real or complex matrix and define the

linear operator $\text{vec}(\cdot)$ by

$$\text{vec}(A) = [a_{11}, a_{21}, \dots, a_{m1}, a_{12}, a_{22}, \dots, a_{m2}, \dots, a_{1n}, \dots, a_{mn}]^T$$

The following lemma will be used to develop simpler SMS stability criteria as:

Lemma 5.1 (Horn and Johnson, 1991):

(a). $\text{vec}(AX) = (I \otimes A)\text{vec}(X)$, $\text{vec}(AXB) = (B^T \otimes A)\text{vec}(X)$.

(b). If $A_1XB_1 + \dots + A_kXB_k = C$, then

$$\left[B_1^T \otimes A_1 + \dots + B_k^T \otimes A_k \right] \text{vec}(X) = \text{vec}(C).$$

(c). $\text{vec}(AX + YB) = (I \otimes A)\text{vec}(X) + (B^T \otimes I)\text{vec}(Y)$.

By adopting Lemma 5.1, simpler second moment stability criteria can be further obtained.

Corollary 5.1 (Costa and Fragoso, 1993; Fang and Loparo, 2002): Suppose the Markov chain network model is finite state and the probability transition matrix converges to a stationary matrix, SMS stability is guaranteed if an eigenvalue condition formed from the network parameters and switched dynamics mode is satisfied as:

$$p\left(\left(P^T \otimes I_{n^2}\right) \left[\text{diag}(A_i \otimes A_i)_{i=1,2,\dots,N} \right]\right) < 1 \quad (5.32)$$

where p is spectral radius, $P = [p_{ij}]$ is the Markov transition probability matrix, and \otimes represents the Kronecker product operation.

For the situation considered in this work, $N = 2$ corresponding to *OL* and *CL* two system states. By testing either of the conditions in Eq. (5.31) or (5.32) for various transition probabilities, $\rho_{D,R} \in [0,1]$ and $\rho_{R,D} \in [0,1]$, regions of stability and instability can be successfully found.

5.4.2 Example

One particular plant model examined in this motion NCS framework is the rotating base inverted pendulum discretized for the 200 Hz sampling (Kawka and Alleyne, 2009). The state dynamics and LQ control used in the linear switched

systems models obtained as :

$$A = \begin{bmatrix} 1.008 & 5.0001 \times 10^{-3} & 0 & 0 \\ 0.31635 & 1.0008 & 0 & 0 \\ -4.1645 \times 10^{-4} & -6.9404 \times 10^{-7} & 1 & 5.0000 \times 10^{-3} \\ -0.16602 & -4.1645 \times 10^{-4} & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 6.5098 \times 10^{-3} \\ -2.6043 \\ 0.010052 \\ 4.0210 \end{bmatrix} \quad (5.33)$$

$$K = -[2.0941 \quad 0.37835 \quad 0.11654 \quad 0.092198] \quad (5.34)$$

When the eigenvalue condition of Eq. (5.32) is applied to the pendulum model with static state feedback control with one-delay compensator during data dropout given by Eq. (5.15), (5.16), (5.33) and (5.34), the resulting theoretical stability region in the space of network conditions can be plotted as shown in Fig. 5.3.

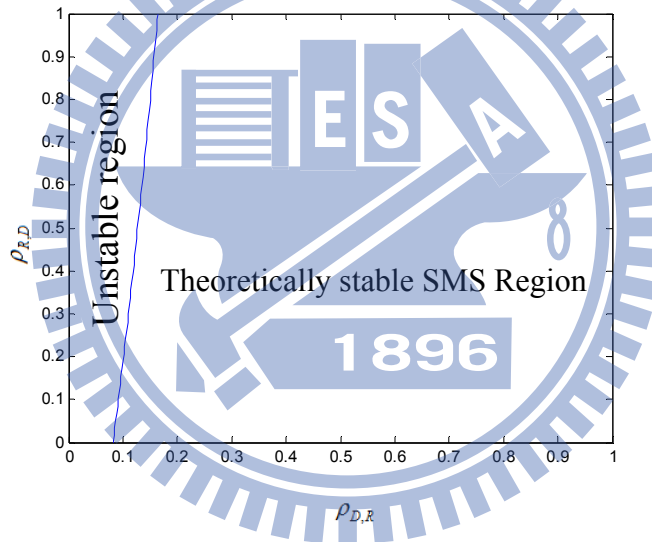


Fig. 5.3 Theoretical SMS stability regions for one-delay compensator

Furthermore, LSE(2,1) LSE(3,2) and LSE(5,3) are adopted to check their SMS stability region, as shown in Fig. 5.4, 5.5 and 5.6, respectively. From these figures, the transition probability $\rho_{D,R}$ is the domination factor to affect SMS stability region. For various data dropout compensators, their suitable network conditions based on $\rho_{D,R}$ have been determined to ensure stability of NCS with the *LSE* in different orders, as shown in Fig. 5.7.

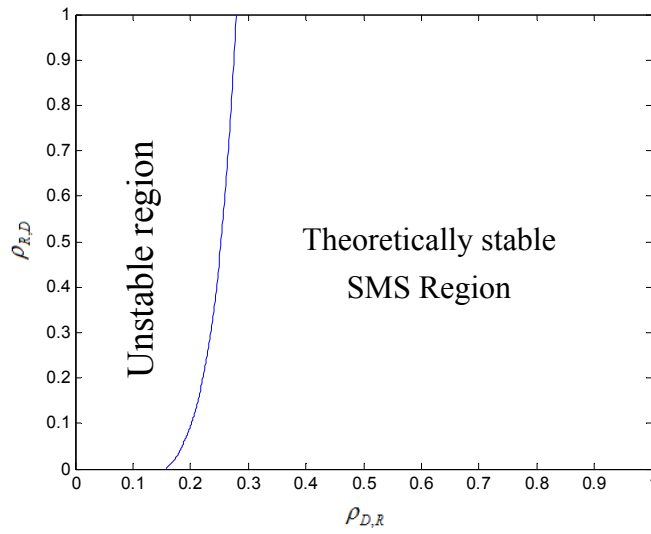


Fig. 5.4 Theoretical SMS stability regions for LSE(2,1)

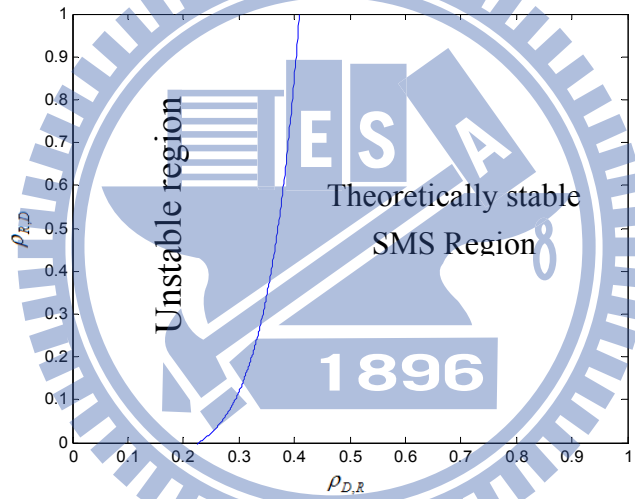


Fig. 5.5 Theoretical SMS stability regions for LSE(3,2)

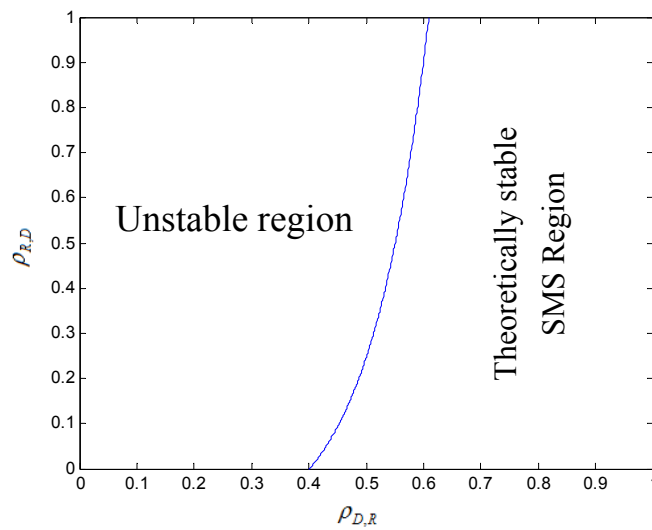


Fig. 5.6 Theoretical SMS stability regions for LSE(3,2)

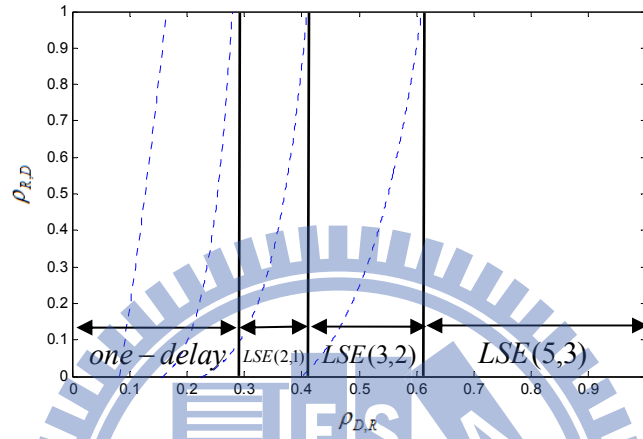


Fig. 5.7 Theoretical SMS stability regions for different LSE

Therefore, the intelligent message estimator (IME) is proposed for motion NCS based on real-time local transition probability. All switching laws are shown in the following:

$$\left\{ \begin{array}{ll} 0.6 \leq \hat{P}_{D,R}(K) \leq 1, & LSE(5,3) \text{ is adopted} \\ 0.45 < \hat{P}_{D,R}(K) \leq 0.6, & LSE(3,2) \text{ is adopted} \\ 0.3 < \hat{P}_{D,R}(K) \leq 0.45, & LSE(2,1) \text{ is adopted} \\ 0 < \hat{P}_{D,R}(K) \leq 0.3, & 1\text{-delay estimator is adopted} \end{array} \right. \quad (5.35)$$

5.5 Discussion

The dynamic model of the DYNA CNC machine tool obtained from the system identification procedure was adopted as (Hsieh and Hsu, 2008)

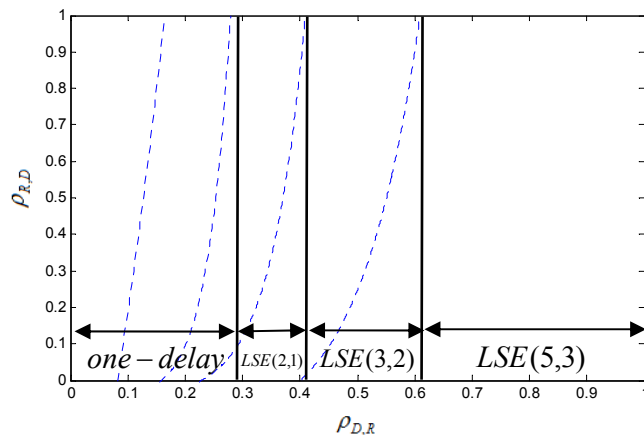
$$P(z) = \frac{0.30554z^{-2} - 0.023766z^{-3} + 0.11104z^{-4} + 0.028834z^{-5} - 0.012243z^{-6} + 0.020811z^{-7} - 0.089113z^{-8}}{1 - 0.70669z^{-1} + 0.1934z^{-2} - 0.15112z^{-3} - 0.02566z^{-4} + 0.028011z^{-5}}$$

And the state dynamics and pole placement design used in the linear switched systems models with all poles are about 0.5 obtained as

$$A = \begin{bmatrix} 0.707 & -0.193 & 0.151 & 0.026 & -0.028 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$K = -[-3.293 \quad 6.806 \quad -6.849 \quad 4.400 \quad -1.778 \quad 0.438 \quad -0.063 \quad 0.004]$$

By using Eq. (5.32), SMS stability regions with different orders of the least square estimator also can be found, as shown in Fig. 5.8(b). By comparing stability regions of the rotating base inverted pendulum discretized is as shown in the Fig. 5.8. The suitable switching region of IME for both plants are listed in Table 5.1. Results indicate that the stability regions of the DYNA CNC machine increases. This reason is that the CNC machine possesses better stability than the rotating base inverted pendulum. To obtain a motion NCS with guaranteed stability, the regions of an unstable plant lie the rotating base inverted pendulum is adopted to construct a more conservative design of NCS.



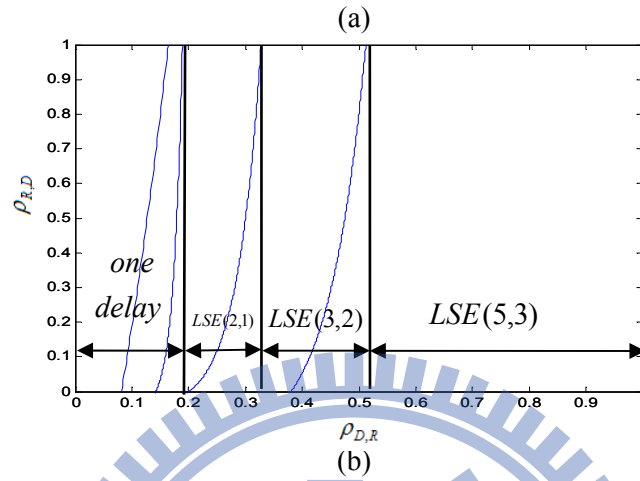
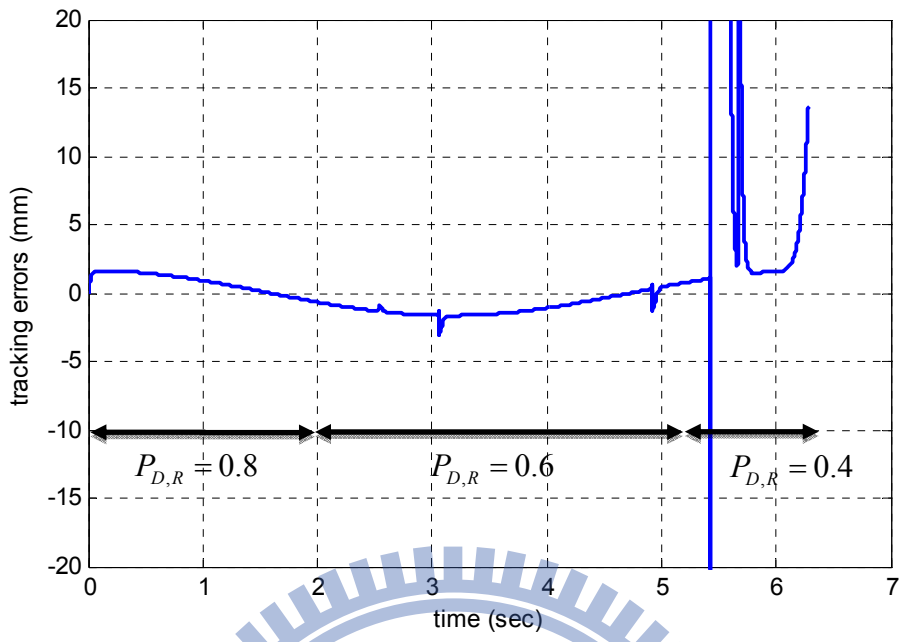


Fig. 5.8 Theoretical SMS stability regions for different LSE (a) the rotating base inverted pendulum, (b) the CNC machine

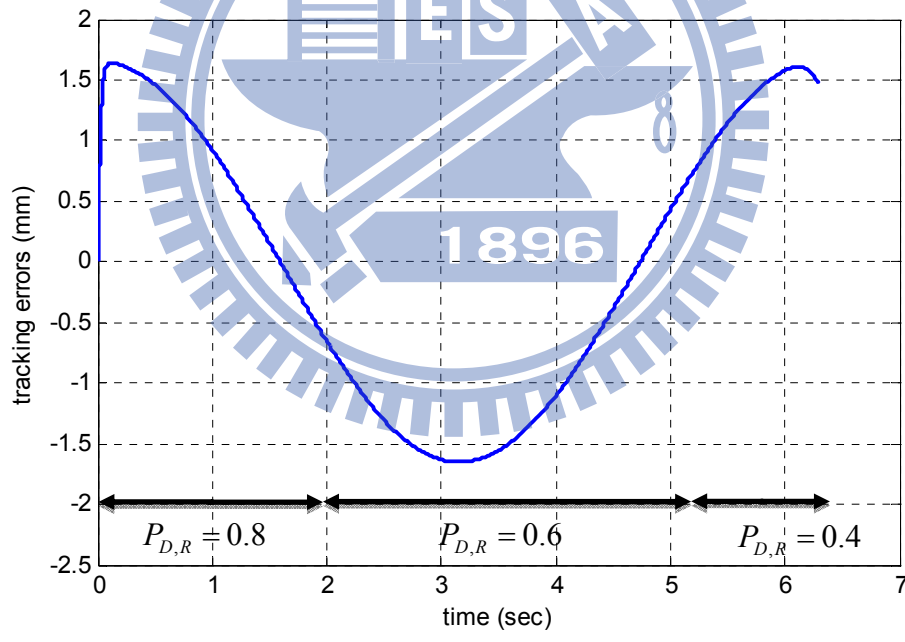
Table 5.3 Evaluation for SMS stability regions of different LSE

Estimator	Plant	inverted pendulum	CNC machine
LSE(5,3)		$0.6 \leq \rho_{D,R} \leq 1$	$0.5 \leq \rho_{D,R} \leq 1$
LSE(3,2)		$0.4 \leq \rho_{D,R} \leq 1$	$0.32 \leq \rho_{D,R} \leq 1$
LSE(2,1)		$0.3 \leq \rho_{D,R} \leq 1$	$0.2 \leq \rho_{D,R} \leq 1$
One-delay estimator		$0.15 \leq \rho_{D,R} \leq 1$	$0.15 \leq \rho_{D,R} \leq 1$

By applying switching law Eq. 5.35, tracking errors with Taylor estimator and IME are shown in the Fig. 5.9(a) and (b), respectively. From these figures, IME can lead to both better tracking accuracy and stability in different transition probabilities.



(a)



(b)

Fig. 5.9 Tracking errors by applying (a) Taylor estimator (b) IME

5.6 Summary

In this chapter, the network infrastructure is introduced in control loop of the servo motor system. The time delay and data dropout simultaneously happens to

degrade the control performance. Simulation results has rendered satisfactory performance to verify feasibility and effectiveness of the proposed IME. Several critical summaries are presented as follows:

- (1) Motion NCS dynamics are successfully modeled as a switched system. Furthermore, Motion NCS applying least square estimators with different order least square estimators are also modeled as switched systems.
- (2) By using SMS stability condition, network region of stability with different least square estimators are obtained. Furthermore, the switching laws of IME can be also determined based on network region of stability.
- (3) In simulation results, the motion NCS can ensure stability in different data dropout distributions by applying IME.



Chapter 6

Integration of delay and dropout compensation for NCS

In real-time motion NCS, most of data dropouts are caused by the networked-induced time delay as it is longer than the system sampling period. Practically, network signals in TCP do not cause data dropout but it may increase networked time delay, as shown in the Fig. 5.9. However, if the motion NCS is constructed in a time-triggered system with a fixed sampling frequency, the increase of the time delay will cause data dropout unavoidably. From this figure, the significant increase of time delay will lead to consecutive data dropout as well as the increase of the probability of $\rho_{D,D}$. On the other hand, the mild increment of the networked time delay will also increase the probability of $\rho_{D,R}$. Furthermore, when networked time delay decreases, more than one data can be received during one sampling period, as called the message rejection because only the last data is adopted. Although system states (R , D and RJ) can represent variation of networked time delays, the base-line of the networked time delay (τ_d) is avoidable and will affect control performance of motion NCS.

6.1 The perfect delay compensation

For network time delay, many previous researches can effectively compensate its effects, for example adaptive smith predictor (Lai and Hsu, 2010), communication disturbance observer (CDOB) (Natori et al., 2008), scattering transformation (Matiakis et al., 2009) and so on. Although these methods are useful for compensating effects of networked time delay, the system dynamics model or networked time delay must be known.

Matlab. The dynamic model of the DYNA CNC machine tool obtained from the system identification procedure was adopted as

$$P(z) = \frac{0.30554z^{-2} - 0.023766z^{-3} + 0.11104z^{-4} + 0.028834z^{-5} - 0.012243z^{-6} + 0.020811z^{-7} - 0.089113z^{-8}}{1 - 0.70669z^{-1} + 0.1934z^{-2} - 0.15112z^{-3} - 0.02566z^{-4} + 0.028011z^{-5}}$$

The command signal is a sinusoid wave as shown in the Fig. 2.6 (a). Fig. 6.3 represents the tracking errors with $\tau_d = 10 \text{ ms}$.

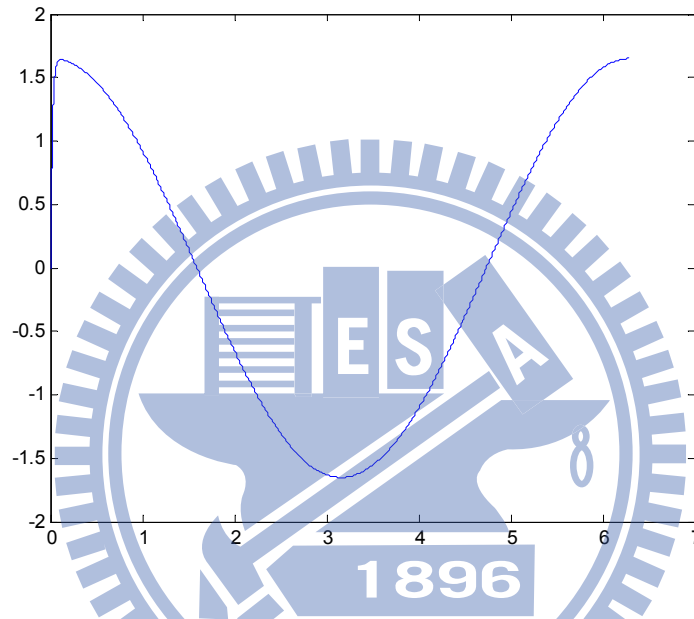


Fig. 6.3 The tracking error without delay compensation as $\tau_d = 10 \text{ ms}$

6.2 Case of invariant dropout distribution and the time delay

When the data dropout signals with $\rho_{D,R} = 0.8$ are introduced, the tracking errors increases and becomes not smooth without IME and PDC, as shown in the Fig. 6.4. Furthermore, Fig. 6.5 and 6.6 respectively represents the tracking errors by applying IME and PDC. Simulation results indicate that IME indeed compensate effects of data dropout but the PDC can not compensate effects of the networked delay when the data dropouts happen. By combining IME with PDC, effects of data dropout and networked time delay can be simultaneously and effectively eliminated to make real-time motion NCS return the original situation without data dropout and networked time delay, as shown in the Fig. 6.7.

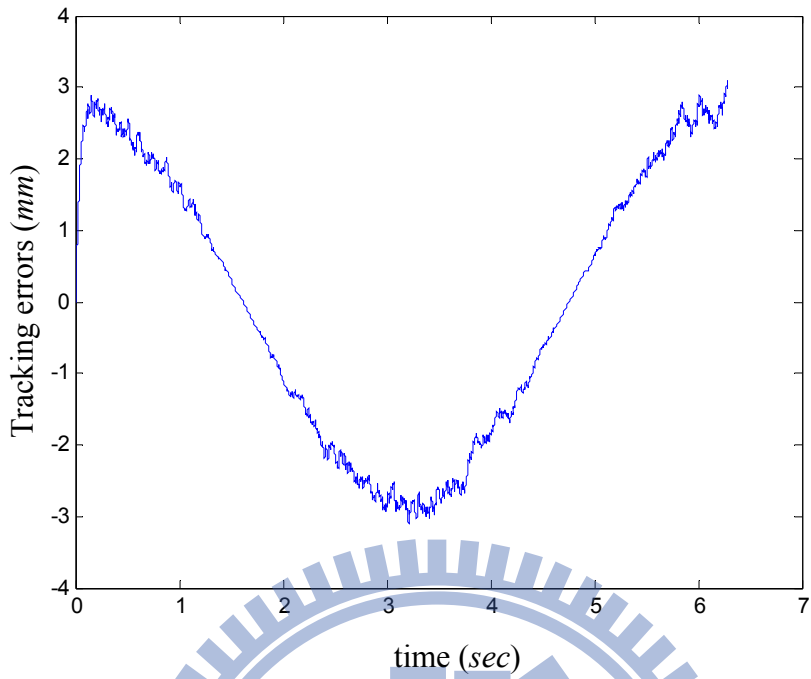


Fig. 6.4 The tracking error without compensation as both dropout and delay induced
 ($\rho_{D,R} = 0.8, \tau_d = 10 \text{ ms}$)

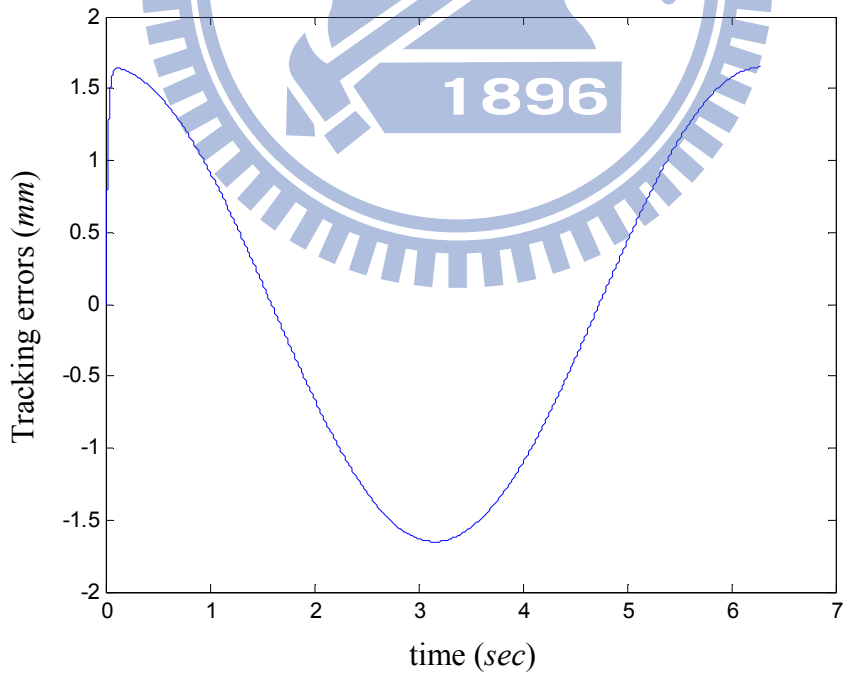


Fig. 6.5 The tracking error by applying IME with $\rho_{D,R} = 0.8$ and $\tau_d = 10 \text{ ms}$

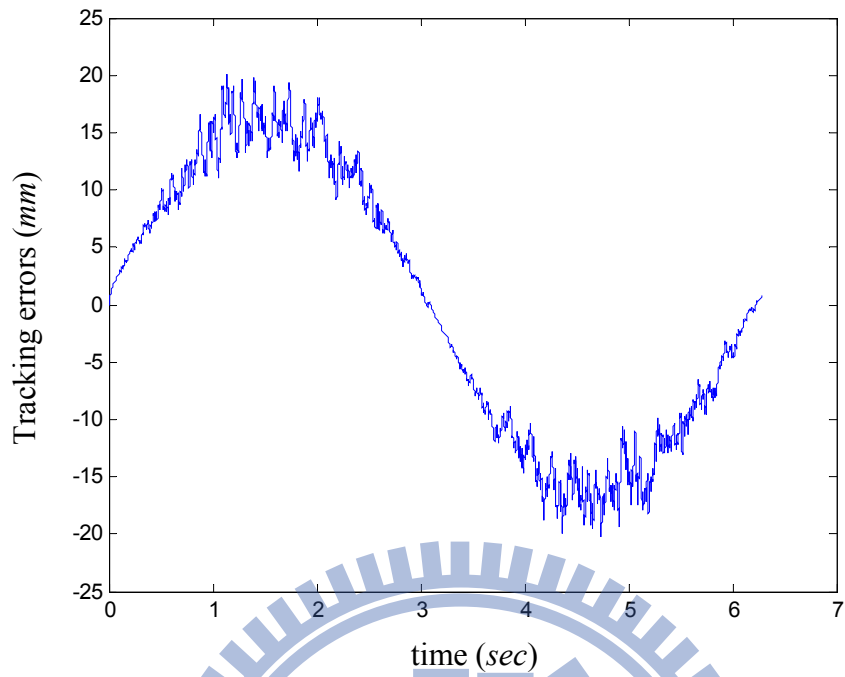
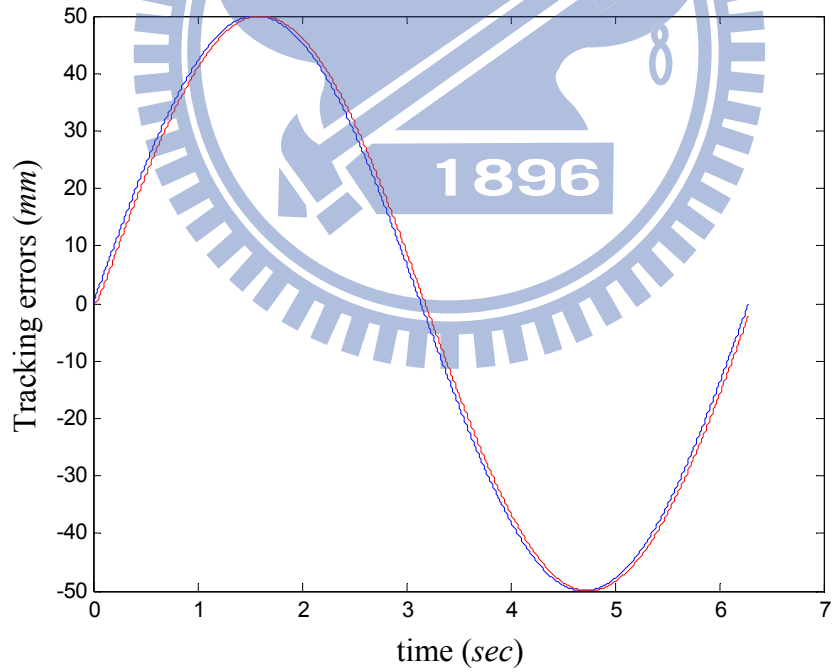


Fig. 6.6 The tracking error by applying PDC with $\rho_{D,R} = 0.8$ and $\tau_d = 10 \text{ ms}$



(a)

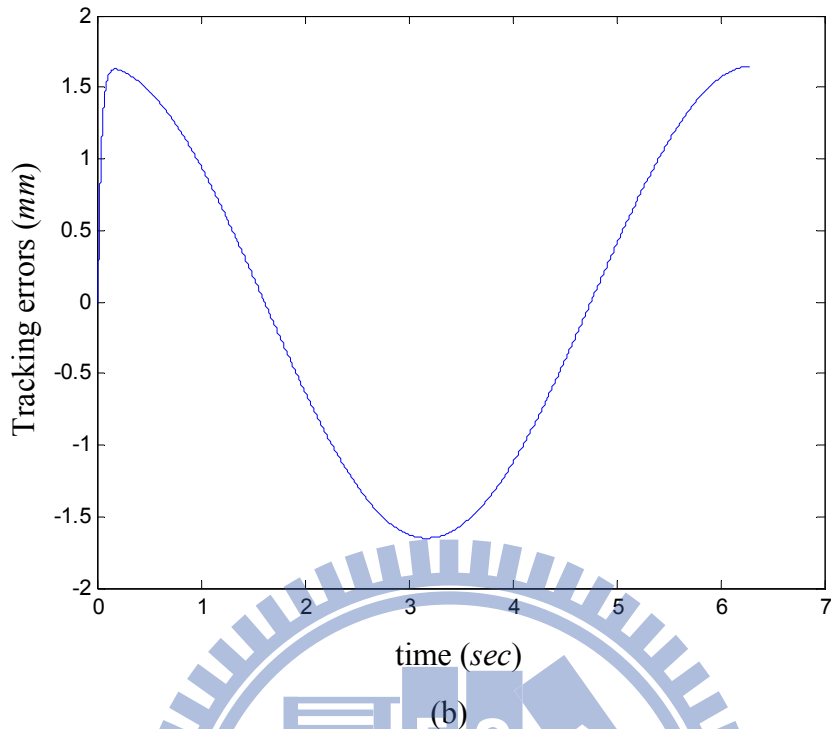


Fig. 6.7 Applying both IME and PDC with $\rho_{D,R}=0.8$ and $\tau_d=10\text{ ms}$

(a) the system response (b) the tracking error

Further, the longer networked time delay ($\tau_d=50\text{ ms}$) is introduced. Fig. 6.8 represents the tracking errors with $\tau_d=50\text{ ms}$ and the system performance happens serious shake because of networked time delay is too long. When the data dropout signals with $\rho_{D,R}=0.8$ are introduced, the tracking errors without IME and PDC are shown in the Fig. 6.9. Furthermore, Fig. 6.10 and 6.11 respectively represents the tracking errors by applying IME and PDC. Simulation results also indicate that IME indeed compensate effects of data dropout but the PDC cannot compensate effects of the networked delay when the data dropout happen. By combining IME with PDC, effects of data dropout and networked time delay can be simultaneously and effectively eliminated even system response is not acceptable because of too longer networked time delay, as shown in the Fig. 6.12.

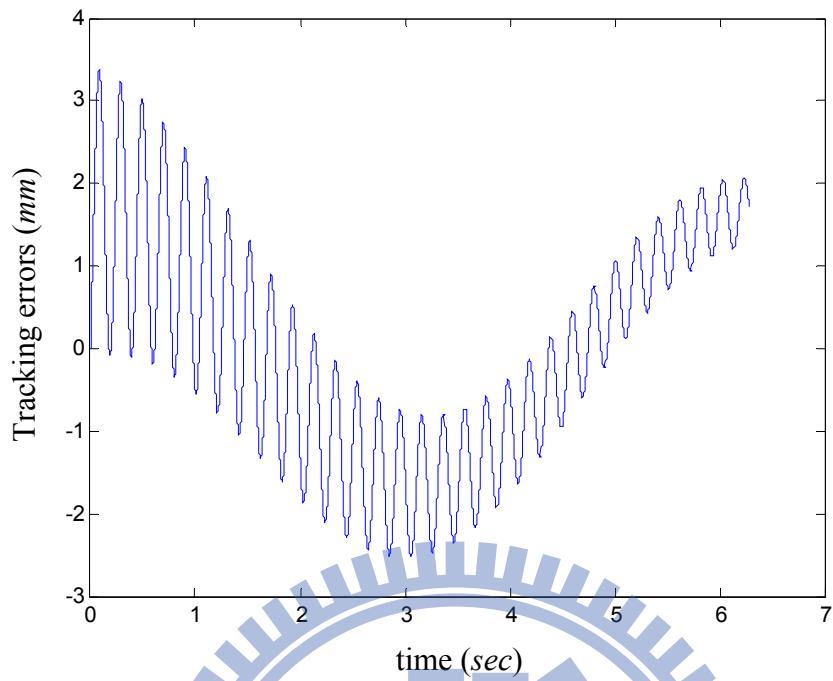


Fig. 6.8 The tracking error for NCS without compensation as $\tau_d = 50 \text{ ms}$

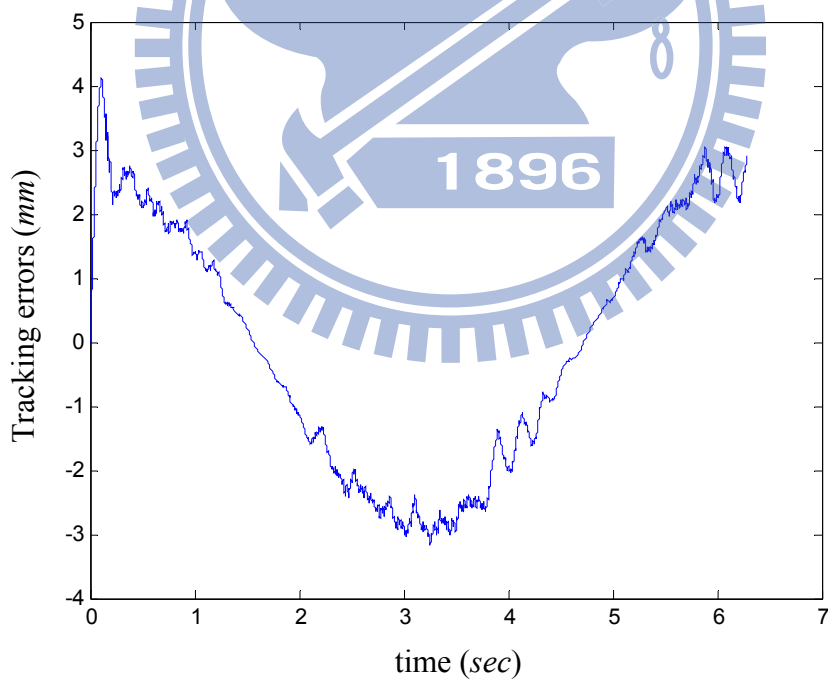


Fig. 6.9 The tracking error without compensation as both dropout and delay induced
 ($\rho_{D,R} = 0.8$, $\tau_d = 50 \text{ ms}$)

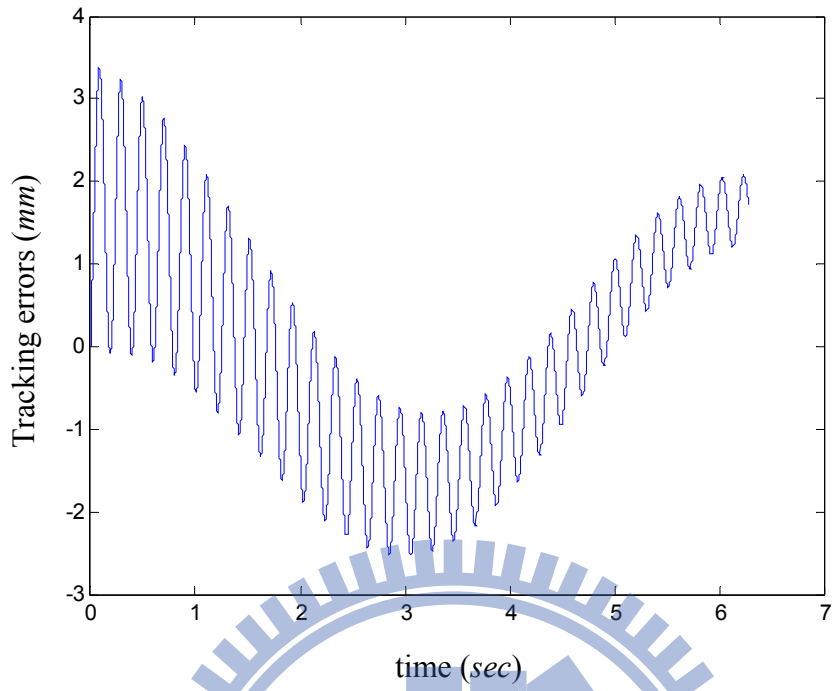


Fig. 6.10 The tracking error by applying IME only ($\rho_{D,R} = 0.8$, $\tau_d = 50 \text{ ms}$)

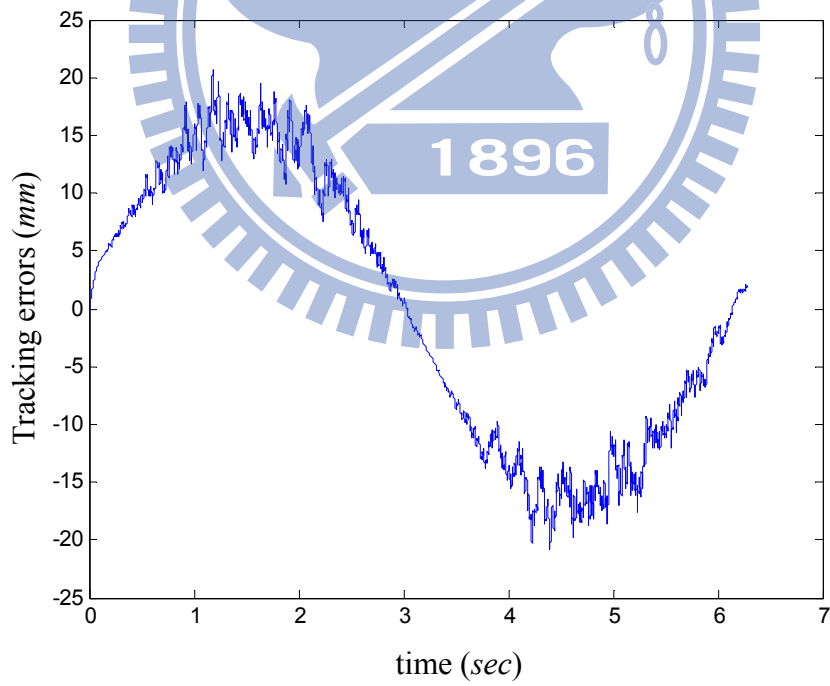
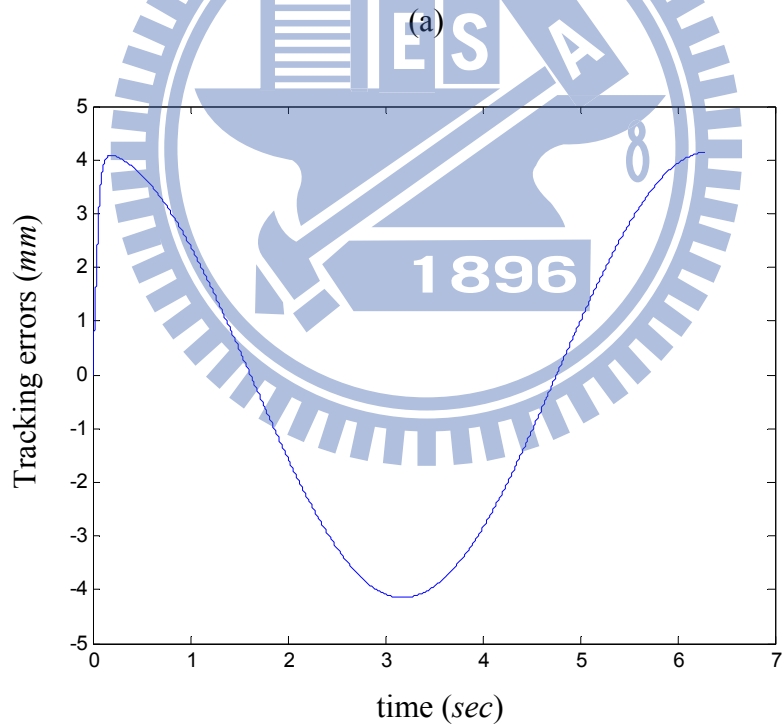
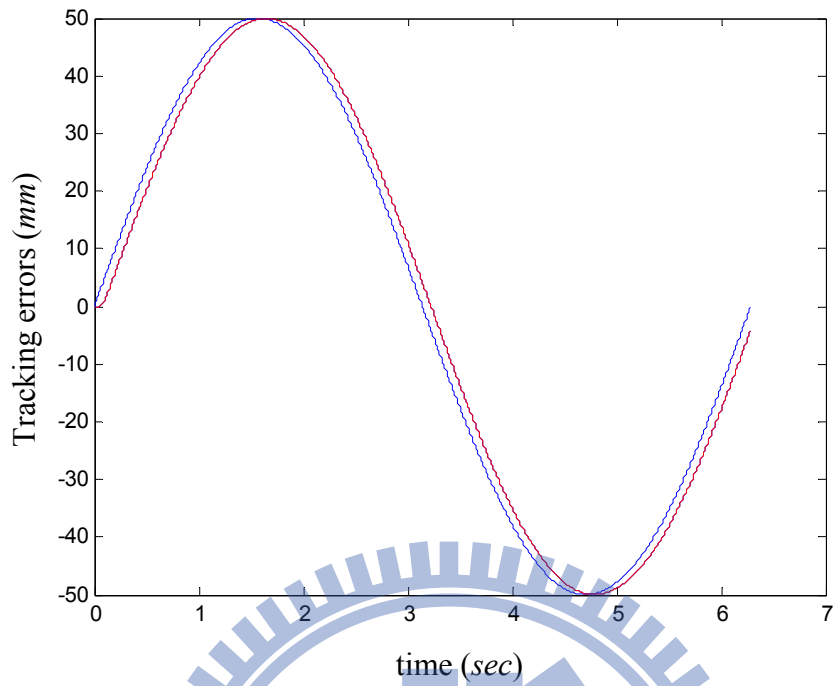


Fig. 6.11 The tracking error by applying PDC only ($\rho_{D,R} = 0.8$, $\tau_d = 50 \text{ ms}$)



(b)

Fig. 6.12 Applying both IME and PDC with $\rho_{D,R} = 0.8$ and $\tau_d = 50 \text{ ms}$
 (a)the system response (b)the tracking error

6.3 Case of varied dropout distribution and the time delay

In above section, effects of fixed networked time delay and distribution data dropouts are discussed and simulation results indicate that the integration IME and PDC can effectively compensate effects of data dropout and network time delay. The system architecture of integration of IME and PDC is shown in the Fig. 6.13. However, the varied networked time delay and data dropout distributions are avoidable existent in real network infrastructure. Therefore, the varied time delay and different dropout distribution are shown in the Fig 6.14 (a) and (b) to be as real network infrastructure situation.

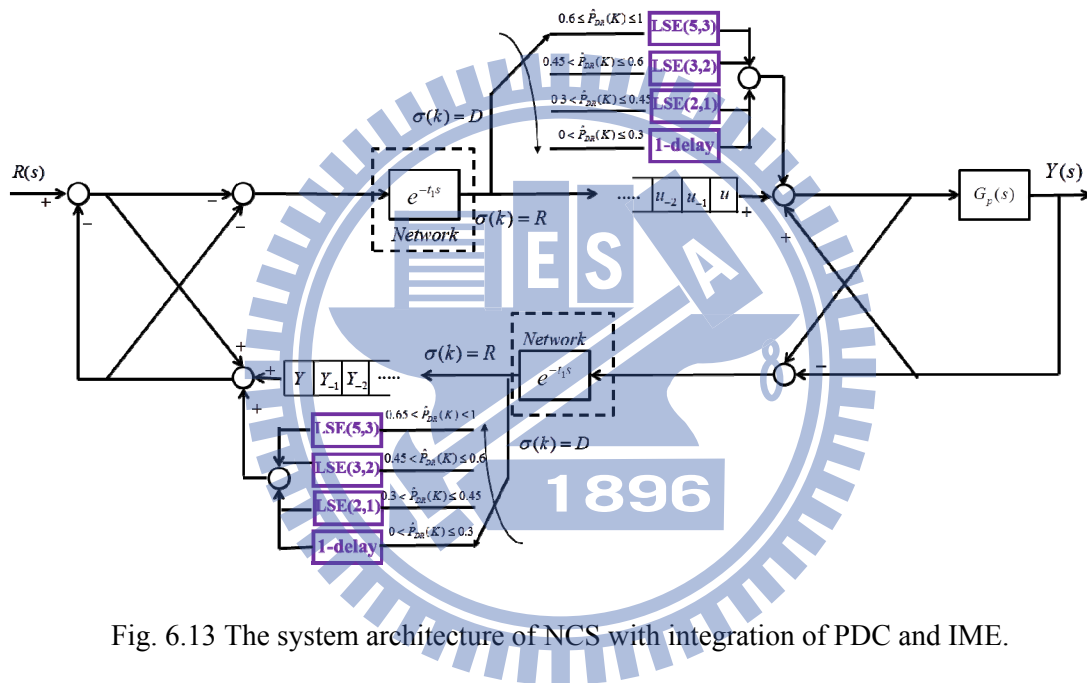


Fig. 6.13 The system architecture of NCS with integration of PDC and IME.

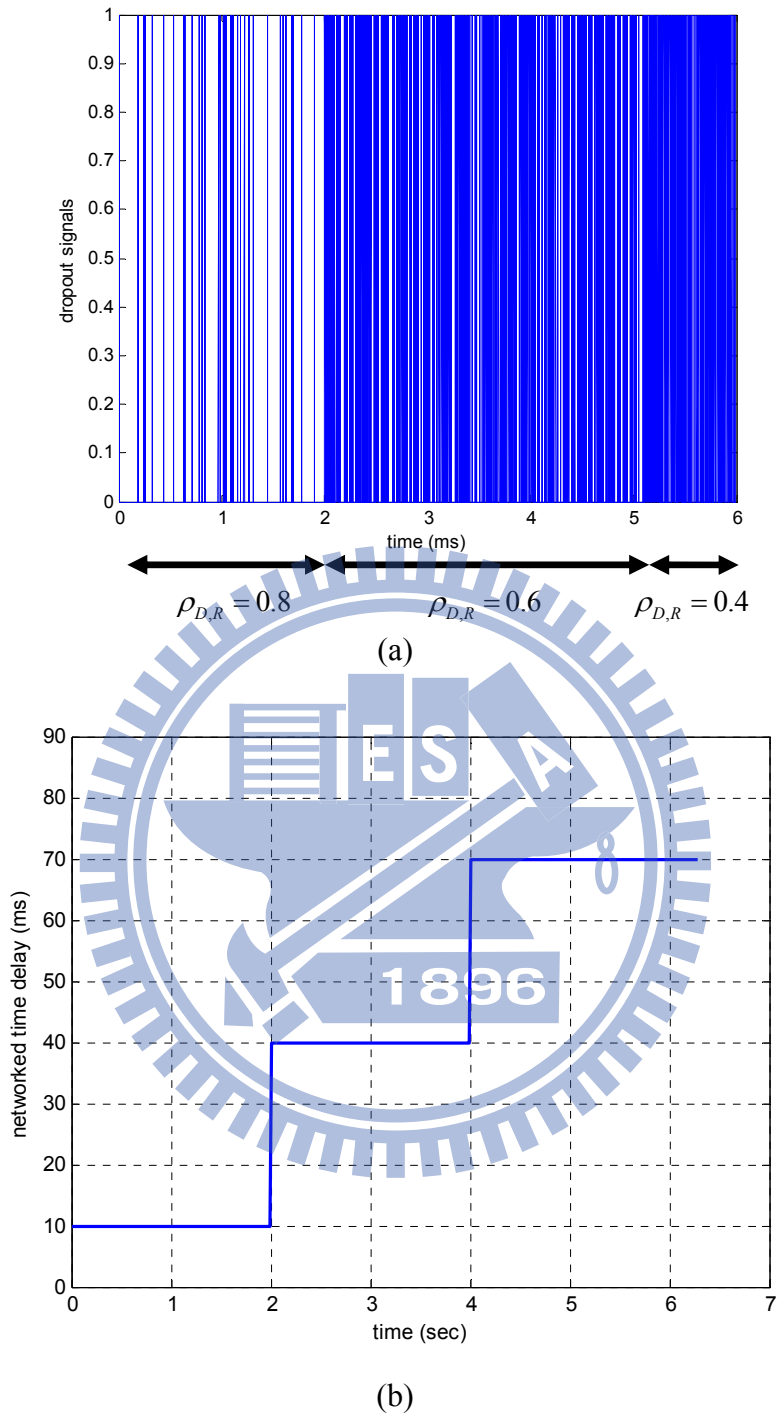


Fig. 6.14 (a) The dropout signal with different distributions (b) the varied networked time delay.

In this section, there are five different network conditions to verified compensation effects of IME and PDC. Total six cases will be tested and discusses as:

1. Ideal Case: no data dropout and no time delay
2. Case A: increasing data dropout and no time delay
3. Case B: no data dropout and increasing time delay
4. Case C: increasing data dropout and constant time delay
5. Case D: constant dropout and increasing time delay
6. Case E: increasing dropout and increasing time delay

Simulation results of all cases are summarized in Table 6.1. Results for each case are introduced as in the following:

- **Ideal case:** No networked time delay and no data dropout happens. Simulation results of the tracking error for the CNC system with a sinusoidal input are shown in Fig. 6.15. Overall control performance of this ideal case without network will be compared with other 5 cases with the NCS implementation.

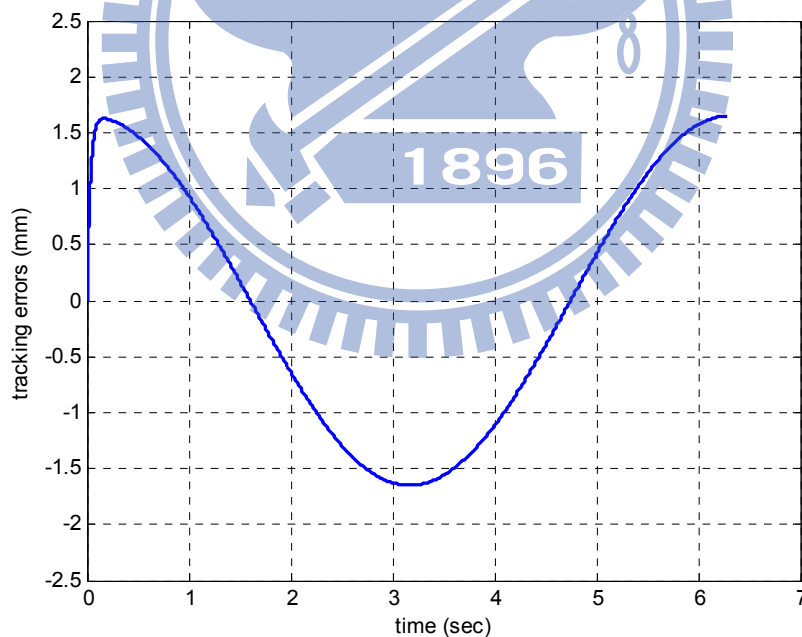


Fig. 6.15 The tracking error of the Ideal Case without time delay and data dropout

- **Case A:** Different dropout distributions are introduced to the motion NCS, as shown in Fig. 6. 13(a). As no message estimator is applied, results show that the system is unstable as shown in Fig. 6.16(a). From Fig. 6. 16 (b), results indicate

that IME can significantly eliminate the of dropout with different distributions.

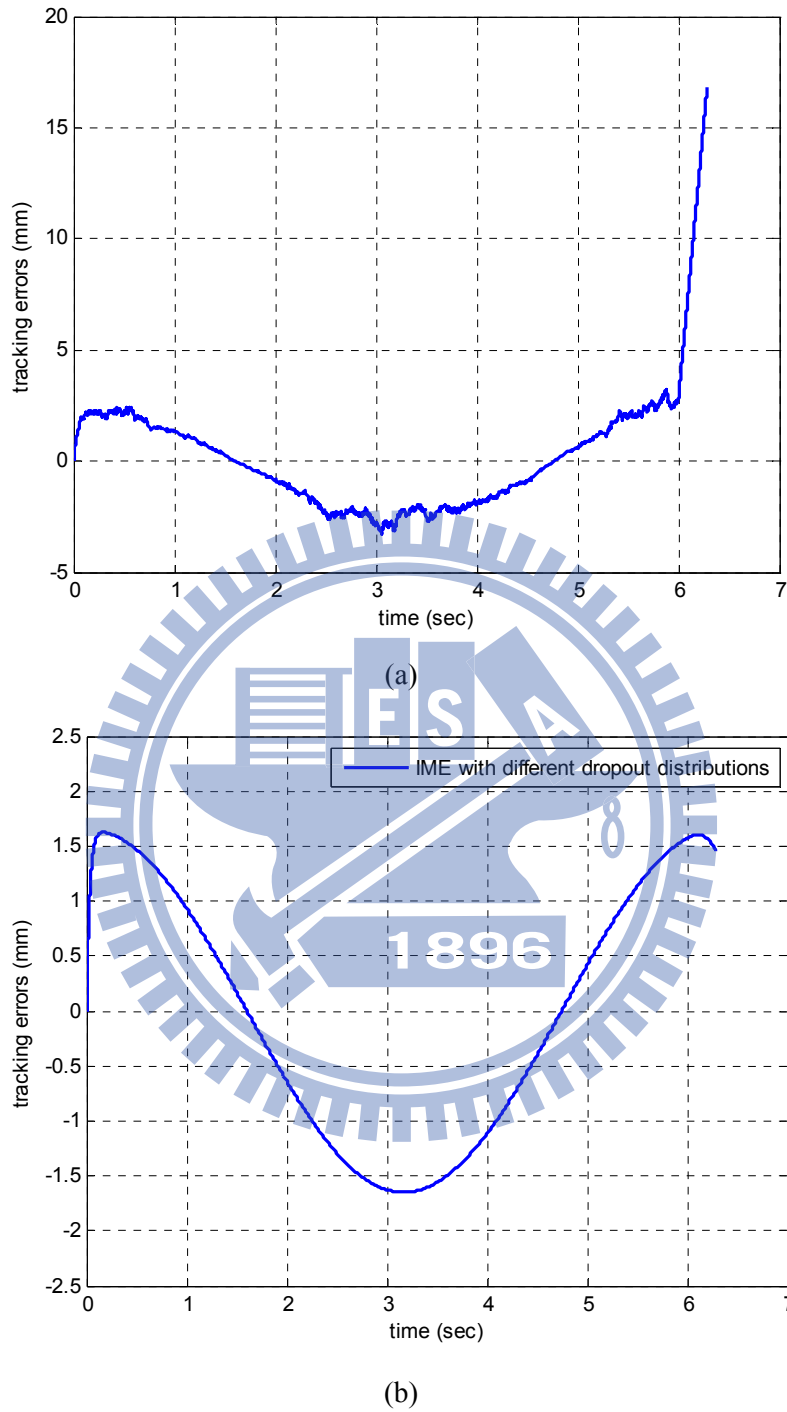
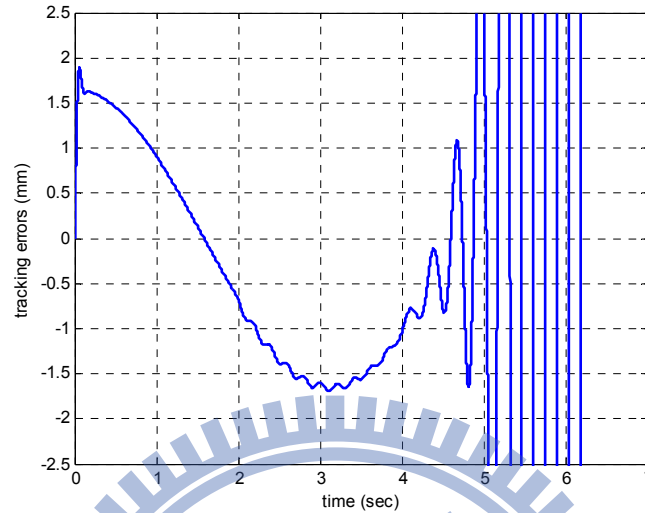


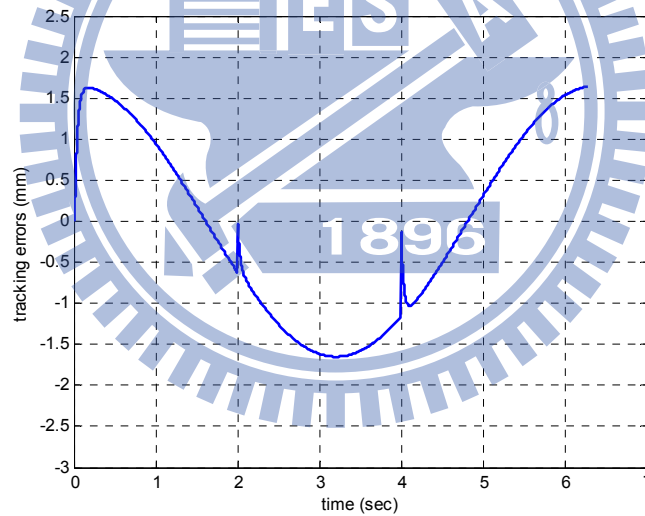
Fig. 6.16 Tracking errors with different distributions (a) without compensation (b) applying IME

- **Case B:** The varied networked time delay is introduced to the motion NCS, as shown in Fig. 6. 13(b). From Fig. 6. 17(a), the effect of network time delay lead

to unstable motion NCS. However, by applying PDC, even the varied network time delay effect is significantly eliminated, as shown in the Fig. 6.17(b), except as the delay changes abruptly at time 2s and 4s.



(a)



(b)

Fig. 6.17 Tracking errors with varied network delay (a) without compensation (b) applying PDC

- **Case C:** Different dropout distribution and constant network time delay ($\tau_d = 50\text{ ms}$) are both introduced to motion NCS. From Fig. 6. 18(a), simulation results indicate that although IME can effectively eliminate the effect of dropout distributions even in different levels, the effect of the network time delay still exists and it will lead motion NCS to be unstable as the oscillation occurs, as

shown in Fig. 6.18(a). Furthermore, the proposed motion NCS by integrating PDC and IME simultaneously eliminates the effect of both data dropout and network time delay, as shown in the Fig. 6.18(b).

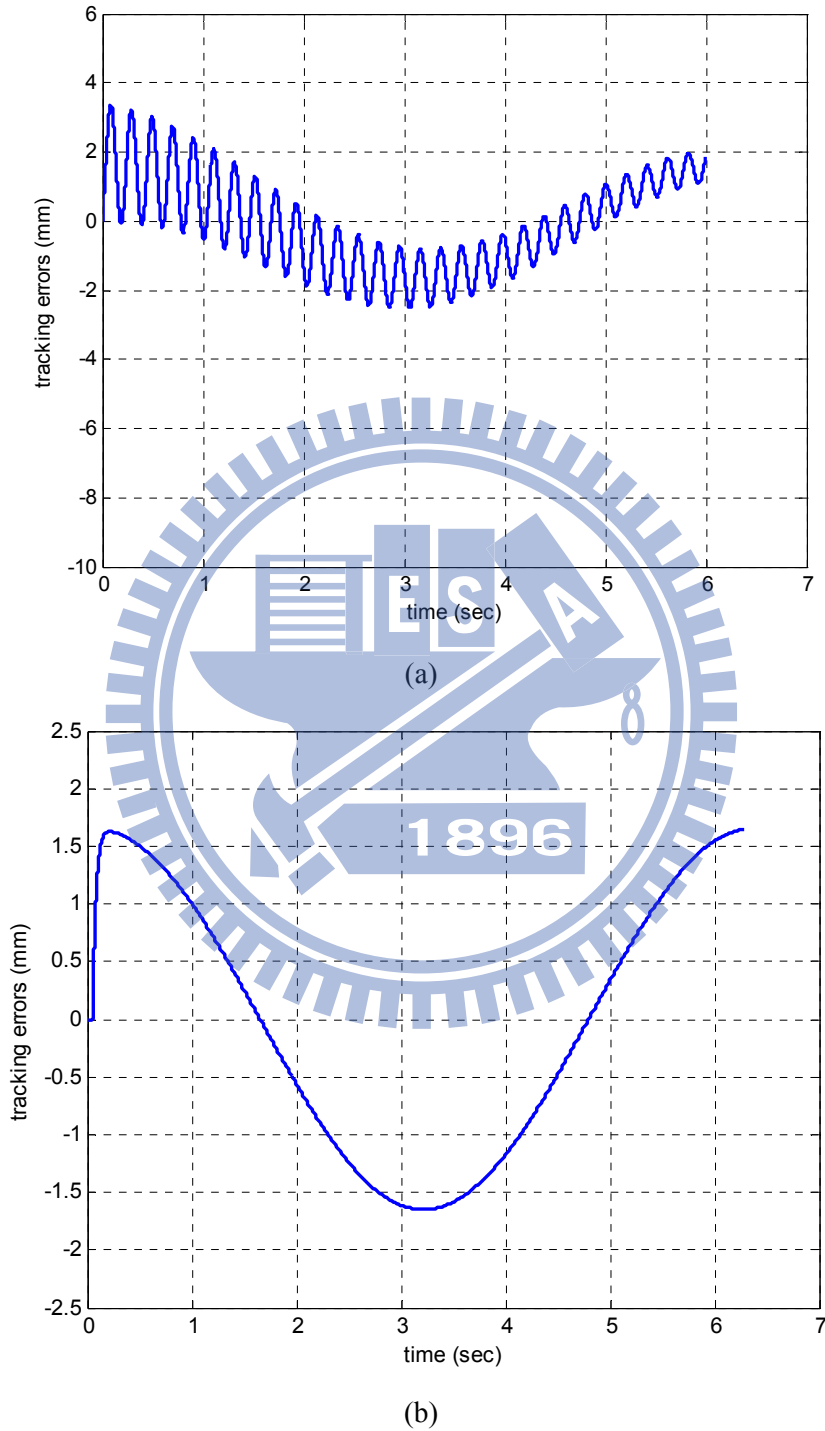


Fig. 6.18 Tracking errors with different dropout distributions and $\tau_d = 50\text{ ms}$ by

(a) applying IME (b) applying both IME and PDC

Case D: With the varied network time delay and a fixed dropout distribution ($\rho_{D,R} = 0.8$) introduced to the motion NCS, simulation results indicate that PDC can not compensate the effect of the time delay as the data dropout simultaneously occurs, as shown in Fig. 6. 19(a). However, integrated PDC and IME can eliminate both the effects of data dropout and network time delay, as shown in the Fig. 6.18(b).

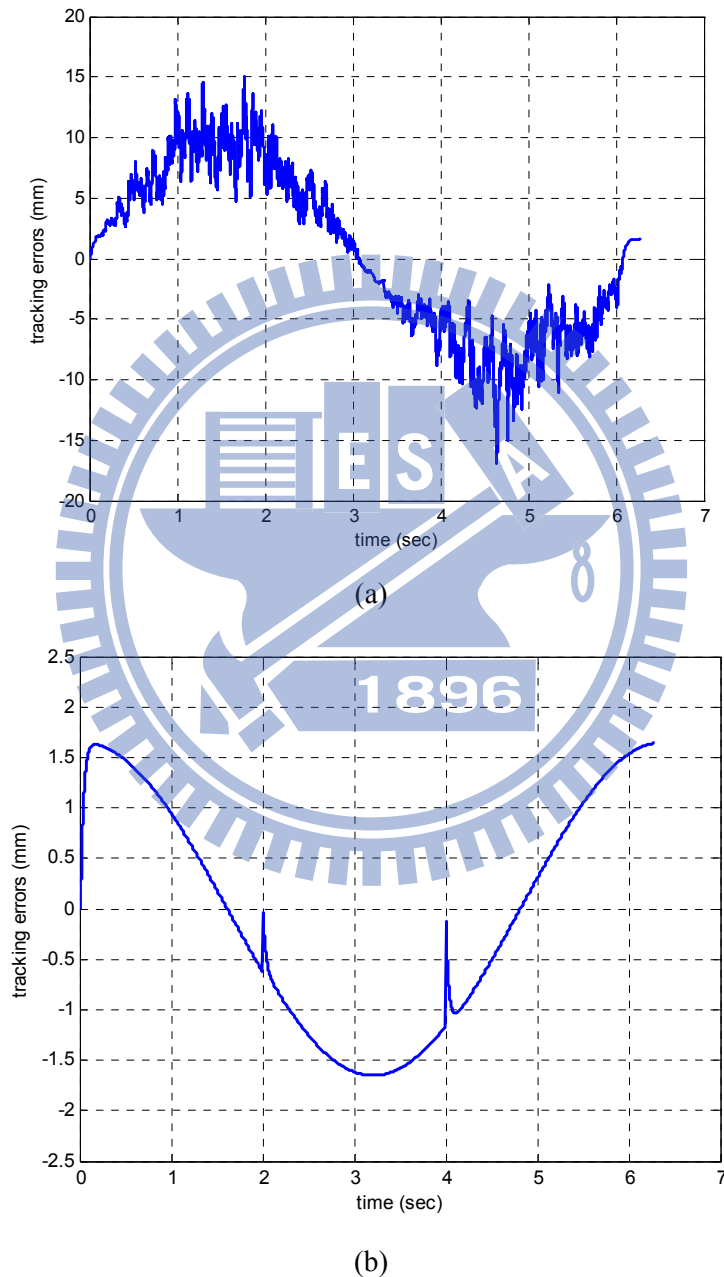


Fig. 6.19 Tracking errors with varied network time delay and $\rho_{D,R} = 0.8$ by
 (a) applying PDC (b) applying both IME and PDC

- **Case E:** Finally, different dropout distributions as shown in Fig. 6.13(a), and the varied networked time delay as shown in Fig. 6.13(b) are all introduced to the motion NCS. Fig. 6.20 indicates that integration of the PDC and the IME can simultaneously handle the most difficult conditions in NCS.

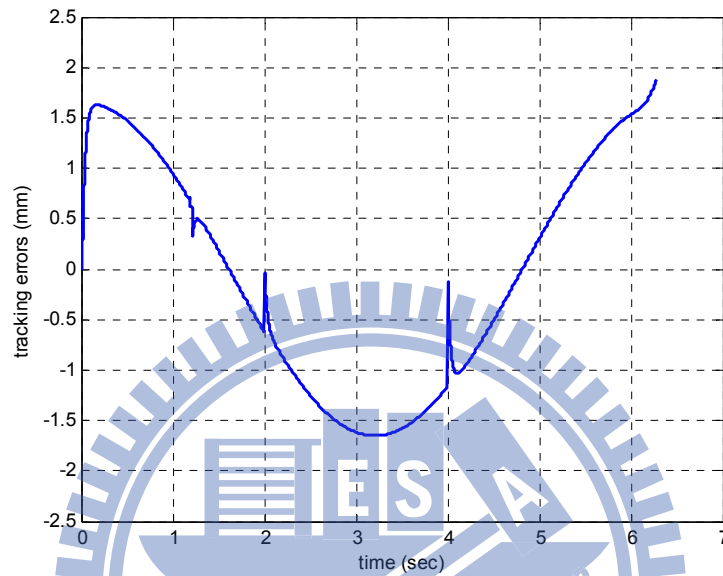


Fig. 6.20 The tracking error with varied network time delay and different dropout distributions by integrating PDC and IME.

Table 6.4 Evaluation for compensated effects of PDC and IME.

Case	Description	Network conditions	Compensator	Evaluation
Case A	Increasing dropout No delay	$\rho_{D,R} = 0.8 \rightarrow 0.6 \rightarrow 0.4$ $\tau_d = 0$	w/o IME	Unstable
			IME	Satisfactory
Case B	No dropout Increasing delay	$\rho_{D,R} = 1$ (no dropout) $\tau_d = 10 \rightarrow 40 \rightarrow 70$ (ms)	w/o PDC	Unstable
			PDC	Satisfactory
Case C	Increasing dropout Constant delay	$\rho_{D,R} = 0.8 \rightarrow 0.6 \rightarrow 0.4$ $\tau_d = 50$ (ms)	IME	Oscillation
			IME+PDC	Satisfactory
Case D	Constant dropout Increasing delay	$\rho_{D,R} = 0.8$ $\tau_d = 10 \rightarrow 40 \rightarrow 70$ (ms)	PDC	Oscillation
			IME+PDC	Satisfactory
Case E	Increasing dropout and Increasing delay	$\rho_{D,R} = 0.8 \rightarrow 0.6 \rightarrow 0.4$ $\tau_d = 10 \rightarrow 40 \rightarrow 70$ (ms)	IME	Unstable
			PDC	Unstable
			IME+PDC	Satisfactory

6.4 Summary

Results show that either the time delay or the data dropout unavoidably degrade control performance and stability. Moreover, their simultaneous occurrences will lead to serious problems in both stability and performance of NCS. In this chapter, both the time delay compensator PDC and the data dropout estimator IME are integrated in the proposed motion NCS to render satisfactory performance. The feasibility and effectiveness of the proposed integrated NCS have been proven for motion control with simulation results as follows:

- (1) The NCS dynamics with data dropout effect have been suitably modeled as a switched system between messages received and dropout status. Furthermore, motion NCS applying least square estimators with different orders can be also modeled as switched systems.
- (2) By using SMS stability condition, the region of stability can be thus obtained depending on the transition probability. Furthermore, the switching laws of IME

can be also determined based on the dominant transition probability $\rho_{D,R}$.

- (3) In simulation results, the motion NCS applying IME successfully achieves the guaranteed stability even with different data dropout distributions.
- (4) With different levels of the time delay, the PDC still renders satisfactory performance and stability. However, when the effect of the data dropout also involves, the PDC alone is no longer suitable for NCS.
- (5) Under both the different levels of the time delay and the different distribution of data dropout, the motion NCS maintains satisfactory control performance and stability by combining both IME and PDC.



Chapter 7

Conclusions and Future Work

7.1 Conclusions

This dissertation presents three newly developed approaches to deal with data dropout effects in motion NCS. The 3rd-order Taylor message estimator is proposed for motion NCS, as in Chapter 2. To estimate data dropout distributions for further compensation, the real-time transition probability estimator is further proposed as in Chapter 3. Moreover, the IME based on the estimated transition probability is then applied to motion NCS by concerning both the tracking accuracy and the guaranteed stability, as in Chapter 4 and 5. As the serious network-induced time delay occurs, control performance of the integration of both IME and PDC is verified, as in Chapter 6. Conclusions of this dissertation are as follows:

- (1). The 3rd-order Taylor message estimator has been successfully applied to CAN based CNC machine. Results indicate that as data dropout rate increases, the proposed estimator still compensate for the dropout effect well to maintain the originally designed motion NCS. Thus, even the feedforward controller can be applied to the motion NCS with a 3rd-order Taylor message estimator to further improve tracking accuracy.
- (2). To cope with dropout with stochastic natures, the Taylor message estimator is no longer to be suitable. Therefore, the real-time transition probability estimator has been developed and it is realized on DSP microcontroller for motion NCS. The proposed estimator provides efficient estimation of the transition probability which is useful to switch different dropout compensators to be the most suitable in tracking accuracy.
- (3). Average of fifty estimated transition probabilities, $\hat{\rho}_{D,D}$ and $\hat{\rho}_{R,R}$, can represent the transition probability of overall network to construct the two-state Markov chain network model and stability analysis.
- (4). Performance and stability of IME in a motion NCS are modeled as a switched

system and its stability is analyzed by the two-state Markov chain network model. By monitoring the dominant transition probability ($\hat{\rho}_{D,R}$), the switching laws of IME are accordingly determined to compensate for the data dropout effect with guaranteed stability of motion NCS.

- (5). The IME is successfully applied to the CAN-bus two-axis AC servo motor control system to efficiently compensate for the data dropout. Experimental results of the proposed IME render the best performance compared to the one-delay or the Taylor estimators.
- (6). As the network-induced time delay presents, the IME alone only is no longer suitable for motion NCS; however, the PDC alone only is also not suitable as the dropout occurs. Simulation results indicate that the condition with either dropout or delay occurrence will cause unstable NCS. By applying the integration of the PDC and IME compensators, it effectively compensates for the time delay and dropout simultaneously. Furthermore, as both the dropout and delay increase simultaneously, simulation results also indicate that the proposed integration of both PDC and IME still control the NCS well with satisfactory performance.

7.2 Future work

In this dissertation, the dropout compensator has already been realized in real industrial applications on CNC machine tools and the 2-axis AC servo motor control systems. The integration of both PDC and IME should be investigated on other applications to prove their superior performance and stability as the delay and dropout present simultaneously in practice.

References

- Adas, A. (1997), "Traffic models in broadband networks," *IEEE Communications Magazine*, vol. 35, no. 7, pp. 82-89
- Akyildiz L. F., Su W., Sankarasubramaniam, Y., and Cayirci, E. (2002), "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102 - 116.
- Antsakls, P. and Baillieul, J. (2004), "Guest editorial: special issue on networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1421 - 1423.
- Azimi, M., Nasiopoulos, P., and Ward, R. K. (2003), "Online Identification of Hidden Semi-Markov Models," *3rd International Symposium on Image and Signal Processing and Analysis*, pp. 991 – 996.
- Azimi, M., Nasiopoulos, P., and Ward, R. K. (2005), "Offline and Online Identification of Hidden Semi-Markov Models," *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2658 – 2663.
- Bertuccelli, L. F. and How, J. P. (2008), "Estimation of Non-stationary Markov Chain Transition Models," *IEEE Conference on Decision and Control*, Cancun, Mexico, pp. 55 – 60.
- Bolzern, P., Colaneri, P. and De Nicolao, G. (2006), "On Almost sure stability of discrete-time Markov jump linear systems," *43rd IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, pp. 14-17.
- Bushnell, L. G. (2001), "Guest editorial: network and control," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 22 - 23.

- Chan, H. and Özgüner, Ü. (1995), “Closed-loop control of systems over communications network with queues,” *International Journal of Control*, vol. 62, no. 3, pp. 493–510.
- Chen, L., Zhang J., and Wang, S. (2006), “Scheduling and Control Co-Design for Delay Compensation in Networked Control System,” *Asian Journal of Control*, vol. 8, no. 2, pp. 124-134.
- Chizeck, H. J., Willsky, A. S., and Castanon, D. (1986), “Discrete-time Markovian jump linear quadratic optimal control,” *International Journal of Control*, vol. 43, no. 1, pp. 213–231.
- Costa, O. L. V. and Fragoso M. D. (1993), “Stability results for discrete-time linear systems with Markovian jump parameters,” *Journal of Mathematical Analysis and Applications*, vol. 179, pp. 154-178.
- Costa, O. L. V. and Fragoso, M. D. (1995), “Discrete-time LQ-optimal control problems for infinite Markov jump parameter systems,” *IEEE Transactions on Automatic Control*, vol. 40, no. 12, pp. 2076–2088.
- Costa, O. L. V., Filho, E. O. A., Boukas, E. K., and Marques, R. P. (1999), “Constrained quadratic state feedback control of discrete-time Markovian jump linear systems,” *Automatica*, vol. 35, pp. 617–626.
- Costa, O. L. V. and Marques, R. P. (2000), “Robust H_2 -control for discrete-time Markovian jump linear systems,” *International Journal of Control*, vol. 73, no. 1, pp. 11 – 21.
- Fang, Y. and Loparo, K. A. (2002), “Stochastic stability of jump linear systems,” *IEEE Transactions on Automatic Control*, vol. 47, no. 7, pp. 1204-1208.
- Gilbert, E. N. (1960), “Capacity of a burst-noise channel,” *Bell Syst. Tech. J.*, vol. 39, pp. 1253 – 1265

- Gopi, M. and Manohar, S. (1997), "A Unified Architecture for the Computation of B-Spline Curves and Surfaces," *IEEE Transactions on Parallel and Distributed Systems*, vol. 8, no. 12, pp.1275 – 1287.
- Graham, A. (1981), *Kronecker Products and Matrix Calculus with Applications*, New York: John Wiley & Sons, pp. 21-35.
- Grenier, M. and Navet, N. (2008), "Fine-tuning MAC-Level protocols for optimized real-time QoS," *IEEE Transactions on Industrial Informatics*, vol. 4, no. 1, pp. 6 - 15.
- Gupta, V., Spanos, D., Hassibi, B., and Murray, R. M. (2005), "On LQG control across a stochastic packet-dropping link," *American control conference (ACC05)*, Portland, OR, pp. 360-365.
- Horn R. A. and Johnson C. R. (1991), *Topics in matrix analysis*. New York:Cambridge University Press.
- Hansen, M. and Yu, G. (2001), "Model Selection and Minimum Description Length Principle," *J. Amer. Statist. Assoc.*, vol. 96, pp. 746-774.
- Hsieh, C. C. and Hsu, P. L. (2005), "The CAN-Based Synchronized Structure for Multi-Axis Motion Control Systems," *IEEE International Conference on Systems, Man and Cybernetics*, vol. 2, pp. 1314-1319.
- Hsieh, C. C., Hsu, P. L., and Wang, B. C. (2006), "The Motion Message Estimator in Real-Time Network Control Systems," *32nd Annual Conference on IEEE industrial Electronics*, pp. 4632-4637.
- Hsieh, C. C. and Hsu, P. L. (2008), "Analysis and applications of the motion message estimator for network control systems," *Asian Journal of Control*, vol. 10, pp. 45-54.

Ji, Y., Chizeck, H. J., Feng, X., and Loparo, K. A. (1991), "Stability and control of discrete time jump linear systems," *Control Theory and Advanced Technology*, vol. 7, no. 2, pp. 58 – 71.

Jiang, X., and Han, Q. L. (2006), "Networked-Induced Delay-Dependent H_{∞} Controller Design for a Class of Networked Control Systems," *Asian Journal of Control*, vol. 8, no. 2, pp. 97-106.

Kawka, P. A. and Alleyne, A. G. (2006), "Stability and performance of packet-based feedback control over a Markov channel," *American control conference (ACC06)*, Minnerapolis, MN, pp. 20807 – 2812.

Kawka, P. A. and Alleyne A. G. (2009), "Robust Wireless Servo Control Using a Discrete-Time Uncertain Markovian Jump Linear Model," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 3, pp. 733 – 742.

Kim, Y. H., Park, H. S., and Kwon, W. H. (1996), "Stability and a Scheduling Method for Network-Based Control Systems," *IEEE IECON 22nd international Conference on Industrial Electronics, Control, and Instrumentation*, vol. 2, pp. 934-939.

Lai, C. L. and Hsu, P. L. (2011), "Design of Multi-rate Remote Control Systems with On-line Estimation of the Delay Time," *International Journal of System, Control and Communications*, vol. 3, no. 2, pp. 124 - 143.

Lee, K. C., Lee, S., and Lee, M. H. (2003), "Remote fuzzy logic control of networked control system via Profibus-DP," *IEEE Transactions on Industrial Electronics*, vol. 50, no. 4, pp. 784- 792.

Leon, S. J. (1998), *Linear Algebra with Applications*, New Jersey: Prentice Hall,

Li, H., Chow, M. Y., and Sun, Z. (2009), "Optimal stabilizing gain selection for networked control systems with time delays and packet losses," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1154 - 1162.

Lian, F. L., Moyne, J., and Tilbury, D. (2002), "Network Design Consideration for Distributed Control Systems," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 2, pp. 297 - 307.

Lian, F. L., Moyne, J. F., and Tilbury, D. M. (2001), "Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet," *IEEE Control System Magazine*, vol. 21, pp. 66–83.

Ling, Q. and Lemmon, M. D. (2002), "Optimal Dropout Compensation in Networked Control Systems," *IEEE Conference on Decision and Control (CDC'02)*, vol. 1, pp. 670-675.

Ling, Q. and Lemmon, M. D. (2002), "Robust Performance of Soft Real-Time Networked Control Systems with Data Dropouts" *IEEE Conference on Decision and Control (CDC'02)*, Las Vegas, Nevada, USA, vol. 2, pp. 1225-1230.

Ling, Q. and Lemmon, M. D. (2003), "Optimal Dropout Compensation in Networked Control Systems" *IEEE Conference on Decision and Control (CDC'03)*, Maui, Hawaii USA, vol. 1, pp. 670-675.

Ling, Q. and Lemmon, M. D. (2004), "Power Spectral Analysis of Networked Control Systems with Data Dropouts" *IEEE Transactions on Automatic Control*, vol. 49, no. 6, pp. 955-959.

Lu, L., Xie, L., and Cai, W. (2004), " H_2 Controller Design for Networked Control System," *Asian Journal of Control*, vol. 6, no. 1, pp. 88-96.

Luck, R. and Ray, A. (1994), "Experimental Verification of a Delay Compensation Algorithm for Integrated Communication and Control Systems," *IEEE Transactions on Control Systems Technology*, vol. 59, no. 6, pp. 1357-1372.

Matiakis, T., Hirche, S., and Buss, M. (2009), "Control of networked systems using the scattering transformation," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 1, pp. 60 - 67.

- Meyer, M. K. and Conant, R. (2005), "Opportunities of wireless sensors and controls for building operation," *Energy Engineering*, vol. 102, no. 5, pp. 27 - 48.
- Natori, K., Oboe, R., and Ohnishi, K. (2008), "Stability analysis and practical design procedure of time delayed control systems with communication disturbance observer," *IEEE Transactions on Industrial Informatics*, vol. 4, no. 3, pp. 185 -197.
- Natori, K. and Ohnishi, K. (2008), "A design method of communication disturbance observer for time delay compensation," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 5, pp. 2152 - 2168.
- Nilsson, J. (1998), *Real-time control systems with delays*, Ph.D. dissertation, Lund Institute of Technology.
- Ogata, K. (1995), *Discrete-Time Control Systems*, 2nd edition, Prentice-Hall: New Jersey.
- Oliveira, M. C. D., Geromel, J. C., and Bernussou, J. (2002), "Extended H_2 and H_∞ norm characterizations and controller parameterizations for discrete-time systems," *International Journal of Control*, vol. 75, no. 9, pp. 666–679.
- Park, B. G., Kwon, W. H., and Lee, J. W. (2001), "Robust receding horizon control of discrete-time Markovian jump uncertain systems," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E84-A, no. 9, pp. 2272–2279.
- Park, B. G. and Kwon, W. H. (2002), "Robust one-step receding horizon control of discrete-time Markovian jump uncertain systems," *Automatica*, vol.38, pp. 1229–1235.
- Peng, C. and Tian, Y. C. (2007), "Network-based H_∞ control of linear systems with state quantization," *Information Sciences*, vol. 177, no. 24, pp. 5763 - 5774.

- Peng, C., Tian, Y. C., and Yue, D. (2007), "Network quality-of-service based guaranteed cost control for networked control systems," *Dynamics of Continuous, Discrete and Impulse Systems Series B*; vol. 14, no. 2, pp. 233 - 247.
- Peng, C. and Tian, Y. C. (2006), "Robust H_∞ control of networked control systems with parameter uncertainty and state-delay," *European Journal of Control*, vol. 11, no. 5, pp. 471 - 480.
- Peng, C., Yue, D., and Sun, J. (2004), "The study of Smith prediction controller in NCS based on time-delay identification," *8th International Conference on Control, Automation, Robotics and Vision*, Kunming, CHINA, pp. 1644 - 1648.
- Piegl, L. and Tiller, W. (1995), *The NURBS Book*, Springer-Verlag, New York.
- Schenato, L. (2006), "Optimal estimation in networked control systems subject to random delay and packet loss," *IEEE Conference on Decision and Control (CDC'06)*, San Diego, CA, USA, pp. 5615 - 5620.
- Sinopoli, B., Schenato, L., Franceschetti, M., Poola, K., and Sastry, S. (2005), "Optimal control with unreliable communication: the TCP case," *American Control Conference (ACC05)*, Portland, OR, pp. 3354 - 3359.
- Smith, S. C. and Seiler, P. (2004), "Estimation with lossy measurements: jump estimators for jump systems," *IEEE Transactions on Automatic Control*, Vol. 48, pp. 640-644.
- Sorenson, H. W. (1970), "Least-square estimation: from Gauss to Kalman", *IEEE Spectrum*, vol.7. pp. 63-68.
- Souza, C. E. D. (2006), "Robust stability and stabilization of uncertain discrete-time Markovian jump linear systems," *IEEE Transactions on Automatic Control*, vol. 51, no. 5, pp. 836-841.

- Soglo, A. B. and Yang, X. (2006), "Compensation for network-induced delays and packet dropping in control systems," *European Journal of Control*, vol. 12, no. 3, pp. 296–306.
- Tang, B., Liu, G. P., and Gui, W. H. (2008), "Improvement of state feedback controller design for networked control systems," *IEEE Transactions on Circuit and Systems-II: Express Briefs*, vol. 55, no. 5, pp. 464 - 468.
- Tian, Y. C. and Levy, D. (2008), "Compensation for control packet dropout in networked control systems," *Information Sciences*, vol. 178, no. 5, pp. 1263-1278.
- Tipsuwan, Y. and Chow, M. Y. (2004), "Gain scheduler middleware: A methodology to enable existing controllers for networked control and teleoperation: PART I: Networked control," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 6, pp. 1218 - 1227.
- Tipsuwan, Y. and Chow, M. Y. (2004), "On the Gain Scheduling for Networked PI Controller over IP Network," *IEEE/ASME Transactions on Mechatronic*, vol. 9, pp. 491-497.
- Tomizuka, M. (1978), "Zero Phase Error Tracking Algorithm for Digital Control," *ASME Transactions on Journal of Dynamic Systems, Measurement and Control*, vol. 109, pp. 65-68.
- Val, J. B. R. D., Geromel, J. C., and Goncalves, A. P. C. (2002), "The H_2 control for jump linear systems: Cluster observation of the Markov state," *Automatica*, vol. 38, pp. 343–349.
- Varma, A., Ganesh, B., Sen, M., Choudhury, S. R., Srinivasan, L., and Bruce, J. (2003), "A control-theoretic approach to dynamic voltage scheduling," *International Conference on Compilers, Architectures, and Synthesis of Embedded Systems*, ACM, San Jose, CA, USA, pp. 255–266.

- Walsh, G. C., Ye, H., and Bushnell, L. (1999), "Stability Analysis of Networked Control Systems," *American Control Conference*, San Diego, pp. 2876 - 2880.
- Wang, H. S. and Moayeri, N. (1995), "Finite-state Markov channel, a useful model for radio communication," *Transactions on Vehicular Technology*, vol. 43, no. 1, pp. 163-171.
- Xie, D., Chen, X., Lv, L., and Xu, N. (2008), "Asymptotical stability of networked control systems: time-delay switched system approach," *IET Control Theory and Applications*, vol. 2, no. 9, pp. 743 - 751.
- Yeh, S. S. and Hsu, P. L. (1999), "An Optimal and Adaptive Design of the Feedforward Motion Controller," *IEEE/ASME Transactions on Mechatronics*, vol. 4, pp. 428-439.
- Yeh, S. S. and Hsu, P. L. (1999), "The Speed-Controlled Interpolator for Machining Parametric Curves," *Computer-Aided Design*, vol. 31, no. 5, pp. 349-357.
- Yue, D., Han, Q. L., and Peng, C. (2004), "State feedback controller design of networked control systems," *IEEE Transactions on Circuits and System II*, vol. 51, no. 11, pp. 640 - 644.
- Zhang, W., Branicky, M. S., and Phillips, S. M. (2001), "Stability of Networked Control Systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84 - 99.
- Zhang, W., Branicky, M. S., and Phillips, S. M. (2001), "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84 - 99.
- Zhao, Y. B., Liu, G. P., and Ress, D. (2009), "Design of a packet-based control framework networked control systems," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 4, pp. 859 - 865.

Vita

PERSONAL DATA

Name : 謝鎮洲, Chen-Chou Hsieh

Date of Birth : Aug. 12, 1978

E-mail : erazor.ece89g@nctu.edu.tw



EDUCATION

2002/9 - 2013/1	Receive the Ph.D. degree in the Institute of Electrical and Control Engineering from National Chiao Tung University, Taiwan, R.O.C.
2000/9 - 2002/6	Receive the M.S. degree in the Institute of Electrical and Control Engineering from National Chiao Tung University, Taiwan, R.O.C.
1996/9 - 2000/6	Receive the B.S. degree in the Department of Electrical Engineering from DAYEH University, Taiwan, R.O.C.

PUBLICATION LIST

A. Journal papers

1. **C. C. Hsieh** and P. L. Hsu, "Analysis and applications of the motion message estimator for network control systems," *Asian Journal of Control*, vol. 10, no. 1, p 45-54, 2008.
2. **C. C. Hsieh** and P. L. Hsu, "The switching message estimator for network-based motion control systems," *Journal of Control Science and Engineering*, vol.2012, Article ID 262378, 13 pages, 2012.

B. Conference papers

1. **C. C. Hsieh**, P. L. Hsu and B. C. Wang, "SDQ-based message estimator for TCP-network motion control system," *International Conference on Intelligent Control and Information*, Dalian, China, pp. 420-424, Aug. 2010.
2. **C. C. Hsieh**, P. L. Hsu and B. C. Wang, "The SDQ-based message estimator for motion NCS," *7th Asian Control Conference (ASCC 2009)*, Hong Kong, PEOPLES R CHINA, pp. 931-936, Aug. 2009.
3. **C. C. Hsieh**, P. L. Hsu and B. C. Wang, "The motion message estimator in networked control systems," *17th IFAC World Congress on Control*, Seoul, Republic of Korea, pp. 352-357, July 2008.
4. P. L. Hsu, **C. C. Hsieh**, and B. C. Wang, "The Integration of the Motion Message Estimator and the Adaptive CCC in Networked Control Systems," *CACS 2007 International Automatic control Conference*, Taichung, Taiwan, pp. 1211 – 1216.

5. **C. C. Hsieh**, P. L. Hsu and B. C. Wang, “The motion message estimator in real-time network control systems,” *32nd Annual Conference of the IEEE Industrial Electronics Society*, Paris, FRANCE, pp. 3946-3951, NOV. 2006.
6. **C. C. Hsieh** and P. L. Hsu, “The CAN-based synchronized structure for multi-axis motion control systems,” *IEEE International Conference on Systems, Man, and Cybernetic*, Waikoloa, HI, pp. 1314-1319, Oct. 2005.
7. **C. C. Hsieh** and P. L. Hsu, “The event-time triggered network control structure for CAN-based motion systems,” *IEEE International Conference on Control Applications*, Toronto, CANADA, pp. 722-726, Aug. . 2005.
8. **C. C. Hsieh**, A. P. Wang and P. L. Hsu, “CAN-based motion control design,” *SICE 2003 Annual Conference*, Fukui, Japan, pp. 2504-2509, Aug. 2003.

