THE HONG KONG POLYTECHNIC UNIVERSITY

Department of Electrical Engineering

Novel Adaptive Control Algorithms via Soft Computing Methods

Ho Ho Fai, James

A thesis submitted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

September 2004

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, expect where due acknowledgement has been made in the text.

______________________(Signed)

HO, HO FAI (Name of student)

Abstract

Typical objectives in control of dynamic systems are to ensure stability, tracking, and disturbance rejection. This thesis focuses on integration soft computing techniques with modem control methodologies for the developing novel adaptive control algorithms. The philosophy behind this study is that soft computing methods complement and not necessarily replace model-based control methodologies. The thesis presents extensive simulation studies and experimental verifications to demonstrate the characteristics of the proposed adaptive control methodologies.

In the first part of the study, we study the problem of a class of nonlinear systems that can be linearized around an operating point and can be represented by a lower-order model with time delay. Many chemical and petrochemical processes fall into this category. A new algorithm to approximate higher order systems with first order plus time delay via neural network is proposed. An on-line proportional plus integral derivative (PID) tuning method is then applied to control such systems.

In the second part, we address the problem of adaptive control of affine nonlinear dynamic systems. Based on a simple variable structure control, the sliding mode control, a novel adaptive fuzzy sliding mode control with chattering elimination has been developed. An algorithm has also been proposed to eliminate chattering at steady state.

Next, an H-infinity control technique incorporating a fuzzy system is studied. We introduce an adaptive fuzzy control with state observer to guarantee a robust performance of the controlled system. The H-infinity control is used to guarantee the robustness in the presence of system uncertainties.

We will also consider the multiple-input multiple-output nonlinear dynamic systems. We introduce the design of robust adaptive fuzzy control, with the aid of integrated sliding mode algorithm, an improved robust adaptive controller is proposed. This algorithm has been applied to a robot manipulator. We have also developed a direct adaptive fuzzy control based on a Takagi-Sugeno fuzzy system. We introduce a direct adaptive fuzzy control with state observer to estimate the unmeasured states of the multiple-input multiple-output controlled system. The controller has been tested for control of a two degree-of-freedom helicopter to track a given trajectory. Moreover, the system stability can be guaranteed based on Lyapunov's principle.

Finally, a robust adaptive fuzzy control for a class of uncertain nonlinear systems is examined. We introduce the design of adaptive fuzzy control for a general class of strict-feedback uncertain nonlinear dynamic systems. The main idea of this method is to apply the fuzzy system to derive a novel robust adaptive tracking controller by use of the input-to-state stability and by combining the backstepping technique.

Acknowledgments

I would like to begin by thanking my thesis supervisors, Dr. Y.K. Wong and Prof. A.B. Rad, for their invaluable guidance, patience and supervision during my study, and especially for their granting me the greatest freedom in my research. They guided me to work rigorously in the theory and practice of modern control and encouraged me to pursue my research on the theoretical ground work of soft-computing control. I would also to thank Dr. K.M. Tsang for his help in leading me to the control system theory.

I would also like to thank my colleagues, Dr. Y.L. Ip, Dr. C.H. Lo, Dr. W.L. Lo, Dr. P.T. Chan, and many others, for their supports.

I am also grateful to external members of my thesis committee, Prof. K.A. Hoo, Prof. F. Gao and Dr. X. Yang for reading through the dissertation and making a number of helpful suggestions.

The financial support by the award of a studentship from the Hong Kong Polytechnic University is also acknowledged.

Finally, I would like to thank the most important person in my life, my fiancée Sandy Kei, for her love, encouragement, patience, and help for these many years and for the years to come.

LIST OF ABBREVIATIONS

AFSMC	Adaptive Fuzzy Sliding Mode Control
BP	Back-Propagation
DOF	Degree of Freedom
FAC	Fuzzy Adaptive Control
FMRLC	Fuzzy Model Reference Learning Control
FOPDT	First Order plus Dead Time
FS	Fuzzy Systems
FSMC	Fuzzy Sliding Mode Control
GA	Genetic Algorithms
IMC	Internal Model Control
ISS	Input-to-State Stable
ISpS	Input-to-State practically Stable
IVSC	Integral Variable Structure Control
LMI	Linear Matrix Inequality
MIMO	Multi-Input Multi-Output
MRAC	Model Reference Adaptive Control
NN	Neural Networks
PCU	Process Control Unit
PD	Proportional Derivative
PI	Proportional Integral
PID	Proportional Integral Derivative

PIP	Predictive Proportional Integral
PT	Process Trainer
rhs	Right Half Side
SISO	Single-Input Single-Output
SMC	Sliding Mode Control
SOC	Self-Organizing Controller
SOPDT	Second Order plus Dead Time
SPR	Strictly Positive Real
STC	Self Tuning Controller
TS	Takagi-Sugeno
VSS	Variable Structure Control
ZN	Ziegler-Nichols

LIST OF SYMBOLS AND NOTATIONS

$\lvert x \rvert$	absolute value of scalar x
$\lVert\cdot\rVert$	Euclidean vector norm
$\dot{y}, y^{(n)}$	first and n th derivative of y with respect to time
R, R^n	the set of real numbers and n-dimensional Euclidean space
$f_1 \circ f_2$	composition of functions
$x \in X$	element x belongs to set X
$X \subset Y$	set X is contained in set Y
$\equiv$	equivalence
$\approx$	approximately equal
$\arg\ \min_a$	the minimum with respect to a
$\mathrm{sgn}(\cdot)$	sign function
$\tanh(\cdot)$	hyperbolic tangent
A, B, C	system, input and output matrix of a continuous system
$f(\cdot), g(\cdot), h(\cdot)$	unknown smooth mapping nonlinear function defined on a compact set
$x \in R^n$	state vector of the control system
DJ	Jocobian matrix
$L_a h(x)$	Lie derivative of $h(x)$ with respect to $a(x)$
u	control signal of the control system
y	output signal of the control system

r	reference input
e	output error
$\hat{e}$	error between the output and estimated state vector
$\hat{x}$	estimated state vector
$\tilde{x}$	error between the state vector and estimated state vector
L	observer gain
y_m	reference trajectory
θ^*	nominal parameter
θ	parameter vector to be identified
$\tilde{\theta}, \phi$	parameter vector error between the identified and nominal parameter
V	Lyapunov function candidate
c	a Hurwitz polynomial
$s(\cdot)$	sliding surface
b	bias term
W^h, W^o	connection weights of the neural network
f^h, f^o	activation functions of the neural network
$R^{(l)}$	the l th fuzzy rule of fuzzy system
A^l, B^l	fuzzy sets of input and output variables in fuzzy rules
y^l	output of the l th fuzzy rule
$\xi(\cdot)$	fuzzy basis function vector
μ_A^l, μ_B^l	membership functions of fuzzy sets A^l and B^l

$Y(s), U(s)$	Laplace transforms of the output and input of the system
a_i, b_j	coefficients of the system transfer function
$\hat{y}_m$	output of the model generator
K	gain of a first order model
T	time constant of a first order plus time delay model
ϑ	dead-time of plant
τ	dead-time of a first order plus time delay model
E	error between y and $\hat{y}_m$
ΔW	previous weight change of neural network
η	learning rate
α	momentum factor
ϕ_n	input vector of reduced order model
K_p, T_i, T_d	proportional gain, integral time constant and derivative time constant of the PID controller
y_f	filtered derivative of the PID controller
K_u, T_u	ultimate gain and ultimate period of a system
$G(s), G_c(s)$	transfer function of the model and PID controller
Θ	nominal parameter of fuzzy dynamic model
y_{pi}	sensitivity of the plant

$P_{i\max}$	maximum value of the fuzzy regressor values
$S_{i\max}$	maximum value of the sensitivity of the plant
d	external disturbance
D	upper bound of disturbance
θ_p	nominal parameter vector of PI control
ψ	regression vector of PI control
M_i	upper bound of estimated parameters
ω, ε	minimum approximation error of fuzzy system
γ_i	learning coefficients
Φ	boundary layer
$W(s), L(s)$	stable transfer functions
$M(\cdot)$	matrix of the moment inertia
$C(\cdot,\cdot)$	matrix of the Coriolis and centrifugal
$G(\cdot)$	vector of gravitational force
τ_i	applied torques
F_v, F_c	viscous and dynamic friction
θ, ϕ	elevation and azimuth angle
Δ_i	unknown Lipschitz continuous friction

z_i	error variables of backstepping control
λ_i	parameters of the hyperbolic tangent function
$\widetilde{\psi}_i$	parameters error between the estimated parameters ψ_i and nominal parameters $K_{\vartheta i}^2$
α_i	stabilizing functions
ψ_i^0, θ_i^0	constant parameters of the δ modification
$\delta_{i1}, \delta_{i2}, \Gamma_{n1}, \Gamma_{n2}$	adaptation gains of the δ modification update law

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENTS

Chapter 3 On-line Lower Order Modeling Via Neural Networks

Chapter 4 On-line Lower Order Modeling via Fuzzy System

Chapter 5 Adaptive Fuzzy Sliding Mode Control Systems with Chattering Elimination

CHAPTER ONE

INTRODUCTION

Modern process industries operate under strict regimes for improved productivity, more flexibility and cost efficiency. It has been suggested that the conventional approaches have been stretched to their limits and can not provide compatible solutions with such requirements. To meet these demands, the role of adaptive control schemes becomes more significant. In addition, the fusion of linear systems theory and the soft-computing methodologies has provided a new alternative strategy for addressing the ever growing problems in advanced process control. In this context, soft-computing techniques are broadly referred to those methodologies that exploit fuzzy systems (FS), artificial neural networks (NN) and genetic algorithms (GA) either standalone or in conjunction with each other.

The studies undertaken in this thesis address some of the control problems associated with the design of linear and nonlinear adaptive control systems by borrowing ideas and concepts from classical control systems theory, fuzzy systems, and artificial neural networks. The direct focus of this thesis is to explore the potentials of modern and intelligent control for the purpose of the design of novel adaptive control systems.

In particular, the following objectives are addressed:

- Design of neural network and fuzzy system modeling algorithm to model a class of nonlinear systems with lower-order model with time delay.

- Design of an adaptive fuzzy sliding mode control (AFSMC) algorithm for nonlinear systems to alleviate the problem of chattering due to sliding mode control.
- Design of an observer-based adaptive fuzzy control algorithm with H^{∞} control performance.
- Design of an indirect adaptive fuzzy control algorithm for multi-input multi-output nonlinear systems.
- Design of an observer-based direct adaptive fuzzy control algorithm for multi-input multi- output nonlinear systems.
- Design of an adaptive fuzzy control algorithm for nonlinear strict-feedback systems.

1.1 Motivation and Background

The linear control methodology has been the backbone of many industries in the last fifty years. It incorporates multitude of powerful methods and has a long history of successful applications. Despite many fruitful results, it may not be able to address emerging control problems for a variety of reasons. It is well known that linear control methods rely on the assumption of small range of operations in order to be valid. Such methods fail if this range of operations exceeds and subsequently the controlled system either performs poorly or becomes unstable. Another important assumption is that the plant is indeed lineariazable (linearization approximation around an equilibrium point) and linear control methodologies are to be adopted in this thesis. In fact, there are many nonlinearities, especially dead-zone, saturation and dead time, the discontinuous nature of which does not allow linearization and the assumption of linearization holds only within some regions.

In contrast, nonlinear control strategies relax the above assumptions. Most of the early methodologies of nonlinear control systems analysis utilized the phase plane method [Slotine and Li 1991, Vidyasagar 1993], the describing function approach [Slotine and Li 1991] or Lyapunov analysis in a trial and error fashion [Vidyasagar 1993]. Such designs were ad hoc and limited in dealing with complex and higher dimensional systems.

Based on the differential geometry technique, another study in nonlinear control system is exact feedback linearization. The basic idea of this methodology is to transform the nonlinear system into a fully or partially linear system by using nonlinear transformation, and then use the well known linear design techniques to complete the control design. Although, this methodology gives a strong impetus to nonlinear systems theory, it does not guarantee robustness of controlled systems that are subject to uncertainties and external disturbances.

Model uncertainties may also come from lack of knowledge about the controlled system or from the choice of a simplified representation of the plant. In general, adaptive control techniques and robust control techniques provide particularly appealing solutions to the problem of modeling uncertainties. A good body of knowledge exists for adaptive control of linear systems, guaranteeing stability and robustness to uncertainties and disturbance [Miller and Davison 1991; Sun 1993]. On the other hand, nonlinear adaptive control is rapidly developing and promise global stability and tracking results for relatively large classes of nonlinear systems [Marino and Tomei 1993, 1995; Krstic *et al.* 1995].

The focus of intelligent control research spans to around three decades. Zadeh (1994) coined the term soft-computing as collective methodologies, which fuse together fuzzy logic, neural network and genetic algorithms. The main feature of soft-computing techniques is their higher capability to tolerate imprecision, inaccuracies and partial truth. The motivation is often that the soft-computing techniques provide alterative solution to traditional modeling and design of control systems when system knowledge and dynamic models are incomplete. Soft-computing has been successfully applied to many commercial products and industrial systems, where less accurate mathematical models are available. In addition, human knowledge or biologically motivated techniques are available to provide inference from limited available information.

The hybrid adaptive control methodology is a set of techniques which combine the modern control theory with soft-computing theories to create adaptive control for dynamical systems. This class of adaptive controllers provide a very powerful methodology and provide control engineers with a framework to construct superior controllers for emerging nonlinear systems *i.e.* process control, mobile robot control, robot manipulation.

1.2 Research Outline

The philosophy in this thesis is that a hybrid adaptive control system is designed to combine the advantages from the soft-computing theory (fuzzy system, neural network) and modern control methodology. Bearing this philosophy in mind, the scope of studies can be classified into two phases. In phase 1, with a view of the fact that the linearization around an operating point can be represented by a lower-order model with time delay, the design of adaptive control system via on-line neural network and fuzzy modeling is presented. In phase 2 of the thesis, the author first addresses a class of nonlinear systems in a normal form for the design of adaptive control systems. Borrowing the concept of sliding mode control (SMC) and H^{∞} control from the modern control theory, a new fuzzy control and observer-based fuzzy control are developed by using adaptive feedback linearization strategy. Moreover, the extension from SISO nonlinear systems to MIMO nonlinear systems have also been attempted via integration of variable structure control with uncertainties bound estimation and observer-based direct adaptive control scheme. Finally, the study of nonlinear strict-feedback systems via model based adaptive fuzzy control is addressed. The control schemes in this thesis can be illustrated diagrammatically in Figure 1.1.

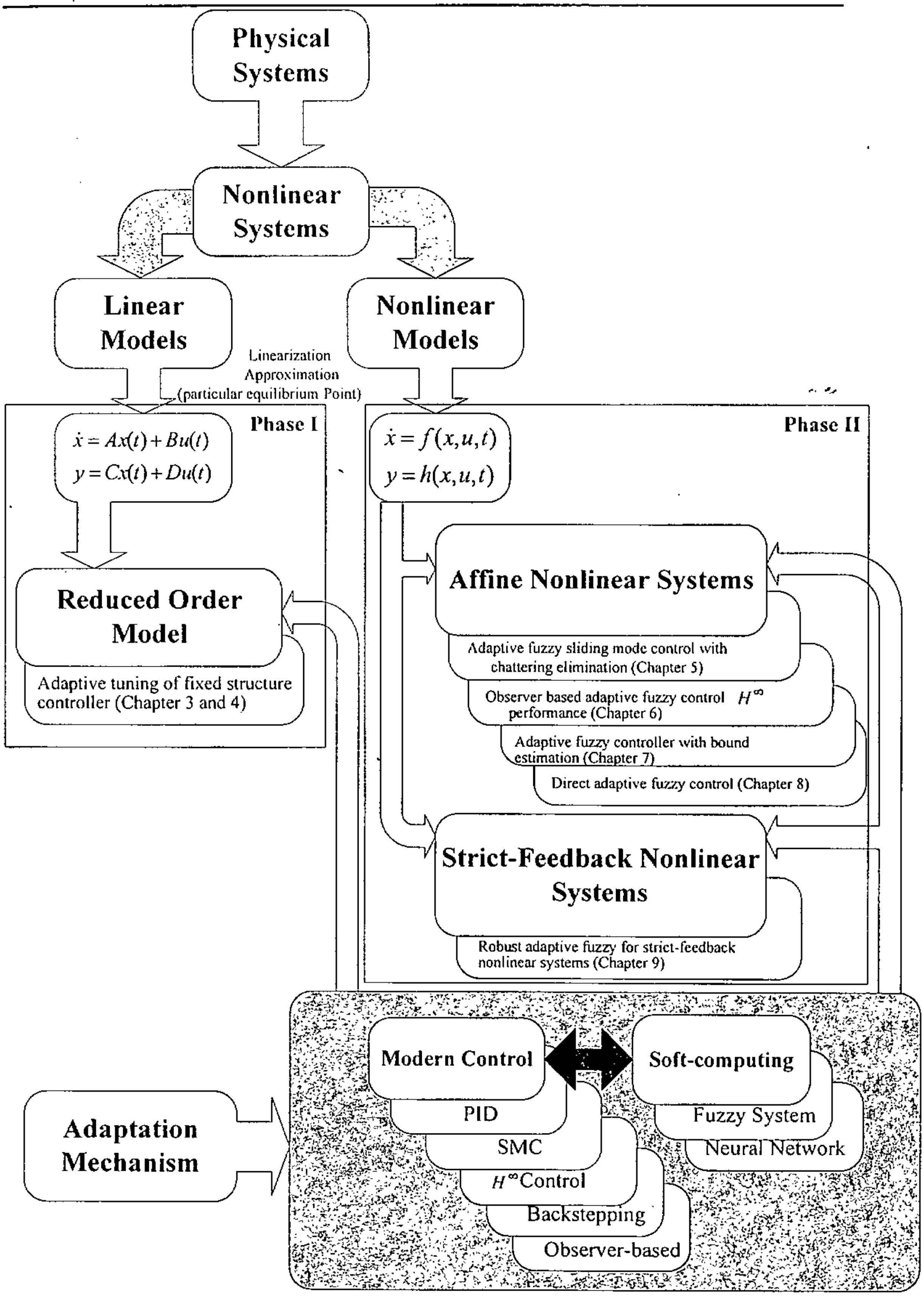


Figure 1.1 The design flow of this thesis

1.3 Layout of thesis

The thesis is organized as follows. The first chapter states the background, motivation, outline of the thesis, main contribution and a list of research outputs generated from this project.

Chapter 2 presents a literature survey on theories and principles of modern control, including nonlinear system theory, state observer, adaptive control, sliding mode control (SMC), H^{∞} control, backstepping control, neural networks and fuzzy systems. It should be noted that due to the extensive literature available, the treatment in this survey is very selective and only those relevant topics are included.

In chapter 3, a novel on-line approximation method based on a neural network architecture is proposed for nonlinear systems that can be linearized around an operating point and can be represented by a first order plus time delay model. An on-line proportional plus integral derivative (PID) tuning method is then applied to control such systems.

In chapter 4, by introducing a Lyapunov function, adaptive fuzzy system modeling is firstly developed for a class of linearized nonlinear systems, which can be represented by a lower-order with time delay model. Then, an on-line control design is developed via a fuzzy system with gradient descent technique. It is proved that under certain conditions, a convergence analysis on the proposed algorithm is achievable for the closed-loop adaptive system. Finally, the performance of the proposed algorithm for real time control of a temperature system with varying dominant time delay is evaluated.

In chapter 5, an adaptive fuzzy sliding mode control is proposed for nonlinear systems in the normal form. The controller is designed based on the fuzzy dynamic model. The control mechanism is basically an indirect adaptive control scheme. Then, a stable fuzzy controller is provided by using sliding mode control technique. The chattering problem of the adaptive fuzzy control is also investigated, and an adaptive PI control method are combined for reducing the systems chattering and zero steady tracking error can be ensured.

In chapter 6 adaptive observer-based fuzzy control is investigated for a class of affine nonlinear systems. By using a state observer for state estimation, an output feedback controller is given for fuzzy systems. The control performance of the systems is guaranteed by the H^{∞} performance criterion. Closed-loop stability is established by Lyapunov stability theory and effectiveness is verified by numerical simulation.

In chapter 7, adaptive controller design is investigated for a class of MIMO nonlinear system (Robot Manipulator). Using integral sliding mode control with estimation bound. The linear parameterization, i.e. nonlinearities of the robot dynamics are in the forms of linear in the parameters, can be solving using model based adaptive control techniques and the unknown nonlinear functions be approximated using fuzzy approximation. The stability of adaptive fuzzy control system has been analyzed by applying Lyapunov theory. The simulation results show how the validity, the performance of the proposed method and superiority to that of the tradition computed torque control scheme.

In chapter 8, another class of observer based fuzzy control system, i.e. TS fuzzy direct adaptive control system has been introduced for a class of MIMO nonlinear systems. The state variables are formed by a state observer which is designed based on the normal form of nonlinear systems. Closed-loop stability is established by Lyapunov direct method. The control algorithm is successfully applied to the laboratory scale 2-DOF helicopter on a hardware test-bed.

In chapter 9, a new controller design is investigated for a class of strict-feedback systems. By introducing a small gain theorem, a robust adaptive fuzzy control structure is developed, in which the cancellation of the nonlinearity $g_i(\cdot)$ is not required such that the control singularity problem can be avoided. It has been proved that the uniform ultimate boundedness is achievable for the closed-loop adaptive systems. Simulation studies are included to verify the performance of the proposed method.

In chapter 10, we conclude the thesis and provide some suggestions for future research.

1.4 Statement of Originality

The original contributions or important developments made by the author in this thesis are stated below:

- A method of on-line approximation of higher-order systems lower-order models with time delay using neural network and fuzzy system techniques. A theorem on convergence analysis of the on-line algorithm is presented (Chapters 3 and 4).
- A method of combining sliding mode controller and PI controller with adaptive fuzzy algorithm is proposed. The problem of chattering is avoided and zero steady state error can be ensure. A theorem on the stability analysis of adaptive fuzzy sliding mode control scheme is proposed (Chapter 5).
- An adaptive observer-based fuzzy control design, variable structure control (VSS) and H^{∞} disturbance attenuation theory are combined together to construct a hybrid indirect

adaptive observer-based robust tracking control scheme. The problem of the unavailable state variables and external disturbance of the system are solved. A theorem on the stability and robustness of the adaptive observer-based fuzzy control is given (Chapter 6).

- A method of combining adaptive fuzzy controller and sliding mode control with estimation bound for a class of MIMO nonlinear system is presented. The problem nonlinearities of the robot dynamics are in the forms of linear in the parameters and the unknown upper bound of the uncertainties are solved (Chapter 7)
- A direct adaptive observer-based fuzzy control system for a class of MIMO nonlinear systems is introduced. The control singularity problem and unavailable state variables can be avoided. A theorem for the closed-loop stability of direct fuzzy adaptive control scheme is established (Chapter 8).
- A robust adaptive fuzzy control scheme is proposed for a class of nonlinear strict-feedback systems. The cancellation of the nonlinearity gain functions is not required and the control singularity problem can be avoided. An on-line update law for the parameters of the fuzzy controller to compensate the unknown nonlinear and uncertainties is developed. A theorem on the stability analysis of robust adaptive fuzzy control scheme is proposed (Chapter 9).

1.5 Publications

At the time of writing this thesis, two papers have been published in international journals, one book chapter has been published, one journal paper has been submitted to international journals and is being review and four other papers are being prepared for submission. Moreover, eight conference papers have been presented in international conferences. These papers are listed below:

Published/Accepted Journal Papers:

1. H.F. Ho, A.B. Rad, Y.K. Wong and W.L. Lo, On-line lower-order modeling via neural networks. *ISA Transactions* 42: 577-593, 2003.
2. H.F. Ho, Y.K. Wong, A.B. Rad and W.L. Lo, State Observer Based Indirect Adaptive Fuzzy Tracking Control. *Simulation Practice and Theory*, 2005. (In press. Paper No. SIMPRA 182)

Book Chapter

3. H.F. Ho, A.B. Rad, Y.K. Wong and W.L. Lo, Adaptive neuro-fuzzy control of systems with unknown time delay, Tzyh-Jong Tarn, Shan Ben Chen and Changjiu Zhou (Eds.), in *Robotic Welding, Intelligence and Automation*, Lecture Notes in Control and Information Sciences LNCIS, Springer, pp. 304-326, 2003.

Journal Papers Submitted/Under Review:

4. H.F. Ho, Y.K. Wong, A.B. Rad and W.L. Lo, Fuzzy sliding mode control: adaptive approach. *IEEE transactions on industrial electronics.* (Submitted in July 2003)
5. H.F. Ho, Y.K. Wong, A.B. Rad, and W.L. Lo, On line low-order model approximation via fuzzy logic. (In final stages of preparation)
6. H.F. Ho, Y.K. Wong and A.B. Rad, Robust fuzzy tracking control for robotic manipulators. (In final stages of preparation)
7. H.F. Ho, Y.K. Wong and A.B. Rad, Direct adaptive fuzzy control for a class of nonlinear MIMO system. (In final stages of preparation)
8. H.F. Ho, Y.K. Wong and A.B. Rad, Robust adaptive fuzzy control for strict-feedback nonlinear systems. (In final stages of preparation)

Conference Papers:

9. H.F. Ho, Y.K. Wong, A.B. Rad and W.L. Lo, Fuzzy Predictive Controller. *The IASTED International Conference Control and Applications*, 2001, pp. 154-159

10. H.F. Ho, Y.K. Wong and A.B. Rad, Adaptive neuro-fuzzy control of systems with time delay. *IFSA World Congress and 20th NAFIPS International Conference*, Vol. 2, 2001, pp. 1044-1049.

11. H.F. Ho, Y.K. Wong and A.B. Rad, Adaptive fuzzy-neural control with state observer for unknown nonlinear systems via H^{∞} approaches, *Proceedings of the 9th International Conference on Neural Information Processing*, Vol. 4, 2002, pp. 1877-1881.

12. H.F. Ho, Y.K. Wong and A.B. Rad, Direct adaptive fuzzy control with state observer for a class of nonlinear systems. *The International Conference on Fuzzy Systems. FUZZ-IEEE 2003*, pp. 1338-1343.

13. H.F. Ho, Y.K. Wong and A.B.Rad, Adaptive fuzzy sliding mode control for SISO nonlinear systems. *The International Conference on Fuzzy Systems. FUZZ-IEEE 2003*, pp. 1344-1349.

14. H.F. Ho, Y.K. Wong, A.B. Rad and W.L. Lo, On line lower order modeling using fuzzy systems. *International Symposium on Advanced Control of Chemical Processes 2003.*

15. H.F. Ho, Y.K. Wong and A.B. Rad, Adaptive PID controller for nonlinear systems with H^{∞} tracking performance. *International Conference PHYSICS and CONTROL, PhysCon 2003*. pp.1315-1319.

16. H.F. Ho, Y.K. Wong and A.B. Rad, Fuzzy Sliding Mode Control: Adaptive Approach. Regional Inter-University Postgraduate Electrical and Electronic Engineering Conference. RIUPEEEC 2003.

Chapter Two

LITERATURE SURVEY

2.1 Introduction

The research studies reported in this thesis covers a wide range of topics in linear and nonlinear systems theory. The main contributions are expected to be on the integration of modern and soft-computing (fuzzy systems, neural network) methodologies in order to design novel adaptive control systems. Hence, the two domains are merged and a bridge is formed to utilize both domains for the purpose of enhanced algorithms that can be applied to solve control problems in real world with a view that the two domains complement each other. Specifically, we propose some new design methodologies for adaptive intelligent control algorithms for nonlinear dynamic systems. We deal with the problem of on-line adaptation and hence tackle systems with uncertainties and provide stability and robustness analysis.

This chapter provides an overview of relevant system analysis and modern control including the general ideas of local linearizatioin, input-output linearizatioin, state observer, sliding mode control, adaptive control, H^{∞} control and backstepping control. The soft-computing methods survey includes the fuzzy systems and neural network. Since the literature in this area is exhaustive, this review is highly selective and only covers parts of modern control and soft computing techniques which are used in the following chapters of this thesis.

This chapter is organized as follows. Section 2.2 presents topics within system theory that will be used in the following chapters. In Section 2.3, the author describes some of modern control methodologies. In Section 2.4 the soft-computing methodologies, neural network (NN) and fuzzy system (FS) will be discussed. Section 2.5 concludes the chapter.

2.2 Nonlinear systems

The representation of physical systems via mathematical terminology allows for a uniform framework of analysis and synthesis. The use of ordinary differential equations is one way of describing dynamic continuous systems. In general, a nonlinear system can be characterized by the state-space model in the following form:

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x) \end{aligned} \tag{2.1}$$

In the special case of input affine form [Khalil 2002, Sastry 1999]:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{2.2}$$

where $x \in R^n$ is the state vector, u and y represent the input and output. The nonlinear function $f: R^n \to R^n$, $g: R^n \to R^n$ are a smooth (C^∞) vector value function and $h: R^n \to R$ is a scalar function.

2.2.1 The Local Linearization Principle

This linearization is based on a Taylor-series expansion of a smooth nonlinear function f around an equilibrium point [Khalil 2002]. Indeed, given a nonlinear of the form $\dot{x} = f(x)$, the following Taylor series expansion is obtained:

$$\begin{aligned} f(x) &= f(x_0) + \left.\frac{\partial f}{\partial x}\right|_{x_0}(x - x_0) + \left.\frac{\partial^2 f}{\partial x^2}\right|_{x_0}\frac{(x - x_0)^2}{2!} + \cdots \\ &= f(x_0) + \left.\frac{\partial f}{\partial x}\right|_{x_0}(x - x_0) + \text{H.O.T.} \end{aligned} \tag{2.3}$$

where H.O.T. denote the higher order terms and $\frac{\partial^i f}{\partial x^i}$ denotes the *i-th* partial derivative of the function f with respect to x. Given that $x_0 = 0$ is an equilibrium point, $f(x_0) = 0$. The linear approximation of the original nonlinear equation $\dot{x} = f(x)$ is then obtained by neglecting the H.O.T.

$$\dot{x} = \left.\frac{\partial f}{\partial x}\right|_{x_0} (x - x_0) + \text{H.O.T.}$$

$$\dot{x} = Ax \tag{2.4}$$

It is easy to extend the principle for the case of a multivariate function $\dot{x} = f(x_1, x_2, \cdots, x_n)$ to obtain

$$\dot{x} = DJ|_{x_0} (x - x_0) \tag{2.5}$$

where DJ is the Jocobian matrix of the function f containing the partial derivative of the function respect to the variable x evaluated at the equilibrium point,

$$DJ|_{x_0} = \left.\frac{\partial f_j}{\partial x_i}\right|_{x_0} \tag{2.6}$$

2.2.2 Input-Output Linearization

In the middle of 1980's, the use of differential geometry led to the development of feedback linearization as a tool for controlling particular types of nonlinear systems [Isidori 1995]. The exact linearization is the use of state feedback and change of coordinates in the state space in order to transform a nonlinear system into linear and controllable one. Therefore any linear controller design method can be used to stabilize the system. Very often, feedback linearization is applied to nonlinear systems in input-affine form:

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{2.7}$$

Differentiating y in equation (2.7) with respect to time

$$\begin{aligned} y^{(1)} &= \frac{\partial h}{\partial x}\dot{x} \\ &= \frac{\partial h}{\partial x}[f(x)+g(x)u] \\ &= \frac{\partial h}{\partial x}f(x)+\frac{\partial h}{\partial x}g(x)u \\ &= L_f h(x) + L_g h(x)u \end{aligned} \tag{2.8}$$

where $L_f h(x): R^n \to R$ and $L_g h(x): R^n \to R$ is the Lie derivative of a scalar function $h(x)$ with respect to a vector function $f(x)$ and $g(x)$ respectively.

Suppose $L_g h(x) \neq 0$ for all x inside a region $\Omega \subseteq R^n$, the above input-output relation between y and u is well defined. In this case, the system is said to have a relative degree of one in region Ω, because the input-output relation was obtained after one differentiation. Suppose $L_g h(x) \equiv 0$ for all points inside region Ω. We differentiate y repeatedly with respect to time. The n^{th} derivative of y is given by

$$L_g L_f^{i-1} h(x) = 0, \qquad \forall i = 1,2,\cdots,(n-1) \tag{2.9}$$

By this, we can obtain a well-defined input-output relation after n differentiation. The dynamics could be expressed into a normal form by the coordinate transformation.

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} = \begin{bmatrix} \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_n(x) \end{bmatrix} = \begin{bmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{n-1} h(x) \end{bmatrix} \tag{2.10}$$

In the new coordinates

$$z_i = \phi_i(x) = L_f^{i-1} h(x), \quad 1 \le i \le n$$

the systems became

$$\begin{aligned}
\dot{z}_1 &= z_2 \\
\dot{z}_2 &= z_3 \\
&\vdots \\
\dot{z}_{n-1} &= z_n \\
\dot{z}_n &= L_f^n h(x) + L_g L_g^{n-1} h(x) u \\
&= a(x) + b(x) u
\end{aligned} \tag{2.11}$$

It is easy to see that the state feedback control law

$$u = \frac{1}{b(x)}(-a(x) + v) \tag{2.12}$$

cancels the nonlinearities to the n-order linear systems from input v to output z :

$$z^{(n)} = v \tag{2.13}$$

This represents a nonlinear system having relative degree $r = n$ (exactly equal to the dimension of the state-space), when in fact for $r < n$, the equation (2.11) must have rendered $(n-r)$ internal unobservable states. This is exactly analogous to the concept of pole-zero cancellation in linear systems theory. The $(n-r)$ dimensional internal states are regarded as the zero dynamics [Isidori 1995, Sastry 1999]. A common assumption in control for feedback linearization systems is that the zero dynamics is exponentially stable.

2.2.3 Observer

The design of observers imply the reconstruction of the internal states from output measurements of the system. For linear systems, the observability and detectability properties are closely connected to the existence of observers with strong properties, such as exponential convergence of the errors. For nonlinear systems, the problem is more difficult and challenging. The practical approach to design of the observer based problems in nonlinear systems is to utilize the linearization in Section 2.2.1. In Section 2.2.2, another linearization method dealt with a class of nonlinear systems that can be transformed into linear systems by using feedback and change of coordinates. In this section, the idea of state observer is briefly reviewed. Consider a state space realization of the form:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{2.14}$$

where $A \in R^{n \times n}, B \in R^{n}$ and $C^{T} \in R^{n}$. Assuming that (A, C) form an observable pair. A observer for the system of equation (2.14) is construct as

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{2.15}$$

where L is the observer gain. Define the observer error

$$\tilde{x} = x - \hat{x} \tag{2.16}$$

we have

$$\begin{aligned} \dot{\tilde{x}} &= \dot{x} - \dot{\hat{x}} \\ &= (Ax + Bu) - (A\hat{x} + Bu + Ly - LC\hat{x}) \\ &= (A - LC)\tilde{x} \end{aligned} \tag{2.17}$$

The observer gain L is designed such that $A - LC$ is Hurwitz. It is well known that if the observability or at least detectability of the pair (A, C) is satisfied, then the eigenvalues of $A - LC$ can be choice by suitable selection of the observer gain [Khalil 2002]. Thus, $\tilde{x} \to 0$ as $t \to \infty$.

2.3 Modern Control

2.3.1 Adaptive control

Research in the field of adaptive control has a long history. This goes back to early 1950's, with the design of autopilots for high performance aircrafts. In the 1970's, adaptive control was widely accepted in control applications, due to the Lyapunov's stability theories and the progress in control theories based on state space techniques [Miller and Davison 1991; Sun 1997; Hsu *et al.* 1997]. The concept and development of a wide class of adaptive control theory with well established stability properties was accompanied by several successful applications [Sastry and Bodson 1989, Ioannou and Sun 1994]. Model reference adaptive control (MRAC) techniques [Sastry and Bodson 1989, Slotine and Li 1991] were designed and analyzed base on Lyapunov's stability approach [Narendra 1989, Narendra and Balakrishnan 1997, Sun and Hoo 1999].

In the late 1980's to early 1990's, the adaptive control theory was used to extend the properties and results of the linear systems to a certain class of nonlinear systems with unknown system parameters. The adaptive control of nonlinear systems has undergone rapid development. The seminal works including those of [Sastry and Isidori 1989, Slotine and Li 1991] led to stabilization and tracking control for some classes of nonlinear systems. In the late 1990's, the focus of intelligent adaptive controllers were to further extend previous results and provide more in depth analysis.

Here, we briefly illustrate basic concepts and design for adaptive control systems [Khalil 2003, Slotine and Li 1991, Narendra 1989]. The control objective is to design a feedback control law such that all closed-loop system signals are bounded and the plant output track the pre-specified reference asymptotically. The general approach of adaptive control systems is illustrated in Figure 2.1.

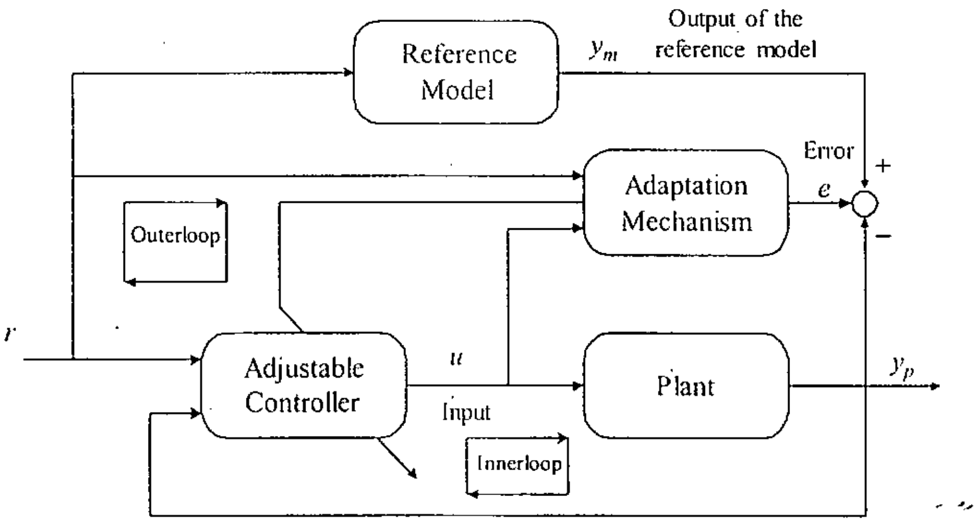


Figure 2.1 Model reference adaptive control scheme

Consider a linear time-invariant plat and reference model [Khalil 2002, Åström and Wittenmark 1995].

$$\begin{aligned} \dot{y}_p &= a_p y_p + k_p u \\ \dot{y}_m &= a_m y_m + k_m r \end{aligned} \qquad (2.18)$$

where a_p and k_p aré the plant parameters, $a_m < 0$ and k_m are the model parameters of a chosen reference model. Furthermore, r is a bounded external input. When the plant parameters a_p and k_p are known, we use the feedback controller

$$u(t) = \theta_1^* r(t) + \theta_2^* y_p(t) \qquad (2.19)$$

where

$$\theta_1^* = \frac{k_m}{k_p}, \quad \theta_2^* = \frac{a_m - a_p}{k_p} \qquad (2.20)$$

when the plant parameters are unknown, we cannot implement the control law u because θ_1^* and θ_2^* are unknown. Instead, we use estimate θ_1 and θ_2 for θ_1^* and θ_2^* to implement the adaptive controller

$$u(t) = \theta_1 r(t) + \theta_2 y_p(t) \qquad (2.21)$$

Define the tracking error as $e = y_p - y_m$, parameters error $\phi_1 = \theta_1 - \theta_1^*$ and $\phi_2 = \theta_2 - \theta_2^*$. In term of the tracking error e, we have

$$\begin{aligned} \dot{e} &= \dot{y}_p - \dot{y}_m \\ &= a_p e + k_p(\theta_1 - \theta_1^*)r + k_p(\theta_2 y_p - \theta_2^* y_m) \\ &= a_m e + k_p \phi_1 r + k_p \phi_2 y_p \end{aligned} \tag{2.22}$$

The choice of adaptive law

$$\begin{aligned} \dot{\theta}_1 &= \dot{\phi}_1 = -\gamma e r \\ \dot{\theta}_2 &= \dot{\phi}_2 = -\gamma e y_m \end{aligned} \tag{2.23}$$

Introduce parameters as measure for the errors e, ϕ_1 and ϕ_2:

$$V = \frac{e^2}{2k_p} + \frac{1}{2\gamma}\phi_1^2 + \frac{1}{2\gamma}\phi_2^2 \tag{2.24}$$

as a Lyapunov function candidate, we obtain

$$\dot{V} = \frac{-a_m}{k_p} e^2 \leq 0 \tag{2.25}$$

It can be proved that $\lim_{t\to\infty} e(t) = 0$ and the output of the plant tracks the desired model output. Since

$$-\int_0^\infty \dot{V} d\tau = V(0) - V(\infty) < \infty \tag{2.26}$$

we have

$$\int_0^\infty e^2(\tau) d\tau < \infty \tag{2.27}$$

Since $\dot{e}(t)$ as given by equation (2.22) is bounded. Hence, $e, \dot{e} \in L^\infty$, use of Barbalat's lemma [Sastry and Bodson 1989], the system is stable and the tracking error in the adaptive control system converges to zero asymptotically.

2.3.2 Sliding Mode Control (SMC)

The simplest variable structure control, the sliding mode control [Utkin 1977] is widely accepted as a powerful control method of tackling uncertain nonlinear systems. [DeCarlo *et al.* 1988, Slotine *et al.* 1984, 1991]. This control algorithm offers good robustness against model uncertainties and external disturbances, provided that the uncertainties and disturbances lie within a bound. The control signal can be driven the system states to the sliding plane and stay on it. Consider an n-th order nonlinear system in the form of :

$$\begin{aligned} x^{(n)} &= f(x) + g(x)u + d(t) \\ y &= x_1 \end{aligned} \tag{2.28}$$

Assuming that the upper bound of the disturbance $d(t)$ is D, that is $|d(t)| \leq D$. Without loss of generality, we assume that $g(x) \neq 0$ and $g(x) > 0$. The control objective is to design a control action for the state x to track a desired reference state y_m in the presence of model uncertainties and external disturbances. The tracking error is defined as follows.

$$e = x - y_m = [e, \dot{e}, \cdots, e^{(n-1)}]^T \in R^n \tag{2.29}$$

The sliding surface in the error state space is defined as:

$$\begin{aligned} s(e) &= c_1 e + c_2 \dot{e} + \cdots + c_{n-1} e^{(n-2)} + e^{(n-1)} \\ &= c^T e \end{aligned} \tag{2.30}$$

where $c = [c_1, c_2, \cdots, c_{n-1}, 1]^T$ are the coefficients of the Hurwitiz polynominal. For the zero initial conditions $e(0) = 0$, the tracking problem $x = y_m$ can be considered as keeping the error state vector on the sliding surface $s(e) = 0$ for all $t \geq 0$. A sufficient condition to achieve this condition is to select the control strategy such that

$$\frac{1}{2}\frac{d}{dt}(s^2(e)) \leq -\eta |s|, \qquad \eta \geq 0 \tag{2.31}$$

The system is controlled in such a way that the state always moves towards the sliding surface and hits it. The sign of the control value must change at the intersection between the state trajectory and sliding surface. Assume the nonlinear function $f(x)$ and $g(x)$ are known, the following SMC control input u^* can guarantee the sliding condition of (2.31).

$$u^* = \frac{1}{g(x)}\left[-\sum_{i=1}^{n-1} c_i e^{(i)} - f(x) + y_m^{(n)} - (\eta + D)\operatorname{sgn}(s)\right] \tag{2.32}$$

where

$$\operatorname{sgn}(s) = \begin{cases} 1 & \text{for} \quad s > 0 \\ 0 & \text{for} \quad s = 0 \\ -1 & \text{for} \quad s < 0 \end{cases} \tag{2.33}$$

The final control is realized as $u^* = u_{eq} + g(x)^{-1} u_{sw}$.

where

$$u_{eq} = g(x)^{-1}\left[-\sum_{i=1}^{n-1} c_i e^{(i)} - f(x) + y_m^{(n)}\right] \tag{2.34}$$

$$u_{sw} = -(\eta + D)\operatorname{sgn}(s) \tag{2.35}$$

Let the Lyapunov function candidate defined as

$$V_1 = \frac{1}{2} s^2(e) \tag{2.36}$$

Differentiating (2.36) with respect to time, $\dot{V}$ along the system trajectory as

$$\begin{aligned} \dot{V}_1 &= s \cdot \dot{s} \\ &= s \cdot (c_1 \dot{e} + c_2 \ddot{e} + \cdots + c_{n-1} e^{(n-1)} + x^{(n)} - y_m^{(n)}) \\ &= s \cdot (\sum_{i=1}^{n-1} c_i e^{(i)} + f(x) + g(x)u + d(t) - y_m^{(n)}) \\ &\le -\eta \,|\, s \,| \end{aligned} \tag{2.37}$$

Hence the SMC input u^* guarantees the sliding condition of (2.31). The controller includes switching between two control structures: an equivalent control and a discontinuous control. The equivalent control governs the system dynamics when it state are on the sliding plane. The discontinuous control handles the uncertainties. The phenomenon of chatting is caused by the discontinuous control action and makes the control signal unnecessary large. The general approach of SMC is illustrated in Figure 2.2. A common method to remove chattering is to use the boundary layer [DeCarlo 1988, Slotine 1991]. This method uses a continuous control to replace the discontinuous control when the system states are inside a pre-defined boundary layer such that chattering is eliminated. However, a finite steady-state error must exist and asymptotically cannot be guarantee. In Chapter 5, the use of adaptive fuzzy system with PI control to form a new adaptive fuzzy SMC with chattering elimination will be discussed.

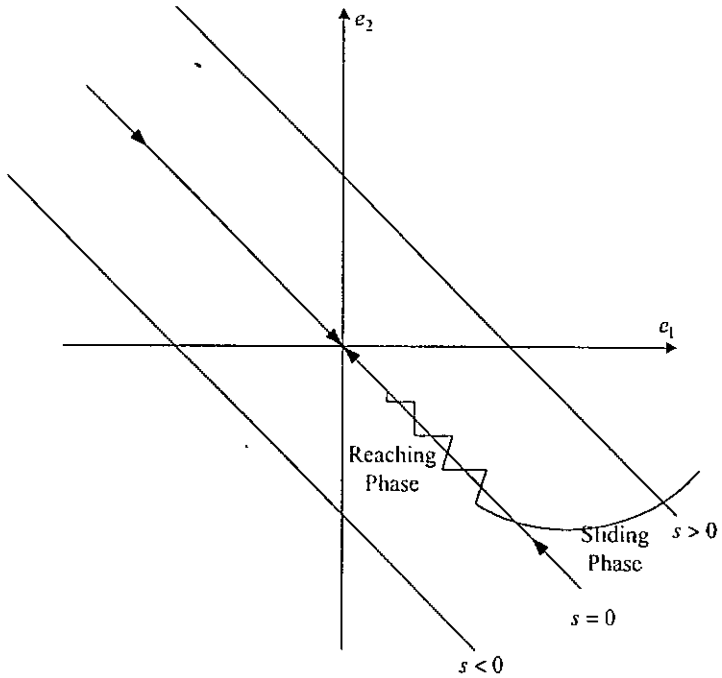


Figure 2.2 Sliding condition in two dimensional error states space

2.3.3 H^{∞} Control

One of the major innovations of control theory has been the development of the H^{∞} control by George Zames [Zames 1981, Francis and Zames 1984], which addresses the issue of the worst-case controller design for systems subjected to unknown disturbances and system uncertainties. The objective of a H^{∞} controller is to find a feedback control law $u = \alpha(x)$ such that the L_2 gain from an exogenous disturbance signal, $w(x)$, to an output signal, $z(x)$, is minimized. Consider the system

$$\dot{x} = f(x) + g(x)w$$
$$z = h(x) \tag{2.38}$$

For some definite function $V(x)$, the Hamilton-Jocobi inequality [Khalil 2003]

$$\frac{\partial V}{\partial x} f(x) + \frac{1}{2\gamma^2} \frac{\partial V}{\partial x} g(x) g^T(x) \left(\frac{\partial V}{\partial x} \right)^T + \frac{1}{2} h^T(x) h(x) \leq 0 \tag{2.39}$$

such that the L_2 gain of system must be less than or equal to γ. Consider

$$V_x f(x) + V_x g(x) w = -\frac{1}{2}\gamma^2 \left\| w - \frac{1}{\gamma^2} g^T(x) V_x^T \right\|^2 + V_x f(x) + \frac{1}{2\gamma^2} V_x g(x) g^T V_x^T + \frac{1}{2}\gamma^2 \|w\|^2 \tag{2.40}$$

where $V_x = \frac{\partial V}{\partial x}$. From equation (2.40), equation (2.39) become

$$V_x f(x) + V_x g(x) w \leq \frac{1}{2}\gamma^2 \|w\|^2 - \frac{1}{2}\|z\|^2 - \frac{1}{2}\gamma^2 \left\| w - \frac{1}{\gamma^2} g^T(x) V_x^T \right\|^2 \leq \frac{1}{2}\gamma^2 \|w\|^2 - \frac{1}{2}\|z\|^2 \tag{2.41}$$

Differentiating V with respect to time, $\dot{V}$ along the system trajectory as

$$\dot{V} = V_x(f(x) + g(x)w) \leq \frac{1}{2}\gamma^2 \|w\|^2 - \frac{1}{2}\|z\|^2 \tag{2.42}$$

$$V(x(t)) - V(x_0) \leq \frac{1}{2}\gamma^2 \int_0^t \|w\|^2 d\tau - \frac{1}{2}\int_0^t \|z\|^2 d\tau \tag{2.43}$$

implies

$$\frac{1}{2}\int_0^t \|z\|^2 d\tau \leq \frac{1}{2}\gamma^2 \int_0^t \|w\|^2 d\tau + 2V(x_0)$$
$$\Rightarrow \|z\|_{L_2} \leq \gamma \|w\|_{L_2} \tag{2.44}$$

In the special case of linear time-invariant systems

$$\dot{x} = Ax + Bw$$
$$z = Cx \tag{2.45}$$

Equation (2.26) can be obtained

$$V(x) = \frac{1}{2}x^T Px \tag{2.46}$$

the solution of which can be obtained by solving the following algebraic Riccati equation:

$$PA + A^T P + \frac{1}{\gamma^2} PB^T BP + C^T C \le 0 \tag{2.47}$$

the valid solutions are ones in which $P = P^T > 0$.

2.3.4 Backstepping Control

Backstepping control [Kanellakopoulos *et al.* 1991, Krstic *et al.* 1995] is a systematic method for nonlinear control design, which can be applied to a broad class of systems. Backstepping refer to the recursive nature of the design procedure.The main idea is to start out with the stabilization problem for a first-order subsystem based on Lyapunov stability techniques and step-by-step increase the order of the subsystem considered. The idea of backstepping is briefly reviewed in this section. Consider the system

$$\dot{x}_1 = f_1(x_1) + x_2$$
$$\dot{x}_2 = u \tag{2.48}$$
$$y = x_1$$

where $f_1(x_1)$ is known function. $x_1, x_2 \in R$ are the state variables and $u \in R$ is the control input. Define the first backstepping variable $z_1 = x_1$, if x_2 as a virtual control input, choosing $x_2 = -x_1 - f_1(x_1)$ would achieve global stabilization of x_1. However, since x_2 is not the actual control input, the virtual control is introduced as $z_2 = x_2 - \alpha_1$, where α_1 is stabilizing function to be specified later. Hence z_1 subsystem can be written as

$$\dot{z}_1 = f_1(z_1) + \alpha_1 + z_2 \tag{2.49}$$

Consider the Lyapunov function candidate for this system

$$\begin{aligned} V_1(z_1) &= \frac{1}{2} z_1^2 \\ \dot{V}_1(z_1) &= z_1 \dot{z}_1 \\ &= z_1(f_1(z_1) + \alpha_1) + z_1 z_2 \end{aligned} \tag{2.50}$$

Choose the stabilizing function

$$\alpha_1 = -f_1(z_1) - k_1 z_1, \quad k_1 > 0 \tag{2.51}$$

such that it stabilizes the z_1 subsystem

$$\dot{V}_1 = -k_1 z_1^2 + z_1 z_2 \tag{2.52}$$

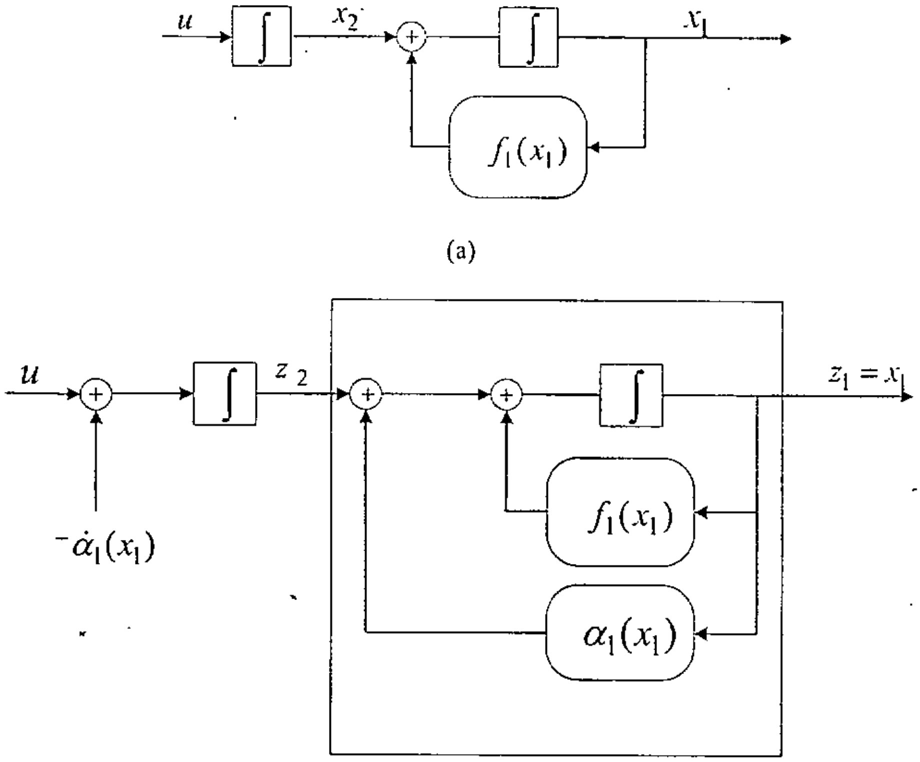
If $z_2 = 0$ then the z_1 subsystem is stable. Consider the Lyapunov function candidate for z_2 subsystem

$$\begin{aligned} V_2 &= V_1 + \frac{1}{2} z_2^2 \\ \dot{V}_2 &= \dot{V}_1 + z_2 \dot{z}_2 \\ &= -k_1 z_1^2 + z_1 z_2 + z_2 \dot{z}_2 \\ &= -k_1 z_1^2 + z_2(\dot{z}_2 + z_1) \\ &= -k_1 z_1^2 + z_2(u - \dot{\alpha}_1 + z_1) \\ &= -k_1 z_1^2 - k_2 z_2^2 < 0, \quad (z_1, z_2) \neq (0,0) \end{aligned} \tag{2.53}$$

if u is chosen as $u = \dot{\alpha} - z_1 - k_2 z_2, k_2 > 0$ where

$$\begin{aligned} \dot{\alpha}_1 &= -\frac{\partial f_1(x_1)}{\partial x_1} \dot{x}_1 - k_1 \dot{x}_1 \\ &= -\left(\frac{\partial f_1(x_1)}{\partial x_1} + k_1 \right) (f_1(x_1) + x_2) \end{aligned} \tag{2.54}$$

Figure 2.3 illustrate the approach of backstepping control [Krstic *et al.* 1995].



(b)

Figure 2.3 (a) The block diagram of the system (2.48) (b) "backstepping" of $-\alpha_1$ through the integrator and the feedback loop in the box with α_1

Thus, by recursively applying backstepping, global stabilizing control laws can be constructed for systems of the following pure-feedback system [Krstic *et al.* 1995]

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2) \\ \dot{x}_2 &= f_2(x_1, x_2, x_3) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, x_2, \cdots, x_n, u) \end{aligned} \tag{2.55}$$

Also systems which can be written on strict-feedback form can be handled

$$\begin{aligned} \dot{x}_1 &= f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 &= f_2(x_1, x_2) + g_2(x_1, x_2)x_3 \\ &\vdots \\ \dot{x}_n &= f_n(x_1, x_2, \cdots, x_n) + g_n(x_1, x_2, \cdots, x_n)u \end{aligned} \tag{2.56}$$

strict-feedback systems are also called triangular systems. In chapter 9, the use of robust adaptive fuzzy control by backstepping will be discussed for a class of strict-feedback system in the presence uncertainties.

2.4 Soft-computing methods

The application of soft-computing theory (fuzzy system, neural network and genetic algorithm) to the solution of control problems has been the focus of numerous studies and research [Linkens *et al.* 1996, Song *et al.* 1996, Zadeh 1994]. The motivation is often that the soft-computing an alternative way to the traditional modeling, optimization and design of control systems when system knowledge and models in the traditional sense are uncertain, inaccurate or complex.

2.4.1 Artificial Neural Networks

Since McCulloch and Pitts in 1943 [McCulloch and Pitts 1943] studied the potential and capabilities of the interconnection of several basic components based on the model of neuron. Neural system has become the focus of many research investigations by scientists, mathematicians and engineers from all around the world. Hebb [Hebb 1949] was concerned with the adaptation algorithm involved in neural system. Subsequently, Rosenblatt [Rasenblatt 1958] coined the term Perceptron and devised its architecture.

Neural network are a class of technology attempt to mimic the theoretical operation of the human brain as opposed to following the pre-programmed rules of a sequential digital computer inspired by the ability of learning of solving problems especially those which are not amenable to convection programming. From control theory point of view, the advantage of neural network is the ability to represent nonlinear mappings. Therefore, a nonlinear modeling in the form of simulation date or experimental data can be represented by a neural network system, which is the feature to be most readily exploited in the synthesis of nonlinear system.

In this section, the structure of the feed-forward neural network as shown in figure 2.4

for nonlinear mapping will be considered. The basic elements neurons as shown in the figure are grouped into several layers: input layer, hidden layer(s) with sigmoid function, output layer. Thus, a multi-layer network in Figure 2.5 is usually used for nonlinear mapping applications. The activation function used here is the standard sigmoid function $f(x) = 1/1 + e^{-x}$ with range between 0and 1. Other choice of activation functions can be used if it is required [Freeman and Skapura 1991].

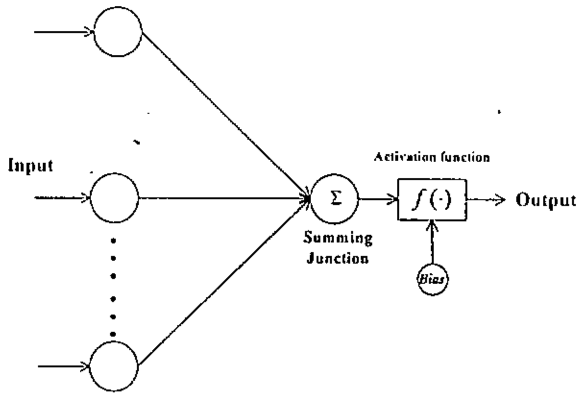


Figure 2.4 Single neuron

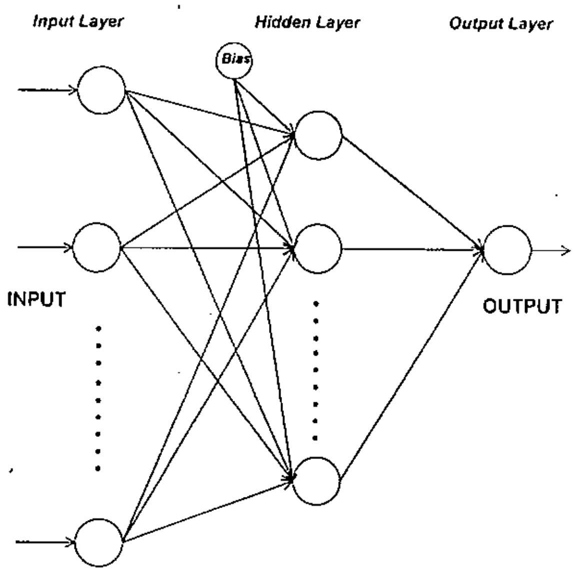


Figure 2.5 General feed-forward network

The step by step description for a feed-forward neural network are as follows [Freeman and Skapura 1991]:

Step 1: Calculate the NN input to the hidden layer units

$$\text{net}_j^h = \sum_{i=1}^{n} W_{ji}^h \cdot x_i + b_j^h \tag{2.57}$$

where x_i is the input vector, n is the number of input node of the network, j is the j^{th} hidden unit, h superscript refers to quantities on the hidden layer and b is the bias term.

Step 2: Calculate the output from the hidden layer

$$i_j = f_j^h(\text{net}_j^h) \tag{2.58}$$

where f^h is sigmoid function

Step 3: Calculate the output layer receives a hidden layer input

$$\text{net}_k^o = \sum_{j=1}^{L} W_{kj}^o \cdot i_j \tag{2.59}$$

where L is the number of hidden node of the network, k is the k^{th} output unit, o superscript refers to quantities on the output layer

Step 4: Calculate the output

$$y_k = f_k^o(\text{net}_k^o) \tag{2.60}$$

where f^o is linear activation function.

From the control theory point of view the ability of neural networks to deal with nonlinear systems is significant. Hence to model nonlinear systems is the feature to be most readily exploited in the synthesis of nonlinear controller.

2.4.2 Fuzzy Systems

Fuzzy logic theory was first introduced by Lotfi. A. Zadeh in 1965 [Zadeh 1965] as a generalization of classical binary logic theory. Unlike the characteristic function of a convectional set, which take values in True and False or 0 and 1, the membership function of a fuzzy set can takes values anywhere between 0 and 1. This multi-value logic is the basis of fuzzy logic systems. Fuzzy logic theory gave a strong impulse to the area and provided the catalyst for much of the subsequent research and commercial products in this field. [Mendel 1995, Jamshidi *et al.* 1993, Sugeno 1985].

Fuzzy control, especially in nonlinear control area, is considered as one of the most attractive strategies in solving control and modeling of complex systems. In term of a logical point of view, a fuzzy system is a decision making machine that has a knowledge base composed of fuzzy IF-THEN rules. Therefore, a nonlinear system with inaccurate and/or uncertain knowledge of system can translated into fuzzy system easily. In case the mathematical model of the controlled system is difficult to obtain, the important information comes from numerical measurements of system variables or human experts who provide linguistic knowledge about the system. Hence, fuzzy systems provide a systematic and efficient framework for incorporating linguistic fuzzy information from human experts. In term of mathematical point of view, Li. X. Wang [Wang 1992] coined the name fuzzy basis functions (FBF) and fuzzy system can be regarded as a universal approximator. Therefore, a nonlinear system in the form of experimental data or expert knowledge can be sufficient formulated by a fuzzy system. In contrast to convectional control, a fuzzy control system is one, which designed based on the heuristic and experience knowledge of experts, while the convectional control design starts with mathematical model which describe the behavior of a controlled system.

The fuzzy design approach ability to provide explicit model-free control of complex system, particularly from expert knowledge or experimental data would be sufficient for formulation and available for compilation of the fuzzy IF-THEN rule base [Mamdani 1974, Sugeno 1985]. Such designs were *ad hoc* and lacked the property robustness and stability analysis of the closed-loop controlled system. On the other hand, modern control theory is mature branch design method which is powerful for its rigorous mathematics, convincing stability proof and reliable system analysis.

In 1974, the earliest application of fuzzy system was to control a steam engine [Mamdani 1974], where the fuzzy IF-THEN rules were obtain from expert knowledge and linguistic concept. The fuzzy logic theory and its practical realizations have become popular and generated unprecedented research interests due to the impressive growth in consumer products and industrial system from early 1990's [Terano *et al.* 1994 and Sugeno 1985] etc, where no accurate mathematical models of the system are available, but human experts are available to provide linguistic information about the system. Although, fuzzy theory is widely applied in industrial systems (blast furnace control, temperature control, automatic train operation, speech and image recognition, *etc* [Terano *et al.* 1994, Moon *et al.* 2004]. Fuzzy control has not been viewed as rigorous due to a lack of formal synthesis techniques which guarantee the basic requirements for control systems such as robustness and global stability.

Earlier research on the stability analysis of the model base fuzzy control systems required a mathematical model of the plant [Tanaka 1992]. This contradicted the model free or poor understanding of the control processes. In fact, if the mathematical model of a system is known, then the modern model control methods should be given a higher priority. Wang [Wang 1994] provided an in-depth and through analysis of adaptive fuzzy system. This work gave a strong impulse to fuzzy system technology. Especially, base on the Universal Approximation Theorem, an adaptive fuzzy control method can provide stabilizing controller even for nonlinear system with dominant nonlinearities.

The fuzzy system regarded as knowledge based or rule based system. The main part of a fuzzy system is a knowledge base consisting of the fuzzy IF-THEN rules. A fuzzy rule is an IF-THEN statement in which some words are characterized by fuzzy membership functions. We will consider a fuzzy system consisting of the product-inference value, singleton fuzzifier, center average defuzzifer, and Gaussian membership function. Then the resulting fuzzy system will be denoted as a linear combination of adjustable parameters and fuzzy basis functions. Figure 2.6 shows an adaptive fuzzy system. An adaptive fuzzy system is defined as a fuzzy system equipped with a adaptation or learning algorithm where the fuzzy system is developed from a set of fuzzy IF-TEHN rules, and the adaptation algorithm adjusts the parameters of the fuzzy system based on the training information. Adaptive fuzzy systems can be viewed as fuzzy systems whose fuzzy rules are automatically generated or modified through training process. There are four principle components for fuzzy system: Fuzzifer: Mapping the measured crisp input variables into fuzzy sets described in linguistic expression. Fuzzy Inference Engine: Implementing the fuzzy implication relation expressed by the fuzzy IF-THEN rules. Fuzzy Rule Base: Representing the collection of a fuzzy rule base or a data base. Defuzzifier: Converting the fuzzy values into the crisp output variable.

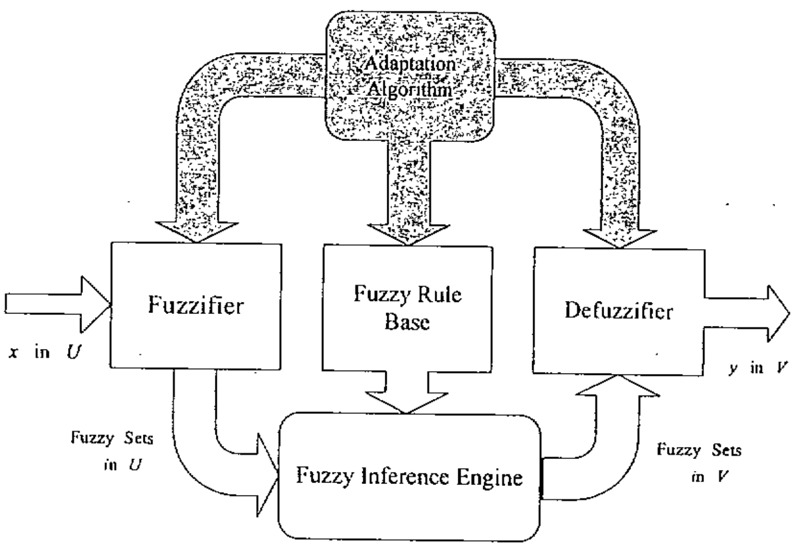


Figure 2.6 Adaptive fuzzy system

2.4.2.1 Mamdani type fuzzy system

The basic configuration of a fuzzy system consists of a collection of fuzzy IF-THEN rules:

$$R^{(l)}: \text{IF } x_1 \text{ is } A_1^l \text{ and } \cdots \text{ and } x_n \text{ is } A_n^l$$
$$\text{Then } y \text{ is } B^l$$

The fuzzy system performs a mapping from $U = U_1 \times \cdots \times U_n \subseteq R^n$ to $V \subseteq R$, where $x = [x_1, \cdots, x_n]^T \in U$ and $y \in V \subseteq R$ are the input and output of the fuzzy system, respectively. A_i^l and B^l denote the linguistic variables of the input and output of the fuzzy set in U and V, respectively. In general, there are many different choices for the design of fuzzy system if the mapping is static. More detailed information of these fuzzy systems can be found in [Wang 1996]. Usually, fuzzy systems with a single value consequent are regarded as Mamdani type fuzzy systems.

The fuzzy logic systems with singleton fuzzifier, product inference engine, center average defuzzifier are in the following form,

$$f(x) = \frac{\sum_{l=1}^{m} y^l \left(\prod_{i=1}^{n} \mu_{A_i^l}(x_i) \right)}{\sum_{l=1}^{m} \prod_{i=1}^{n} \mu_{A_i^l}(x_i)} \tag{2.61}$$

where $\mu_{A_i^l}(x_i)$ is the membership function of the linguistic variable x_i, and y^l represents a crisp value at which the membership function μ_{B^l} for output fuzzy set reaches its maximum. By introducing the concept of fuzzy basis function vector or the antecedent function vector. Equation (2.61) can be rewritten as

$$f(x) = \theta^T \xi(x) \tag{2.62}$$

$$\xi^l(x) = \frac{\prod_{i=1}^{n} \mu_{A_i^l}(x_i)}{\sum_{l=1}^{m} \prod_{i=1}^{n} \mu_{A_i^l}(x_i)} \tag{2.63}$$

where $\theta = [y^1, \cdots, y^m]^T \in R^m$ is called the parameter vector and $\xi(x) = [\xi^1(x), \cdots, \xi^m(x)]^T \in R^m$ is called the fuzzy basis function vector. One of the most important advantages of fuzzy logic system is that the fuzzy logic system has the capability to approximate nonlinear mappings. The fuzzy logic system described above is for single-output system. However, it is straightforward to show that a multi-output system can always be approximated by a group of single-output approximation systems.

2.4.2.2 Takagi-Sugeno (TS) Type Fuzzy system

The Takagi-Sugeno fuzzy system [Takagi and Sugeno, 1985] consists of fuzzy IF-THEN rules with the following structure

$$\begin{aligned} R^{(l)}: \text{IF } x_1 \text{ is } A_1^l \text{ and } \cdots \text{ and } x_n \text{ is } A_n^l \\ \text{Then } y \text{ is } a_0^l + a_1^l x_1 + \cdots + a_n^l x_n \end{aligned} \tag{2.64}$$

where $a_i^l, i = 0,1,\cdots,n, l = 1,2,\cdots,m$ are the unknown constant parameters. By using singleton fuzzifier, product inference engine, center average defuzzifier, the final output value is

$$f(x) = \frac{\sum_{l=1}^{m} y^l (\prod_{i=1}^{n} \mu_{A_i^l}(x_i))}{\sum_{l=1}^{m} \prod_{i=1}^{n} \mu_{A_i^l}(x_i)} \tag{2.65}$$

where $y^l = a_0^l + a_0^l x_1 + \cdots + a_n^l x_n$ and $\xi^l(x) = \prod_{i=1}^{n} \mu_{A_i^l}(x_i) / \sum_{l=1}^{m} \prod_{i=1}^{n} \mu_{A_i^l}(x_i)$ is the fuzzy basis function. Equation (2.65) can be rewritten as

$$f(x)=\sum_{l=1}^{m} y^l \xi^l(x) \tag{2.66}$$

Let $\xi(x)=[\xi^1(x),\cdots,\xi^m(x)]$, $Z=[1,x_1,x_2,\cdots,x_n]^T$, $A_z=\begin{bmatrix} a_0^1 & a_1^1 & \cdots & a_n^1 \\ a_0^2 & a_1^2 & \cdots & a_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ a_0^m & a_1^m & \cdots & a_n^m \end{bmatrix}$, then

equation (2.65) can be rewritten as

$$f(x)=\xi(x)A_z Z=\xi(x)A_z^0+\xi(x)\overline{A}_z x \tag{2.67}$$

where $x=[x_1,x_2,\cdots,x_n]^T$, $A_z^0=[a_0^1,a_0^2,\cdots,a_0^m]^T$ and $\overline{A}_z=\begin{bmatrix} a_1^1 & a_2^1 & \cdots & a_n^1 \\ a_1^2 & a_2^2 & \cdots & a_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ a_1^m & a_2^m & \cdots & a_n^m \end{bmatrix}$.

The consequent of TS fuzzy system rule is a linear system. This linear system is very useful in designing identification and control as compared with a single value consequent The main advantages of TS fuzzy system is that a stability analysis base on the linear matrix inequality (LMI) techniques for the fuzzy systems. A sufficient condition to ensure the stability of the overall closed-loop is to find a common Lyapunov function for all fuzzy sub-systems [Tanaka 1992, 1996].

Theorem 2.1: (Universal Approximation Theorem) [Wang 1997]

For any given real continuous function f on a compact set $U\subset R^n$ and arbitrary $\varepsilon>0$, there exists a fuzzy system $f^*(x)$ in the form of equation (2.49) or equation (2.54) such that

$$\sup_{x\in U}\left\|f^*(x)-f(x)\right\|\le\varepsilon \tag{2.68}$$

A general fuzzy control design does not rely on fuzzy model or adaptation algorithm in the control system and only implement in term of trial and error approach. Hence, its design and implementation can be very simple. Much work on fuzzy control is base on PID control and sliding mode control (SMC). Later on, researchers used the well-established design method and analysis of modern control theory to aid the design of fuzzy control. A brief review will be discussed in the following three sub-sections.

2.4.2.3 Fuzzy PID Control

The fuzzy system is designed as a controller related to the PID or three term controller [Ying 1990, Mizumoto 1995]. The controlled plant is a nonlinear dynamical system, the input variable are the output error, the rate of change or error and the time integral of the error. The output variable is the rate of change of the control value. The properties of these PID-like fuzzy controller can be regarded as a PID control with nonlinear gains. During the control system design, the shape and the distribution of the input and output membership functions to be non-uniformly spaced in the domain of interested. It is difficult to apply fine tuning to specific membership functions without affecting others. Moreover, the rule base of the fuzzy controller is pre-defined. Such pre-defined structure may not lead to a satisfactory performance in time-varying environments, the control rules are refined after a number of trial-and-error cycles by experienced human expert. The system stability of these types of fuzzy control is difficult to ensure.

2.4.2.4 Fuzzy Sliding Mode Control (FSMC)

Once the design and analysis of the fuzzy control is merged with the sliding mode control (SMC), it can be regarded as fuzzy sliding mode control. The first kind of fuzzy sliding control is based on the state space of input error and/or change of error. The state space partitions approach is usually used [Palm 1994, Kim and Lee 1995] with the properties of avoiding the chattering problems. The state space is divided into many small partitions according to the input membership functions. The analysis basically carried out for all partitions, the design of such fuzzy control is time consuming. Moreover, these fuzzy sliding mode controllers are difficult to apply to higher order systems with more that two states. The second kind of fuzzy sliding mode control with the input s , s is a variable proportional to the distance between the state vector and the sliding surface. These methods are similar to a convectional sliding mode control with a boundary layer about the sliding surface. These fuzzy sliding mode control can be handle a high order system and stability analysis can be carried out by using Lyapunov's analysis in term of the sliding surface and assume the system model is exactly known.

2.4.2.4 Self-Organizing and Adaptive Fuzzy Control

Self-organizing controller (SOC) introduced by Procky and Mandani [Procky and Mandani 1979], which possesses self-learning and adaptation for the knowledge base such that a given performance was minimized. The adaptation basically use a performance evaluation and fuzzy system to access the controller's performance and adapt the fuzzy relations in the controller. Borrowing ideas from convectional model reference adaptive, fuzzy model reference learning control (FMRLC) [Passino *et.al.* 1993] is introduced to deal with nonlinear control problems with adjustment of the fuzzy control rule by using a stable reference model and a fuzzy inverse model. However, these fuzzy controller usually focus on improving the performance; the system stability may not be guaranteed.

An adaptive fuzzy control with stability consideration was proposed by Wang [Wang 1994]. According to modern adaptive control approach, model based fuzzy adaptive controllers are designed by using certainty equivalent principle [Slotine 1991, Åström 1995]. The unknown plant parameters are estimated by adaptive fuzzy systems. The stability analysis is used to prove the closed-loop system and performance. The disadvantage of this method is that the robustness and zero steady state error of the control system cannot be guaranteed.

2.5 Conclusions

In this chapter, an attempt was made to give a informative summary and a literature survey on various topics that will be used throughout this thesis. Some remarks and discussions on the modern control theories have also been elaborated. Various types of control schemes based on combining the modern control and the fuzzy system counterparts in the structure have been summarized. This coverage has been selective and by no means claimed to be exhaustive.

CHAPTER THREE

ON-LINE LOWER ORDER MODELING VIA NEURAL NETWORKS

3.1 Introduction

The ubiquitous Proportional-Integral-Derivative (PID) controller is regarded as a jack of all trades in many process industries and is the most widely used controller due to its simple structure, easy implementation and robust performance. However, the dynamics of many systems exhibit complex characteristics such as non-linearity, time-varying parameters, as well as time-delay, *etc.* This often leads to a poor control performance if one uses a conventional fixed parameters PID controller to handle the above mentioned problems. A possible way to circumvent the control problem in these situations is to employ some form of adaptive control. There are some alternatives for tuning PID controllers adaptively. Self-Tuning Controllers (STC) and Model Reference Adaptive Controllers (MRAC) are reported in literature whereby the parameters of a PID controller are adjusted on-line [Astrom *et al.* 1993, Smith and Corripio 1997, Rad *et al.* 1997,Ho *et al.* 2000].

In this chapter, we present an on-line PID tuning controller based on the parameters of a *first-order plus dead-time* (FOPDT) model, which are obtained by using neural networks. Here the complete knowledge of the system may not be required especially if a fixed structure controller like a PID is to be employed. PID structure is inherently a second-order and as such can not exploit the higher dynamics, which is provided by complete system identification. A lower-order model of the system is sufficient and can be obtained with less complexity and computation. Based on this premise, the proposed on-line approximating approach is aimed at identifying a lower-order model for the purpose of the controller design.

This chapter is based on the papers 1, 3, 5 and 9, 10 on chapter one, pages 11 and 12

The rest of this chapter is organized as follows: The approximating method of high-order systems with a first-order plus dead time using neural networks is described in Section 3.2. The proposed on-line PID tuning control method is derived in Section 3.3. Simulation studies for a control system undergoing dynamic change are included in Section 3.4. The performance of the proposed algorithm for real-time flow control of a laboratory test rig is evaluated in Section 3.5. Finally, the chapter is concluded in Section 3.6.

3.2 Lower-order Approximation of Systems with Neural Networks

The discussion will be restricted to Single-Input / Single-Output (SISO) systems. It is well known that a higher order system can be approximated by a first order plus dead time model in form of:

$$\frac{Y(s)}{U(s)} = e^{-s\vartheta} \frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_m}{s^n + a_1 s^{n-1} + \cdots + a_n} \approx \frac{K \cdot e^{-\tau s}}{Ts+1} \tag{3.1}$$

Here, K, T, τ are the gain, dominant time constant and the apparent dead-time respectively and $m \le n$ and the system gain is assumed positive $K > 0$. $U(s)$ and $Y(s)$ are the Laplace transformed input and output signals respectively. T is the system time delay and a_i and b_j are the coefficients of the system transfer function. The parameters K, T and τ can be determined from various methods [Ziegler and Nichols 1942, Smith 1972, Sundaresan and Krishnaswamy 1978]. Smith's method [Smith 1972] is a popular approach whereby the parameters are determined from the process reaction curve which is obtained by injecting a step signal to the system. Although, this method is convenient and accurate, however, it should be carried out off-line and it is very sensitive to noise. Furthermore, in case of time varying parameters, this process should be repeated frequently.

3.2.1 On-line Neural Networks modeling

The proposed methodology is inspired by the process reaction curve method which is a popular technique in process control to obtain the step response of a higher-order system. The architecture of the on-line approximating approach is shown in Figure 3.1. As shown in this figure, the control signal *u(t)* is applied to the high-order system, the neural network, and the FOPDT model generator at the same time. The outputs of neural network are the three parameters, namely, the gain K, the time constant T, the dead-time τ of the approximated FOPDT model of the high-order system. These three parameters are sent to the first-order-plus-dead-time model generator to get an output of the model. The error between output of the plant and the output of the model is used to train the weights of the neural networks. The training process tends to force the output of the FOPDT model generator to approximate the output of the system. Thus, the inputs of the FOPDT model generator are the approximating parameters of the first-order model for the high-order system. The output of the FOPDT model is expected to match the output of the high-order system after the neural network converges.

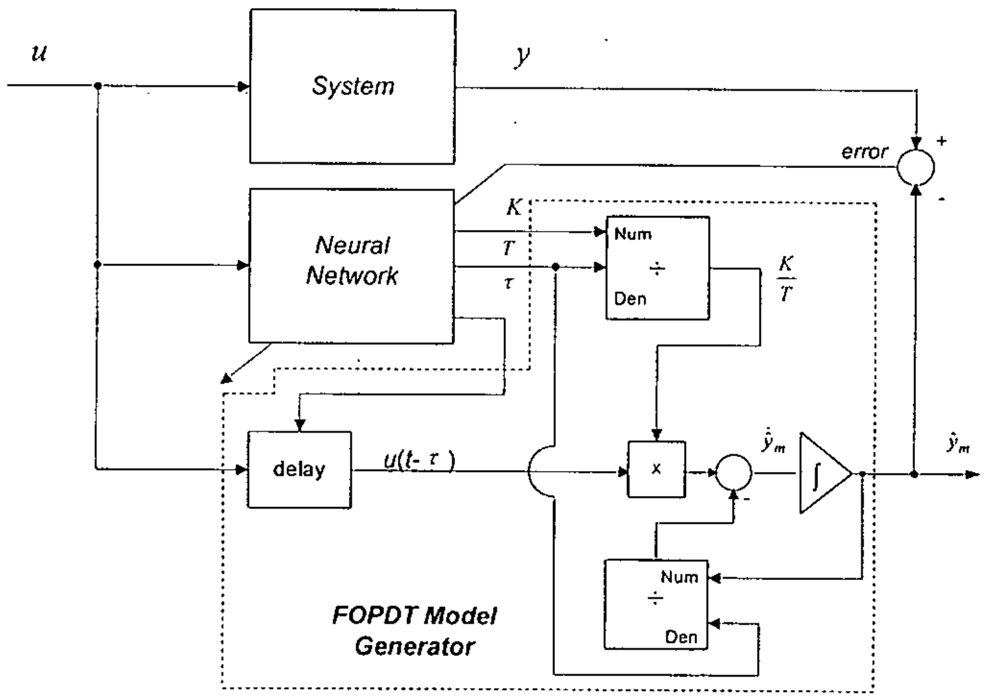


Figure 3.1: The structure of the on-line lower-order modeling of high-order systems

The transfer function of the FOPDT model generator is rewritten below:

$$\frac{\hat{Y}_m(s)}{U(s)} = \frac{K \cdot e^{-\tau s}}{Ts+1} \tag{3.2}$$

According to the convolution theorem, the output of the FOPDT model generator can be obtained in the time domain as:

$$\hat{y}_m(t) = L^{-1}[\frac{K \cdot e^{-\tau \cdot s}}{Ts+1} \cdot U(s)] \tag{3.3}$$

where L^{-1} stands for inverse Laplace transform. Therefore, we get:

$$\hat{y}_m(t) = K \cdot u(t-\tau) - T\frac{d\hat{y}_m(t)}{dt} \tag{3.4}$$

where, *u(t)* is the input of the FOPDT model generator. The model output $\hat{y}_m(t)$ depends on the parameters K, T, and τ at each time instant.

3.2.2 Neural Network Structure and Training Algorithm

The neural network architecture used for plant modeling is a three-layer feed-forward network with *1* node in the input layer, *8* nodes in the hidden layer, and *3* nodes in the output layer. The basic structure, having one hidden layer with sigmoid function, has been shown to be powerful enough to produce an arbitrary mapping among variables [Cybenko 1989]. Thus, a three layers network is usually used for control applications. In general, three layers are sufficient, however if we want the network learns faster (converge faster), the network may more than one hidden layer. In such cases, the computation load will increase [Freeman and Skapura 1991]. The activation function used here is the standard sigmoid function with range between *0* and *1*.

To train the above neural network, a direct learning strategy is employed for on-line training. As the desired outputs of the neural network are unknown, the FOPDT model generator is considered as an additional but not modifiable layer of the neural network. Back-Propagation (BP) algorithm is used to update the weights of the neural network.

The algorithm consists of two passes, forward pass and backward pass. The calculation of the forward pass and updating the connection weights from the input layer to hidden layer are the same as those in the standard BP algorithm. To update the connection weights from the hidden layer to the output layer, the momentum technique [Battiti 1989] is employed. The weight adjustment in each iteration is derived below. The error function E is defined as:

$$E = \frac{1}{2}\sum_{t=1}^{r}(y - \hat{y}_m)^2 \tag{3.5}$$

where r is the number of input/output pairs available for training the network, y and y_m are the output of the plant and the output of the FOPDT model at any time instant t. Within each time interval from t to $t+1$, the BP algorithm is used to update the connection weights, according to the following relationship:

$$W_{ij}(t+1) = W_{ij}(t) - \eta \cdot \frac{\partial E}{\partial W_{ij}(t)} + \alpha \cdot \Delta W_{ij}(t) \tag{3.6}$$

Here, η is the learning rate; α is the momentum factor; Δw_{ij} is the amount of the previous weight change. Using the chain rule, one has

$$\begin{aligned}\frac{\partial E}{\partial W_{ij}(t)} &= \frac{\partial E}{\partial \hat{y}_m(t)} \cdot \frac{\partial \hat{y}_m(t)}{\partial \phi_n(t)} \cdot \frac{\partial \phi_n(t)}{\partial W_{ij}(t)} \\ &= -(y(t) - \hat{y}_m(t)) \cdot \frac{\partial \hat{y}_m(t)}{\partial \phi_n(t)} \cdot X_i(t) \cdot (1 - X_i(t)) \cdot X_j(t)\end{aligned} \tag{3.7}$$

Here, X_i is the output of the i^{th} node of the output layer, X_j is the input vector of the nodes of the j^{th} output layer. $\phi_n(t)$ is $3x1$ input vector of the FOPDT model (the output of vector of the neural network)

$$\frac{\partial \hat{y}_m(t)}{\partial \phi_n(t)} = \frac{\partial \hat{y}_m(t)}{\partial K} \cdot \frac{\partial \hat{y}_m(t)}{\partial T} \cdot \frac{\partial \hat{y}_m(t)}{\partial \tau} \tag{3.8}$$

Next problem is to find $\frac{\partial \hat{y}_m(t)}{\partial K}$, $\frac{\partial \hat{y}_m(t)}{\partial \tau}$ and $\frac{\partial \hat{y}_m(t)}{\partial T}$ which are the partial derivatives of the output $\hat{y}_m(t)$ of the model generator (FOPDT) w.r.t. gain (*K*), dominant time constant (*T*) and apparent dead-time (τ), respectively, are given by

$$\frac{\partial \hat{y}_m(t)}{\partial K} = L^{-1}\left[\frac{e^{-\tau s}}{Ts+1}u(s)\right] \tag{3.9}$$

$$\frac{\partial \hat{y}_m(t)}{\partial T} = L^{-1}\left[\frac{-sKe^{-\tau s}}{(Ts+1)^2}u(s)\right] \tag{3.10}$$

$$\frac{\partial \hat{y}_m(t)}{\partial \tau} = L^{-1}\left[\frac{-sKe^{-\tau s}}{Ts+1}u(s)\right] \tag{3.11}$$

The neural network used in this chapter consists of three output nodes for the first order with time delay model parameters (*K, T, τ*) and one input node for the control signal (*u(t)*). The selection of the number of hidden layers and hidden nodes of the neural network depends on the complexity and the non-linearity of the system parameters with respect to operation range. Selection of hidden layers and hidden nodes is a compromise between accuracy and computation efficiency. After extensive simulation studies of slow time varying higher order systems, we decided to employ a neural network with one hidden layer and 8 to 10 hidden nodes. This architecture was found to track the changes of system parameters and time delay. Simulation and experimental studies indicated that around $10-15\%$ change in system parameter could be tolerated. Large number of hidden layers and hidden nodes trend to slow down the convergence of estimated parameters. In our proposed, 1 hidden layer and 8 hidden nodes are used for the proposed algorithm.

Remarks 3.1:

- The method can be extended to approximate a Second-Order Process with Dead Time (SOPDT) by changing the model generator to a second-order. Such a model is especially useful for oscillatory systems, which are associated with the second-order-underdamped systems. The derivation of the equations (3.9-3.11) is similar to the derivation as outlined above.

- The proposed method can be used for self-regulating processes and can be used for non self-regulating processes (run-away) by modifying the structure of the on-line lower-order approximator. The method is also applicable to non-minimum phase systems. In such cases, the numerator polynomial has zeros in the *rhs* of the *s*-plane. The time delay can incorporate the effect of numerator dynamics in such systems.
- The method can be further simplified by reducing the number of estimated parameters. The gain of the system can be easily calculated at steady-state conditions as the ratio of output over input excitation.

The proposed controller can be regarded as a gain-scheduling adaptive PID controller for which its parameters are scheduled according to input variable $u(t)$. As such, the performance of the controller depends on the dynamic of input signal [Rough 1997].

3.3 On-line PID Tuning Control Method using Neural Network

The control structure for the on-line PID tuning is shown in Figure 3.2 There are two parts in the control structure of the on-line PID tuning method. The first part, which was described in the previous section, is the approximation of high-order systems with FOPDT using neural networks, and the second part is the design of the PID controller. Normally, the parameters of a PID controller can be obtained after the corresponding parameters of a FOPDT model of the high-order system are known. There are many tuning methods based on the parameters of FOPDT model, such as, Ziegler-Nichols (ZN) ultimate cycle tuning formulae [Ziegler and Nichols 1942], Minimum Error Integral Tuning Formulas, Minimum IAE, 5% Overshoot tuning formulas [Smith and Corripio 1997], and Refined Ziegler-Nichols tuning formulae [Hang *et al.* 1991]. Although ZN ultimate cycle tuning algorithm is not the best tuning method, it is the most widely known PID tuning algorithm. In order to get better control performance, the so-called refined Ziegler-Nichols tuning method [Hang *et al.* 1991] can be used. In principle, any other tuning method can be integrated with the proposed identification algorithm; however, to demonstrate the merits of this approach, we have used the ZN method. The parameters of the PID controller suggested by ZN algorithm is:

$$K_p = 0.6 * K_u \qquad T_I = 0.5 * T_u \qquad T_d = 0.125 * T_u \tag{3.12}$$

Here, K_p, T_i, T_d, K_u and T_u are the proportional gain, integral time constant, derivative time constant, the ultimate gain and the ultimate period respectively, which are calculated from the FOPDT model of the high-order plant [Smith and Corripio 1997].

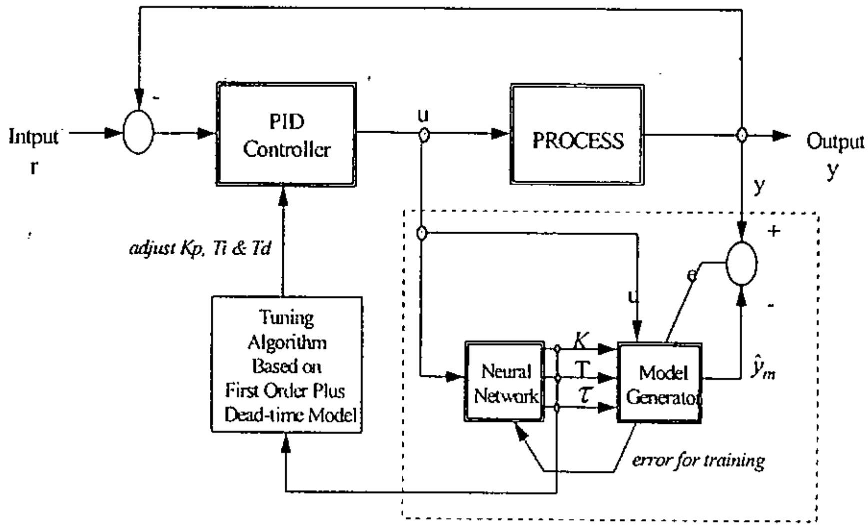


Figure 3.2: Overall structure of the auto-tuning PID controller

The output of the PID controller is in the form of

$$u(t) = K_p(e(t) + \frac{1}{T_i}\int e(t)\,dt + T_d \cdot \frac{dy_f}{dt})$$

$$e(t) = r(t) - y(t)$$

$$y_f(s) = \frac{1}{1 + \frac{T_d}{10}s} y(s) \tag{3.13}$$

where $u(t)$, $y(t)$, $r(t)$, $y_f(t)$ are the controller output, process output, set-point, and filtered derivative. The filtered derivative term with the filter time constant 10 is generally agreed by both academics and practitioners respectively [Åström 1995, Tan 1999].

3.4 Performance of the proposed on-line controller

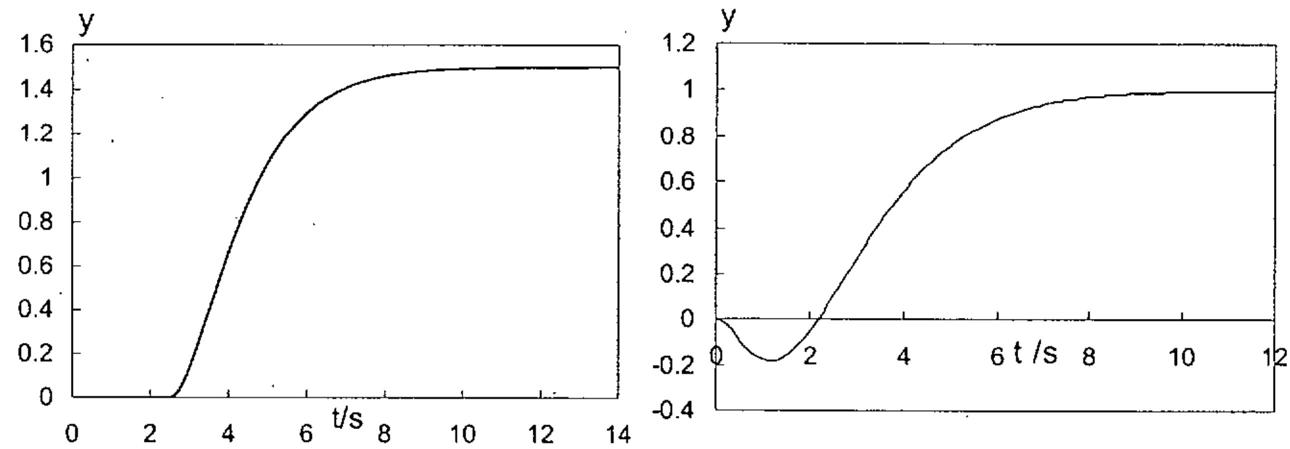
3.4.1 System Variations

To show the adaptive behavior of the algorithm, let us consider two processes as:

Process I $$\frac{Y(s)}{U(s)} = \frac{1.5\,e^{-2.5s}}{(1+s)^2}$$

Process II $$\frac{Y(s)}{U(s)} = \frac{1-1.4s}{(1+s)^3}$$

The first system is a system with time delay and the second process is a non-minimum phase system. If the time delay is approximated by a rational transfer function, it will be in the form of non-minimum phase system. Figure 3.3 shows the open loop response for the two simulated process. In this algorithm, we identify the exact time delay and there is no need for approximating it with a rational transfer function, i.e., to expand the time delay as a higher order rational transfer function approximation. The set point was chosen to be a square-wave with amplitude of *0.5* and a period of *40 s*. In order to get a more realistic environment, a Gaussian noise with mean zero and variance of *0.01* is injected at the output of the system. We employed a fourth-order Runge-Kutta numerical integration algorithm for all time responses and the integration interval was selected to be *0.01s*. The NN also used the same time interval for updating its parameters. The simulation proceeded as follows: the PID controller was initialized with $K_p = 1$, $T_i = 1000$, $T_d = 0.0$. The architecture of the NN was (1,8,3) and a bias term of 0.5 was added to all the hidden nodes. This structure of NN is generic for this algorithm and works well in most cases. The number of hidden layers was the only variable that could be manipulated if needed. The number of hidden layers was determined by a trial and error procedure. The connection weights were randomly initialized. The learning rate and the momentum parameter were set at 0.8 and 0.1 respectively. The updating of the PID started at time $t = 40s$ based on the estimated parameters $\hat{K}, \hat{T}$ and $\hat{\tau}$ (The symbol ^ over a parameter indicates estimated parameter) by the FOPDT neural model. K_u and T_u were then calculated according to these estimated parameters [Smith and Corripio 1997]. The corresponding PID parameters K_p, T_i and T_d were updated from ZN ultimate cycle method [Equation (3.11)]. The control law like other self-tuning methods is based on *certainty equivalent* [Astrom *et al.* 1993], i.e., the controller uses the estimated parameters and does not check the validity of these estimates. However, as the time goes on, the estimated parameters of the approximating FOPDT neural model for the corresponding systems tend to converge to a final value. Therefore, the performance of the controller improves with time. Figure 3.4 shows the overall performance of the proposed algorithm. In this figure, the set-point and the output, the controller signal -output of the PID controller- and the estimated parameters of gain, apparent time delay and the dominant time constant are shown in top, middle and bottom curves respectively.



Step Response of Process I

Step Response of Process II

Figure 3.3: Open loop response for the simulated processes

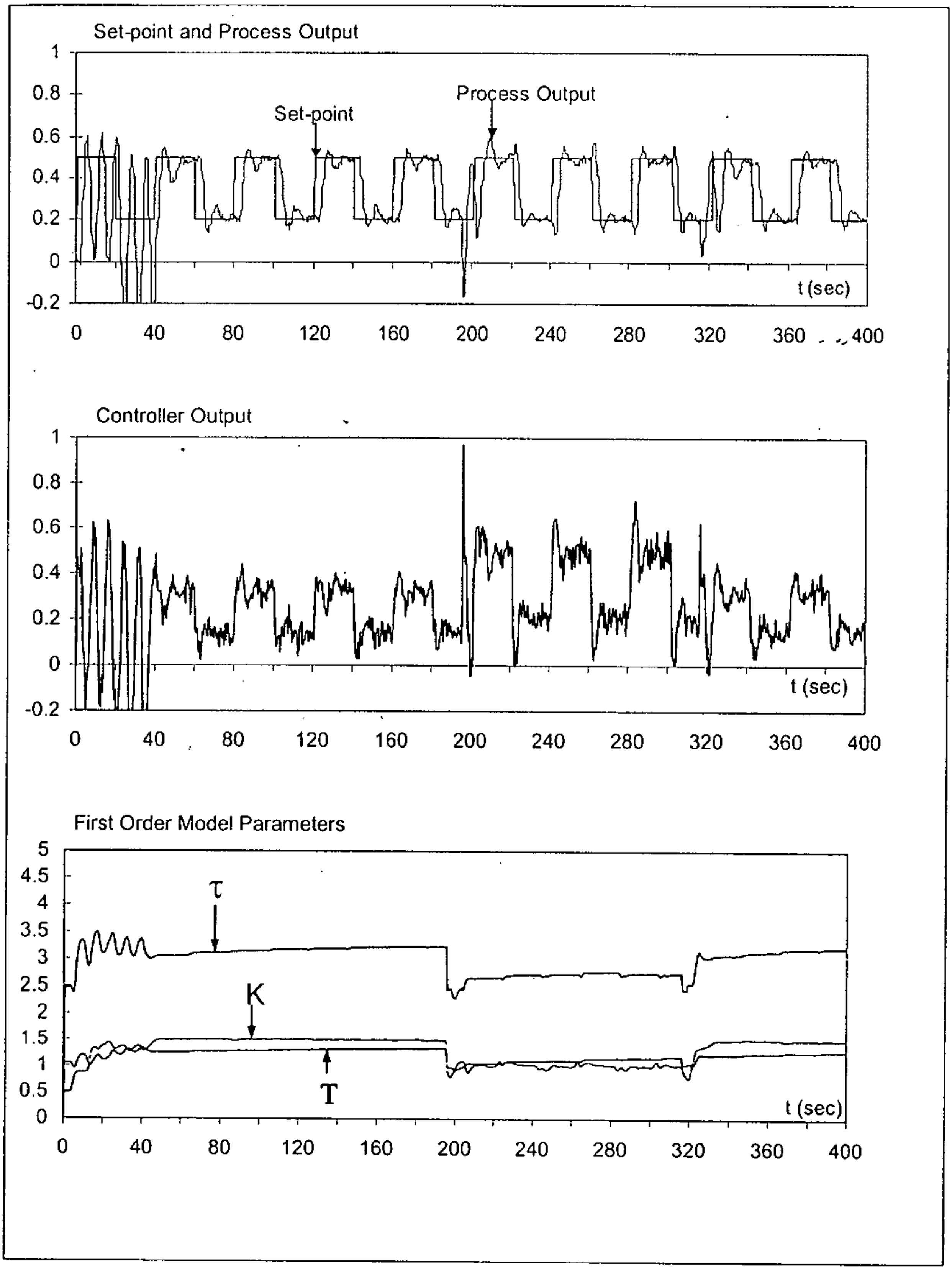


Figure 3.4: Adaptive PID control under noise and system variation

In order to demonstrate the adaptive property of the algorithm, at $t = 195s$, the system was changed to process 2 and again at $t = 320s$, it was switched back to process 1. Furthermore, it should be noted that the gain in system 1 and 2 is different (1.5 and 1). It is known that some adaptive controllers can not cope with change in steady-state gain of the controlled system. However, as it is seen in Figure 3.4, the proposed method can successfully track the system change. These simulation studies demonstrate the adaptive property of the proposed algorithm. In all these system changes, the neural networks converged and the estimated parameters of the FOPDT also converged to their steady-state values and the closed-loop stability was also maintained under these conditions. It is generally accepted that a PID controller tuned by Ziegler and Nichols algorithm exhibits good robustness and stability properties under various conditions. Since the underlying controller in the proposed algorithm is PID, it is expected to share these characteristics. This has already been demonstrated in simulation and later on will be verified in experimental studies. Tables 3.1 show the parameters of FOPDT model approximated by several other methods such as Smith's, minimized error, *etc.*, and the corresponding ultimate gain and the ultimate period for process 1 and 2 respectively. It should be noted that the parameters from all other methods except the proposed algorithm were obtained off-line from an open-loop excitation with unit step and were noise-free. The values quoted for the proposed algorithm is based on the last measurement before each system change and not the average value. Nevertheless, these measurements match the original system very well especially with the non-minimum phase system.

3.4.2 Stability Assessment

In the preceding simulation study, the PID controller was tuned on-line based on the model obtained from the neural network approximator. This model is inevitably based on incomplete information about the dynamic behavior of the process. In the simulation study, the actual system was a second-order with time delay, which was switched to a third order non-minimum phase system. In practical process control applications, the parameters may change due to many factors such as failure or deterioration of system components, driving components and transducers. This is related to robustness of the control system and in turn affects the stability of the overall system. It is therefore a must for the final controlled system not to be sensitive to parameter variations in order to ensure a robust and stable performance. Here, we address this problem in a rather unconventional approach. We study a statistical measure of probability of instability of the closed-loop system via *Monte Carlo simulation*.

	Process I					**Process II**				
	K	**T**	**τ**	**K_u**	**T_u**	**K**	**T**	**τ**	**K_u**	**T_u**
Smith Method	1.5	1.65	3.00	1.06	8.38	1.0	1.89	2.43	1.93	7.22
Minimized-error	1.5	1.46	3.11	0.98	8.44	1.0	1.67	2.55	1.74	7.35
Tsang-Rad	1.5	1.5	3.09	1.0	8.45	1.0	1.78	2.44	1.85	8.45
Proposed method	1.49	1.33	3.23	0.95	8.56	0.99	1.22	2.61	1.47	7.08
Theoretical calculations*	-	-	-	1.04	8.44	-	-	-	1.54	6.83

*Values obtained from theoretical calculations

Table 3.1: Parameters for process I & II

The stability of the closed-loop system under the adaptive PID controller given by Table 3.1 are determined by calculating the eigenvalues of the corresponding characteristic equation. A Monte Carlo simulation is used to estimate the probability of instability of the closed-loop system [Ray and Stegenl 1993]. The closed-loop eigenvalues are evaluated J times with each element of system parameters specified by a random generator whose individual output are shaped by a desired probability density function. The estimate probability of stability will be more accurate as J tends to infinity. The estimated probability of instability is defined as the number of cases with right half plane roots over total number of evaluation J.

$$Pr(unstable) = 1 - Pr(stable) \qquad where \qquad Pr(stable) = \lim_{j \to \infty} \frac{N(\sigma_{max} \leq 0)}{j}$$

The numerator term denotes the number of cases for which all real part of eigenvalues remains in the left half plane. The subscript "max" indicates the maximum value of real part of system eigenvalues. σ_{max} is the maximum real part of all the roots of the closed-loop characteristic equation $N(\sigma_{max} < 0)$ is the number of case for which all the roots of the characteristic equation lie in the left-half s-plane. A step by step procedure for carrying out the Monte Carlo simulation is as follows:

(1) *Initialize a random generator*

(2) *Give a random fluctuation of ±25% for each system parameter of the system model.*

(3) *Derive the loop transfer function $L(s)=G(s)G_c(s)$ where $G(s)$ and $G_c(s)$ are model and PID controller transfer function, search the gain cross-over frequency of $L(s)$ and estimated the gain margin ϕ_m.*

(4) *Repeat step 2 and 3 for J iteration, count the number of case of unstable closed-loop system (N_s) in which $\phi_m < 0$*

(5) *The estimated probability of instability is defined as $Pr(\lambda)= N_s/J$ and Pr(stable)=1-Pr(unstable).*

We first evaluated the stability of a PID controller designed by each of the methods in Table 3.1 (process 1 and process 2) via the above procedure using the corresponding (K_u, T_u) pairs. The Monte Carlo simulation was then carried out for each set of ZNPID controller parameters in order to compare the robustness of the closed-loop control systems for each method (Smith, Minimized error, Tsang-Rad). In the Monte Carlo analysis, a uniform probability density function with ±25% models the system parameter uncertainties. The first-order with time delay model is applicable to many practical systems since this model can be designed to cover the dominant system dynamics within the system bandwidth. The system parameters are approximated online by the proposed neural network algorithm. After the system parameters converge to their optimal values, the parameter uncertainties and output error, which are due to neglected dynamics and noise disturbance, should not be too large for a proper selection of system model structure. In this chapter, the maximum uncertainty of system parameters is chosen as ±25%. Monte Carlo evaluation was then repeated for J times (J = 25,000) for different ZNPID controllers. Tables 3.2 summarize the results of estimated probability of instability and the interval estimates lower and upper (L, U) with 95 % confidence interval for different off-line tuning methods for process 1 and process 2 respectively.

However, the Monte Carlo analysis for the proposed method can not be carried out the same way. This is due to the fact that the proposed controller is an adaptive PID controller and at each instant of time employs a new set of PID parameters. Therefore, we decided to run the Monte Carlo simulation for the proposed algorithm as a function of time. That is every time new sets of PID parameters were available; we computed the probability of instability. The simulation in Figure 3.3 was repeated with the addition of the Monte Carlo simulation routine (with $j = 5000$) to determine the probability of instability versus time (Figure 3.5). These simulations indicated that the probability of instability ($Pr(\lambda)$) approached to steady state around $4.8\text{x}10^{-3}$ for $50 \le t \le 195$. When the system was changed to process II at t=195s, $Pr(\lambda)$ jumped to 0.3 and after sometime moved to a new steady state of 0.0136 for $230 \le t \le 320$. At t=320s, the system changed back to process I. $Pr(\lambda)$ increased to 0.25 and then back to $4.8\text{x}10^{-3}$ for $320 \le t \le 400$. These simulations demonstrate the behavior of the proposed algorithm and verify that it exhibits very desirable robustness and stability characteristics.

Modelling Method	Process I		Process II	
	Pr(unstable)	(L,U)	Pr(unstable)	(L,U)
Smith method	7.4×10^{-3}	$(5.92\times10^{-3}, 8.88\times10^{-3})$	0.1924	(0.15392,0.23088)
Minimized-error	5.6×10^{-3}	$(4.48\times10^{-3}, 6.72\times10^{-3})$	0.1032	(0.08256,0.12384)
Tsang-Rad	6.8×10^{-3}	$(5.44\times10^{-3}, 8.16\times10^{-3})$	0.2264	(0.18112,0.27168)
Exact Model	5.2×10^{-3}	$(4.16\times10^{-3}, 6.24\times10^{-3})$	0.0201	(0.01608,0.02412)

Table 3.2 Process I & II, estimated Pr(unstable) by 25,000 Monte Carlo evaluations for ±25 % system parameter uncertainties

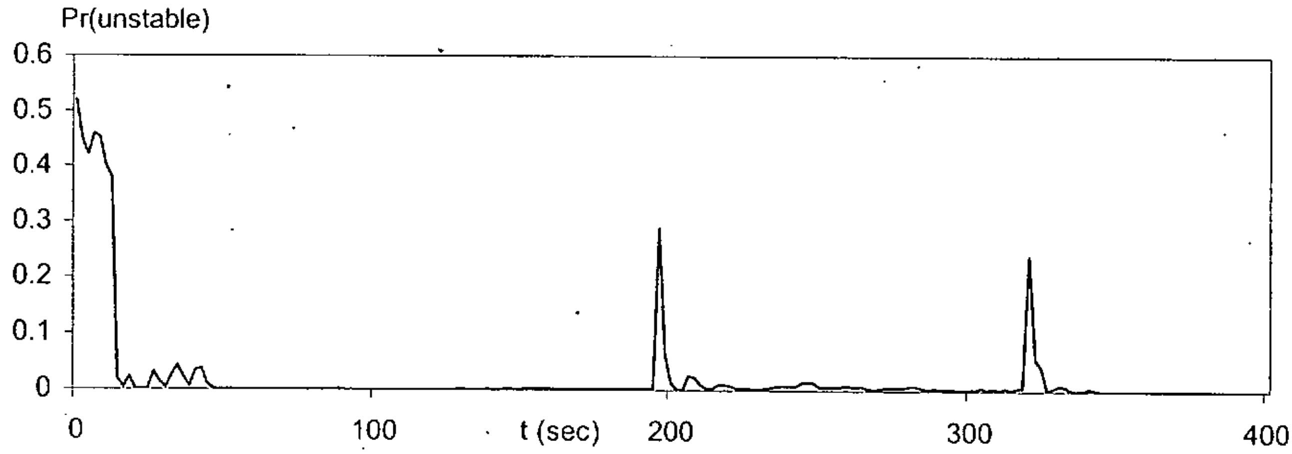


Figure 3.5: Probability of instability for simulation example

It should be noted that the Pr(unstable) is about 0.5 at t = 0, this is due to a large discrepancy between the optimal model parameters and the initial model parameters at t=0. The large impulse of Pr(unstable) is only a short transient and under the adaptive control of the proposed algorithm, the Pr(unstable) drops below 0.05 after 10 sec. As there is a system change at t=195s, the Pr(unstable) rises to 0.28 at t=195. Therefore, the Pr(unstable) is regulated back to 0.06 within 10s. From Figure 3.5, it can be observed that the proposed algorithm has a fast response to system change and the Pr(unstable) is kept below 0.06 in steady state.

3.4.3 Comparison with Relay Auto-tuning Approach

In this section, the performance of the proposed algorithm is compared with that of the relay auto-tuning approach [Astrom *et. al.* 1993]. In order to demonstrate the performance of the proposed algorithms for system with non-linear gain, the simulation example as shown in Figure 3.7 is considered. The system process is a first order with time delay model and the linear output y_o(t) is then passed into a non-linear block in which the linear output $y_o(t)$ is modulated by the following sigmoid function $s(X_n)$.

$$G(s)=\frac{Y_o(s)}{U(s)}=\frac{e^{-0.6s}}{50s+1}; \qquad y(t)=(y_{max}-y_{min})s\left(\frac{y_o(t)}{y_{max}-y_{min}}\right)$$

$$X_n=\frac{y_o}{y_{max}-y_{min}}; \qquad Y_n=\frac{y}{y_{max}-y_{min}}; \qquad Y_n=s(X_n)=\frac{1}{1+e^{-10(X_n-0.5)}}$$

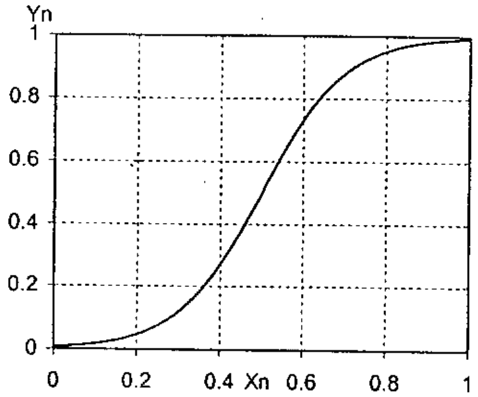


Figure 3.6 Sigmoid function.

where X_n and Y_n are the normalized value of the linear output y_o and the non-linear output y. The actual output is de-normalized by multiply Y by $(y_{max}-y_{min})$. In this chapter, $y_{max}=100\%$ and $y_{min}=0\%$. The graph of the sigmoid function is shown in Figure 3.6. It can be noted that the sigmoid function follows the "S" shape curve and simulated the non-linear gain characteristic of pH control process [Jacobs *et al.*]. The relay auto-tuner PI control loop is simulated in this example. In Figure 3.7, the switch "SW" is switched to "B" when a relay oscillation test is triggered, the ultimate gain and ultimate period are then estimated and the PI controller parameters can be designed based on these estimated parameters. As the process in

Figure 3.7 exhibits non-linear gain characteristics; the controller parameters need to be re-tuned as the process changes its operating condition. The results of the simulation are shown in Figure 3.8 and 3.9. In particular, Figure 3.8 shows the simulation results of a relay auto-tuned ZN-PI controller. Figure 3.9 shows the simulation results of the proposed algorithm with on-line ZN-PI controller. The architecture of the NN was selected as [1,8,3] and the learning rate and momentum factors were chosen 0.4 and 0.3 respectively. The set point considered in this example covered the operating region of 20%, 40% and 80%. It should be noted that the relay oscillation test was triggered at t_r=*0s* and t_r=*60s* as the system changed its operating points. The estimated Ultimate Gain and Ultimate Period for the relay test at t=0s were 64.02 and 2.6s respectively whereas their estimated values at t=80s were 62.5 and 2.2s. The output performances of the two approaches are similar in term of rise time and maximum overshoot. However, from the simulation results, it can be noted that the time duration for the relay oscillation test is about five time constants as this is the transient period for the oscillations to come to steady state. For non-time varying non-linear system, gain scheduling is an effective approach but it needs human intervention for re-tuning the controller parameter as the system change its operating points. After designing the controller parameters for the whole operating region, PI controller parameters can be scheduled according to the operating point. However, in case the system components deteriorate or the system changes its characteristics, the designed procedures have to be repeated again and usually this re-tuning procedures needs human intervention. On the other hand, the algorithm proposed in this chapter is an adaptive approach, which can track the changes of the system parameters and has the ability to re-tune the PI controller on-line.

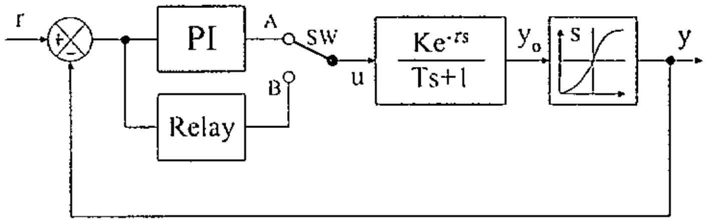


Figure 3.7 Auto tuning PI controller via relay feedback approach

The steps for relay auto-tuning are summarized as follows.

1. *The switch SW is switched to position "B", the system is under closed-loop control of a relay.*

2. *After the limit cycle oscillations come to steady state, the ultimate period is estimated and the ultimate gain is estimated by the follow formula.*

   $K_u = \dfrac{4d}{\pi a}$; T_u = *Oscillation Period*;

3. *The controller parameters can be designed according to the estimated ultimate gain (K_u) and period (T_u). In this example, the PID controller parameters are designed by standard ZNPI.* $K_p = 0.45K_u$; $T_i = 0.85T_u$;

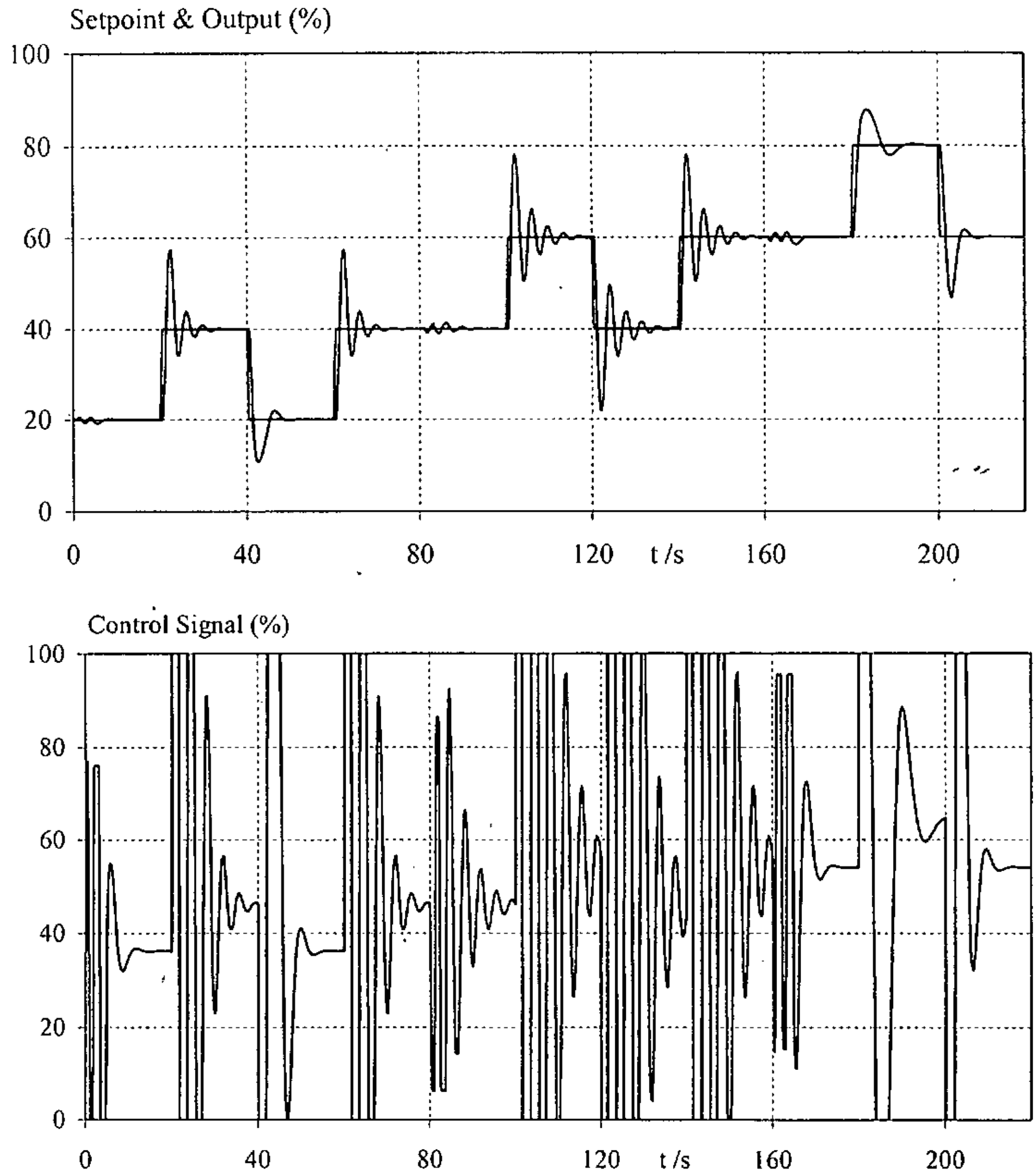
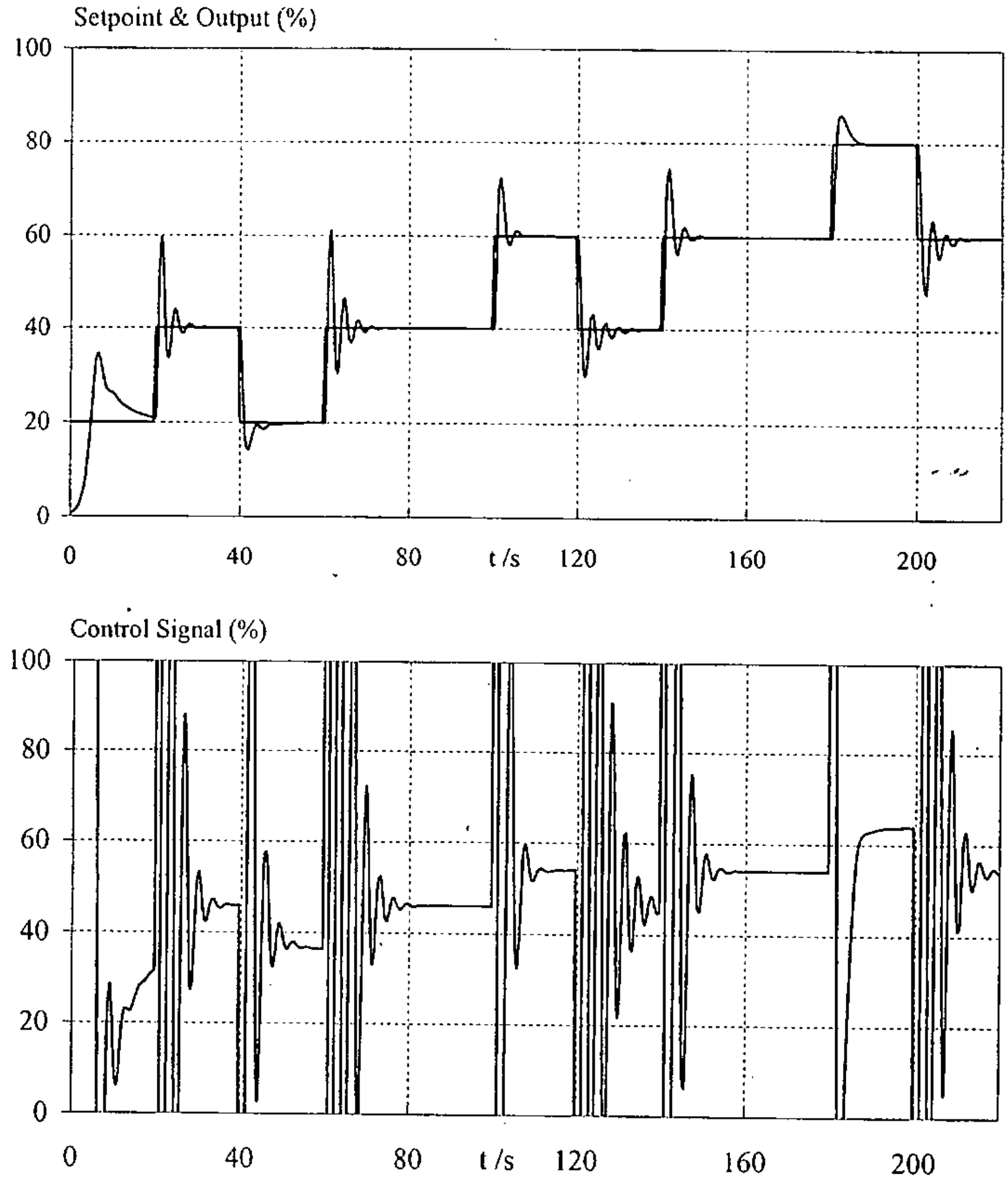
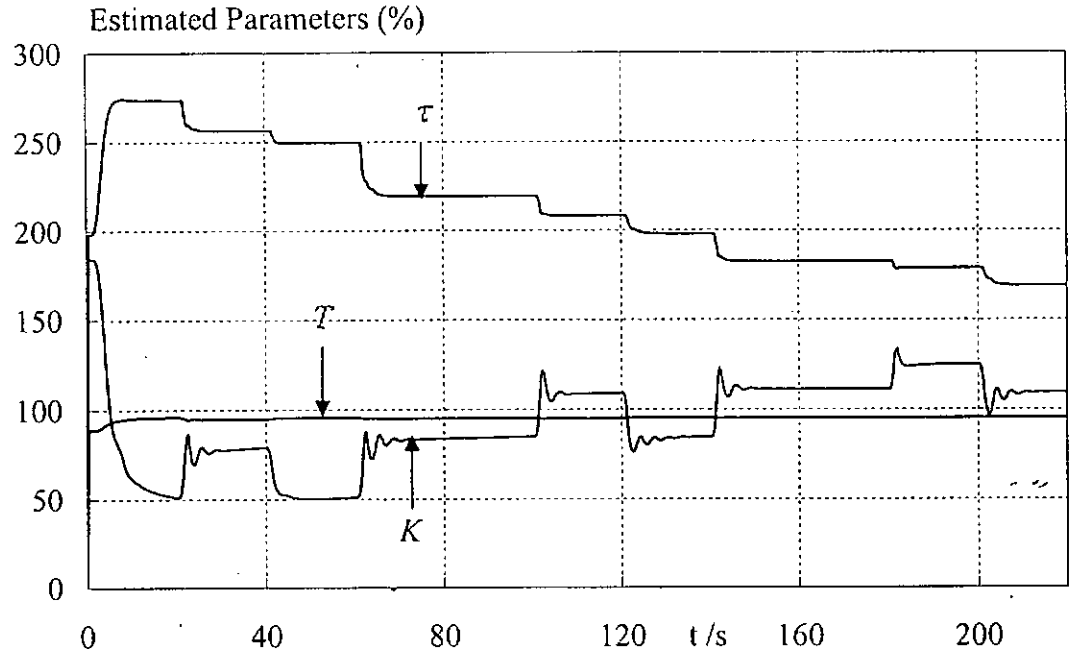


Figure 3.8 Simulation results for relay auto-tuned ZN PI


Setpoint & Output (%)
100
80
60
40
20
0
0
40
80
t /s
120
160
200
Control Signal (%)
100
80
60
40
20
0
0
40
80
t /s
120
160
200



Gain = K/1.0 x 100%, Time Constant = T/50x100%, Delay=τ/0.6 x 100%

Figure 3.9 Simulation results for the proposed PI controller

3.5 Experimental Studies

A laboratory-scale process control unit (PCU) from Bytronic [Byronic 1994] was used in this experiment. The system rig consists of a sump, a pump, manual/computer control diverting valve and drain valve. The sump water is pumped through the pipeline and the manual flow control valve to the process tank. An impeller-type flowmeter is located near the process tank. The water is fed back to the sump via the drain valve, thus completing the cycle. The rig can be used for level, temperature and flow control. The objective in our study was to control the water flow by manipulating the pump voltage. Figure 3.10 shows the schematic diagram of the process rig. The purpose of the experiment was to demonstrate the performance of the algorithm on a real time flow control system. The real time control experiment is somewhat different from the simulation studies because of presence of noise and other non-linearity due to the dead-zone in pump and other components in the system. The flow rate is zero if the pump voltage is less than a threshold value and the steady state flow would increase with pump voltage when the pump voltage is larger than that threshold level. The consideration of dead-zone associated with the pump characteristics is out of the scope of this chapter.

The set point was chosen as a staircase signal with amplitude of *0.4 l/min, 0.8 l/min and 1.2 l/min* and period of *40 sec*. The sample time was chosen *0.05 sec*. The on-line process computer sampled the water flow rate at every sample interval, and the system parameters were identified by the NN from the pump voltage and the water flow rate. Due to the dead zone in the motor dynamic, the pump drive voltage minimum and maximum were $0.8V$ and $5V$ respectively. The controller signal was limited ($0-5V$) in order to avoid the windup problem [Åström and Wittenmark 1995]. It was noticed that at times around 80 and 160 seconds, the change in set point forced the control signal to within the dead-zone range. In such cases, the pump was unable to react properly and hence oscillations occurred for few seconds after which the operation returned to normal. This effect of this is shown in Figure 3.11 at times around 80 and 160 seconds. Most flow control systems employ a PI rather than a PID controller. Since the derivative action amplifies the inherently noisy flow control system. In this experiment, we also modified the control structure to PI by removing the derivative action. The tuning algorithm was also modified to ZN-PI algorithm ($K_p=0.45\ K_u$, and $T_i=0.85T_u$). The architecture of the NN was selected as [1,8,3] and the learning rate and momentum factors were chosen 0.4 and 0.8 respectively. Figure 3.11 shows the results for real-time flow control system: It can be seen that the performance of the adaptive control system is very satisfactory. It should also be noted that the model of the plant changes at different operating points. Nevertheless, the control system maintains a very acceptable performance. The measure of performance is not obtained by quantitative means such as ISE (integral of squared error) or IAE (integral of absolute error) or similar performances index. Instead a qualitative measures such as "lower overshoot", "higher rise time", *etc*. are used.

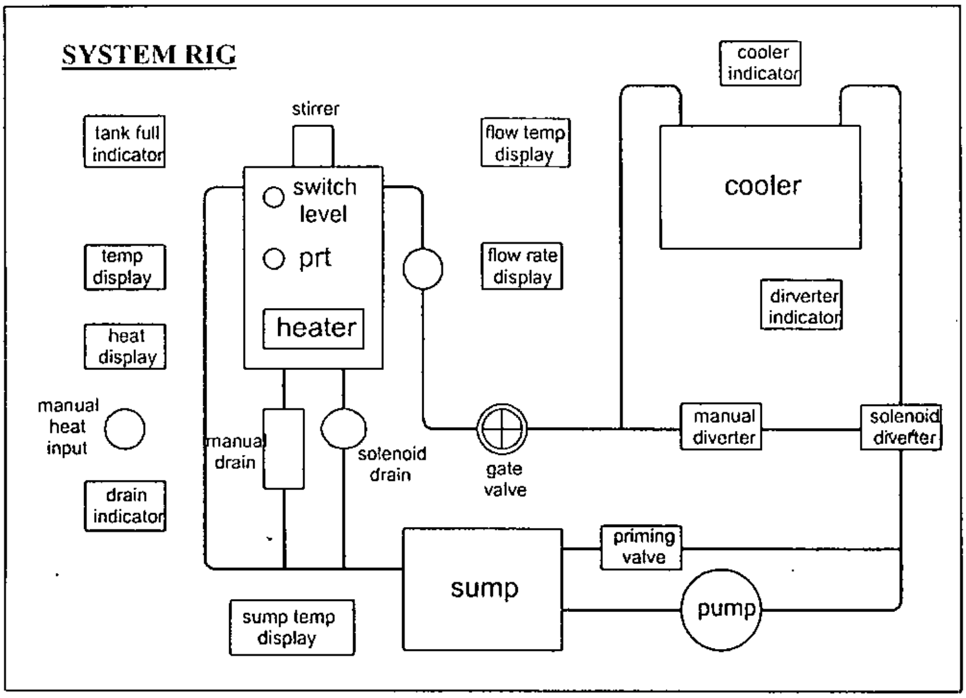


Figure 3.10: Schematic diagram of the process control unit

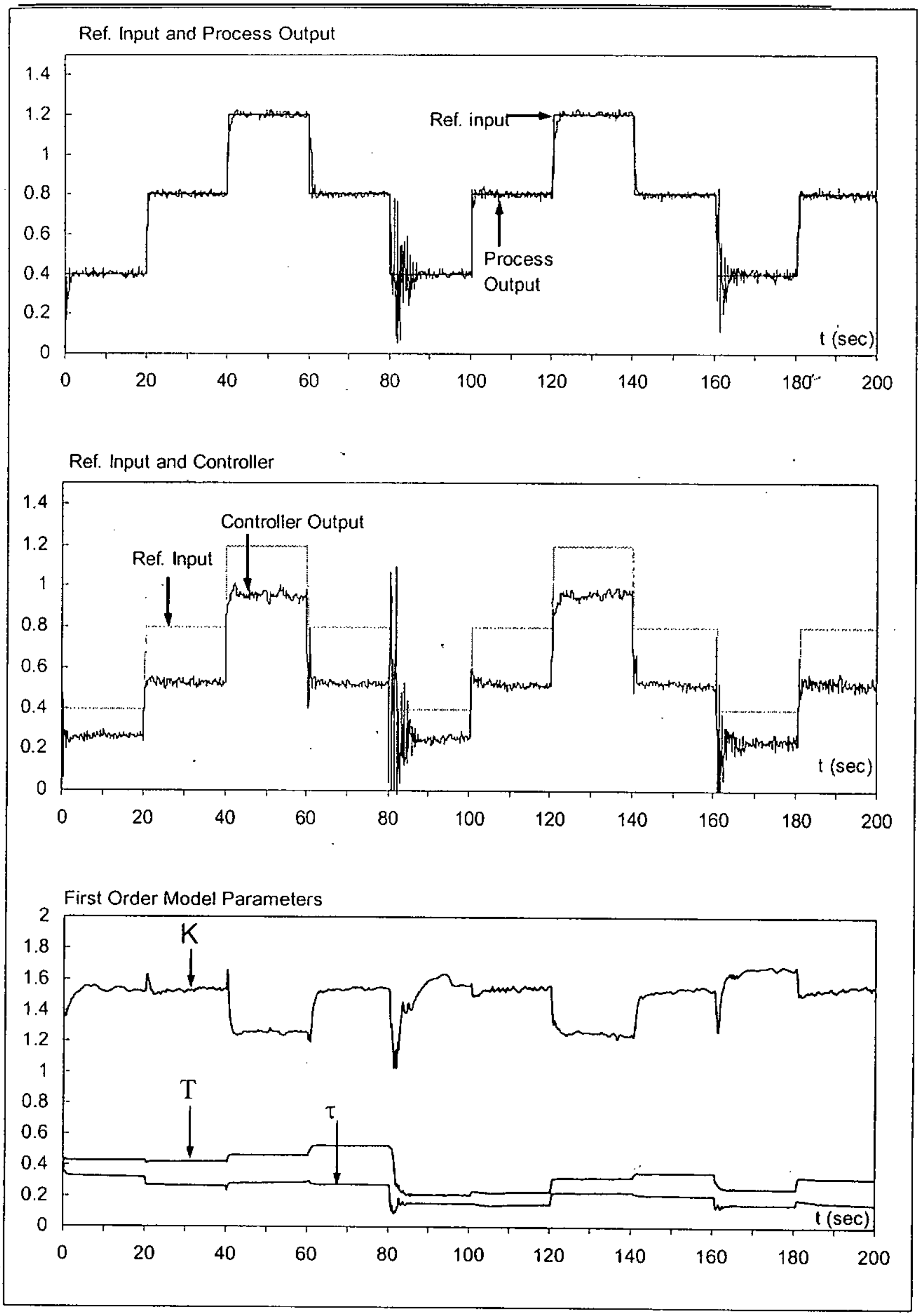


Figure 3.11 Adaptive PI for flow-rate control system

3.6 Conclusions

In this chapter, an on-line PID tuning control algorithm based on the parameters of a FOPDT model which are obtained by using neural networks has been proposed. The neural network is implemented as an approximator rather than an identifier or controller. The outputs of the neural network are the three parameters of the FOPDT model for the plant to be controlled. Combining with a conventional ZN PID controller (or any other similar tuning algorithm), an on-line adaptive control using neural networks is proposed. The method can be integrated with any other tuning algorithm, which utilizes the parameters of a FOPDT model. The simplicity and feasibility of the scheme for real-time control provides a new approach for implementing neural network applications for a variety of on-line industrial tracking control problems. The same idea can be extended to fuzzy system. In the next chapter, an on-line PID tuning algorithm based on fuzzy system will be discussed. The algorithm combines numerical and linguistic information to improve system performance with modest computation.

CHAPTER FOUR

ON-LINE LOWER ORDER MODELING VIA FUZZY LOGIC

4.1 Introduction

In the previous chapter, a class of on-line approximation of higher-order systems with a FOPDT using neural networks was introduced. From this chapter onwards, fuzzy design concepts, *i.e.* fuzzy logic reduced order modeling, direct, and indirect fuzzy adaptive control will be investigated.

In this chapter, a simple and novel fuzzy on line lower-order model approximation method is proposed. The main idea is to combine a fuzzy system with a model generator for the purpose of on-line approximation of the FOPDT model parameters. This method is useful for the adaptive tuning of fixed structure PID controller and the proposed method is applicable to the higher order systems with unknown time delay. The parameter learning is achieved by using the gradient descent algorithm [Wang *et. al.* 1997, Lee and Teng 2000]. The proposed algorithm can guarantee the system stability in Lyapunov sense if the learning rates are properly selected to ensure the values of lower-order model parameters converge. In contrast to the classical fuzzy approach, the proposed algorithm has the following distinctive features:

- Without requiring *a priori* knowledge of the system, the proposed method is an on-line adaptive control algorithm using fuzzy system. This is achieved by combining the on-line fuzzy approximation method with the fixed structure controller (PID, IMC, etc).
- The proposed scheme addresses the problem of time delay identification, which is generally overlooked in the adaptive fuzzy control research.

This chapter is based on the papers 5 and 14 on chapter one, pages 11 and 12

- The proposed method is characterized by its simplicity of controller structure and the feasibility for real-time implementation.

The rest of this chapter is organized as follow: Section 4.2 presents the idea of approximating a high-order system with a FOPDT model by using the fuzzy system. In section 4.3, the stability analysis of the fuzzy system is given. The on-line PID tuning method using fuzzy system is included in Section 4.4. In Section 4.5, simulation studies for controlling a system with changing parameters are presented. The performance of the proposed algorithm for real time control of a temperature system with varying dominant time delay is evaluated in Section 4.6. Finally, the chapter is concluded in Section 4.7

4.2 Lower Order Approximation of Higher-Order Systems with Fuzzy System

As in Chapter 3, the high-order processes dynamic can be represented with sufficient accuracy by a first order plus delay time model (FOPDT).

$$\frac{Y(s)}{U(s)} = e^{-s\vartheta}\frac{b_0 s^n + b_1 s^{n-1} + \cdots + b_m}{s^n + a_1 s^{n-1} + \cdots + a_n} \approx \frac{K \cdot e^{-\tau s}}{Ts+1} \tag{4.1}$$

where K is the system gain, T is the dominant time constant, τ is the apparent dead time and *U(s)* and *Y(s)* are the Laplace transform of the system input and output signals respectively. The proposed approach is conceptually simple and is realized by cascading a fuzzy system with a model generator in parallel with the process to be identified as shown in figure 4.1. For further details please refer to Chapter 3, Section 3.2.1).

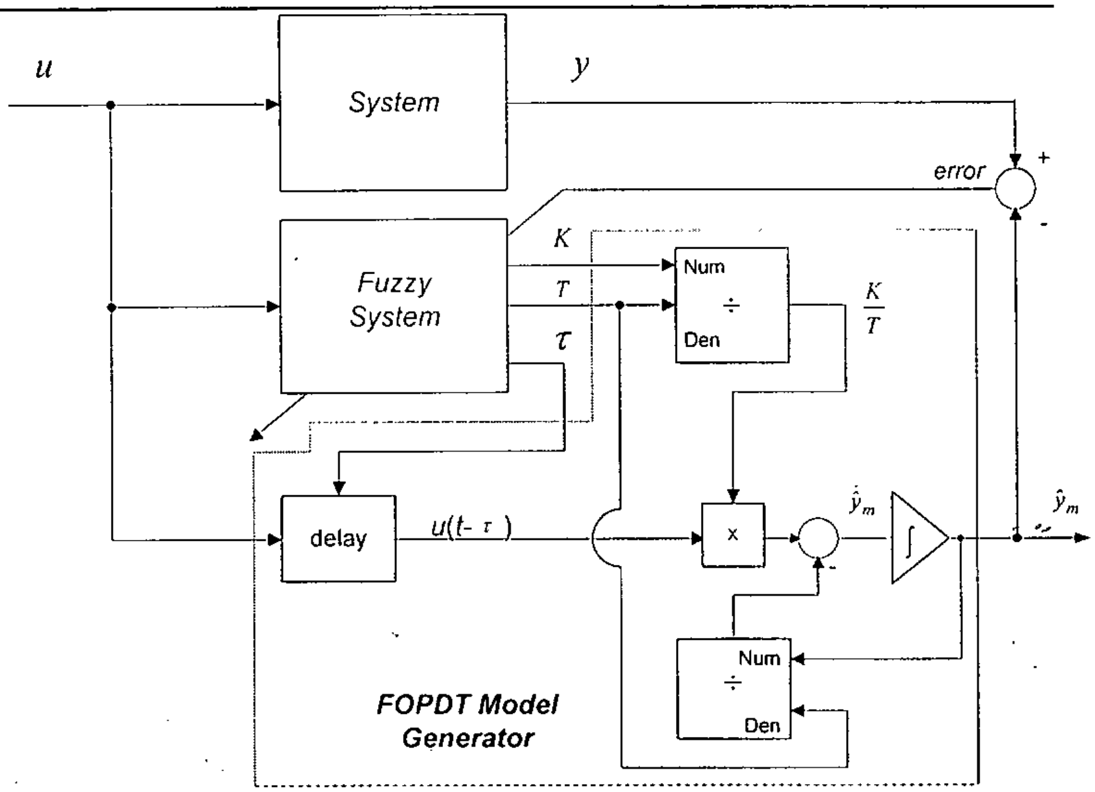


Figure 4.1 The structure of the on-line lower-order modeling of higher-order systems

The transfer function of the FOPDT model generator is given by:

$$\frac{\hat{Y}_m(s)}{U(s)} = \frac{K \cdot e^{-\tau s}}{Ts+1} \tag{4.2}$$

4.2.1 The Fuzzy System Structure

In this section, we apply the fuzzy logic system [Wang *et al.* 1997] to obtain the process model parameters. The basic configuration of the fuzzy logic system consists of a collection of fuzzy IF-THEN rules, which can be written as:

$$\begin{aligned} R^{(l)}: \text{IF} \quad x_1 \quad \text{is} \quad F_1^l \quad \text{and} \quad \cdots \quad \text{and} \quad x_n \quad \text{is} \quad F_n^l \\ \text{Then} \quad y \quad \text{is} \quad B^l \end{aligned} \tag{4.3}$$

The fuzzy logic system performs a mapping from $U = U_1 \times \cdots \times U_n \subseteq R^n$ to R, where the input vector $x=[x_1,\cdots,x_n]^T \in R^n$ and the output variable $y \in R$ denote the linguistic variables associated with the inputs and output of the fuzzy logic system. F_k^l and B^l are labels of the input and output fuzzy sets respectively. Let $k=1,2,\cdots,n$ denotes the number of input for fuzzy logic system and $l=1,2,\cdots,m$ denotes the number of the fuzzy IF-THEN rules. By using the singleton fuzzification, product inference and center average defuzzification, the output value of the fuzzy system is

$$\Theta(x)=\frac{\sum_{l=1}^{m} y^l\left(\prod_{k=1}^{n}\mu_{F_k^l}(x_k)\right)}{\sum_{l=1}^{m}\prod_{k=1}^{n}\mu_{F_k^l}(x_k)} \tag{4.4}$$

where $\mu_{F_k^l}(x_k)$ is the membership function of the linguistic variable x_k , and y^l represents a crisp value at which the membership function μ_{B^j} for output fuzzy set reaches it maximum value. As a usual practice, we assume that $\mu_{B^j}(y^l)=1$. By introducing the concept of fuzzy basis function (FBF) [Wang *et al.* 1997], Equation (4.4) can be rewritten as

$$\Theta(x)=\theta^T\xi(x)=\xi(x)^T\theta \tag{4.5}$$

where $\theta=[\theta^1,\cdots,\theta^m]^T$ is the fuzzy system parameter vector and $\xi(x)=[\xi^1(x),\cdots,\xi^m(x)]^T$ is a regressive vector defined as

$$\xi^l(x)=\frac{\prod_{k=1}^{n}\mu_{F_k^l}(x_k)}{\sum_{l=1}^{m}\prod_{k=1}^{n}\mu_{F_k^l}(x_k)} \tag{4.6}$$

4.2.2 Learning Algorithm

To train the above fuzzy system, a direct gradient decent learning algorithm [Wang *et al.* 1997, Lee and Teng 2000] is employed. The error function E for the learning processes is defined as:

$$E(t) = \frac{1}{2}(y(t) - \hat{y}_m(t))^2 = \frac{1}{2}e^2(t) \tag{4.7}$$

where $y(t)$ and $\hat{y}_m(t)$ are the actual system output and the FOPDT model output at any time instant t. The gradient of error in (4.7) with respect to the consequent vector is as follows:

$$\begin{aligned} \frac{\partial E(t)}{\partial \theta} &= -\frac{\partial E(t)}{\partial y_m(t)} \cdot \frac{\partial \Theta}{\partial \theta} \\ &= -\frac{\partial E(t)}{\partial \hat{y}_m(t)} \cdot \frac{\partial \hat{y}_m(t)}{\partial \Theta} \cdot \frac{\partial \Theta}{\partial \theta} \end{aligned} \tag{4.8}$$

where $\Theta = [K \quad T \quad \tau]^T = [\Theta_1, \Theta_2, \Theta_3]^T \in R^3$ is the input vector of the FOPDT model (the output value of three fuzzy sub-systems) and the term $\frac{\partial \hat{y}_m(t)}{\partial \Theta}$ represents the sensitivity of the plant with respect to its FOPDT model parameters where

$$\frac{\partial \hat{y}_m(t)}{\partial \Theta} = \left[\frac{\partial \hat{y}_m(t)}{\partial K}, \frac{\partial \hat{y}_m(t)}{\partial T}, \frac{\partial \hat{y}_m(t)}{\partial \tau} \right]^T = [y_{p1}(t), y_{p2}(t), y_{p3}(t)]^T \in R^3 \tag{4.9}$$

The suffices 1, 2 and 3 are used to indicate the FOPDT model parameters, system gain, time constant and time delay. The partial derivatives of the output $y_m(t)$ of the FOPDT model generator respect to system gain (K), dominant time constant (T) and apparent dead-time (τ), respectively, are given by

$$\frac{\partial \hat{y}_m(t)}{\partial K} = L^{-1}\left[\frac{e^{-\tau s}}{Ts+1} u(s) \right] \tag{4.10}$$

$$\frac{\partial \hat{y}_m(t)}{\partial T} = L^{-1}\left[\frac{-sKe^{-\tau s}}{(Ts+1)^2} u(s) \right] \tag{4.11}$$

$$\frac{\partial \hat{y}_m(t)}{\partial \tau} = L^{-1}\left[\frac{-sKe^{-\tau s}}{Ts+1}u(s)\right] \tag{4.12}$$

Hence, from (4.8) and (4.9) the error gradient $\frac{\partial E(t)}{\partial \theta}$ can be written as

$$\frac{\partial E(t)}{\partial \theta} = -e(t)\begin{bmatrix} y_{p1} & 0 & 0 \\ 0 & y_{p2} & 0 \\ 0 & 0 & y_{p3} \end{bmatrix}\begin{bmatrix} \frac{\partial \Theta_1}{\partial \theta_1} \\ \frac{\partial \Theta_2}{\partial \theta_2} \\ \frac{\partial \Theta_3}{\partial \theta_3} \end{bmatrix} \tag{4.13}$$

where

$$\frac{\partial \Theta_i}{\partial \theta_i} = \xi_i, \quad i = 1, \cdots, 3 \tag{4.14}$$

With the fuzzy logic systems (4.5) and the error function defined in (4.7), derive the update laws

$$\theta_1(t+1) = \theta_1(t) - \eta_1 \frac{\partial E(t)}{\partial \theta_1} \tag{4.15}$$

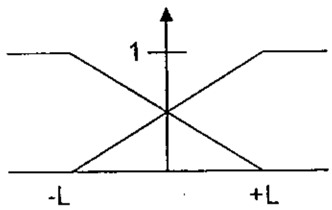
$$\theta_2(t+1) = \theta_2(t) - \eta_2 \frac{\partial E(t)}{\partial \theta_2} \tag{4.16}$$

$$\theta_3(t+1) = \theta_3(t) - \eta_3 \frac{\partial E(t)}{\partial \theta_3} \tag{4.17}$$

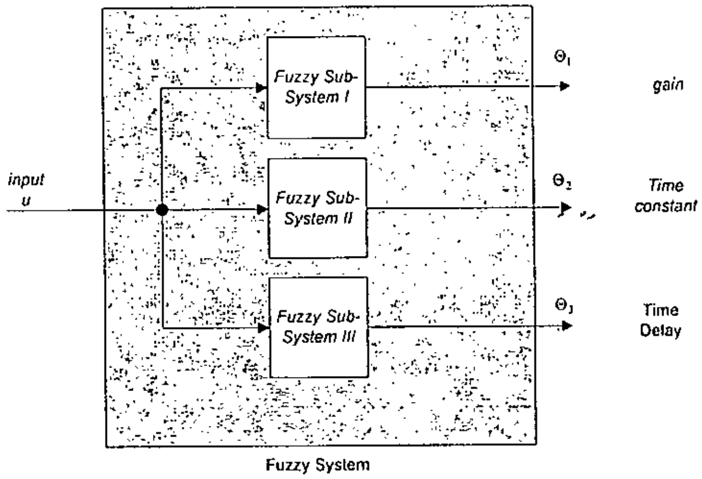
where η_i and θ_i for $i = 1, \cdots, 3$ represent the learning rate and tuning parameters of each fuzzy sub-system respectively. Figure 4.2 shows the membership function of each sub-system. The value of L for the membership function is designed for covering the input domain of interest. Each fuzzy sub-system has only one linguistic input value and two fuzzy rules. Moreover, the membership functions of each fuzzy sub-system are chosen to be the same, as shown in Figure 4.2. Therefore the fuzzy system consists of three fuzzy sub-systems and the output value can be obtained from equation (4.4).

$$\Theta_1 = \theta_1^T \xi_1 \quad \Theta_2 = \theta_2^T \xi_2 \quad \Theta_3 = \theta_3^T \xi_3 \tag{4.18}$$

where $\theta_1 = [\theta_1^1 \quad \theta_1^2]^T$, $\theta_2 = [\theta_2^1 \quad \theta_2^2]^T$ and $\theta_3 = [\theta_3^1 \quad \theta_3^2]^T$ are the regression vector of each sub-system with the regressor as given in equation (4.6).



(a)



(b)

Figure 4.2 Fuzzy logic system. (a) Fuzzy input membership function. (b) Block diagram of fuzzy logic based FOPDT parameters identification

4.3 Stability Analysis of the Fuzzy Logic System

This section develops the selection criterion of the learning rate parameters. If a small learning rate η is used, convergence will be guaranteed but speed of convergence may be very slow. On the other hand, if a large learning rate η may results in an unstable system. In this section, the selection of learning rate are derived by discrete-Lyapunov function analysis, which guarantee the convergence of estimated model parameters [Hoo *et al.* 2002].

Consider a discrete Lyapunov function as follows:

$$V(t) = E(t) = \frac{1}{2} e^2(t) \tag{4.19}$$

where $e(t)$ represents the error in the learning process. The change of the Lyapunov function due to the training process can be given by

$$\Delta V(t) = V(t+1) - V(t) = \frac{1}{2}[e^2(t+1) - e^2(t)] \tag{4.20}$$

The error difference due to the learning can be represented by [Lee and Teng 2000]

$$\begin{aligned} \Delta e = e(t+1) - e(t) &\approx \left[\frac{\partial e(t)}{\partial \theta}\right]^T \Delta\theta \\ &= \left[\frac{\partial e(t)}{\partial \theta_1} \quad \frac{\partial e(t)}{\partial \theta_2} \quad \frac{\partial e(t)}{\partial \theta_3}\right] \begin{bmatrix} \Delta\theta_1 \\ \Delta\theta_2 \\ \Delta\theta_3 \end{bmatrix} \end{aligned} \tag{4.21}$$

where $\Delta\theta$ represents a change in consequent parameter vector. From (4.15-4.17), we have

$$\begin{aligned} \Delta\theta &= -\eta \cdot e(t) \cdot \frac{\partial e(t)}{\partial \theta} \\ &= e(t) \begin{bmatrix} \eta_1 & 0 & 0 \\ 0 & \eta_2 & 0 \\ 0 & 0 & \eta_3 \end{bmatrix} \begin{bmatrix} y_{p1} & 0 & 0 \\ 0 & y_{p2} & 0 \\ 0 & 0 & y_{p3} \end{bmatrix} \begin{bmatrix} \frac{\partial \Theta_1(t)}{\partial \theta_1} \\ \frac{\partial \Theta_2(t)}{\partial \theta_2} \\ \frac{\partial \Theta_3(t)}{\partial \theta_3} \end{bmatrix} \end{aligned} \tag{4.22}$$

where $\theta = [\theta_1, \theta_2, \theta_3]^T$ and $\eta = [\eta_1, \eta_2, \eta_3]^T$ are the tuning parameters and the corresponding learning rates in each fuzzy sub-system respectively, $\Theta = [\Theta_1, \Theta_2, \Theta_3]^T$ is the output vector of the fuzzy sub-system at time t.

Theorem 4.1: Let $\eta = [\eta_1, \eta_2, \eta_3]^T$ be the learning rates for the fuzzy sub-system and let P_{max} be defined as follows:

$$\begin{aligned} P_{max} &= [P_{1max} \quad P_{2max} \quad P_{3max}]^T \\ &= \left[\max_t \left|\frac{\partial \Theta_1}{\partial \theta_1}\right| \quad \max_t \left|\frac{\partial \Theta_2}{\partial \theta_2}\right| \quad \max_t \left|\frac{\partial \Theta_3}{\partial \theta_3}\right|\right]^T \end{aligned} \tag{4.23}$$

Then asymptotic convergence is guaranteed if η_i are chosen to satisfy the following conditions.

$$0<\eta_i<\frac{2}{(y_{pi}P_{i\max})^2},\quad i=1,\cdots,3 \tag{4.24}$$

Proof: From (4.20-4.22) the change in Lyapunov function can be represents as

$$\begin{aligned}
\Delta V(t) &= \frac{1}{2}[e^2(t+1)-e^2(t)] \\
&= \Delta e(t)[e(t)-\frac{1}{2}\Delta e(t)] \\
&= \left[\frac{\partial e(t)}{\partial\theta}\right]^T \eta\cdot e(t)\cdot y_p\cdot\frac{\partial\Theta}{\partial\theta}\cdot\left\{e(t)+\frac{1}{2}\left[\frac{\partial e(t)}{\partial\theta}\right]^T \eta\cdot e(t)\cdot y_p\cdot\frac{\partial\Theta}{\partial\theta}\right\} \\
&= \left[-y_p\cdot\frac{\partial\Theta}{\partial\theta}\right]^T \eta\cdot e(t)\cdot y_p\cdot\frac{\partial\Theta}{\partial\theta}\cdot\left\{e(t)+\frac{1}{2}\left[-y_p\cdot\frac{\partial\Theta}{\partial\theta}\right]^T \eta\cdot e(t)\cdot y_p\cdot\frac{\partial\Theta}{\partial\theta}\right\} \\
&= -e^2(t)\cdot\sum_i^3\left[\frac{1}{2}\eta_i(y_{pi}(t))^2\left(\frac{\partial\Theta_i}{\partial\theta_i}\right)^2\right]\cdot\left\{2-\eta_i(y_{pi}(t))^2\left(\frac{\partial\Theta_i}{\partial\theta_i}\right)^2\right\} \\
&\equiv -\lambda e^2(t)
\end{aligned} \tag{4.25}$$

where

$$\lambda=\sum_i^3\left[\frac{1}{2}\eta_i(y_{p_i}(t))^2\left(\frac{\partial\Theta_i}{\partial\theta_i}\right)^2\right]\cdot\left\{2-\eta_i(y_{pi}(t))^2\left(\frac{\partial\Theta_i}{\partial\theta_i}\right)^2\right\} \tag{4.26}$$

the convergence of the fuzzy logic system is guaranteed if $\lambda>0$. We obtain

$$0<\eta_i<\frac{2}{(y_{pi}P_{i\max})^2},\quad i=1,\cdots,3 \tag{4.27}$$

Theorem 4.2: Let $\eta=[\eta_1,\eta_2,\eta_3]^T$ be the learning rates for the tuning parameters of the fuzzy sub-system. Then asymptotic convergence is guaranteed if the learning rates satisfy the conditions.

$$0<\eta_i<\frac{2}{(y_{pi})^2 R_i},\quad i=1,\cdots,3 \tag{4.28}$$

where R_i is the number of rules of each fuzzy sub-system.

Proof: Let $P_i = \frac{\partial \Theta_i}{\partial \theta_i} = [Z_{1_i}, Z_{2_i} \cdots Z_{R_i}]^T$, in which Z_{l_i} is the regressor values of each fuzzy sub-systems. Since $Z_{l_i} \le 1$ for all l and $|P_i|^2 \le R_i$ then we have $P_{i\max}^2 = R_i$. From Theorem 4.1, we obtain $0 < \eta_i < 2/((y_{pi})^2 R_i),\ \ i = 1,\cdots,3$.

Remark 4.1: The factor $\frac{\partial \hat{y}_m(t)}{\partial K}, \frac{\partial \hat{y}_m(t)}{\partial T}$ and $\frac{\partial \hat{y}_m(t)}{\partial \tau}$ represents the sensitivity of the plant to its FOPDT model values. Once the identification process is done, we assume the sensitivity can be approximated as

$$\frac{\partial y(t)}{\partial K} \approx \frac{\partial \hat{y}_m(t)}{\partial K} \quad \frac{\partial y(t)}{\partial T} \approx \frac{\partial \hat{y}_m(t)}{\partial T} \quad \frac{\partial y(t)}{\partial \tau} \approx \frac{\partial \hat{y}_m(t)}{\partial \tau} \tag{4.29}$$

Remark 4.2: In previous discussion, the sensitivity of the plant to its FOPDT model values is obtained by equation (4.10-4.12). However, in the convergent conditions of Theorem 4.1 and 4.2, the values of y_{p_i} must be replaced by $S_{i,\max}$, where

$$S_{i,\max} = \max_t [y_{pi}(t)] \quad i = 1,\cdots,3 \tag{4.30}$$

4.4 On Line PID Tuning Method Using Fuzzy System

In order to show the effectiveness of the proposed method, we combine the fuzzy algorithm with a standard PID controller to make an adaptive control algorithm. The control structure is shown in Figure 4.3. There are two parts in the control structure of the on line PID tuning method. The first part, which was described in the previous section, is the approximation of high order systems with FOPDT using fuzzy system, and the second part is the design of the PID controller. The parameters of the PID controller can be obtained from the corresponding parameters of the estimated FOPDT by fuzzy system. We have used the

Ziegler-Nichols ultimate cycle tuning method [Ziegler and Nichols 1942] to compute the parameters of the PID controller:

$$K_p = 0.6\,K_u \qquad T_I = 0.5\,T_u \qquad T_d = 0.125T_u \tag{4.31}$$

Here, K_p, T_i, T_d, K_u and T_u are the proportional gain, integral time constant, derivative time constant, the ultimate gain and the ultimate period respectively. The ultimate gain and the ultimate period are calculated from the FOPDT model of the high order plant [Rad *et al.* 1997]. It should be emphasized that other control algorithms could also be used. The PID controller is implemented in the following form:

$$u(t) = K_p(e_t(t) + \frac{1}{T_i}\int e_t(t)dt + T_d\frac{dy_f}{dt})$$

$$e_t(t) = r(t) - y(t) \tag{4.32}$$

$$y_f(s) = \frac{1}{1+\frac{T_d}{10}s}y(s)$$

where $u(t)$, $y(t)$, $r(t)$, $y(t)$ and $y_f(t)$ are the controller output, process output, set-point and filtered derivative, respectively. The implementation of the adaptive PID is as follows:

1. *Derive the first-order with time delay model (FOPDT) parameters by fuzzy system.*
2. *Determine the ultimate gain (K_u) and ultimate period (T_u) by the FOPDT model.*
3. *Find the PID controller parameters K_p, T_i and T_d from equation (4.31) and calculate u(t.)*
4. *Find the FOPDT model output $y_m(t)$ from the FOPDT Model Generator.*
5. *Calculate error between the (FOPDT) model output and the process output.*
6. *Update ϑ_i by using equation(4.15-4.17) (Gradient descent algorithm).*
7. *Update the error between the set-point and the process output. Go to step (1)*

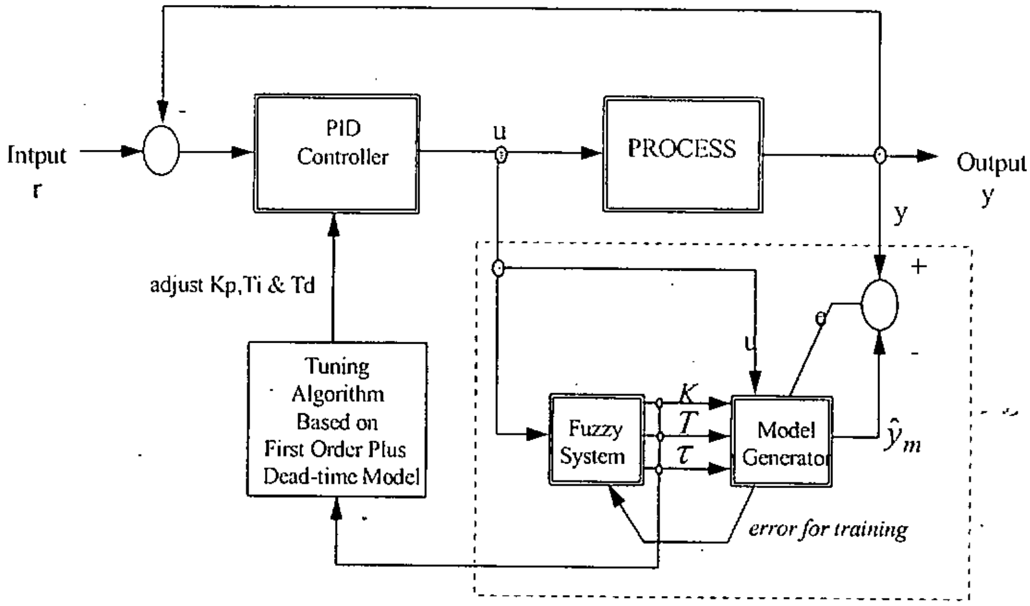


Figure 4.3 On-line PID tuning using fuzzy system

4.5 Simulation Results

To show the performance of the proposed algorithm, let us consider the following three processes:

Process I $\frac{Y(s)}{U(s)} = \frac{1.5e^{-2.5s}}{(s+1)^2}$

Process II $\frac{Y(s)}{U(s)} = \frac{1-1.4s}{(s+1)^3}$

Process III $\frac{Y(s)}{U(s)} = \frac{1.5e^{-3.0s}}{(s+1)^4}$

Process I is a second order with time delay system, the Process II is a non-minimum phase system and the Process III is a fourth order time delay system. First, adaptive control of Process I was simulated for t =120s after which the system was changed from Process I to

Process II. For t =320s the system was switched from process II to process III. Furthermore, it should be noted that the gain in Process I, II and III are different (1.5 and 1.0). It is known that some adaptive controllers cannot cope with such change in steady-state gain of the controlled system. However, as it is seen in Figure 4.4, the proposed method can successfully track the system change. In the simulation, the set-point was selected to be a square wave with amplitude 0.6 and a period of 80s. A Gaussian noise with mean zero and variance of 0.001 was injected at the output of the system. We employed a fourth order Runge Kutta numerical integration algorithm for all time responses' simulation and the integration interval was selected to be 0.1s. The fuzzy system also used the same time interval for updating its parameters. The simulation proceeded as follows: the PID controller was initialized with K_p = *1*, T_i *=1000*, T_d = *0.0*. The consequent values of fuzzy system were initialized with $\theta_1 = 1.0$, $\theta_2 = 1.8$ and $\theta_3 = 3.5$. The value of $L = 1$ and $S_{i,\max} = 1$ for all i. The learning rates were chosen as $\eta_1 = 0.25$, $\eta_2 = 0.8$ and $\eta_3 = 0.8$. Figure 4.4 shows the overall performance of the three controlled systems. In this figure, the set-point and the output, the controller signal and the estimated parameters of gain, apparent time delay and the dominant time constant are shown in top, middle and bottom curves respectively. In all these system changes, the fuzzy system converged and the estimated parameters of the FOPDT also converged to their steady state values. The proposed method is shown to provide stable and robust control under various conditions. Tables 4.1, 4.2 and 4.3 show the parameters of FOPDT model approximated by several other methods such as Smith's [Smith *et al.* 1967], minimized error [Sundaresan and Krishnaswamy 1978], and the corresponding ultimate gain and the ultimate period for processes I, II and III respectively. It should be noted that the parameters from all other methods except the proposed one were obtained off-line, from open loop excitation with unit step and were noise free. Furthermore, the values quoted for the proposed algorithm is based on the last measurement before each system change and not the average value.

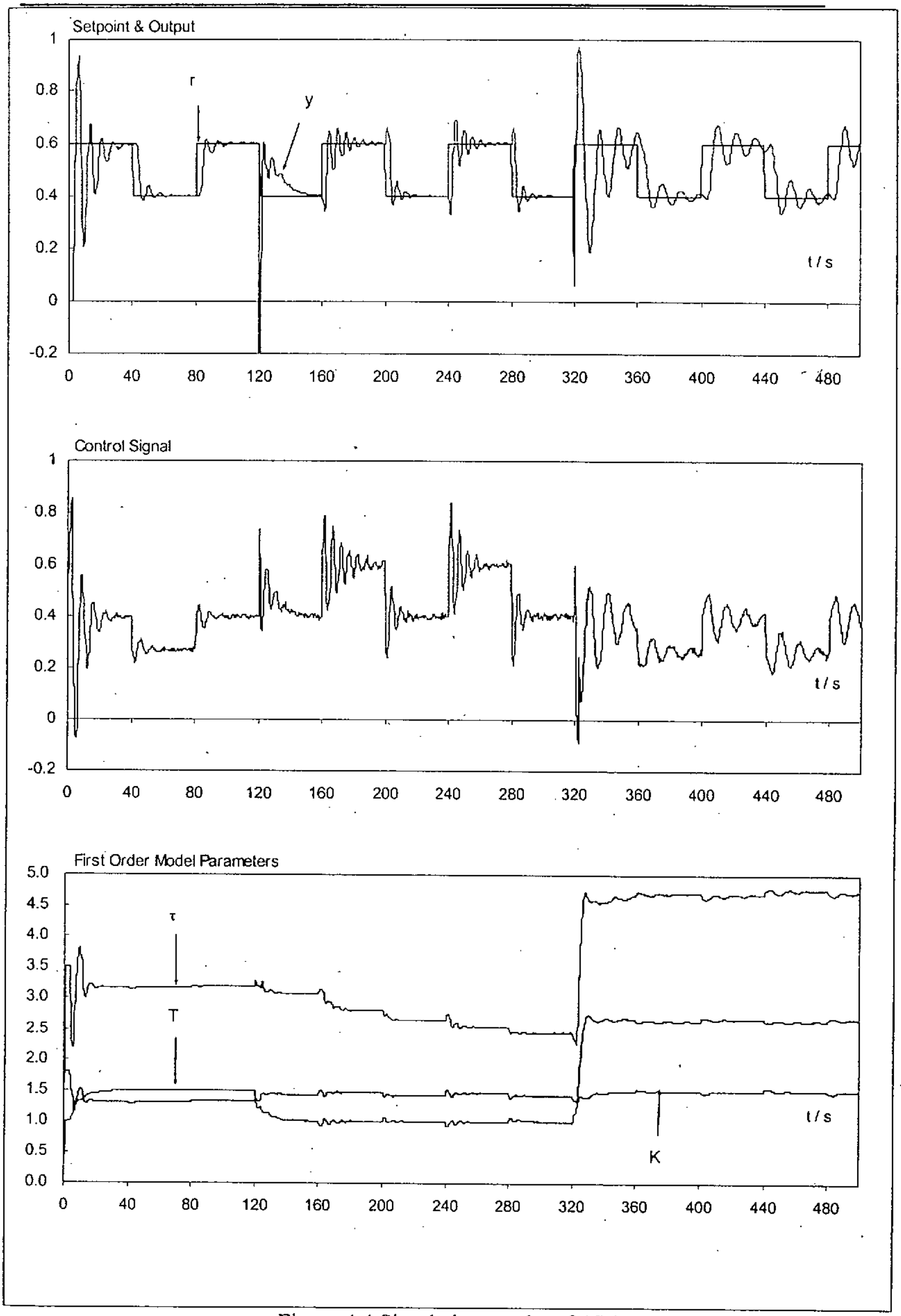


Figure 4.4 Simulation results of APID

	K	T	τ	K_u	T_u
Smith Method	1.5	1.65	3.00	1.06	8.38
Minimized-error	1.5	1.46	3.11	0.98	8.44
Proposed method	1.5	1.33	3.19	0.936	8.48
Process I	-	-	-	1.036	8.438

Table 4.1: Process I FOPDT model parameters

	K	T	τ	K_u	T_u
Smith Method	1.0	1.89	2.43	1.93	7.22
Minimized-error	1.0	1.67	2.55	1.74	7.35
Proposed method	1.0	1.39	2.45	1.64	6.91
Process II	-	-	-	1.54	6.83

Table 4.2: Process II FOPDT model parameters

	K	T	τ	K_u	T_u
Smith Method	1.5	2.49	4.86	1.03	13.39
Minimized-error	1.5	2.057	5.1	0.923	13.49
Proposed method	1.5	2.66	4.75	1.07	13.33
Process III	-	-	-	0.987	13.48

Table 4.3: Process III FOPDT model parameters

4.6 Experimental Studies

In this section, a process trainer PT326 from Feedback [Feedback 1982] was used to demonstrate the performance of the algorithm on a variable time delay system. The system consisted of an adjustable air blower, a length of tube and a heater. The air was drawn through the throttle opening by the centrifugal blower. It was then warmed up by the heater as it passed through the tube length and sent back to the atmosphere. The system input was the heater drive voltage and with its output the air temperature which was measured by a thermistor fitted to the end of the tube. The schematic diagram of the heater is shown in Figure 4.5. An open loop test was then carried out on the system and from the process reaction curve, a first-order model was deduced as:

$$G(s)=\frac{Y(s)}{U(s)}=\frac{1.1e^{-0.35s}}{0.5s+1}$$

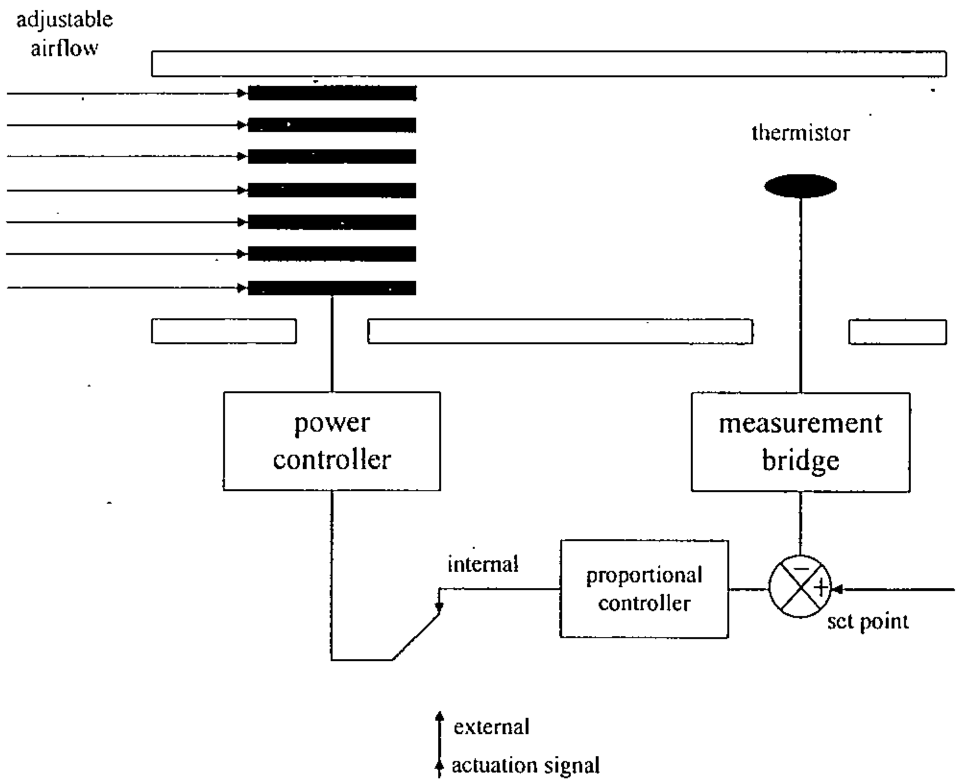


Figure 4.5 Block diagram for the feedback processes trainer (PT326)

The time delay of the system was fixed and very small as compared to its time constant. In order to demonstrate the performance of the proposed algorithm for a system with dominant time delay, a variable delay was added to the system output signal. Therefore, the system transfer function was

$$G(s)=\frac{Y(s)}{U(s)}=\frac{1.1e^{-\tau s}}{0.6s+1}$$

$\tau = 1.0$ for $0 \le t \le 80$
$\tau = 2.0$ for $80 \le t \le 160$
$\tau = 3.0$ for $160 \le t \le 240$

In this experiment, we also modified the control algorithm to predictive PI controller (PIP) [Hagglund 1992] that is specially suited to dominant time delay systems. The system is regarded as a dominant time delay process if $\tau \ge 5T$. The output of the PIP controller is in the from of

$$u(t) = K_p(e(t) + \frac{1}{T_i}\int e(t)dt) - \frac{1}{T_i}\int (u(t) - u(t-\tau))dt$$

Here, K_p, T_i are the proportional gain and the integral time constant, which are calculated based on FOPDT model static gain and time constant as

$$K_p = \frac{1}{K} \quad \text{and} \quad T_i = T$$

the parameters of the PIP controller was obtained on-line from the corresponding parameter of the estimated FOPDT model. The set-point was a square wave with amplitude of 1.0 and period of 80 second superimposed on DC offset level 6. The PIP controller was initialized with $K_p = 1$, $T_i = 1000$. The consequent values of fuzzy system were initialized with $\theta_1 = 1.0$, $\theta_2 = 0.5$ and $\theta_3 = 2.0$. The value of $L = 5$ for the operating range for the control input and $S_{i,\max} = 1$ for all i. The learning rates were chosen as $\eta_1 = 0.25$, $\eta_2 = 0.4$ and $\eta_3 = 0.8$. We compared the performance of the adaptive PIP with the adaptive Ziegler-Nichols PI ($K_p = 0.45K_u$ and $T_i = 0.85T_u$) under the same condition. Figures 4.6 and 4.7 show the results for real time temperature control system. It can be seen that the FOPDT fuzzy model can track the variable time delay. It is observed that the comparison between the performance of the adaptive PIP and adaptive PI, the tracking error of adaptive PIP is less than adaptive PI under the same condition and adaptive PIP control can maintain a better tracking performance for this system.

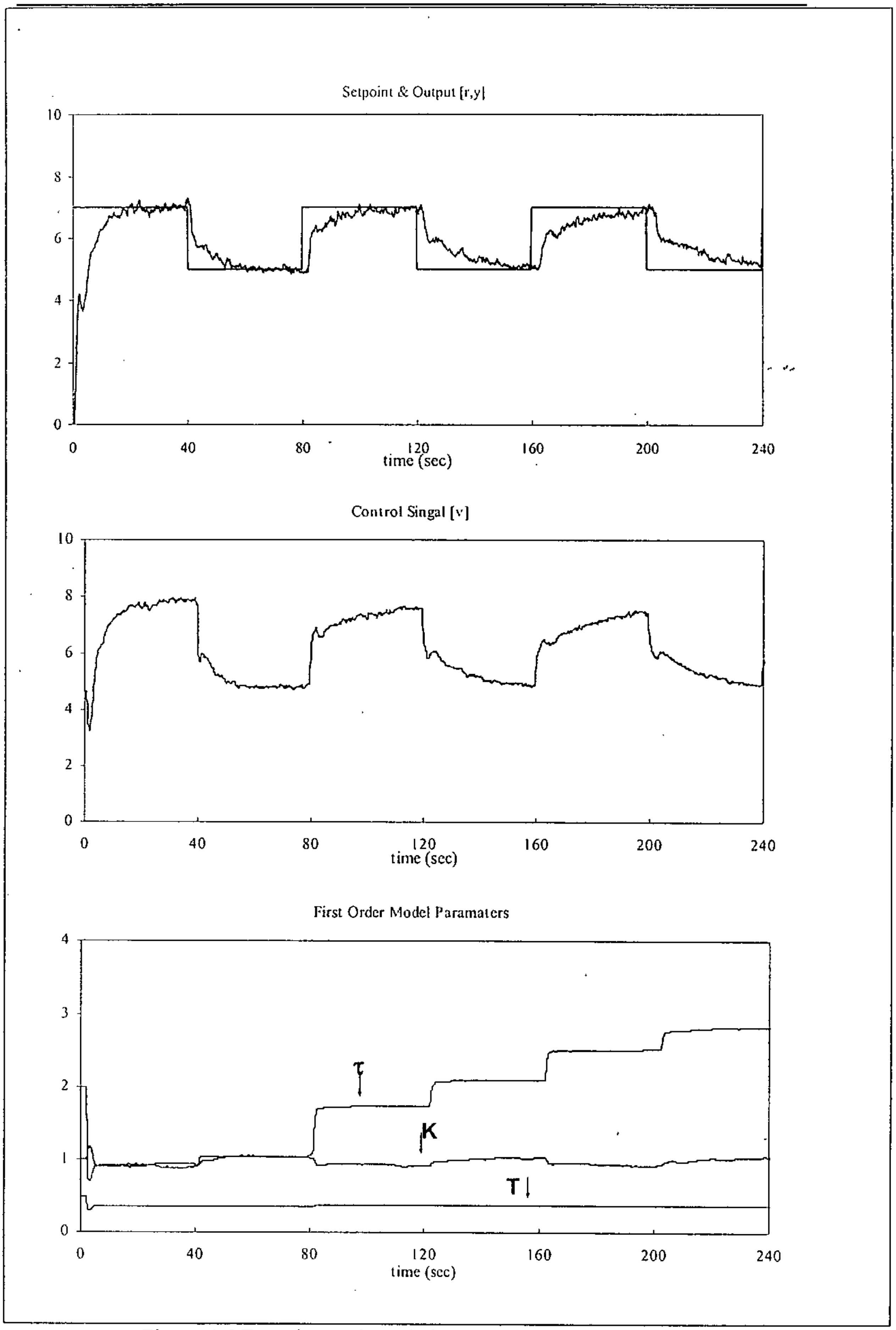


Figure 4.6 Experimental result for Adaptive PI with variable delay.

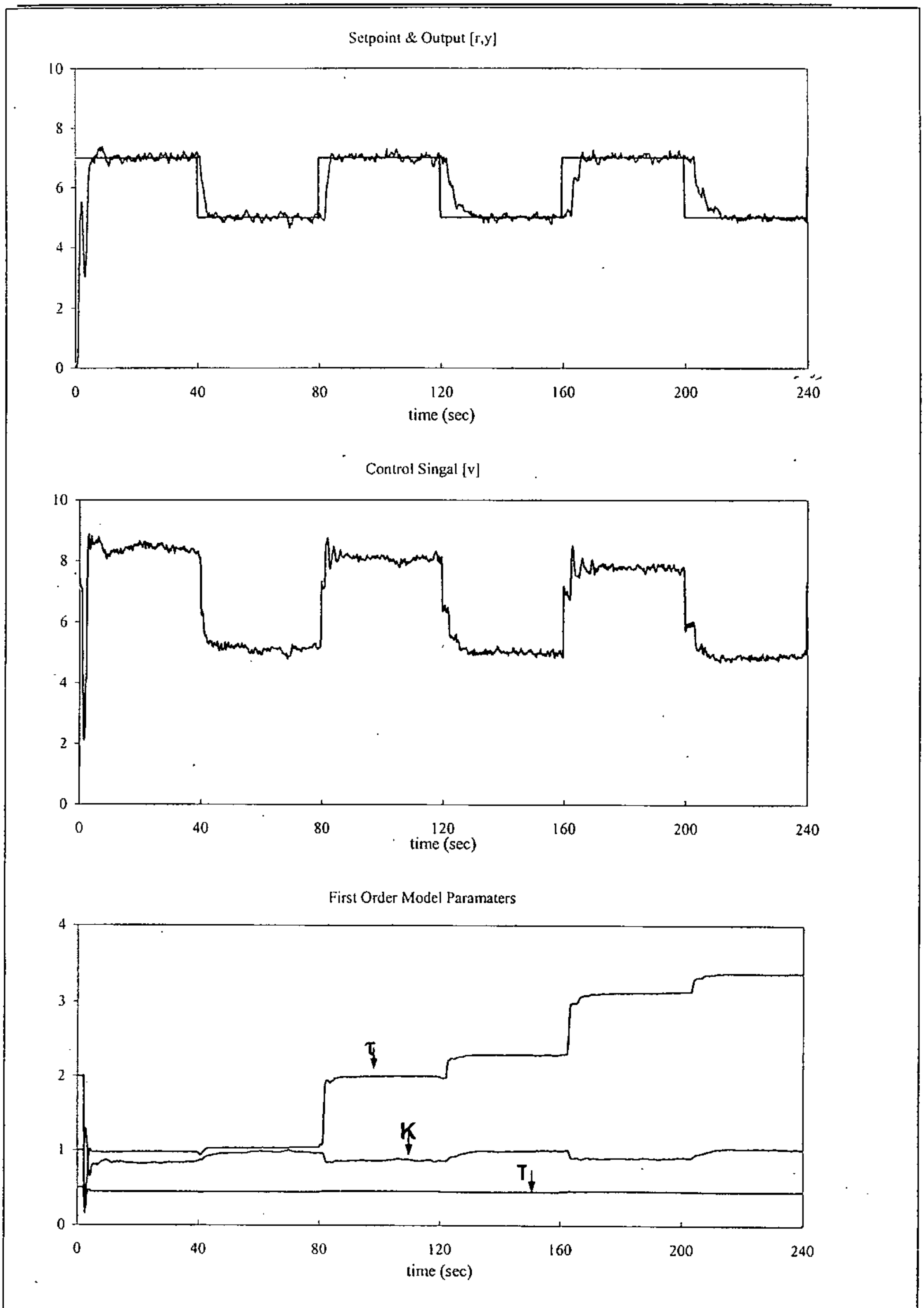


Figure 4.7 Experimental result for Adaptive PIP with variable delay.

4.7 Conclusions

In this chapter, we discussed an on-line FOPDT modeling method in which the model parameters are obtained by using fuzzy logic system. The proposed method is different from conventional fuzzy identification, the fuzzy system outputs are the three parameters of the FOPDT model. Combining the proposed FOPDT identification algorithm with a fixed structure controller, an on-line adaptive control using fuzzy system has been proposed. The stability analysis for the FOPDT model identification has been stated and Lyapunov direct method has been applied to show the convergence of the identification algorithm. The adaptive control scheme has been successfully applied to temperature control system with variable time delay. Simulation and experimental results show that the proposed method can track the system parameters' changes and gives satisfactory closed-loop performance.

This chapter also concludes the first stage of this project. In fact, both artificial neural network and fuzzy system approaches for on-line lower-order modeling are equivalent. The difference between them is mainly in the structure. It is noticed that the NN is highly parallel architecture consisting of simple processing elements which are able to learn from given data. In addition, contract to fuzzy system, it is not possible to integrate prior knowledge to simplify the learning process, or to extract knowledge in form of rules from the trained network. Moreover, it is observed that due of the network architecture, the computation load of NN is higher than fuzzy system.

In the next phase, general adaptive fuzzy control systems for special classes of non-linear systems will be investigated.

CHAPTER FIVE

ADAPTIVE FUZZY SLIDING MODE CONTROL WITH CHATTERING ELIMINATION

5.1 Introduction

A novel fuzzy on-line lower order model approximation method was presented in Chapter 4. In the following chapters, we will move onto more general fuzzy logic systems. More design methods and stability analysis techniques for nonlinear systems will be explored. In this chapter, we will discuss a new adaptive fuzzy control algorithm which combine sliding mode controller (SMC) and a proportional plus integral (PI) controller into a single adaptive fuzzy sliding mode controller (AFSMC). The AFSMC can tackle the control problem of unknown nonlinear systems and systems with modeling uncertainties and external disturbance. The role of the controller is to schedule control under different operating conditions. In this way, the advantage of each control method before combining can be retained while any disadvantages are removed. The proposed control scheme provides good transient and robust performance. Moreover, the drawback of chattering phenomenon in SMC can be avoided. The AFSMC scheme will be proved to guarantee the global stability of the closed-loop system.

The rest of this chapter is organized as follow: Section 5.2 gives a brief review of SMC. The adaptive fuzzy sliding mode controller is included in Section 5.3. Computer simulation results for the proposed control algorithm are illustrated in Section 5.4. Finally, the chapter is concluded in Section 5.5.

This chapter is based on the papers 4 and 13, 16 on chapter one, pages 11 and 12

5.2 A Brief Review on Sliding Mode Control (SMC) for Uncertain Systems

Consider the n-th order nonlinear Single-input Single-Output (SISO) systems in the following form:

$$\begin{aligned} x^{(n)} &= f(x) + g(x)u + d(t) \\ y &= x \end{aligned} \tag{5.1}$$

where $f(x), g(x)$ are smooth, unknown nonlinear functions and $g(x) > 0$ $x = [x, \dot{x}, \cdots, x^{(n-1)}]^T = [x_1, x_2, \cdots, x_n]^T \in R^n$ is the state vector of the systems which is assumed to be available for measurement, $u \in R$ and $y \in R$ are the input and the output of the system, respectively, and $d(t)$ is the unknown external disturbance. Assuming that the upper bound of the disturbance $d(t)$ is D, that is $|d(t)| \leq D$. As in Chapter 2, Section 2.3.2 the control objective is to design a control action for the state x to track a desired reference state y_m in the presence of model uncertainties and external disturbances. Assume the nonlinear function $f(x)$ and $g(x)$ are known, the following SMC control input u^* can guarantee the sliding condition (for further details please refer to Section 2.3.2).

$$u^* = \frac{1}{g(x)}\left[-\sum_{i=1}^{n-1} c_i e^{(i)} - f(x) + y_m^{(n)} - \eta\,\mathrm{sgn}(s)\right] \tag{5.2}$$

5.3 Adaptive Fuzzy Sliding Mode Control Scheme

In this section, a sliding model control (SMC) and a PI control are combined into a single fuzzy sliding mode controller. Obviously, the nonlinear function $f(x)$ and $g(x)$ are generally unknown and the ideal controller (2.19 or 5.2) cannot be implemented in practice. Also, the switching control term u_{sw} will cause chattering problem. We replace $f(x)$ and $g(x)$ by the fuzzy logic system (2.49) (for further details please refer to Section 2.4.2.1). To avoid chattering phenomenon and ensure zero steady-state error for the closed-loop system, a PI controller is used. The input and output of the continuous time PI controller is in the form of:

$$u_p = k_p s + k_i \int s dt \tag{5.3}$$

where k_p and k_i are control gains to be designed. Equation (5.3) can be rewritten as

$$\hat{p}(s \mid \theta_p) = \theta_p^T \psi(s) \tag{5.4}$$

where $\theta_p = [k_p, k_i]^T \in R^2$ is an adjustable parameter vector and $\psi^T(s) = [s, \int s dt] \in R^2$ is a regressive vector. In order to derive the SMC law (5.2), we use fuzzy logic system to approximate the unknown functions $f(x)$, $g(x)$ and an adaptive PI control term is used to attenuate the chattering action and improve the steady state performance.

Hence, we have:

$$u = \frac{1}{\hat{g}(x \mid \theta_g)} \left[-\hat{f}(x \mid \theta_f) - \sum_{i=1}^{n-1} c_i e^i + y_m^{(n)} - \hat{p}(s \mid \theta_p) \right] \tag{5.5}$$

$$\hat{f}(x \mid \theta_f) = \theta_f^T \xi(x) \tag{5.6}$$

$$\hat{g}(x \mid \theta_g) = \theta_g^T \xi(x) \tag{5.7}$$

The switching control u_{sw} is replaced by the PI control when the state is within a boundary layer $|s| < \Phi$ and the control action is kept at the saturated value when the state is outside the boundary layer. Hence, we set $|\hat{p}(s \mid \theta_p)| = D + \eta + \omega_{max}$ where $|s| \geq \Phi$, Φ is the widthness of the boundary layer and ω_{max} is the maximum approximation error of the fuzzy system.

Theorem 5.1: Consider the control problem of the nonlinear system (5.1). If the control action equation (5.5) is used, the functions $\hat{f}$, $\hat{g}$ and $\hat{p}$ are estimated by equation (5.4), (5.6) and (5.7) the parameters vector θ_f, θ_g and θ_p are adjusted by the adaptive law equation (5.8)-(5.10), the closed loop system signals will be bounded and the tracking error will converge to zero asymptotically.

$$\dot{\theta}_f = \gamma_1 s \xi(x) \tag{5.8}$$

$$\dot{\theta}_g = \gamma_2 s \xi(x) u \tag{5.9}$$

$$\dot{\theta}_p = \gamma_3 s \psi(s) \tag{5.10}$$

The proof is given as follows:

Proof: Define the optimal parameters of fuzzy systems

$$\theta_f^* = \arg \min_{\theta_f \in \Omega_f} \left(\sup_{x \in R^n} | \hat{f}(x | \theta_f) - f(x) | \right) \tag{5.11}$$

$$\theta_g^* = \arg \min_{\theta_g \in \Omega_g} \left(\sup_{x \in R^n} | \hat{g}(x | \theta_g) - g(x) | \right) \tag{5.12}$$

$$\theta_p^* = \arg \min_{\theta_p \in \Omega_p} \left(\sup_{s \in R} | \hat{p}(s | \theta_p) - u_{sw} | \right) \tag{5.13}$$

where Ω_f , Ω_g and Ω_p are constraint sets for θ_f , θ_g and θ_p ,respectively. Define the minimum approximation error.

$$\omega = f(x) - \hat{f}(x | \theta_f^*) + (g(x) - \hat{g}(x | \theta_g^*))u \tag{5.14}$$

Assumption 5.1: $$\Omega_f = \{\theta_f \in R^n \| \theta_f | \le M_f\} \tag{5.15}$$

$$\Omega_g = \{\theta_g \in R^n | 0 < \varepsilon \le | \theta_g | \le M_g\} \tag{5.16}$$

and $$\Omega_p = \{\theta_p \in R^2 \| \theta_p | \le M_p\} \tag{5.17}$$

where M_f, ε, M_g and M_p are pre-specified parameters for estimated parameters' bound.

Assuming that the fuzzy parameters θ_f, θ_g and the PI control parameter θ_p never reach the boundaries. Then, we have

$$
\begin{aligned}
\dot{s} &= \sum_{i=1}^{n-1} c_i e^{(i)} + x^{(n)} - y_m^{(n)} \\
&= \sum_{i=1}^{n-1} c_i e^{(i)} + f(x) + g(x)u + d(t) - y_m^{(n)} \\
&= \sum_{i=1}^{n-1} c_i e^{(i)} + f(x) - \hat{f}(x \mid \theta_f) + (g(x) - \hat{g}(x \mid \theta_g))u \\
&\quad - \sum_{i=1}^{n-1} c_i e^{(i)} - \hat{p}(s \mid \theta_p) + d(t) \\
&= f(x) - \hat{f}(x \mid \theta_f) + (g(x) - \hat{g}(x \mid \theta_g))u - \hat{p}(s \mid \theta_p) \\
&\quad + d(t) \\
&= \hat{f}(x \mid \theta_f^*) - \hat{f}(x \mid \theta_f) + (\hat{g}(x \mid \theta_g^*) - \hat{g}(x \mid \theta_g))u - \hat{p}(s \mid \theta_p) \\
&\quad + \hat{p}(s \mid \theta_p^*) - \hat{p}(s \mid \theta_p^*) + d(t) + \omega \\
&= \phi_f^T \xi(x) + \phi_g^T \xi(x) \cdot u + \phi_p^T \psi(s) - \hat{p}(s \mid \theta_p^*) + d(t) + \omega
\end{aligned}
\tag{5.18}
$$

where $\phi_f = \theta_f^* - \theta_f, \phi_g = \theta_g^* - \theta_g, \phi_p = \theta_p^* - \theta_p$

Now consider the Lyapunov candidate

$$
V = \frac{1}{2}s^2 + \frac{1}{2}\phi_f^T\phi_f + \frac{1}{2}\phi_g^T\phi_g + \frac{1}{2}\phi_p^T\phi_p \tag{5.19}
$$

The time derivative of V along the error trajectory (5.19) is

$$
\begin{aligned}
\dot{V} &= s\dot{s} + \frac{1}{\gamma_1}\phi_f^T\dot{\phi}_f + \frac{1}{\gamma_2}\phi_g^T\dot{\phi}_g + \frac{1}{\gamma_3}\phi_p^T\dot{\phi}_p \\
&= s(\phi_f^T\xi(x) + \phi_g^T\xi(x)u + \phi_p^T\psi(s) - \hat{p}(s \mid \theta_h^*) + \omega + d(t)) \\
&\quad + \frac{1}{\gamma_1}\phi_f^T\dot{\phi}_f + \frac{1}{\gamma_2}\phi_g^T\dot{\phi}_g + \frac{1}{\gamma_3}\phi_p^T\dot{\phi}_p \\
&= s\phi_f^T\xi(x) + \frac{1}{\gamma_1}\phi_f^T\dot{\phi}_f + s\phi_g^T\xi(x)u + \frac{1}{\gamma_2}\phi_g^T\dot{\phi}_g + s\phi_p^T\psi(s) \\
&\quad + \frac{1}{\gamma_3}\phi_p^T\dot{\phi}_p - s\hat{p}(s \mid \theta_h^*) + s\omega + sd(t) \\
&= \frac{1}{\gamma_1}\phi_f^T(\gamma_1 s\xi(x) + \dot{\phi}_f) + \frac{1}{\gamma_2}\phi_g^T(\gamma_2 s\xi(x)u + \dot{\phi}_g) \\
&\quad + \frac{1}{\gamma_3}\phi_p^T(s\psi(s) + \dot{\phi}_p) - s\hat{p}(s \mid \theta_p^*) + s(\omega + d(t))
\end{aligned}
$$

$$\begin{aligned} &\le \frac{1}{\gamma_1}\phi_f^T(\gamma_1 s\xi(x)+\dot{\phi}_f)+\frac{1}{\gamma_2}\phi_g^T(\gamma_2 s\xi(x)u+\dot{\phi}_g) \\ &+\frac{1}{\gamma_3}\phi_p^T(s\psi(s)+\dot{\phi}_p)-s(D+\eta)\operatorname{sgn}(s)+sd(t)+s\omega \\ &< \frac{1}{\gamma_1}\phi_f^T(\gamma_1 s\xi(x)+\dot{\phi}_f)+\frac{1}{\gamma_2}\phi_g^T(\gamma_2 s\xi(x)u+\dot{\phi}_g) \\ &+\frac{1}{\gamma_3}\phi_p^T(s\psi(s)+\dot{\phi}_p)-\eta|s|+s\omega \end{aligned} \tag{5.20}$$

where $\dot{\phi}_f=-\dot{\theta}_f$, $\dot{\phi}_g=-\dot{\theta}_g$ and $\dot{\phi}_p=-\dot{\theta}_p$. Substitute (5.8)-(5.10) into (5.20), then we have

$$\dot{V}\le s\omega-\eta|s|\le 0 \tag{5.21}$$

Since ω it the minimum approximation error, condition in (5.21) is the best result that can be obtained. Therefore, all signals in the system are bounded. Obviously, if $e(0)$ is bounded, then $e(t)$ is also bounded for all t. Since the reference signal y_m is bounded, then the system states $x(t)$ is bounded as well. To complete the proof and establish asymptotic convergence of the tracking error, we need proving that $s\to 0$ as $t\to\infty$. Assume that $|s|\le\eta_s$, then equation (5.21) can be rewritten as

$$\dot{V}\le|s||\omega|-|s|\eta\le\eta_s|\omega|-|s|\eta \tag{5.22}$$

Integrating both sides of (5.22), we have

$$\int_0^\tau|s|\,d\tau\le\frac{1}{\eta}(|V(0)|+|V(t)|)+\frac{\eta_s}{\eta}\int_0^\tau|\omega|\,d\tau \tag{5.23}$$

then we have $s\in L_1$. Form (5.21), we know that s is bounded and every term in (5.18) is bounded. Hence, $s,\dot{s}\in L_\infty$, use of Barbalat's lemma [Sastry and Bodson 1989]. We have $s(t)\to 0$ as $t\to\infty$, the system is stable and the error will asymptotically converge to zero. The above stability result is achieved under the assumption 5.1 that all the parameters boundness is ensured. To guarantee the parameters are bounded. The adaptive laws (5.8)-(5.10) can be modified by using the projection algorithm [Wang 1993, 1997]. The modified adaptive laws are given as follows.

For θ_f, use

$$\dot{\theta}_f = \begin{cases} \gamma_1 s\xi(x) & \text{if } (|\theta_f| < M_f) \\ & \text{or } (|\theta_f| = M_f \text{ and } s\theta_f^T\xi(x) \geq 0) \\ P_f[\gamma_1 s\xi(x)] & \text{if } (|\theta_f| = M_f) \\ & \text{and } s\theta_f^T\xi(x) < 0) \end{cases} \tag{5.24}$$

For θ_g, use

whenever an element θ_{gi} of θ_g equals ε, use

$$\dot{\theta}_{gi} = \begin{cases} \gamma_2 s\xi_i(x)u & \text{if } s\xi_i(x)u < 0 \\ 0 & \text{if } s\xi_i(x)u \geq 0 \end{cases} \tag{5.25}$$

where $\xi_i(x)$ is the i-th component of $\xi(x)$. Otherwise, use

$$\dot{\theta}_g = \begin{cases} \gamma_2 s\xi(x)u & \text{if } (|\theta_g| < M_g) \\ & \text{or } (|\theta_g| = M_g \text{ and } s\theta_g^T\xi(x)u \geq 0) \\ P_g[\gamma_2 s\xi(x)u] & \text{if } (|\theta_g| = M_g) \\ & \text{and } s\theta_g^T\xi(x)u < 0) \end{cases} \tag{5.26}$$

$$\dot{\theta}_p = \begin{cases} \gamma_3 s\psi(s) & \text{if } (|\theta_p| < M_p) \\ & \text{or } (|\theta_p| = M_p \text{ and } s\theta_p^T\psi(s) \geq 0) \\ P_p[\gamma_3 s\psi(s)] & \text{if } (|\theta_p| = M_p) \\ & \text{and } s\theta_p^T\psi(s) < 0) \end{cases} \tag{5.27}$$

where the projection operator $P_f[*]$, $P_g[*]$ and $P_p[*]$ are defined as

$$P_f[\gamma_1 s\xi(x)] = \gamma_1 s\xi(x) - \gamma_1 s\frac{\theta_f\theta_f^T\xi(x)}{|\theta_f|^2} \tag{5.28}$$

$$P_g[\gamma_2 s\xi(x)u] = \gamma_2 s\xi(x)u - \gamma_2 s\frac{\theta_g\theta_g^T\xi(x)u}{|\theta_g|^2} \tag{5.29}$$

$$P_p[\gamma_3 s\psi(s)] = \gamma_3 s\psi(s) - \gamma_3 s\frac{\theta_p\theta_p^T\psi(s)}{|\theta_p|^2} \tag{5.30}$$

Then, the overall adaptive fuzzy sliding mode control scheme is shown in Figure 5.1.

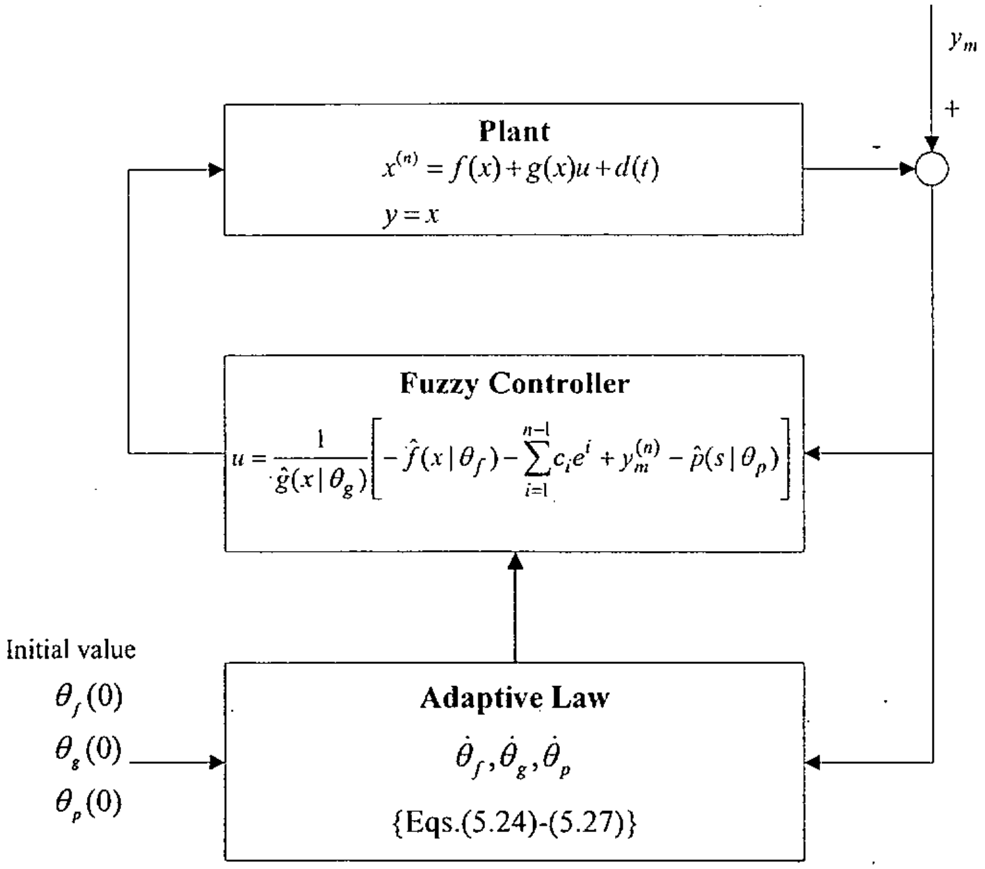


Figure 5.1 Overall Scheme of the Adaptive Fuzzy Sliding Mode Control System

To summarize the above analysis, the step-by-step procedure for the adaptive fuzzy sliding mode control algorithm is proposed as follow.

Design Procedure:

Step 1. Select proper initial values of PI parameters.

Step 2. Specify the desired coefficients $c_1, c_2, \cdots, c_{n-1}$ such as in (2.30).

Step 3. Select the learning coefficients γ_1, γ_2 and γ_3.

Step 4. Define m_i fuzzy sets $F_i^{'}$ for linguistic variable x_i and the membership functions μ_{F_i} is uniformly cover the universe of discourse, for $i = 1,2,\cdots,n$.

Step 5. Construct the fuzzy rule bases for the fuzzy system $\hat{f}(x \mid \theta_f)$ and $\hat{g}(x \mid \theta_g)$.

Step 6. Construct the fuzzy systems $\hat{f}(x \mid \theta_f) = \theta_f^T \xi(x)$ and $\hat{g}(x \mid \theta_g) = \theta_g^T \xi(x)$ in (2.49).

Step 7. Construct the control law (5.5) with the adaptive law in (5.24-5.27).

Step 8. Obtain the control and apply to the plant, then compute the adaptive law (5.24-5.27) to adjust the parameter vector θ_f, θ_g and θ_p.

5.4 Simulation Examples

In this section, we apply our proposed adaptive fuzzy controller for three cases. The first example is a regulation problem to let the output of a first order nonlinear system to track a constant trajectory. The second example is to let a Duffing forced-oscillation system to track a sin-wave trajectory. The third example is to let the inverted pendulum to track a desired trajectory.

***Example 5.1*: State Regulation**

In this example, we verify at the validity of the design approach on the regulation control of a first order nonlinear system. The dynamic equation of such system is given by [Wang 1997].

$$\dot{x}(t) = \frac{1 - e^{-x(t)}}{1 + e^{-x(t)}} + u(t) \tag{5.31}$$

Two regulation simulations ($y_m \equiv 0$) were used for convectional sliding mode controller and the proposed controller. Firstly, we defined $y_m = 0$, $s = e$ the initial state $x(0) = 1.5$ and step size 0.02s for both controllers. Choose $\hat{f}(x)=1, \Delta f(x)=(-2e^{-x(t)}/(1+e^{-x(t)})), |\Delta f| \leq F=2$. Hence, the convectional sliding control law is $u(t)=-\hat{f}(x)-(\eta+F)\,\mathrm{sgn}(s)$, where $\eta=0.1$. For the proposed controller, the initial values of parameters θ_p are set by $k_P(0)=4$ and $k_i(0)=10$. Choose six fuzzy sets over the interval $[-3,3]$ for state x. The membership functions are

$$\mu_{NB}(x)=1/(1+\exp(5(x+2)))$$

$$\mu_{NM}(x)=\exp(-(x+1.5)^2)$$

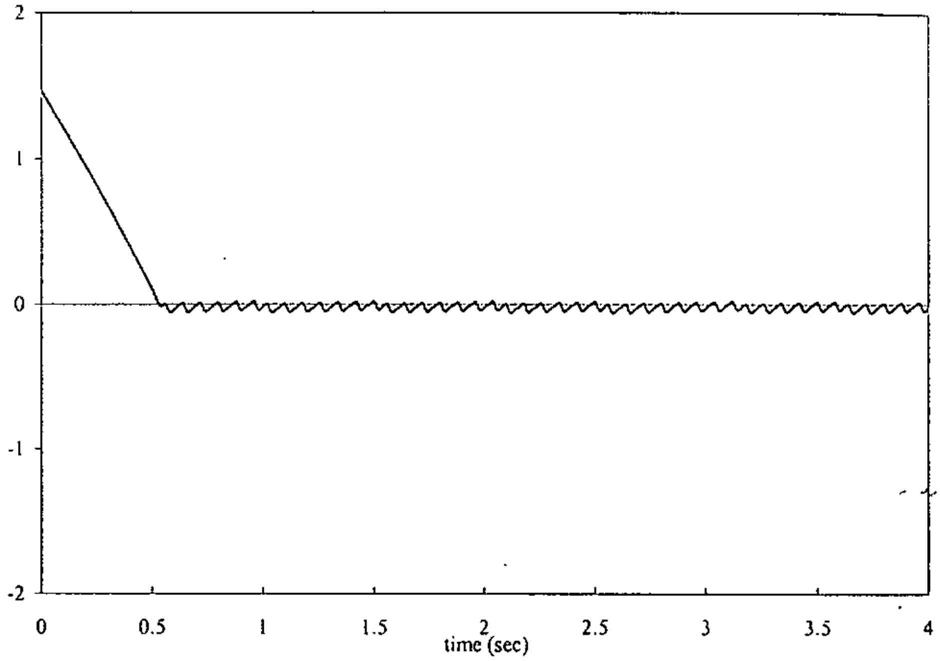
$$\mu_{NS}(x)=\exp(-(x+0.5)^2)$$

$$\mu_{PS}(x)=\exp(-(x-0.5)^2)$$

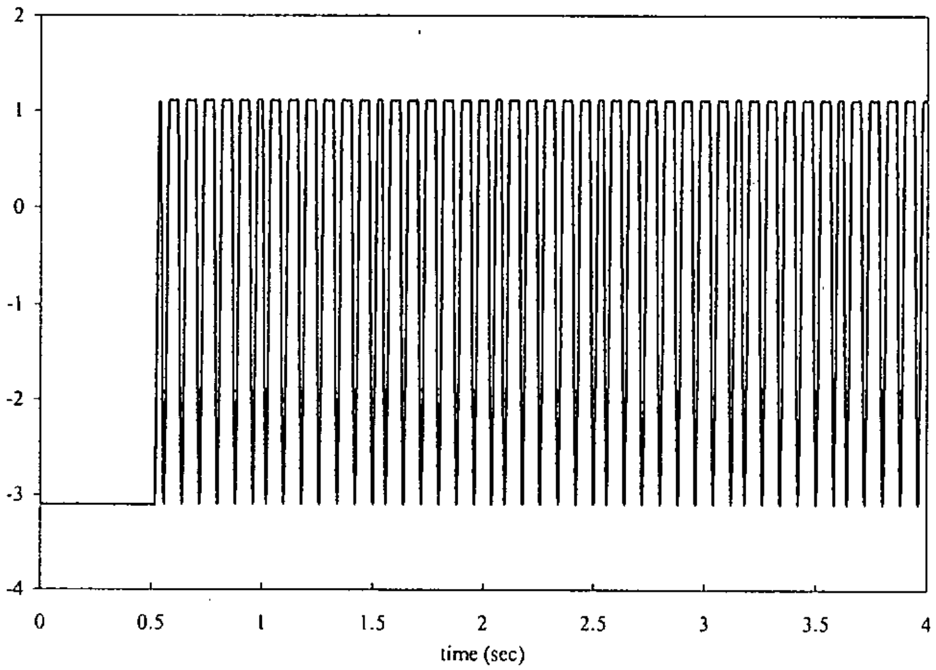
$$\mu_{PM}(x)=\exp(-(x-1.5)^2)$$

$$\mu_{NB}(x)=1/(1+\exp(5(x-2)))$$

The initial consequent parameters of fuzzy are chosen randomly in the interval $[-2,2]$. The width of the boundary layer $\Phi=0.25$ and the learning rate $\gamma_1=40, \gamma_3=120$. A Gaussian noise with mean zero and variance of 0.0002 was injected at the output of the system. Figure 5.2 and Figure 5.3 show the simulation results for convectional SMC and proposed controller, respectively. From the results, we find that for the convectional SMC can track the desired value. However, the chattering and the control signal ringing are obvious. Comparing Figure 5.2 and Figure 5.3, we see that the chattering disappeared and the steady tracking error can be removed using the proposed control algorithm.

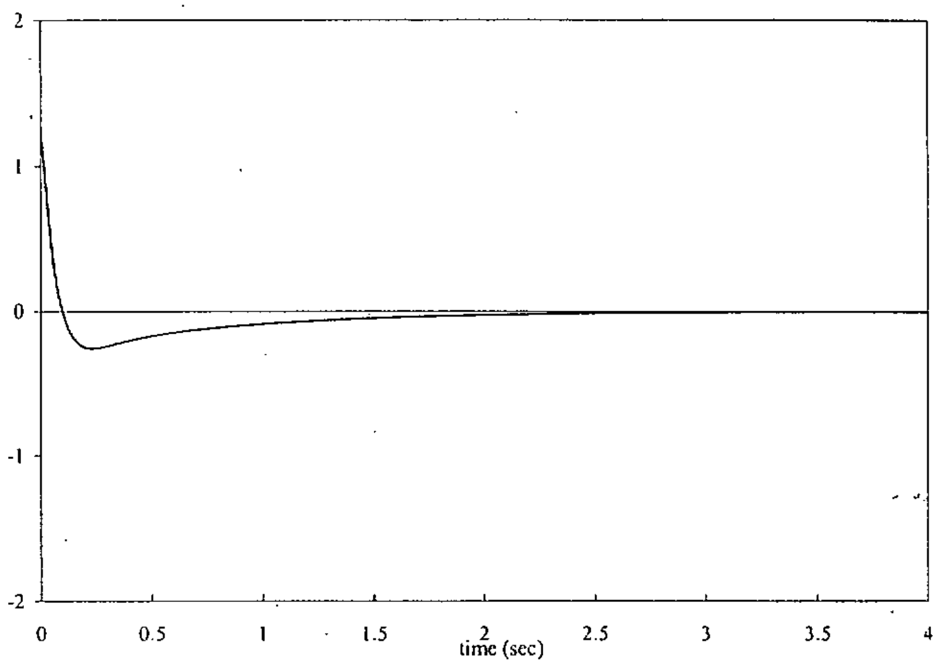


(a)

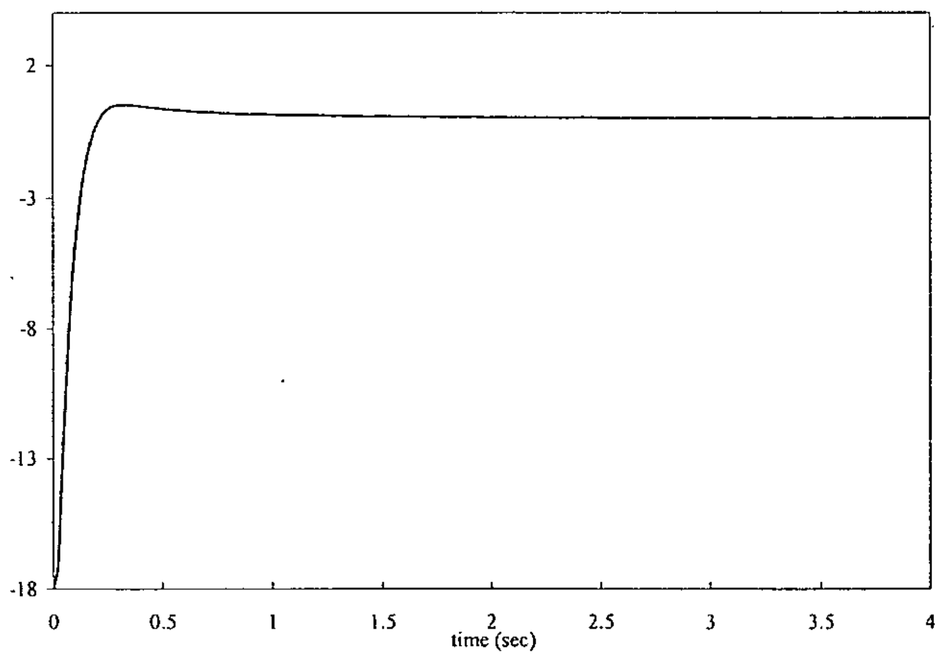


(b)

Figure 5.2 Simulation results of convectional sliding mode control applied to first order nonlinear system: (a) desired output y_m and system output x (b) control signal u .



(a)



(b)

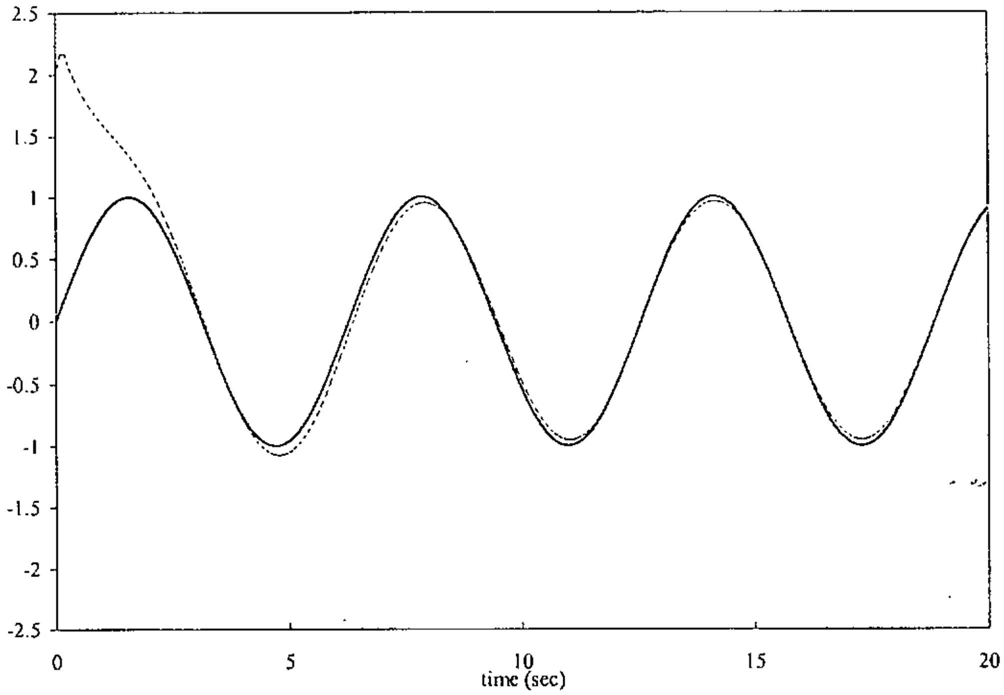
Figure 5.3 Simulation results of adaptive fuzzy sliding mode control applied to first order nonlinear system: (a) desired output y_m and system output x (b) control signal u .

***Example 5.2*: A Duffing forced-oscillation system**

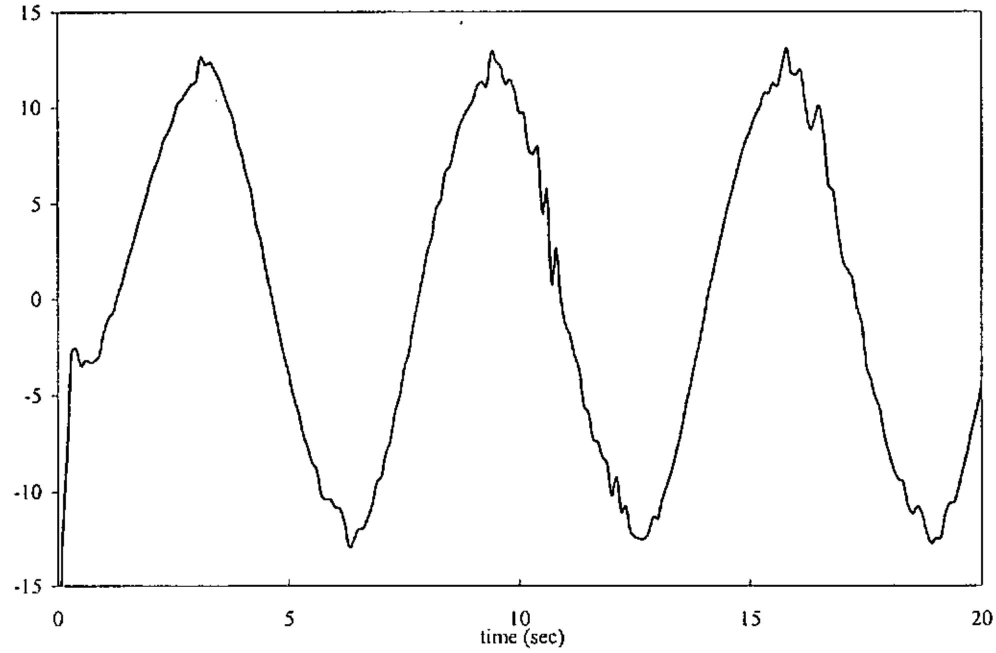
In this example, we verify at the validity of the design approach on the tracking control of a Duffing forced-oscillation system. The dynamic equations of such system are given by [Wang 1997].

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -0.1x_2 - x_1^3 + 12\cos(t) + u(t) + d(t) \end{aligned} \tag{5.32}$$

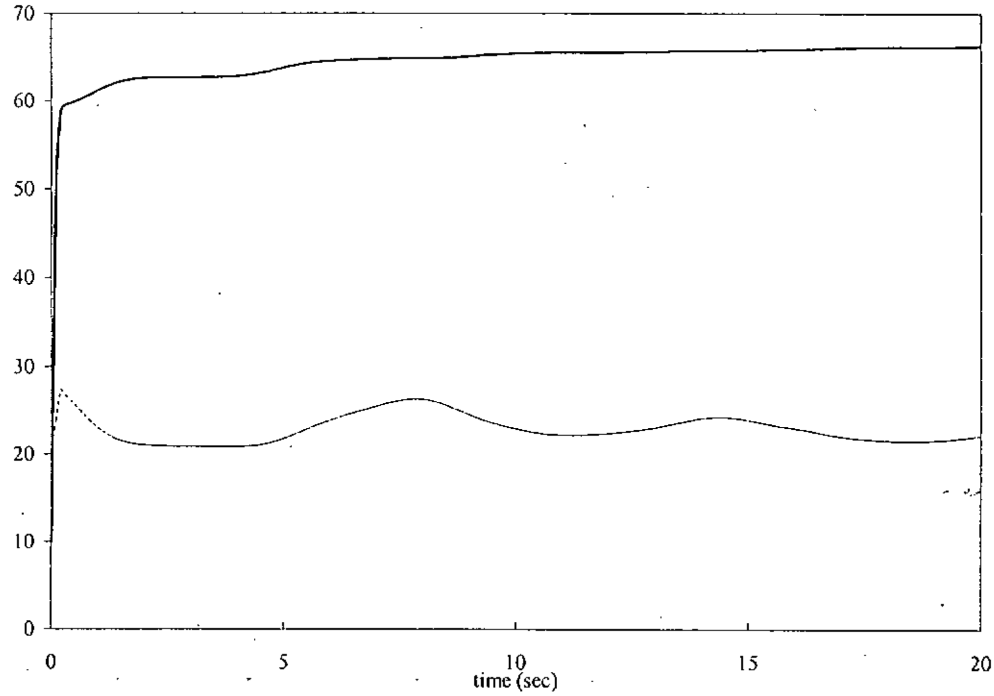
This system is chaotic without control (with sensitive dependence on initial condition. A small difference in initial position will make the outcome completely different) [Alligood *et al.* 1997]. The control objective is to maintain the system to track the desired angle trajectory, $y_m = \sin(t)$ and d is assumed to be a square wave with amplitude ± 0.5 and the period 2π. Choose the sliding surface as $s = c_1 e + \dot{e}$, $c_1 = 4$. The initial values of parameters θ_p are set by $k_p(0) = 10$ and $k_i(0) = 20$. The membership functions for system state x_i, $i = 1,2$ are chosen as in Example 5.1, then there are 36 rules to approximate the system functions f. And initial consequent parameters of fuzzy rules are chosen randomly in the interval $[-2,2]$. We select $M_f = 30$ and $M_p = 80$. Let the width of the boundary layer $\Phi = 0.3$ and the learning rate $\gamma_1 = 25$ and $\gamma_3 = 80$. Choose the initial condition $x = [2,2]^T$ and step size 0.02s, Moreover, a Gaussian noise with mean zero and variance of 0.025 was injected at the output of the system. Figure 5.4 shows the simulation results. It can be seen that the tracking performance can achieve.


2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
5
10
15
20
time (sec)

(a)


15
10
5
0
-5
-10
-15
0
5
10
15
20
time (sec)

(b)



(c)

Figure 5.4 Simulation results of adaptive fuzzy sliding mode control applied to the Duffing forced-oscillation system: (a) desired output y_m (solid line) and system output x_1 (dash line) (b) control signal u (c) trajectory of gains K_p (solid line) and K_i (dash line)

***Example 5.3*: An inverted pendulum system**

In this example, we test the adaptive fuzzy controller on the tracking control of the benchmark control problem of inverted pendulum in Figure 5.5. Let $x_1 = \theta$ be the angle of the pendulum with respect to the vertical line and $x_2 = \dot{\theta}$. The dynamic equations of such system are given by [Wang 1997].

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= \frac{g \sin x_1 - mlx_2 \cos x_1 \sin x_1 /(m_c + m)}{l(4/3 - m\cos^2 x_1 /(m_c + m))} \\
&\quad + \frac{\cos x_1 /(m_c + m)}{l(4/3 - m\cos^2 x_1 /(m_c + m))} u + d
\end{aligned} \tag{5.33}$$

where

g acceleration due to gravity

m_c mass of the cart

m mass of the pole

l half-length of pole

u applied force

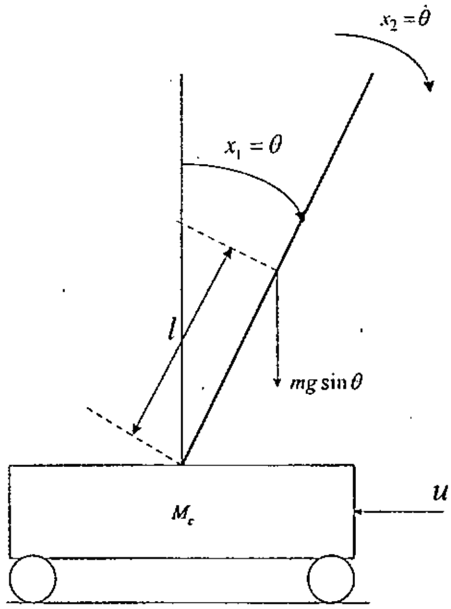


Figure 5.5 The inverted pendulum system.

The control objective is to maintain the system to track the desired angle trajectory, $y_m = \theta_m = \pi/10(\sin(t) + 0.3\sin(3t))$. The system parameters are given as m_c=1kg, m=0.1kg, l=0.5m, g=9.8m/s^2, and d is assumed to be a square wave with amplitude ± 0.5 and the period 2π. Choose the sliding surface as $s = c_1 e + \dot{e}$, $c_1 = 6$. The initial values of parameters θ_p are set by $k_p(0) = 10$ and $k_i(0) = 5$. The membership functions for system state x_i, $i = 1,2$ are selected as:

$$\mu_{NM}(x_i) = \exp[-((x_i + \pi/6)/(\pi/24))^2]$$

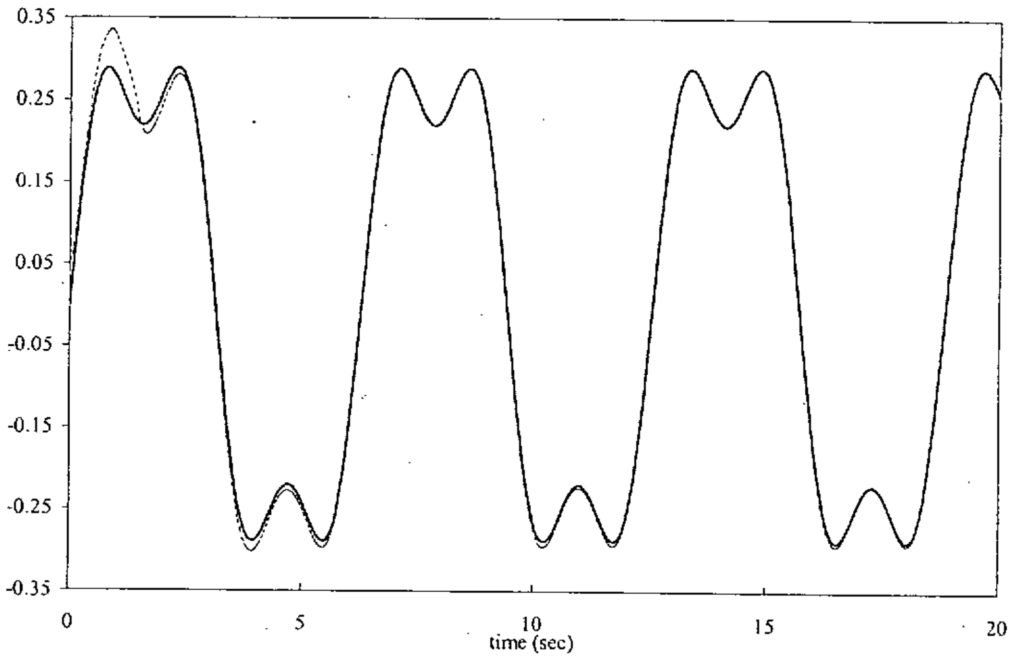
$$\mu_{NS}(x_i) = \exp[-((x_i + \pi/12)/(\pi/24))^2]$$

$$\mu_Z(x_i) = \exp[-(x_i/(\pi/24))^2]$$

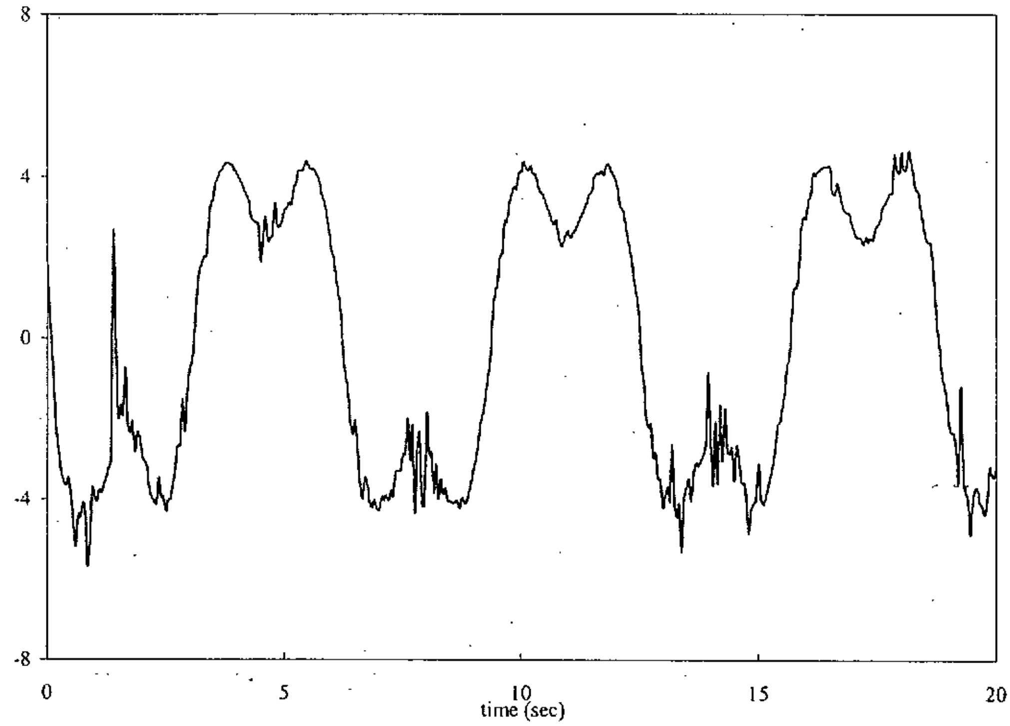
$$\mu_{PS}(x_i) = \exp[-((x_i - \pi/12)/(\pi/24))^2]$$

$$\mu_{PM}(x_i) = \exp[-((x_i - \pi/6)/(\pi/24))^2]$$

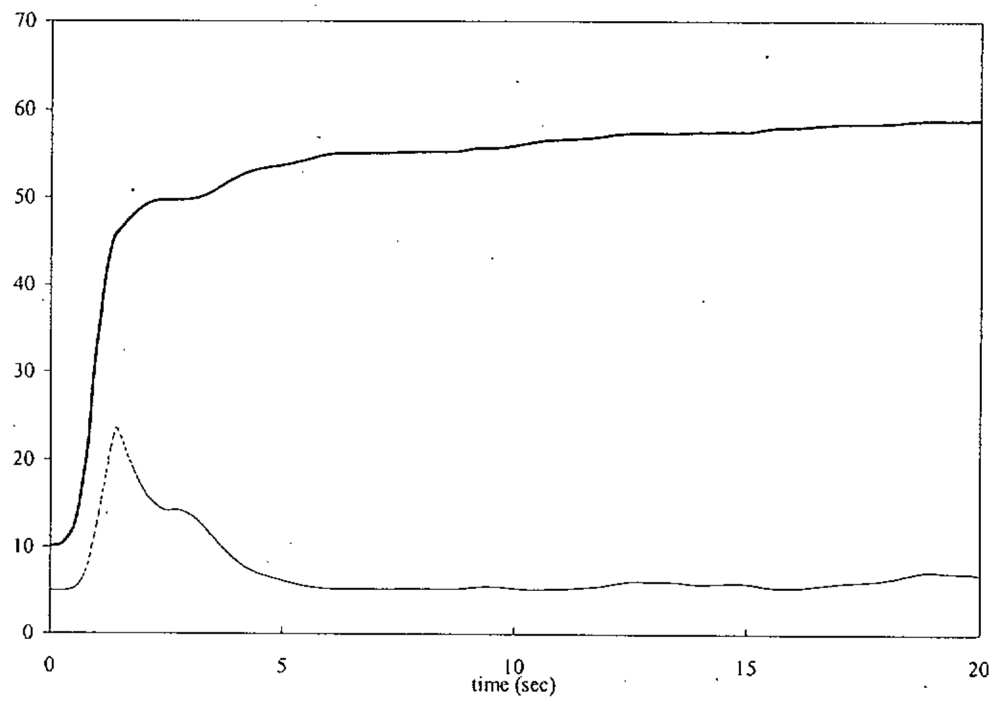
then there are 25 rules to approximate the system functions f and g respectively. And initial consequent parameters of fuzzy rules are chosen randomly in the interval [0.5, 2]. We select $M_f = 20$, $M_g = 20$ and $M_p = 80$. Let the width of the boundary layer $\Phi = 0.2$ and learning rate $\gamma_1 = 60, \gamma_2 = 2$ and $\gamma_3 = 200$. Choose the initial condition $x = [\pi/60, 0]^T$ and step size 0.01s. A Gaussian noise with mean zero and variance of 0.04 was injected at the output of the system. Figure 5.6 shows the simulation results. It can be seen that the tracking performance is good even in the presence of noise and disturbance.



(a)



(b)



(c)

Figure 5.6 Simulation results of adaptive fuzzy sliding mode control applied to the inverted pendulum system: (a) desired output y_m (solid line) and system output x_1 (dash line) (b) control signal (c) trajectory of gains K_p (solid line) and K_i (dash line)

5.5 Conclusions

In this chapter, we introduced the fuzzy sliding mode control and proposed the robust control using the adaptive control strategy. Moreover, based on the Lyapunov synthesis approach, the PI control parameters can be tuned on-line by the adaptive law. The drawback of chattering in sliding mode control is avoided and zero steady tracking error can be ensured. As compared with the single controller approach, the combined adaptive control approach by employing the adaptive fuzzy system and modern control techniques (SMC, PI) gives better performance for a class of nonlinear systems. The closed loop system is stable in the sense of Lyapunov. Finally, the proposed method has been applied to control three simulated nonlinear systems to track a reference trajectory. The simulation results show that the adaptive controller achieves desired performance.

In the next chapters, we will consider the output feedback case for nonlinear systems by using other modern control technique i.e. H^{∞} with adaptive fuzzy system.

CHAPTER SIX

STATE OBSERVER BASED ROBUST INDIRECT ADAPTIVE FUZZY TRACKING CONTROL

6.1 Introduction

Based on the universal approximation theorem and by incorporating fuzzy system into adaptive control scheme, a new fuzzy sliding mode control was presented in Chapter 5. The adaptive fuzzy control approach demonstrated a good performance under the assumption that the state variables of the system were known, or available for feedback. However, sometimes the state variables cannot be fully measured. Moreover, in the previous chapter, the fuzzy approximation error and external disturbance are assumed to be bounded. But in some cases, the external disturbance may be of finite-energy, but not bounded.

In this chapter, a robust adaptive fuzzy controller is designed under the constraint that only the output of the plant is available for measurement. According to this constraint, an adaptive observer-based fuzzy control design, variable structure control (VSS) and H^{∞} disturbance attenuation theory are combined together to construct a hybrid indirect adaptive observer-based robust tracking control scheme.

The state-observer is introduced to resolve the problem of the unavailable state variables. Robustness of the closed-loop system is guaranteed by the incorporate of variable structure control (VSC) and H^{∞} control techniques. In particular, the effect on the tracking error due to fuzzy approximation error is eliminated by the use of variable structure control (VSC). The

This chapter is based on the papers 2, 12 and 15 on chapter one, pages 11 and 12

H^{∞} controller, computed form a Riccati type equation, is employed to attenuate efficiently the effect on the tracking error due to the external disturbance.

The rest of this chapter is organized as follow: The observer-based control problem of the nonlinear SISO systems is described in Section 6.2. The design and stability analysis for the robust adaptive fuzzy control included in Section 6.3. The simulation examples for the proposed control algorithm are included in Section 6.4. Finally, the chapter is concluded in Section 6.5.

6.2 Description of Adaptive Control Problem

Consider the n-th order nonlinear SISO systems in the following form:

$$\begin{aligned} x^{(n)} &= f(x) + g(x)u + d(t) \\ y &= x \end{aligned} \tag{6.1}$$

where $f(x)$ and $g(x)$ are unknown but bounded nonlinear functions, $u \in R$ and $y \in R$ are the input and the output of the system, respectively, and d is the unknown external disturbance.

In particular, we consider the nonlinear systems in the following form:

$$\begin{aligned} \dot{x} &= Ax + B[f(x) + g(x)u + d] \\ y &= C^T x \end{aligned} \tag{6.2}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{n\times n}, B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{n\times 1}, C = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}_{n\times 1}^{T} \tag{6.3}$$

$x = [x, \dot{x}, \cdots, x^{(n-1)}]^T = [x_1, x_2, \cdots, x_n]^T \in R^n$ is the state vector where not all variables are assumed to be available for measurement. Only the system output y is assumed to be measurable. As system (6.1) is required to be controllable, the nonzero condition of input gain $g(x) \neq 0$ is necessary. Without loss of generality, it is assume that $0 < g_L(x) \leq g(x) < \infty$. The control problem is to force the system output y to follow a given bounded reference signal y_m. First, define the reference signal vector y_m, the tracking error vector e and the estimation error vector $\hat{e}$

$$y_m = [y_m, \dot{y}_m, \cdots, y_m^{(n-1)}]^T \in R^n$$

$$e = y_m - x = [e, \dot{e}, \cdots, e^{(n-1)}]^T \in R^n \tag{6.4}$$

$$\hat{e} = y_m - \hat{x} = [\hat{e}, \dot{\hat{e}}, \cdots, \hat{e}^{(n-1)}]^T \in R^n$$

where $\hat{x}$ and $\hat{e}$ denote the estimates of x and e, respectively. If the functions $f(x)$ and $g(x)$ are know and the system is free of external disturbance d, the control law can be designed [Wang 1997] as follows.

$$u^* = \frac{1}{g(x)}[-f(x) + y_m^{(n)} + k_c^T e] \tag{6.5}$$

where $k_c = [k_1^c, k_2^c, \cdots, k_n^c]^T \in R^n$, substituting (6.5) into (6.1) we have $e(t) \to 0$ as $t \to \infty$. Since in general case $f(x)$ and $g(x)$ are unknown functions and not all system states x are measurable, we proposed to estimate $f(x)$ and $g(x)$ by a fuzzy logic system same as Section 5.3 and to design an observer for estimating the state vector x in this chapter. Replacing the functions $f(x)$, $g(x)$ and error vector e in (6.5) by their estimates $\hat{f}(\hat{x})$, $\hat{g}(\hat{x})$ and $\hat{e}$, the certainty equivalent controller becomes:

$$u_I = \frac{1}{\hat{g}(\hat{x})}[-\hat{f}(\hat{x}) + y_m^{(n)} + k_c^T \hat{e}] \tag{6.6}$$

Applying (6.6) to (6.1) and after some simple manipulations

$$\begin{aligned} \dot{e} &= Ae - Bk_c^T\hat{e} + B[\hat{f}(\hat{x}) - f(x) + (\hat{g}(\hat{x}) - g(x))u_f - d] \\ e_1 &= C^T e \end{aligned} \tag{6.7}$$

where $e_1 = y_m - x_1$ denotes the output tracking error. Thus, we have converted the tracking problem into the problem of designing a state observer for estimating the error vector e in (6.7). Consider the following observer that estimates the error vector e in (6.7)

$$\begin{aligned} \dot{\hat{e}} &= A\hat{e} - Bk_c^T\hat{e} + k_o(e_1 - \hat{e}_1) \\ \hat{e}_1 &= C^T\hat{e} \end{aligned} \tag{6.8}$$

where $k_o = [k_1^o, k_2^o, \cdots, k_n^o]^T \in R^n$ is the observer gain vector. We define the observation errors $\tilde{e} = e - \hat{e}$ and $\tilde{e}_1 = e_1 - \hat{e}_1$, subtracting (6.8) from (6.7), we have

$$\begin{aligned} \dot{\tilde{e}} &= A_o\tilde{e} + B[\hat{f}(x) - f(x) + (\hat{g}(\hat{x}) - g(x))u_f - d] \\ \tilde{e}_1 &= C^T\tilde{e} \end{aligned} \tag{6.9}$$

where

$$A_o = A - k_oC^T = \begin{bmatrix} -k_1^o & 1 & 0 & \cdots & 0 \\ -k_2^o & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -k_{n-1}^o & 0 & 0 & \cdots & 1 \\ -k_n^o & 0 & 0 & \cdots & 0 \end{bmatrix}_{n\times n} \tag{6.10}$$

Since (C, A_o) is observable, the observer gain vector k_o can be chosen such that the characteristic polynomial of A_o is strictly Hurwitz. (i.e. the roots of the closed-loop system are in the left-half s-plane) and there exists a positive definite symmetric $n \times n$ matrix P which satisfies the Lyapunov equation $A_o^T P + PA_o = -Q$, where Q is an arbitrary $n \times n$ positive definite matrix.

6.3 Observer-Based Robust Indirect Adaptive Fuzzy Control

The result in (6.5) is possible only while $f(x)$, $g(x)$ and state vector x are well known and free of external disturbance. However, $f(x)$ and $g(x)$ are unknown and not all system state x are measurable. We replace the $f(x)$ and $g(x)$ by their fuzzy system (2.49) and $\hat{x}$ is estimated by the observer. Due to the presence of approximation error and external disturbance, an equivalence control law u_I cannot ensure the stability of the system. We employ another control term u_S and u_h in order to deal with it:

$$u_S = -\lambda_T \operatorname{sgn}(B^T P\tilde{e}) \tag{6.11}$$

$$u_h = -\frac{1}{r \cdot \hat{g}(\hat{x} \mid \theta_g)} B^T P\tilde{e} \tag{6.12}$$

where $\lambda_T = \lambda_1 + \lambda_2$, $\lambda_1 \geq g_L^{-1}(\sup_{t\geq 0} |\omega_f|)$ and $\lambda_2 \geq [g_L^{-1}(\sup_{t\geq 0} |\omega_g|)] |\hat{g}^{-1} \cdot (-\hat{f} + y_m^{(n)} - k_c^T \hat{e})|$, ω_f and ω_g are the estimation errors of $f(x)$ and $g(x)$ respectively. The parameter $r > 0$ is a robust control gain and the matrix $P = P^T > 0$ is a solution of the following Riccati-like equation for any given positive matrix $Q = Q^T > 0$.

$$A_o^T P + PA_o + Q - \frac{2}{r} PBB^T P + \frac{1}{\rho^2} PBB^T P = 0 \tag{6.13}$$

It is noticed that the Riccati equation (6.13) has a solution if and only if $2\rho^2 > r$ [Chen *et al.* 1996].

Thus the resulting control law is

$$u = u_I + u_s + u_h \tag{6.14}$$

In order to adjust the parameters in the fuzzy logic systems, we have to derive adaptive laws.

Let us define the optimal parameter estimates θ_f^* and θ_g^* as follows:

$$\theta_f^* = \arg\min_{\theta_f \in \Omega_f} \left[\sup_{x\in\Omega_x, \hat{x}\in\Omega_{\hat{x}}} |\hat{f}(\hat{x} | \theta_f) - f(x)| \right] \tag{6.15}$$

$$\theta_g^* = \arg\min_{\theta_g \in \Omega_g} \left[\sup_{x\in\Omega_x, \hat{x}\in\Omega_{\hat{x}}} |\hat{g}(\hat{x} | \theta_g) - g(x)| \right] \tag{6.16}$$

where $\Omega_f, \Omega_g, \Omega_x$ and $\Omega_{\hat{x}}$ denote the compact sets of suitable bounds on θ_f, θ_g, x and $\hat{x}$ respectively. We assume that θ_f, θ_g, x and $\hat{x}$ never reach the boundary $\Omega_f, \Omega_g, \Omega_x$ and $\Omega_{\hat{x}}$. We can define the minimum approximation error as:

$$\omega = \omega_f + \omega_f u_I \tag{6.17}$$

where

$$\omega_f = \hat{f}(\hat{x} | \theta_f^*) - f(x) \tag{6.18}$$

$$\omega_g = \hat{g}(\hat{x} | \theta_g^*) - g(x) \tag{6.19}$$

and the parameter adaptation laws are chosen as

$$\dot{\theta}_f = -\gamma_1 \xi_f(\hat{x}) B^T P \widetilde{e} \tag{6.20}$$

$$\dot{\theta}_g = -\gamma_2 \xi_g(\hat{x}) B^T P \widetilde{e} u_I \tag{6.21}$$

Theorem 6.1: Consider the control problem of nonlinear systems (6.2) with control law (6.14), $\hat{f}(\hat{x} | \theta_f)$ and $\hat{g}(\hat{x} | \theta_g)$ are given by (2.49), and the parameter vector θ_f and θ_g are adjusted by adaptation law (6.20) and (6.21). The adaptive control scheme guarantee the following properties:

(i) $\widetilde{e}$, x, $\hat{x} \in L_\infty$, $\lim_{t\to\infty} \widetilde{e} = 0$.

(ii) The following H^∞ tracking performance is achieved [Francis 1997, Chen *et al.* 1996].

$$\int_0^T \widetilde{e}^T Q \widetilde{e} dt \le \widetilde{e}^T(0) P \widetilde{e}(0) + \frac{1}{r} \widetilde{\theta}^T(0) \widetilde{\theta}(0) + \rho^2 \int_0^T d^2(t) dt \qquad \forall T \in [0, \infty) \tag{6.22}$$

The proof is given as follows:

Proof: Taking into account the minimum approximation errors (6.17) and final control law (6.14), the error dynamics (6.9) can be rewritten as:

$$\dot{\tilde{e}} = A_o\tilde{e} + B[\xi^T(\hat{x})\tilde{\theta}_f + \xi^T(\hat{x})\tilde{\theta}_g u_I + \omega + \hat{g}(\hat{x})u_h + \hat{g}(\hat{x})u_s - d] \tag{6.23}$$

where $\tilde{\theta}_f = \theta_f - \theta_f^*$ and $\tilde{\theta}_g = \theta_g - \theta_g^*$.

Choose the Lyapunov function candidate

$$V = \frac{1}{2}\tilde{e}^T P\tilde{e} + \frac{1}{2\gamma_1}\tilde{\theta}_f^T\tilde{\theta}_f + \frac{1}{2\gamma_2}\tilde{\theta}_g^T\tilde{\theta}_g \tag{6.24}$$

where γ_1, γ_2 are positive constant and P is a positive matrix satisfying the Riccati equation (6.13).

The time derivative of V along the error trajectory (6.23) is

$$\dot{V} = \frac{1}{2}\dot{\tilde{e}}^T P\tilde{e} + \frac{1}{2}\tilde{e}^T P\dot{\tilde{e}} + \frac{1}{\gamma_1}\tilde{\theta}_f^T\dot{\tilde{\theta}}_f + \frac{1}{\gamma_2}\tilde{\theta}_g^T\dot{\tilde{\theta}}_g$$

$$= \frac{1}{2}\Big[\tilde{e}^T A_o^T P\tilde{e} + \theta_f^T\xi(\hat{x})B^T P\tilde{e} + u_I^T\theta_g^T\xi(\hat{x})B^T P\tilde{e} + \omega^T B^T P\tilde{e} - \frac{1}{r}\tilde{e}^T PBB^T P\tilde{e}$$
$$u_s^T\hat{g}(\hat{x})B^T P\tilde{e} - d^T B^T P\tilde{e}\Big]$$
$$+\frac{1}{2}\Big[\tilde{e}^T PA_o\tilde{e} + \tilde{e}^T PB\xi^T(\hat{x})\theta_f + \tilde{e}^T PB\xi^T(\hat{x})\theta_g u_I + \tilde{e}^T PB\omega - \frac{1}{r}\tilde{e}^T PBB^T P\tilde{e}$$
$$\tilde{e}^T PB\hat{g}(\hat{x})u_s - \tilde{e}^T PBd\Big] + \frac{1}{\gamma_1}\tilde{\theta}_f^T\dot{\tilde{\theta}}_f + \frac{1}{\gamma_2}\tilde{\theta}_g^T\dot{\tilde{\theta}}_g$$

$$\le \frac{1}{2}\tilde{e}^T\left[PA_o + A_o^T P - \frac{2}{r}PBB^T P\right]\tilde{e} + \frac{1}{2}\left[d^T B^T P\tilde{e} + \tilde{e}^T PBd\right]$$
$$+ (\omega_f + \omega_g u_I)B^T P\tilde{e} + u_s\hat{g}(\hat{x})B^T P\tilde{e} \tag{6.25}$$
$$+ \frac{1}{\gamma_1}\tilde{\theta}_f^T(\gamma_1\xi(\hat{x})B^T P\tilde{e} + \dot{\tilde{\theta}}_f) + \frac{1}{\gamma_2}\tilde{\theta}_g^T(\gamma_2\xi(\hat{x})B^T P\tilde{e}u_I + \dot{\tilde{\theta}}_g)$$

where $\dot{\tilde{\theta}}_f = \dot{\theta}_f$ and $\dot{\tilde{\theta}}_g = \dot{\theta}_g$

From the adaptive laws (6.20) and (6.21), and the definition of u_s in (6.11), we get

$$\begin{aligned}&(\omega_f+\omega_g u_I)B^TP\tilde{e}+u_s\hat{g}(\hat{x})B^TP\tilde{e}\\&\le\omega_fB^TP\tilde{e}-\lambda_1\hat{g}(\hat{x})\,|\,B^TP\tilde{e}\,|+\omega_g u_IB^TP\tilde{e}-\lambda_2\hat{g}(\hat{x})\,|\,B^TP\tilde{e}\,|\le 0\end{aligned}\tag{6.26}$$

Consequently take into account the Riccati-like equation (6.13), the derivative $\dot{V}$ can be bounded as

$$\begin{aligned}\dot{V}&=\frac{1}{2}\tilde{e}^T\left[-Q-\frac{1}{\rho^2}PBB^TP\right]\tilde{e}+\frac{1}{2}\left[-d^TB^TP\tilde{e}-\tilde{e}^TPBd\right]\\&=-\frac{1}{2}\tilde{e}^TQ\tilde{e}-\frac{1}{2}\left[\frac{1}{\rho}B^TP\tilde{e}-\rho d\right]^T\left[\frac{1}{\rho}B^TP\tilde{e}-\rho d\right]+\frac{1}{2}\rho^2d^2\\&\le-\frac{1}{2}\tilde{e}^TQ\tilde{e}+\frac{1}{2}\rho^2d^2\end{aligned}\tag{6.27}$$

If $d(t)\in L_2$, then we have $e,\hat{e},x,\hat{x},u\in L_\infty$ and $\lim_{t\to\infty}\tilde{e}=0$. Integrating the above inequality (6.27) from $t=0$ to $t=T$ yields

$$V(T)-V(0)\le-\frac{1}{2}\int_0^T\tilde{e}^TQ\tilde{e}dt+\frac{1}{2}\rho^2\int_0^Td^2(t)dt\tag{6.28}$$

because $V(T)\ge 0$, from (6.24), the above inequality is equivalent to the following

$$\frac{1}{2}\int_0^T\tilde{e}^TQ\tilde{e}dt\le\frac{1}{2}\tilde{e}^T(0)P\tilde{e}(0)+\frac{1}{2\gamma_1}\tilde{\theta}_f^T(0)\tilde{\theta}_f(0)+\frac{1}{2\gamma_2}\tilde{\theta}_g^T(0)\tilde{\theta}_g(0)+\frac{1}{2}\rho^2\int_0^Td^2(t)dt\tag{6.29}$$

The above inequality is equivalent to the inequality in equation (6.22) which implies the H^∞ tracking performance is achieved.

In general, the adaptive laws in (6.20) and (6.21) are unable to guarantee that $\theta_f\in\Omega_f$ and $\theta_g\in\Omega_g$. In order to guarantee them bounded, the adaptive laws have to be modified by the projection algorithm [Wang 1997]. Define the constraint sets that the parameters concerned belong to

$$\Omega_f=\{\theta_f\in R^n\,\|\,\theta_f\,|\le M_f\}\tag{6.30}$$

$$\Omega_g = \{\theta_g \in R^n \| \theta_g | \le M_g\} \tag{6.31}$$

where M_f and M_g are pre-specified parameters. The modified projection adaptive laws are given as follows

For θ_f, use

$$\dot{\theta}_f = \begin{cases} -\gamma_1 \xi_f(\hat{x}) B^T P \tilde{e} & \text{if } (|\theta_f| < M_f) \\ & \text{or } (|\theta_f| = M_f \text{ and } \tilde{e}^T P B \xi_f^T(\hat{x})\theta_f \ge 0) \\ P_f[-\gamma_1 \xi_f(\hat{x}) B^T P \tilde{e}] & \text{if } (|\theta_f| = M_f) \\ & \text{and } \tilde{e}^T P B \xi_f^T(\hat{x})\theta_f < 0) \end{cases} \tag{6.32}$$

For θ_g, use

whenever an element θ_{gi} of θ_g equals ε, use

$$\dot{\theta}_{gi} = \begin{cases} -\gamma_2 \xi_{gi}(\hat{x}) B^T P \tilde{e} u_I & \text{if } \tilde{e}^T P B \xi_{gi}(\hat{x}) u_I < 0 \\ 0 & \text{if } \tilde{e}^T P B \xi_{gi}(\hat{x}) u_I \ge 0 \end{cases} \tag{6.33}$$

where $\xi_{gi}(\hat{x})$ is the i-th component of $\xi_g(\hat{x})$. Otherwise, use

$$\dot{\theta}_g = \begin{cases} -\gamma_2 \xi_g(\hat{x}) B^T P \tilde{e} u_I & \text{if } (|\theta_g| < M_g) \\ & \text{or } (|\theta_g| = M_g \text{ and } \tilde{e}^T P B \xi_g^T(\hat{x})\theta_g u_I \ge 0) \\ P_g[-\gamma_2 \xi_g(\hat{x}) B^T P \tilde{e} u_I] & \text{if } (|\theta_g| = M_g) \\ & \text{and } \tilde{e}^T P B \xi_g^T(\hat{x})\theta_g u_I < 0) \end{cases} \tag{6.34}$$

Where the projection operator $P_f[*]$ and $P_g[*]$ are defined as

$$P_f[-\gamma_1 \xi_f(\hat{x}) B^T P \tilde{e}] = -\gamma_1 \xi_f(\hat{x}) B^T P \tilde{e} + \gamma_1 \tilde{e}^T P B \frac{\xi_f^T(\hat{x})\theta_f}{|\theta_f|^2} \theta_f \tag{6.35}$$

$$P_g[-\gamma_2 \xi_g(\hat{x}) B^T P \tilde{e} u_I] = -\gamma_2 \xi_g(\hat{x}) B^T P \tilde{e} u_I + \gamma_2 \tilde{e}^T P B \frac{\xi_g^T(\hat{x})\underline{\theta}_g u_I}{|\theta_g|^2} \theta_g \tag{6.36}$$

Remark 6.1: In [Chen *et al.* 1996, Tong *et al.* 2004], a robust H^{∞} control algorithm is tried to attenuate the influence of fuzzy logic approximation error w on the tracking error to a prescribed level by assuming that the approximation error is square-integrable. However, this property is difficult to show for given system and this calculation may require knowledge of system dynamics, which defects the model free approach using fuzzy system. Moreover, it is obvious that the disturbance signal w is influenced by the control input [Kang *et al.* 1998]. In this case, the resulting H^{∞} formulation differs from the standard H^{∞} disturbance attenuation formulation. In contrast to the above work, the effect of the approximation error is compensated by the robust controller u_s. Consequently, the H^{∞} disturbance attenuation performance from the external disturbance to the tracking error can be achieved [van der Schaft 1992], and the attenuation level can be made arbitrarily small.

Remark 6.2: Theorem 6.1 is based on whether there exists a positive-definite matrix P for Riccati-like equation (6.13). If a positive-definite matrix exists, $B^T P\tilde{e}$ can be known and the state estimate $\hat{\underline{e}}$ and $\hat{e}_1$ are available to make the proposed control scheme realizable. If a positive-definite matrix P does not exist for (6.13), then (6.23) can be converted to the strictly positive-real (SPR) [Ioannou and Sun 1996] dynamic system in the same way in [Leu *et al.* 1999].

First, the output error dynamics of (6.27) can be expressed as

$$\tilde{e}_1 = C^T(sI-(A-K_oC^T))B[\tilde{\theta}_f^T\xi(\hat{x})+\theta_g^T\xi(\hat{x})u_I+\omega+\hat{g}(\hat{x})u_h+\hat{g}(\hat{x})u_s-d] \qquad (6.37)$$

where s denotes the Laplace operator.

Let $W(s)$ is a known stable transfer function defined by

$$\frac{1}{W(s)} = C^T(sI-(A-K_o^T))B \tag{6.38}$$

By introducing the state transfer function

$$L(s) = s^{(n-1)} + b_1 s^{(n-2)} + \cdots + b_{n-1} \tag{6.39}$$

into (6.37), the error $\widetilde{e}_1$ can be written as

$$\widetilde{e}_1 = \left[\frac{L(s)}{W(s)}\right]\cdot[\xi_1(\hat{x})\widetilde{\theta}_f^T + \xi_1(\hat{x})\widetilde{\theta}_g^T u_I + \omega_1 + \hat{g}(\hat{x})u_{h1} + \hat{g}(\hat{x})u_{s1} - d_1] \tag{6.40}$$

and $\left[\frac{L(s)}{W(s)}\right]$ is a proper strictly-positive-real (SPR) transfer function with

$$[\xi_1, \omega_1, u_{h1}, u_{s1}, d_1] = \left[\frac{1}{L(s)}\right][\xi, \omega, u_h, u_s, d] \tag{6.41}$$

then the state-space realization of (6.40) can be rewritten as

$$\begin{aligned} \dot{\widetilde{e}} &= A_o\widetilde{e} + B_1[\xi_1(\hat{x})\widetilde{\theta}_f^T + \xi_1(\hat{x})\widetilde{\theta}_g^T u_I + \omega_1 + \hat{g}(\hat{x})u_{h1} + \hat{g}(\hat{x})u_{s1} - d_1] \\ \widetilde{e}_1 &= C^T\widetilde{e} \end{aligned} \tag{6.42}$$

where

$$A_o = [A - K_o C^T] \in R^{n\times n}$$
$$B_1 = [1 \;\; b_1 \cdots b_{n-1}] \in R^n$$
$$C^T = [1 \;\; 0 \cdots 0] \in R^n$$

After this transformation (A_o, B_1, C) is a SPR system. For the given positive matrix $Q = Q^T > 0$, a positive-definite matrix P exists for equation (6.13).

To summarize the above analysis, a step-by-step procedure for the observer-based robust indirect adaptive fuzzy control algorithm is proposed as follow.

Design Procedure:

Step 1. Select the feedback and observer gain vector k_c and k_o, such that the matrices $A - Bk_c^T$ and $A - k_o C^T$ are Hurwitz matrices.

Step 2. Select Q and the desired attenuation level ρ and λ.

Step 3. Solve the Riccati equation (6.13) to obtain a positive definite matrix P

Step 4. Select the parameter values $\gamma_1, \gamma_2, M_f, M_g$ and λ_T.

Step 5. Solve the state observer in (6.8) and obtain the estimated state vector.

Step 6. Select the membership functions $\mu_{F_i^l}(\hat{x}_i)$ and $\mu_{G_i^l}(\hat{x}_i)$ for $i = 1,2,\cdots,n$ and construct the fuzzy systems $\hat{f}(\hat{x} \mid \theta_f) = \theta_f^T \xi(\hat{x})$ and $\hat{g}(\hat{x} \mid \theta_g) = \theta_g^T \xi(\hat{x})$.

Step 7. Obtain the control law (6.14) and apply to the plant, then compute the adaptive law (6.32-6.34) to adjust the parameter vectors θ_f and θ_g.

Figure 6.1 shows the overall scheme of the observer-based indirect adaptive fuzzy logic control proposed in this chapter.

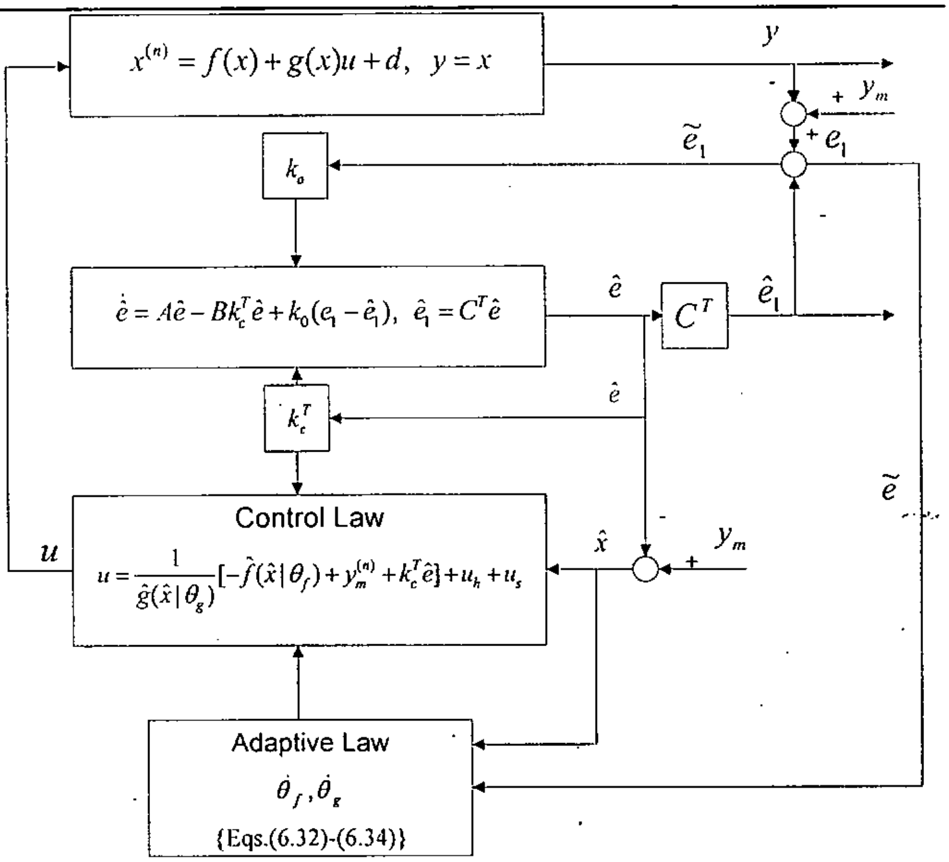


Figure 6.1 observer-based indirect adaptive fuzzy logic control system.

6.4 Simulation Examples

In this section, we illustrate our adaptive controller by the following examples. The first example is letting the Duffing forced-oscillation system track a sin-wave trajectory. The second example is to letting the inverted pendulum system track a sin-wave trajectory.

Example 6.1: **A Duffing forced-oscillation system**

We illustrate the validity of the design approach by an example of tracking control of a Duffing forced-oscillation system.

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -0.1x_2 - x_1^3 + 12\cos(t) + u(t) + d(t) \end{aligned} \tag{6.43}$$

The control objective is to maintain the system to track the desired angle trajectory, $y_m = \sin(t)$, and d is assumed to be a square wave with amplitude $\pm$ 0.5 and the period 2π.

The feedback and observer gain vectors are given as $k_c^T = [30 \;\; 50]$ and $k_o^T = [60 \;\; 140]$. Select the matrix Q in (6.20) as $10I$ and choose the desired attenuation level $\rho = 0.25$, obtain $r = 0.125$, solve the Riccati equation and obtain a positive definite matrix

$$P = \begin{bmatrix} 11.75 & -5 \\ -5 & 2.22 \end{bmatrix}$$

Let the learning rate $\gamma_1 = 20, M_f = 30$ and step size be 0.02sec. The membership functions for system state $\hat{x}_i$ for $i = 1,2$ are chosen as:

$$\mu_{F_i^1}(\hat{x}_i) = 1/(1+\exp(5(\hat{x}_i + 2)))$$

$$\mu_{F_i^2}(\hat{x}_i) = \exp(-(\hat{x}_i + 1.5)^2)$$

$$\mu_{F_i^3}(\hat{x}_i) = \exp(-(\hat{x}_i + 0.5)^2)$$

$$\mu_{F_i^4}(\hat{x}_i) = \exp(-(\hat{x}_i - 0.5)^2)$$

$$\mu_{F_i^5}(\hat{x}_i) = \exp(-(\hat{x}_i - 1.5)^2)$$

$$\mu_{F_i^6}(\hat{x}_i) = 1/(1+\exp(5(\hat{x}_i - 2)))$$

then there are 36 rules to approximate the system functions f. The initial consequent parameters of fuzzy rules are chosen randomly in the interval $[-1,1]$. The initial values of $x(0)$ and $\hat{x}(0)$ are given as $[2 \;\; 2]^T$ and $[-2 \;\; 0]^T$, respectively. Moreover, a Gaussian noise with mean zero and variance of 0.0004 was injected at the output of the system. Figures 6.2-6.4 show the simulation results. It can be seen that the tracking performance can be achieved.

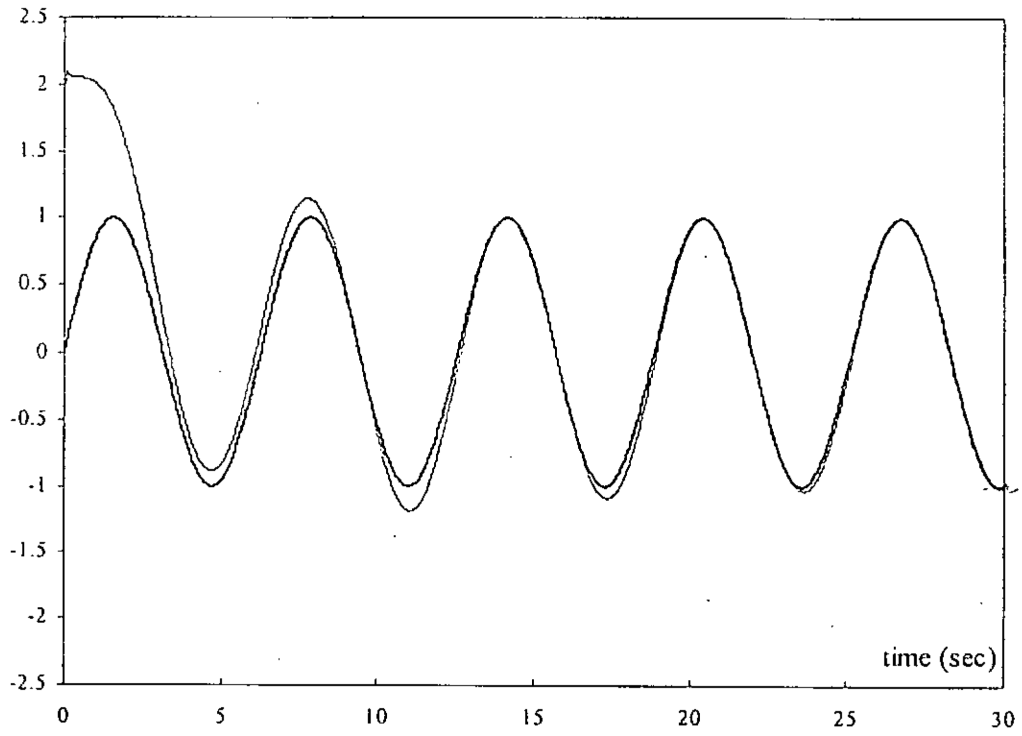


Figure 6.2 Trajectories of the state $x_1(t)$ (dash line) of the tracking control of the desired $y_m(t)$ (solid line) for the Duffing forced-oscillation system.

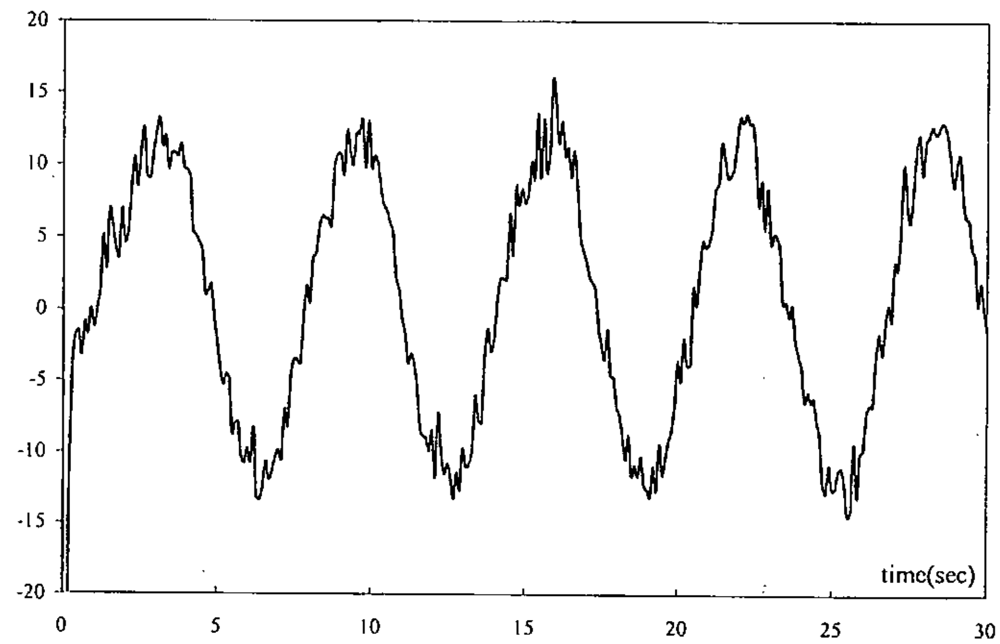


Figure 6.3 Trajectories of the control input $u(t)$ of the tracking control for the Duffing forced-oscillation system.

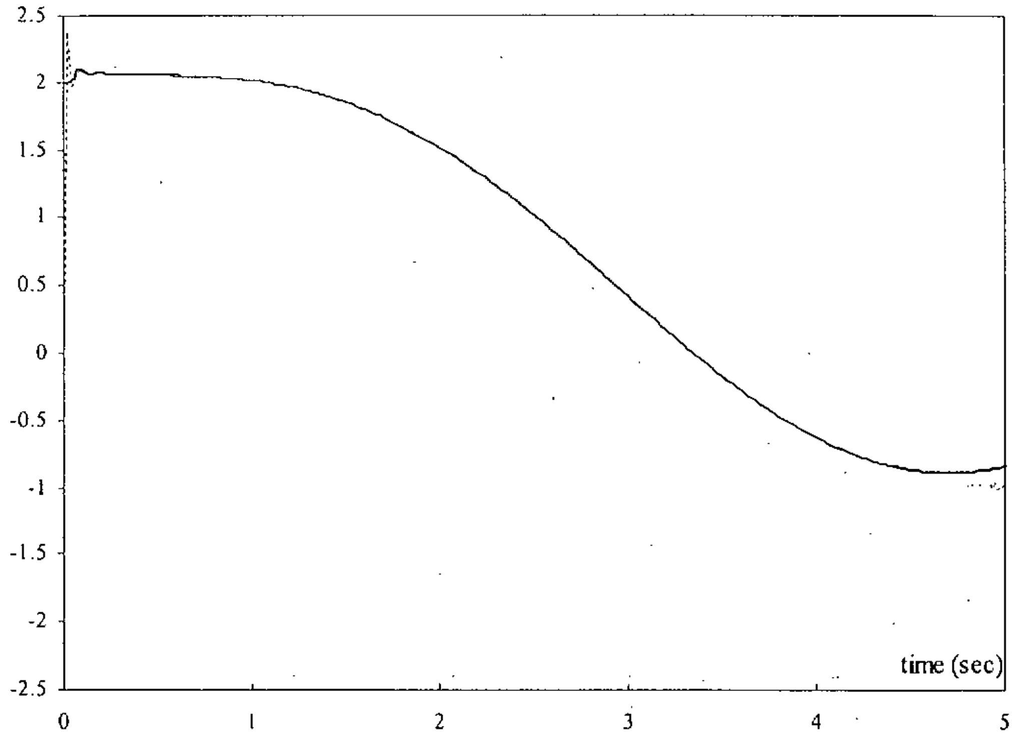


Figure 6.4 Trajectories of the state $x_1(t)$ (solid line) and $\hat{x}_1(t)$ (dash line) of the tracking control for the Duffing forced-oscillation system.

Example 6.2: **An inverted pendulum system**

We test the proposed controller on the tracking control of the benchmark control problem of the inverted pendulum in Figure 5.5. Let $x_1 = \theta$ be the angle of the pendulum with respect to the vertical line and $x_2 = \dot{\theta}$. The dynamic equations of such a system are given by [Wang 1997].

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [f(x_1, x_2) + g(x_1, x_2)u + d] \\ y &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned} \tag{6.44}$$

where

$$f(x_1, x_2) = \frac{g \sin x_1 - mlx_2 \cos x_1 \sin x_1 /(m_c + m)}{l[4/3 - m\cos^2 x_1 /(m_c + m)]} \tag{6.45}$$

$$g(x_1,x_2)=\frac{\cos x_1/(m_c+m)}{l[4/3-m\cos^2 x_1/(m_c+m)]} \tag{6.46}$$

where m_c is the mass of cart, m is the mass of the pole, l is half length of the pole, $g=9.8m/s^2$ is the acceleration of gravity, u is applied force and d is the external disturbance. In the simulation, the system parameters are given as m_c=1kg, m=0.1kg, l=0.5m. The control objective is to maintain the system to track the desired angle trajectory y_m if only the system output y is measurable. For the convenience of the simulation, the reference signal is selected as $y_m=0.1\sin(t)$. The feedback and observer gain vectors are given as $k_c^T=[30\ \ 50]$ and $k_o^T=[4\ \ 10]$. Select the matrix Q in (6.20) as $10I$ and choose the desired attenuation level $\rho=0.15$, obtain $r=0.045$, solve the Riccati equation and obtain a positive definite matrix

$$P=\begin{bmatrix}13.75 & -5\\ -5 & 3.375\end{bmatrix}$$

Let the learning rate $\gamma_1=500$, $\gamma_2=10$, $M_f=20$, $M_g=20$ and step size be 0.01sec. The membership functions for system states $\hat{x}_i$ for $i=1,2$ are chosen as:

$$\mu_{F_i^1}(\hat{x}_i)=\exp[-((\hat{x}_i+\pi/6)/(\pi/24))^2]$$

$$\mu_{F_i^2}(\hat{x}_i)=\exp[-((\hat{x}_i+\pi/12)/(\pi/24))^2]$$

$$\mu_{F_i^3}(\hat{x}_i)=\exp[-(\hat{x}_i/(\pi/24))^2]$$

$$\mu_{F_i^4}(\hat{x}_i)=\exp[-((\hat{x}_i-\pi/12)/(\pi/24))^2]$$

$$\mu_{F_i^5}(\hat{x}_i)=\exp[-((\hat{x}_i-\pi/6)/(\pi/24))^2]$$

then there are 25 rules to approximate the system functions f and g respectively. The initial values of initial consequent parameters of fuzzy rules are chosen randomly in the interval [0.5, 1]. The initial values of $x(0)$ and $\hat{x}(0)$ are given as $[-\pi/60 \;\; 0]^T$ and $[\pi/60 \;\; 0]^T$, respectively. Moreover, an external disturbance d is assumed to be $0.8\sin(2t)e^{-0.1t}$. A Gaussian noise with mean zero and variance of 0.0000125 was injected at the output of the system. Figures 6.5-6.7 are the simulation results, which show that the tracking performance is good even in the presence of disturbance.

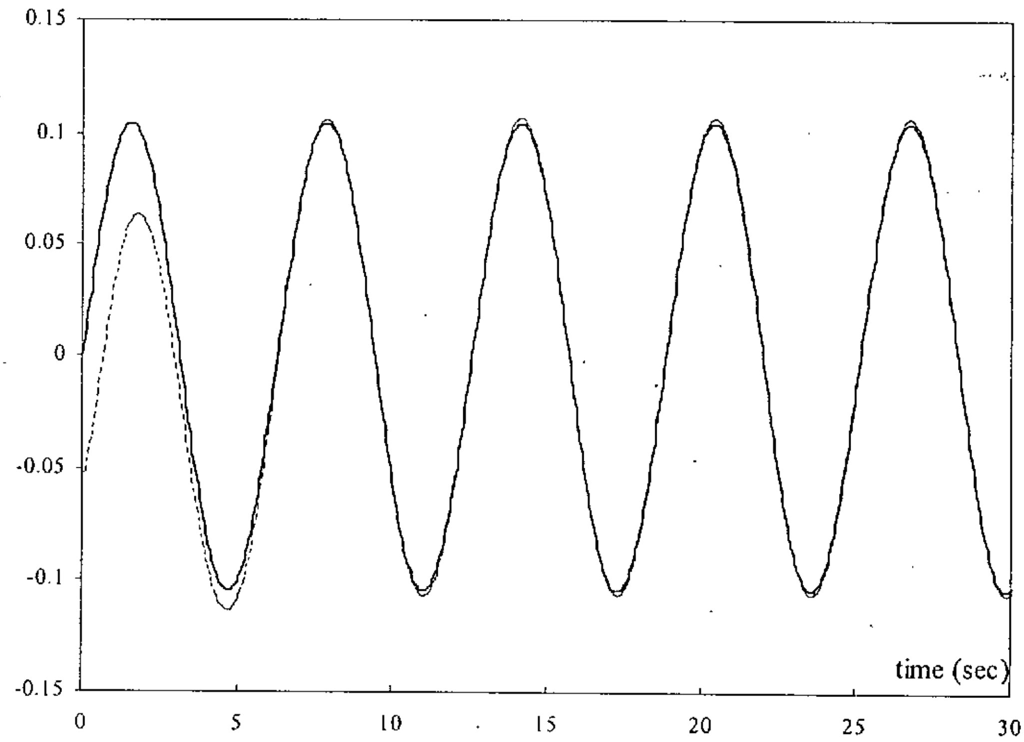


Figure 6.5 Trajectories of the state $x_1(t)$ (dash line) of the tracking control of the desired $y_m(t)$ (solid line) for the inverted pendulum system.

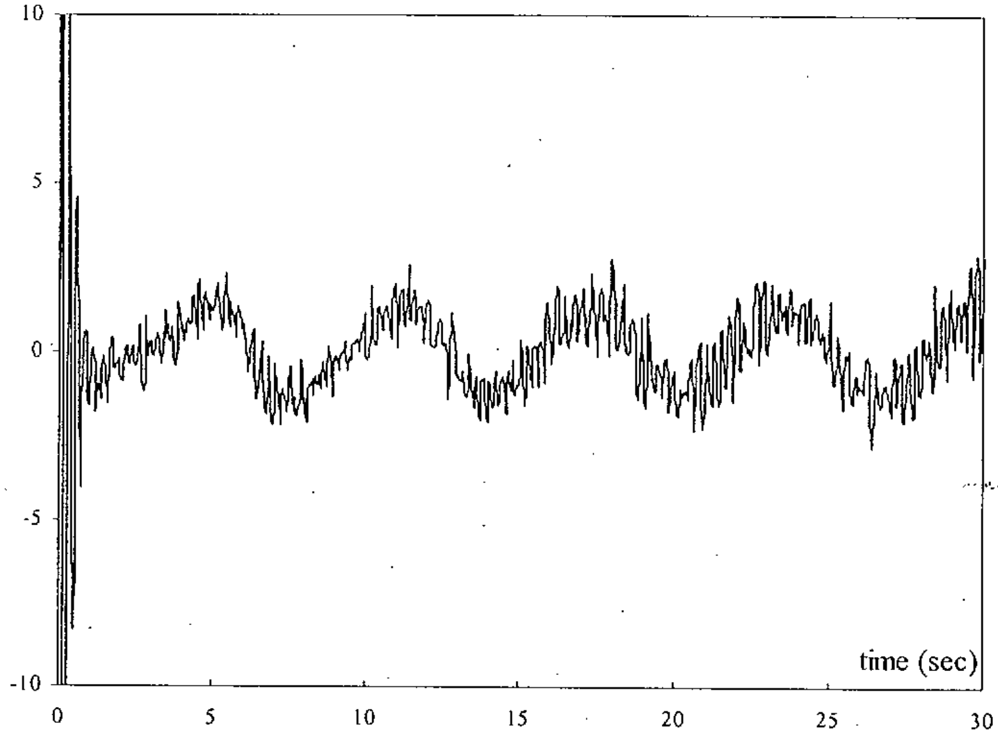


Figure 6.6 Trajectories of the control input $u(t)$ of the tracking control for inverted pendulum system.

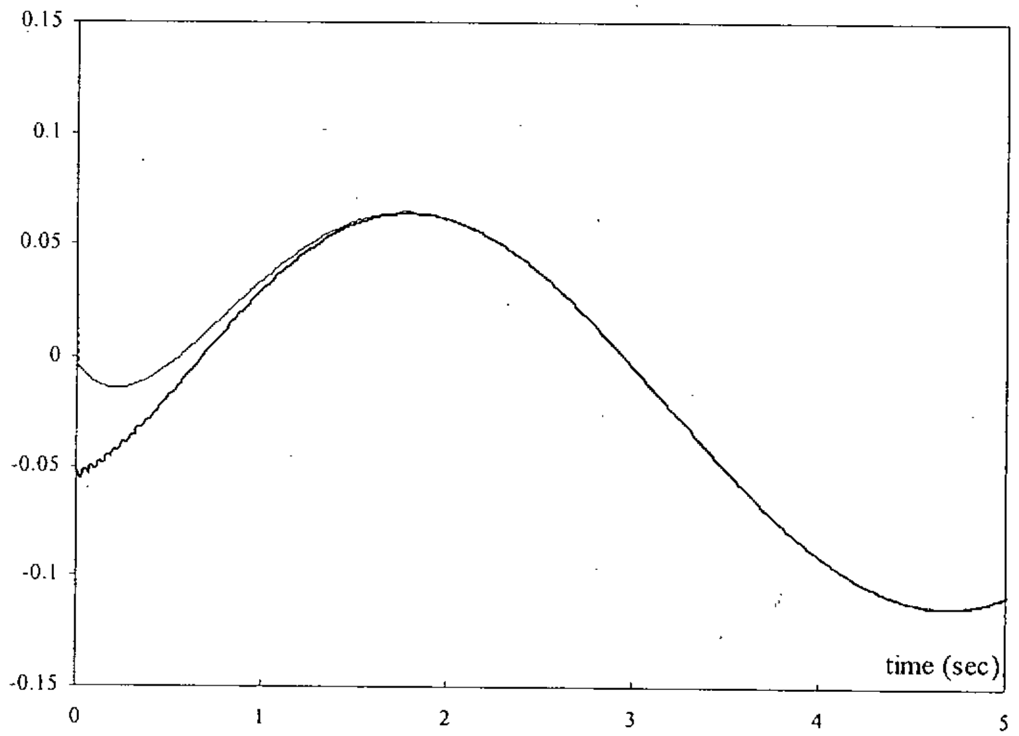


Figure 6.7 Trajectories of the state $x_1(t)$ (solid line) and $\hat{x}_1(t)$ (dash line) of the tracking control for the inverted pendulum system.

6.5 Conclusions

In this chapter, an observer-based adaptive fuzzy control design incorporated with a VSC algorithm and a H^{∞} control algorithm has been proposed for a class of uncertain nonlinear system. First, a state observer is introduced to overcome the problem of state variables unavailability. Then, VSC algorithm is introduced to attenuate efficiently the effect on tracking error due to approximation error. Moreover, an H^{∞} controller which is computed from a Riccati like equation is employed to attenuate the effect of the external disturbances to a prescribed level. The adaptive fuzzy control algorithm stability is guaranteed according to the Lyapunov stability theorem. Application of the proposed method has been applied to control a Duffing forced-oscillation system and an inverted pendulum to track a reference trajectory. The simulation results show that the adaptive controller can achieve the desired performance.

In the next two chapters, we will consider multi-input multi-output (MIMO) nonlinear systems by using adaptive fuzzy system incorporate with modern control techniques.

CHAPTER SEVEN

ADAPTIVE FUZZY CONTOLLER WITH BOUND ESTIMATION FOR ROBOT MANIPULATORS

7.1 Introduction

In Chapter 5, we presented an adaptive fuzzy sliding mode control for a class of SISO nonlinear systems. In this chapter, the adaptive fuzzy sliding mode design method with the bound estimation is extended to a class of MIMO nonlinear system, *i.e.* robot manipulators. The tracking control of robot manipulators is one of the challenging tasks for control engineers, especially when manipulator are required to maneuver very quickly under changing payload and external disturbances.

Many control algorithms such as computed torque method and proportional derivative (PD) with gravity compensation [Spong and Vidyasagar 1989] have been proposed to deal with this robotic control problem. Computed torque control is developed on the basis of the feedback linearization. However, these designs are possible only when the dynamics of the robotic dynamic are well known. The conventional adaptive control schemes can be employed to deal with the unknown robotic dynamics [Slotine and Li 1989, Craig 1989]. In these approaches, the linear parameterizations must be assumed. *i.e.* the unknown parameters must be of linear structure. Moreover, the unknown parameters are assumed to be constant or slowly varying. However, as the robotic dynamic systems are nonlinear, highly coupled, and time varying, the linear parameterization property may not be applicable. Also the implementation requires a precise knowledge of the structure of the dynamic model.

Generally, uncertainties may not be known in practical robotic systems such as changing

This chapter is based on the paper 6 on chapter one, pages 11 payload, nonlinear friction, unknown disturbance, and the high-frequency part of the dynamics. Therefore, it is necessary to consider these effects containing both structured uncertainties (parametric) and unstructured uncertainties (un-modeled dynamics). Variable structure control (VSC) is one of the robust control strategies to compensate these uncertainties in robotic dynamics. In these robust control design approaches [Stepanenlo and Su 1993, Su and Leung 1993], a fixed control law based on a priori bound of uncertainty is designed to compensate the effects of system uncertainties. However, the assumptions in these approaches may be restrictive and difficult to be evaluated. On the other hand, extensive fuzzy control approaches have been developed as a feasible technique to achieve consistent performance in the presence of configuration and uncertainties, owing to the adaptive and nonlinear capabilities of fuzzy system.

In this chapter, a novel control algorithm is developed by combining the fuzzy approach with the sliding mode control method. The proposed method combines the adaptive fuzzy algorithm and robust control technique to guarantee a robust tracking performance for uncertain robotic system. In the proposed algorithm, the adaptive fuzzy systems are used to cancel the nonlinear robot dynamics, which do not need to have a linear parameterized structure as in the case of conventional adaptive control scheme's assumption. Moreover, by combining the integral variable structure control (IVSC) [Bail *et al.* 2000] with uncertainties bound estimation, the proposed control scheme becomes a new robust fuzzy control algorithm for robot manipulators. It is proved that the closed-loop system is globally stable in the Lyapunov sense if all the signals are bounded and the system output can track the desired reference output asymptotically with modeling uncertainties and external disturbances.

This chapter is organized as follows. In Section 7.2, the robot dynamics, its property and control design is described. A robust fuzzy control with bound estimation is developed in Section 7.3. Simulation results for the proposed control algorithm are included in Section 7.4. Finally, the paper is concluded in Section 7.6.

7.2 Robot manipulator dynamics and Control

A robotic manipulator is defined as an open kinematics chain of rigid links. According to the Lagrangian formulation, the dynamic equation of an n-joint robotic manipulator with revolute joints can be formulates as dynamical model [Spong and Vidyasagar 1989].

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{7.1}$$

where

$q,\dot{q},\ddot{q} \in R^n$	vectors of joint position, velocities and accelerations
$M(q) \in R^{n\times n}$	matrix of the moment inertia
$C(q,\dot{q})\dot{q} \in R^{n\times n}$	matrix of the Coriolis and centrifugal
$G(q) \in R^n$	vector of gravitational force
$\tau \in R^n$	vector of applied joint torques

In general, a robotic manipulator is always presented of uncertainties such as frictions and disturbances. Then, (7.1) can be rewritten as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + D = \tau \tag{7.2}$$

where D is the uncertainties of the dynamics, including frictions $F_r(\dot{q})$ and disturbance τ_d. Several fundamental properties of the robot model (7.2) have been obtained as follow:

Property 7.1: The inertia matrix $M(q)$ is a positive definite symmetric matrix, e.g. non-singular and bounded by $m_{\min}\|x\|^2 \le x^T M(q)x \le m_{\max}\|x\|^2$, $\forall x \in R^n$, where m_{min} and m_{max} are minimum and maximum eigenvalues of M.

Property 7.2: $\dot{M}(q) - 2C(q,\dot{q})$ is skew-symmetric matrix, i.e., $x^T(\dot{M} - 2C)x = 0$, $\forall x \in R^n$.

Property 7.3: The unknown disturbance τ_d are assumed to be unknown but bounded. *i.e.* $\|\tau_d\| < \beta_d$.

Property 7.4: The friction in the dynamic equation (7.2) is in the form $F_r(\dot{q}) = F_v\dot{q} + F_c \operatorname{sgn}(\dot{q})$ with F_v the coefficient matrix of viscous friction and F_c a dynamic friction term. The friction is depended on the angular velocity and the bound of the friction terms may be assumed to be in the form of $\|F_v(\dot{q}) + F_c(\dot{q})\| \le \beta_{r_1}\|\dot{q}\| + \beta_{r_2}$, $\beta_{r_1}, \beta_{r_2} > 0$.

In the following analysis, it will be assumed that the nonlinear dynamic model of the robot manipulator to be controlled is well known and uncertainties are negligible. As a consequence, equation (7.1) can be rewritten as

$$\ddot{q} = F(q,\dot{q}) + M^{-1}(q)\tau \tag{7.3}$$

where $F(q,\dot{q})$ is a $n \times 1$ vector defined by

$$F(q,\dot{q}) = -M^{-1}(q)[C(q,\dot{q})\dot{q} + G(q)] \tag{7.4}$$

If $F(q,\dot{q},\ddot{q})$ and $M^{-1}(q)$ are known, we can use the state feedback control law

$$\tau = M(q)[-F(q,\dot{q}) + v'] \tag{7.5}$$

to linearize and decouple the robot dynamic system (7.1), where v' is an external input vector $v' = [v_1', \cdots, v_n']^T$, such that the control law (7.5) apply to system (7.1) results in a closed-loop dynamics with $\ddot{q} = v'$. The objective of control is to follow a given continuously differentiable and uniformly bounded trajectory in the joint space q_d and the tracking error $e = q - q_d$ should be kept as small as possible.

Define a sliding surface in the space of the error state vector $S = R^n$ as

$$S = \begin{bmatrix} s_1(e_1) \\ \vdots \\ s_n(e_n) \end{bmatrix} = \begin{bmatrix} c_1 e_1 + \dot{e}_1 + k_1 \int_0^t e_1 dt \\ \vdots \\ c_n e_n + \dot{e}_n + k_n \int_0^t e_n dt \end{bmatrix} \tag{7.6}$$

where e_i are the tacking error defined by $e_i = q_i - q_{di}$, q_i and q_{di} are the joint and desired output trajectories for each joint. The coefficients c_i and k_i $(i = 1, \cdots, n)$ are positive constants. The tracking problem of the robot manipulator in the joint space implies that the error states should stay on the sliding surface $S = 0$ as the time goes to infinity. A sufficient condition to achieve this behavior is to select the control strategy such that

$$\frac{1}{2}\frac{d}{dt}(s_i) \le -\eta_{\Delta i} \,|\, s_i \,|, \qquad \eta_{\Delta i} \ge 0 \tag{7.7}$$

If the sliding condition equation (7.7) is satisfied, the system is controlled in such a way that the trajectories of the closed-loop system moves towards the sliding surface and hit it. In order to satisfy the sliding reachability condition, the external input vector v' is defined as

$$v'(q, \dot{q}) = \begin{bmatrix} \ddot{q}_{d1} - c_1 \dot{e}_1 - k_1 e_1 - \eta_{\Delta 1} \operatorname{sgn}(s_1) \\ \vdots \\ \ddot{q}_{dn} - c_n \dot{e}_n - k_n e_n - \eta_{\Delta n} \operatorname{sgn}(s_n) \end{bmatrix} \tag{7.8}$$

and $\operatorname{sgn}(\cdot)$ is the usual sign function. Since the control equation (7.5) contains the sign function, direct application of such control signals to the robotic system may result in chattering caused by the signal discontinuity. To overcome this problem, the control law is smooth out within a thin boundary layer Φ by replacing the sign function by a saturation function defined as

$$sat(s_i/\Phi_i)=\begin{cases}1 & \text{if} \quad s_i>\Phi_i\\ s_i & \text{if} \quad -1\le s_i/\Phi_i\le 1\\ -1 & \text{if} \quad s_i<\Phi_i\end{cases} \tag{7.9}$$

From (7.6) and (7.9), it can be noted that the steady state error due to the boundary layer can be removed and there is no reaching phase problem.

As described above, the plant uncertainties are neglected for the controller design. In order to eliminate the influence due to the frictions and disturbance, the positive constant $\eta_{\Delta i}$ are replaced by $\eta_i^* + \eta_{\Delta i}$ to guarantee the existence of sliding condition. η_i^* is the upper bound of uncertainties, i.e. $|D_i| \le \eta_i^*$. Hence, differentiate equation (7.6) with respect to time, the dynamics of the system (7.6) can be rewritten in term of S as follow

$$\dot{S}=\begin{bmatrix}\dot{s}_1\\ \vdots\\ \dot{s}_n\end{bmatrix}=\begin{bmatrix}k_1e_1+c_1\dot{e}_1+\ddot{e}_1\\ \vdots\\ k_ne_n+c_n\dot{e}_n+\ddot{e}_n\end{bmatrix}=F(q,\dot{q})+M^{-1}(q)\tau+D-v(q,\dot{q}) \tag{7.10}$$

where

$$D=M^{-1}(q)(F_v(\dot{q})+\tau_d) \tag{7.11}$$

and

$$v(q,\dot{q})=\begin{bmatrix}k_1e_1+c_1\dot{e}_1-\ddot{q}_{d1}\\ \vdots\\ k_ne_n+c_n\dot{e}_n-\ddot{q}_{dn}\end{bmatrix} \tag{7.12}$$

If the dynamical model of the robot manipulator to be controlled and the bounded of uncertainties are known, then we can use the control law equation (7.4) for the robotic dynamic. However, the dynamics of the robotic dynamic are generally unknown in practice and there are system uncertainties. To solve these problems, the robust fuzzy control algorithm is proposed in Section 7.4.

7.3 Adaptive Fuzzy Control of Robot Manipulator

In section 7.3, the dynamic model of the robot manipulator is assumed to be known, then we can use the control law in (7.5) to linearize and control the robot dynamic system (7.2). However, the robotic model is unknown and the control law is unrealizable. In this section, we propose to use a fuzzy logic system to approximate the unknown system dynamics. Moreover, we employ the integral sliding mode control to compensate both the structured and the unstructured uncertainties. In order to take into account the unknown uncertainties bounds, an adaptive term $\hat{\eta}$ are provided to estimate these parameters online. If the robotic dynamic model is unknown, this implies that the elements of the matrices $F(q,\dot{q})$ and $M(q)$ of (7.2) are also unknown.

$$F(q,\dot{q}) = [f_i(q,\dot{q})] \tag{7.13}$$

$$M(q) = [m_{ij}(q)] \tag{7.14}$$

where $i,j = 1,\cdots,n$. We shall propose the fuzzy logic system to model the unknown function $f_i(q,\dot{q})$ and $m_{ij}(q)$, with fuzzy logic system $\hat{f}_i(q,\dot{q}\,|\theta_i)$ and $\hat{m}_{ij}(q,\dot{q}\,|\theta_{ij})$ for n-link robotic system defined as:

$$\hat{f}_i(q,\dot{q}\,|\,\theta_i) = \theta_{f_i}^T \xi(q,\dot{q}) \tag{7.15}$$

$$\hat{m}_{ij}(q,\dot{q}\,|\,\theta_{ij}) = \theta_{m_{ij}}^T \xi(q) \tag{7.16}$$

Hence, the control law (7.5) can be defined as

$$\tau = \hat{M}(q)[-\hat{F}(q,\dot{q}) + v'] \tag{7.17}$$

where $\hat{M}(q)$ and $\hat{F}(q,\dot{q})$

$$\hat{F}(q,\dot{q}) = \begin{bmatrix} \hat{f}_1(q,\dot{q} \mid \theta_{f_1}) \\ \hat{f}_2(q,\dot{q} \mid \theta_{f_2}) \\ \vdots \\ \hat{f}_n(q,\dot{q} \mid \theta_{f_n}) \end{bmatrix} = \begin{bmatrix} \theta_{f_1}^T \xi(q,\dot{q}) \\ \theta_{f_2}^T \xi(q,\dot{q}) \\ \vdots \\ \theta_{f_n}^T \xi(q,\dot{q}) \end{bmatrix} \tag{7.18}$$

$$\hat{M}(q) = \begin{bmatrix} \hat{m}_1(q \mid \theta_{m_1}) \\ \hat{m}_2(q \mid \theta_{m_2}) \\ \vdots \\ \hat{m}_n(q \mid \theta_{m_n}) \end{bmatrix} = \begin{bmatrix} \theta_{m_1}^T \xi(q) \\ \theta_{m_2}^T \xi(q) \\ \vdots \\ \theta_{m_n}^T \xi(q) \end{bmatrix} \tag{7.19}$$

where

$\hat{m}_i = [\hat{m}_{1i}(q \mid \theta_{m_i}), \cdots, \hat{m}_{ni}(q \mid \theta_{m_i})]^T$ and $\theta_{m_i} = [\theta_{m_{1i}}, \cdots, \theta_{m_{ni}}]^T$

Theorem 7.1.: Consider the control problem of the robotic system equation (7.2). If the control law equation (7.17) is used, the nonlinear functions $f_i(q,\dot{q})$, $m_{ij}(q)$ are estimated by equation (7.15) and equation (7.16), the parameters vector $[\theta_{f_1}, \cdots, \theta_{f_n}]^T$, $[\theta_{m_{ij}}, \cdots, \theta_{m_{ij}}]^T$ and the uncertainties bound estimates $\hat{\eta}_i$ are adjusted by the adaptive law equation (7.20)-(7.22), the closed loop system signals will be bounded and the tracking error will converge to zero asymptotically.

$$\dot{\theta}_{f_i} = \gamma_{f_i} s_i \xi(q,\dot{q}) \tag{7.20}$$

$$\dot{\theta}_{m_{ij}} = \gamma_{m_{ij}} s_i \xi(q) \tau_j \tag{7.21}$$

$$\dot{\hat{\eta}}_i = \gamma_{\eta_i} \mid s_i \mid \tag{7.22}$$

$i, j = 1, \cdots, n$

The proof is given as follows:

Proof: Define the optimal parameters vector $\theta^*_{f_i}, \theta^*_{m_{ij}}$ of fuzzy systems

$$\theta^*_{f_i} = \arg \min_{\theta_{f_i} \in \Omega_{f_i}} \left(\sup_{q,\dot{q} \in R^n} \left| \hat{f}_i(q,\dot{q} \mid \theta_{f_i}) - f_i(q,\dot{q}) \right| \right) \tag{7.23}$$

$$\theta^*_{m_{ij}} = \arg \min_{\theta_{m_{ij}} \in \Omega_{m_{ij}}} \left(\sup_{q \in R^n} \left| \hat{m}_{ij}(q \mid \theta_{m_{ij}}) - m_{ij}(q) \right| \right) \tag{7.24}$$

where Ω_{f_i} , $\Omega_{m_{ij}}$ are constraint sets for θ_{f_i} , $\theta_{m_{ij}}$ defined as

$$\Omega_{f_i} = \{\theta_{f_i} \in R^n \| \theta_{f_i} |\leq M_{f_i}\},\ \Omega_{m_{ij}} = \{\theta_{m_{ij}} \in R^n \| \theta_{m_{ij}} |\leq M_{m_{ij}}\} \tag{7.25}$$

where $M_{f_i}, M_{m_{ij}}$ are pre-specified parameters for estimated parameters bound. Assume the fuzzy parameter vector θ_{f_i} and $\theta_{m_{ij}}$ are never reach the boundaries. Define the minimum approximation error.

$$\omega_i = f_i(q,\dot{q}) - \hat{f}_i(q,\dot{q} \mid \theta^*_{f_i}) + (m_{ij}(q) - \hat{m}_{ij}(q \mid \theta^*_{m_{ij}}))\tau_j \tag{7.26}$$

and assume the approximation errors are upper bounded by $|\,\omega_i\,| < \omega_{i\max}$ .And define $\tilde{\eta}_i = \eta_i^* - \hat{\eta}_i$, where $\eta_i^* = |\,D_i + w_i\,|_{\max}$ is the upper bounded of uncertainties. Then, we have

$$\begin{aligned}
\dot{S} &= M^{-1}(q)\tau + F(q,\dot{q}) + D - v(q,\dot{q}) \\
&= [M^{-1}(q) + \hat{M}^{-1}(q \mid \theta_{m_{ij}}) - \hat{M}^{-1}(q \mid \theta_{m_{ij}})] \times \\
&\quad [\hat{M}(q \mid \theta_{m_{ij}})(-\hat{F}(q,\dot{q} \mid \theta_{f_i}) + v')] + F(q,\dot{q}) + D - v(q,\dot{q}) \\
&= [M^{-1}(q) - \hat{M}^{-1}(q \mid \theta_{m_{ij}})] \cdot \hat{M}(q \mid \theta_{m_{ij}}) \cdot (-\hat{F}(q,\dot{q} \mid \theta_{f_i}) + v') \\
&\quad + F(q,\dot{q}) - v(q,\dot{q}) - \hat{F}(q,\dot{q} \mid \theta_{f_i}) + v' + D \\
&= [M^{-1}(q) - \hat{M}^{-1}(q \mid \theta_{m_{ij}})]\tau + F(q,\dot{q}) - \hat{F}(q,\dot{q} \mid \theta_{f_i}) \\
&\quad + v(q,\dot{q}) - (\hat{p} + \eta)\operatorname{sgn}(S) - v(q,\dot{q}) + D \\
&= [M^{-1}(q) - \hat{M}^{-1}(q \mid \theta_{m_{ij}})]\tau + F(q,\dot{q}) - \hat{F}(q,\dot{q} \mid \theta_{f_i}) - (\hat{p} + \eta)\operatorname{sgn}(S) + D \\
&= [\hat{M}^{-1}(q \mid \theta^*_{m_{ij}}) - \hat{M}^{-1}(q \mid \theta_{m_{ij}})]\tau + F(q,\dot{q} \mid \theta^*_{f_i}) - \hat{F}(q,\dot{q} \mid \theta_{f_i}) \\
&\quad - (\hat{p} + \eta)\operatorname{sgn}(S) + D + \omega \\
&= \tilde{\theta}^T_{f_i}\xi(q,\dot{q}) + \tilde{\theta}^T_{m_{ij}}\xi(q)\tau - (\hat{p} + \eta)\operatorname{sgn}(S) + D + \omega
\end{aligned} \tag{7.27}$$

where $\tilde{\theta}_{f_i} = \theta^*_{f_i} - \theta_{f_i}, \theta_{m_{ij}} = \theta^*_{m_{ij}} - \theta_{m_{ij}}$, $\hat{p} = [\hat{\eta}_1\ \hat{\eta}_2 \cdots \hat{\eta}_n]^T, \eta = [\hat{\eta}_{\Delta 1}\ \hat{\eta}_{\Delta 2} \cdots \hat{\eta}_{\Delta n}]^T$

Now consider the Lyapunov candidate

$$V = \sum_{i=1}^{n} V_i \tag{7.28}$$

where

$$V_i = \frac{1}{2}s_i^2 + \frac{1}{2\gamma_{f_i}}\tilde{\theta}_{f_i}^T\tilde{\theta}_{f_i} + \sum_{j=1}^{n}\frac{1}{2\gamma_{m_{ij}}}\tilde{\theta}_{m_{ij}}^T\tilde{\theta}_{m_{ij}} + \frac{1}{2\gamma_{\eta_i}}\tilde{\eta}_i^T\tilde{\eta}_i \tag{7.29}$$

$\gamma_{f_i}, \gamma_{m_{ij}}$ and γ_{η_i} are design positive constants parameters. The time derivative of V along the error trajectory (7.31) is

$$\begin{aligned}
\dot{V}_i &= s_i\dot{s}_i + \frac{1}{\gamma_{f_i}}\tilde{\theta}_{f_i}^T\dot{\tilde{\theta}}_{f_i} + \sum_{j=1}^{n}\frac{1}{\gamma_{m_{ij}}}\tilde{\theta}_{m_{ij}}^T\dot{\tilde{\theta}}_{m_{ij}} + \frac{1}{\gamma_{\eta_i}}\tilde{\eta}_i^T\dot{\tilde{\eta}}_i \\
&= s_i(\tilde{\theta}_{f_i}^T\xi(q,\dot{q}) + \sum_{j=1}^{n}\tilde{\theta}_{m_{ij}}^T\xi(q,\dot{q})u_j - (\hat{\eta}_i + \eta_{\Delta i})\mathrm{sgn}(s_i) + \omega_i + D_i) \\
&\quad + \frac{1}{\gamma_{f_i}}\tilde{\theta}_{f_i}^T\dot{\tilde{\theta}}_{f_i} + \sum_{j=1}^{n}\frac{1}{\gamma_{m_{ij}}}\tilde{\theta}_{m_{ij}}^T\dot{\tilde{\theta}}_{m_{ij}} + \frac{1}{\gamma_{\eta_i}}\tilde{\eta}_i^T\dot{\tilde{\eta}}_i \\
&= s_i\tilde{\theta}_{f_i}^T\xi(q,\dot{q}) + \frac{1}{\gamma_{f_i}}\tilde{\theta}_{f_i}^T\dot{\tilde{\theta}}_{f_i} + \sum_{j=1}^{n}s_i\tilde{\theta}_{m_{ij}}^T\xi(q,\dot{q})u_j + \sum_{j=1}^{n}\frac{1}{\gamma_{m_{ij}}}\tilde{\theta}_{m_{ij}}^T\dot{\tilde{\theta}}_{m_{ij}} \\
&\quad - s_i(\hat{\eta}_i + \eta_{\Delta i})\mathrm{sgn}(s_i) + s_i\omega_i + s_iD_i + \frac{1}{\gamma_{\eta_i}}\tilde{\eta}_i^T\dot{\tilde{\eta}}_i \\
&\le \frac{1}{\gamma_{f_i}}\tilde{\theta}_{f_i}^T(\gamma_{f_i}s_i\xi(q,\dot{q}) + \dot{\tilde{\theta}}_{f_i}) + \sum_{j=1}^{n}\frac{1}{\gamma_{m_{ij}}}\tilde{\theta}_{m_{ij}}^T(\gamma_{m_{ij}}s_i\xi(q,\dot{q})u_j + \dot{\tilde{\theta}}_{m_{ij}}) \\
&\quad + s_i\omega_i + s_iD_i - |s_i|\eta_i^* + |s_i|(\eta_i^* - \eta_i) - |s_i|\eta_{\Delta i} + \frac{1}{\gamma_{\eta_i}}\tilde{\eta}_i^T\dot{\tilde{\eta}}_i \\
&\le \frac{1}{\gamma_{f_i}}\tilde{\theta}_{f_i}^T(\gamma_{f_i}s_i\xi(q,\dot{q}) + \dot{\tilde{\theta}}_{f_i}) + \sum_{j=1}^{n}\frac{1}{\gamma_{m_{ij}}}\tilde{\theta}_{m_{ij}}^T(\gamma_{m_{ij}}s_i\xi(q,\dot{q})u_j + \dot{\tilde{\theta}}_{m_{ij}}) \\
&\quad + \frac{1}{\gamma_{\eta_i}}\tilde{\eta}_i^T(\gamma_{\eta_i}|s_i| + \dot{\tilde{\eta}}_i) - |s_i|\eta_{\Delta i}
\end{aligned} \tag{7.30}$$

where $\dot{\tilde{\theta}}_{f_1} = \dot{\theta}_{f_1}$ and $\dot{\tilde{\theta}}_{m_{ij}} = \dot{\theta}_{m_{ij}}$. Substitute (7.20)-(7.22) into (7.30), then we have

$$\dot{V}_i \le -\eta_{\Delta i} |s_i| < 0 \tag{7.31}$$

To complete the proof and establish asymptotic convergence of the tracking error, we need to prove that $s_i \to 0$ as $t \to \infty$. Integrating both sides of (7.31), we have

$$\int_0^\infty |s_i| \, dt \le \frac{1}{\eta_{\Delta i}}(V(0) - V(\infty)) < \infty \tag{7.32}$$

Then, we have shown that $s_i \in L_1$, from (7.31), we know that $s_i \in L_\infty$, because we have proved that all the variables on the right-hand side of (7.32) are bounded, we have $\dot{s}_i \in L_\infty$. Using the Collary of Barbalet's Lemma [Sastry and Bodson 1989], if $s_i, \dot{s}_i \in L_\infty$ and $s \in L_p$, for some $p \in [1, \infty]$. We have $s_i \to 0$ as $t \to \infty$, thus $e_i \to 0$ as $t \to \infty$.

The above stability result is achieved under the assumption that all the parameter vectors are within the constraint sets or on the boundaries of the constraint set but moving their interior ($|\theta_{f_i}| = M_{f_i}, |\theta_{m_{ij}}| < M_{m_{ij}}$). To guarantee the parameters are bounded. The adaptive laws equation (7.20) and equation (7.21) can be modified by using the projection algorithm [Wang 1997]. The modified adaptive laws are given as follows.

For θ_{f_i}, we use

$$\dot{\theta}_{f_i} = \begin{cases} \gamma_{f_i} s_i \xi(q,\dot{q}) & \text{if } (|\theta_{f_i}| < M_{f_i}) \\ & \text{or } (|\theta_{f_i}| = M_{f_i} \text{ and } s_i \theta_{f_i}^T \xi(q,\dot{q}) \ge 0) \\ P_{f_i}[\gamma_{f_i} s_i \xi(q,\dot{q})] & \text{if } (|\theta_{f_i}| = M_{f_i}) \\ & \text{and } s\theta_{f_i}^T \xi(q,\dot{q}) < 0) \end{cases} \tag{7.33}$$

For $\theta_{m_{ij}}$, we use

$$\dot{\theta}_{m_{ij}} = \begin{cases} \gamma_{m_{ij}} s_i \xi(q) u_j & \text{if } (|\theta_{m_{ij}}| < M_{m_{ij}}) \\ & \text{or } (|\theta_{m_{ij}}| = M_{m_{ij}} \text{ and } s_i \theta_{m_{ij}}^T \xi(q) u_j \ge 0) \\ P_{m_{ij}}[\gamma_{m_{ij}} s_i \xi(q) u_j] & \text{if } (|\theta_{m_{ij}}| = M_{m_{ij}}) \\ & \text{and } s_i \theta_{m_{ij}}^T \xi(q) u_j < 0) \end{cases} \tag{7.34}$$

where the projection operator, $P_{f_i}[*]$ and $P_{m_{ij}}[*]$ are defined as

$$P_{f_i}[\gamma_{f_i} s_i \xi(q,\dot{q})] = \gamma_{f_i} s_i \xi(q,\dot{q}) - \gamma_{f_i} s_i \frac{\theta_{f_i}\theta_{f_i}^T \xi(q,\dot{q})}{|\theta_{f_i}|^2}$$

$$P_{m_{ij}}[\gamma_{m_{ij}} s_i \xi(q) u_j] = \gamma_{m_{ij}} s_i \xi(q) u_j - \gamma_{m_{ij}} s_i \frac{\theta_{m_{ij}}\theta_{m_{ij}}^T \xi(q) u_j}{|\theta_{m_{ij}}|^2}$$

To summarize the above analysis, a step-by-step procedure for the adaptive fuzzy control of uncertain robotic system is outlined as follow

Design Procedure:

Step 1. The design parameters M_{f_i}, $M_{m_{ij}}$ are specified based on practical constrains.

Step 2. Specify the desired coefficients $c_1,\cdots,c_n$, $k_1,\cdots,k_n$ in equation (7.6).

Step 3. Select the learning coefficients γ_{f_i}, $\gamma_{m_{ij}}$ and γ_{n_i}.

Step 4. Define fuzzy sets A_i for linguistic variable $q,\dot{q}$ and the membership functions μ_{A_i} is uniformly cover the universe of discourse.

Step 5. Construct the fuzzy rule bases for the fuzzy system $\hat{f}_i(q,\dot{q}\,|\,\theta_{f_i})$ and $\hat{m}_{ij}(q\,|\,\theta_{m_{ij}})$.

Step 6. Construct the fuzzy systems $\hat{F}(q,\dot{q}) = \theta_{f_i}^T \xi(q,\dot{q})$ and $\hat{M}(q) = \theta_{m_{ij}}^T \xi(q)$ in equation (7.18) and equation (7.19).

Step 7. Construct the control law equation (7.17) with the adaptive law in equation (7.20-722).

Step 8. Obtain the control and apply to the robot dynamic, then compute the adaptive law equation (7.20-7.22) to adjust the parameter vector θ_{f_i}, $\theta_{m_{ij}}$ and the estimate bound $\hat{\eta}_i$.

Then, the overall adaptive fuzzy control scheme is shown in Figure 7.1.

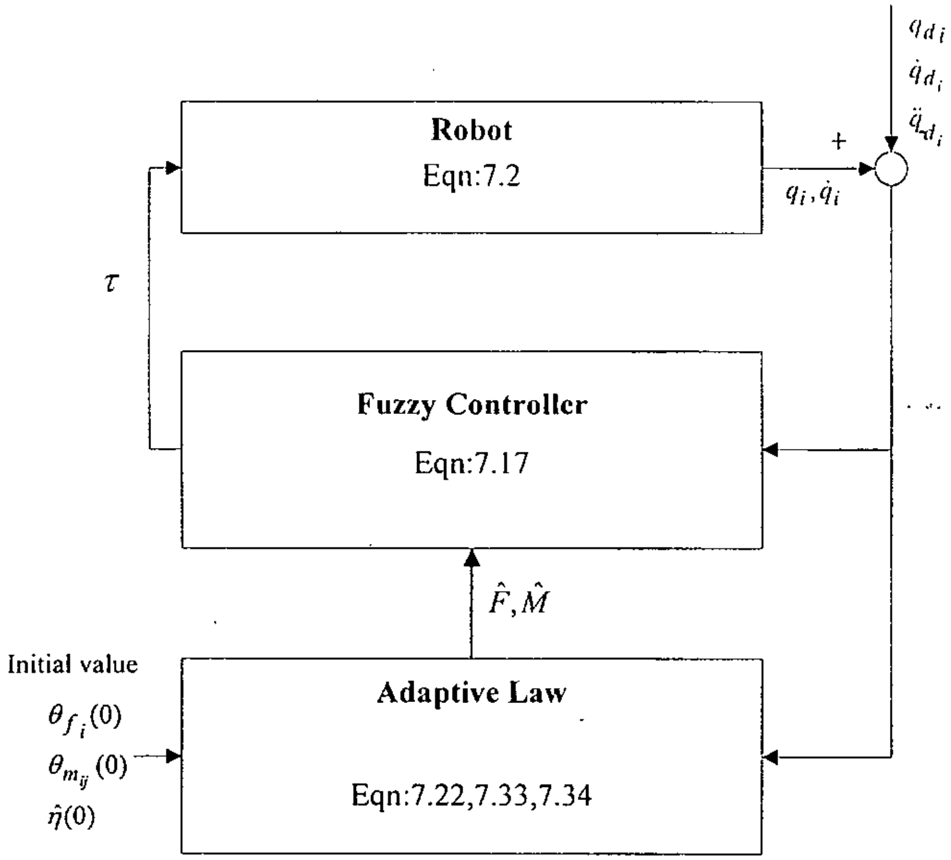


Figure 7.1 Overall Scheme of the Adaptive Fuzzy Control Scheme for Robot Manipulator.

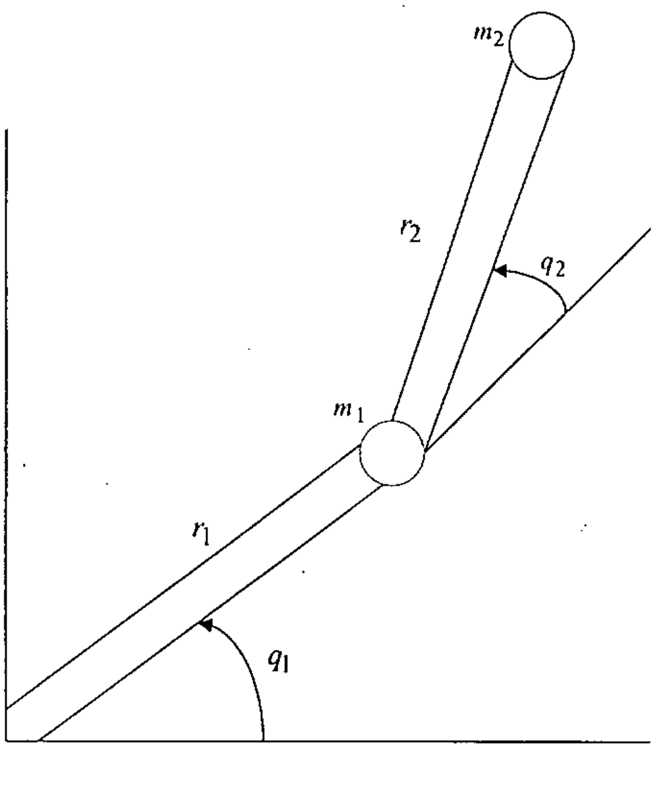


Figure 7.2 Two Degrees of Freedom Robot Manipulator

7.4 Simulation Examples

To verify the theoretical results, simulations were carried out in two degrees of freedom robot manipulator as shown in Figure 7.2 described by [Ge *et al.* 1997].

The dynamics are given as:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F_v(\dot{q}) + F_c(\dot{q}) = \tau \tag{7.35}$$

where

$$M(q) = \begin{bmatrix} m_1 + m_2 + 2m_3\cos q_2 & m_2 + m_3\cos q_2 \\ m_2 + m_3\cos q_2 & m_2 \end{bmatrix}$$

$$C(q,\dot{q}) = \begin{bmatrix} -m_3\dot{q}_2\sin q_2 & -m_3(\dot{q}_1 + \dot{q}_2)\sin q_2 \\ m_3\dot{q}_1\sin q_2 & 0.0 \end{bmatrix}$$

$$G(q) = \begin{bmatrix} m_4\cos q_1 + m_5\cos(q_1 + q_2) \\ m_5\cos(q_1 + q_2) \end{bmatrix}$$

where the parameters m_i are defined by

$$M = P + p_l R$$

$$M = [m_1 \ \ m_2 \ \ m_3 \ \ m_4 \ \ m_5]^T$$

$$P = [p_1 \ \ p_2 \ \ p_3 \ \ p_4 \ \ p_5]^T$$

$$R = [r_1^2 \ \ r_2^2 \ \ r_1 r_2 \ \ r_1 \ \ r_2]^T$$

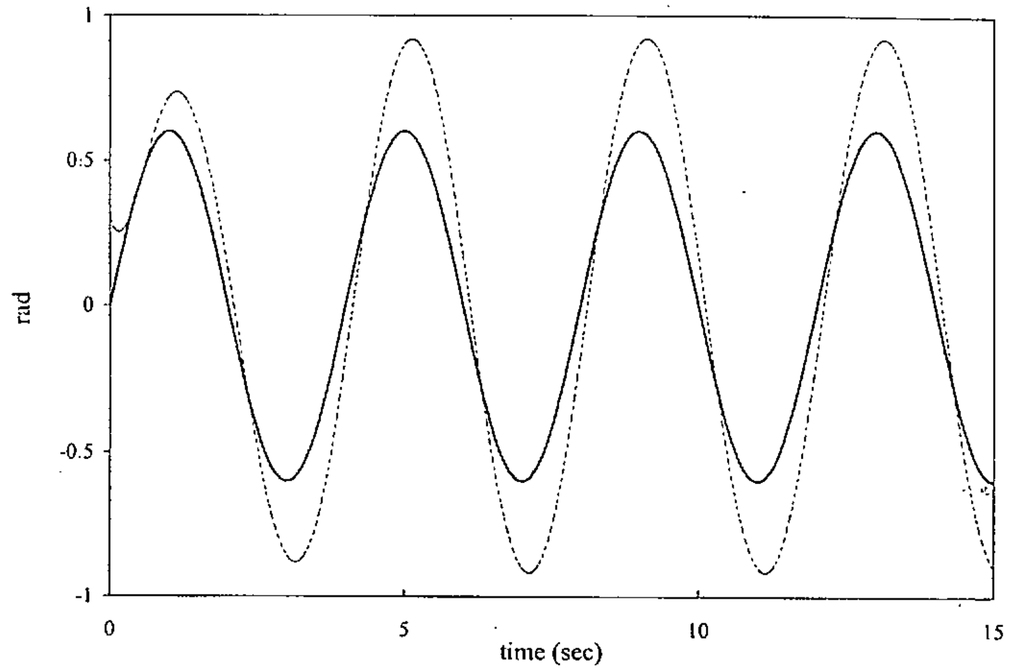
where p_l is the payload, $r_1 = 1m$ and $r_2 = 1m$ are the lengths of link 1 and 2, respectively. And P is the parameter of the robot manipulator. The parameters of the robot used for simulation are

$$P = [1.66 \ \ 0.42 \ \ 0.63 \ \ 3.75 \ \ 1.25]^T \ kg \cdot m^2$$

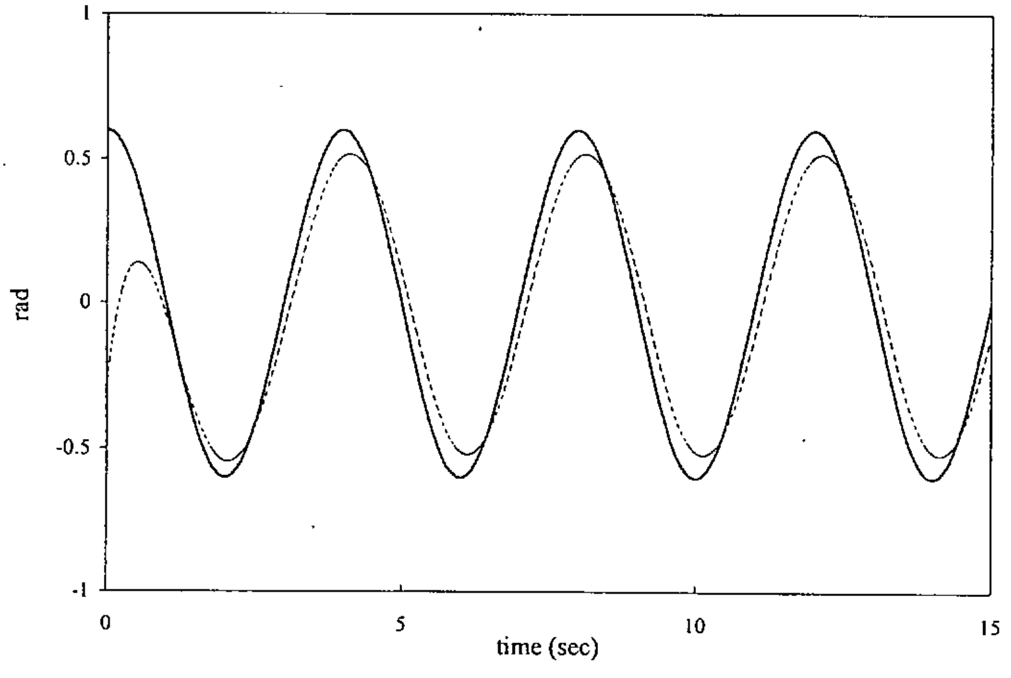
and assume that we have no payload, $p_l = 0.0kg$. $F_v(\dot{q})$ and $F_c(\dot{q})$ are the viscous and coulomb friction have been chosen.

$$F_v = \begin{bmatrix} k_{v_1}\dot{q}_1 \\ k_{v_2}\dot{q}_2 \end{bmatrix}, F_c = \begin{bmatrix} k_{c_1}\operatorname{sgn}(\dot{q}_1) \\ k_{c_2}\operatorname{sgn}(\dot{q}_2) \end{bmatrix}$$

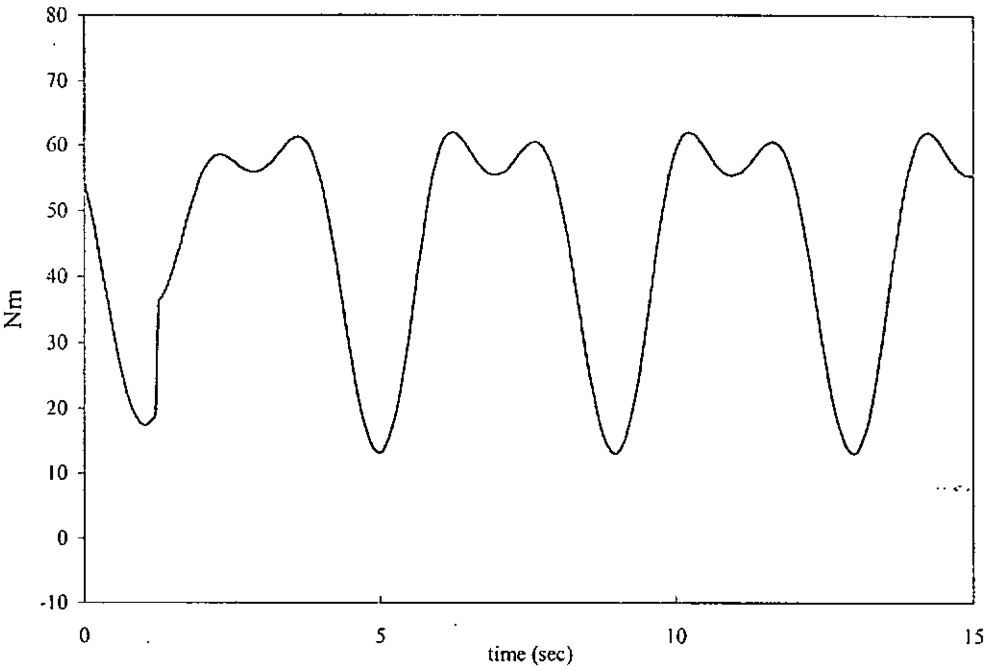
where $k_{v_1} = 0.1$, $k_{v_1} = 0.2$, $k_{c_1} = k_{c_1} = 0.15$. The unknown nonlinearities $f_i(q,\dot{q})$ and $m_{ij}(q)$, $i,j = 1,\cdots,2$, are estimated using three triangular fuzzy sets for q and $\dot{q}$ define in $U = [-3,3]$. No prior knowledge is assumed in this simulation and the consequent parameters are initialized to zero. Select $M_{f_i} = 40$ and $M_{m_{ij}} = 80$. The controller parameters $\gamma_{f_i=50}$, $\gamma_{m_{ij}} = 0.5$ and $\gamma_{p_i} = 0.1$, $i,j = 1,\cdots,2$ .The width of the boundary layer $\Phi_i = 0.1$, $\eta_{\Delta i} = 0.01$, $i = 1,\cdots,2$ and the sliding surface coefficient $c_1 = 8$, $c_1 = 5$, $k_1 = k_2 = 0.5$. The desired reference trajectory are chosen as $q_{d_1} = 0.6\sin(t)$, $q_{d_2} = 0.6\sin(t)$, respectively. The initial conditions $q_1(0) = 0.3$, $q_2(0) = -0.3$, $\dot{q}_1(0) = \dot{q}_2(0) = 0$ and $\hat{\eta}_1(0) = \hat{\eta}_2(0) = 0.2$. In order to verify the robustness of the controller for payload variation, the payload $p_l = 1.0\,kg$ was added at time $t = 5\sec$. For comparison, the conventional computed torque control $\tau = k_p e + k_v \dot{e} + G(q,\dot{q})$ under the same conditions is also demonstrated. The gains are chosen as $k_p = diag[20,20], k_v = diag[50,50]$. Figure 7.3 shows the results with computed torque control. It can be seen the controller cannot drive the joints to reach the desired positions and steady-state tracking error exist. Figure 7.4 shows the results for the proposed fuzzy controller. It is observed the tracking errors go to small values after some transient, which is cause because of the initial choice of the consequent parameters. However, the tracking error decreases quickly since of the on-line learning of fuzzy logic system, and the effect of uncertainties are successfully compensated by the robust control term. The simulation results thus demonstrate the propose robust adaptive fuzzy control can effectively control the rigid robot system with uncertainties



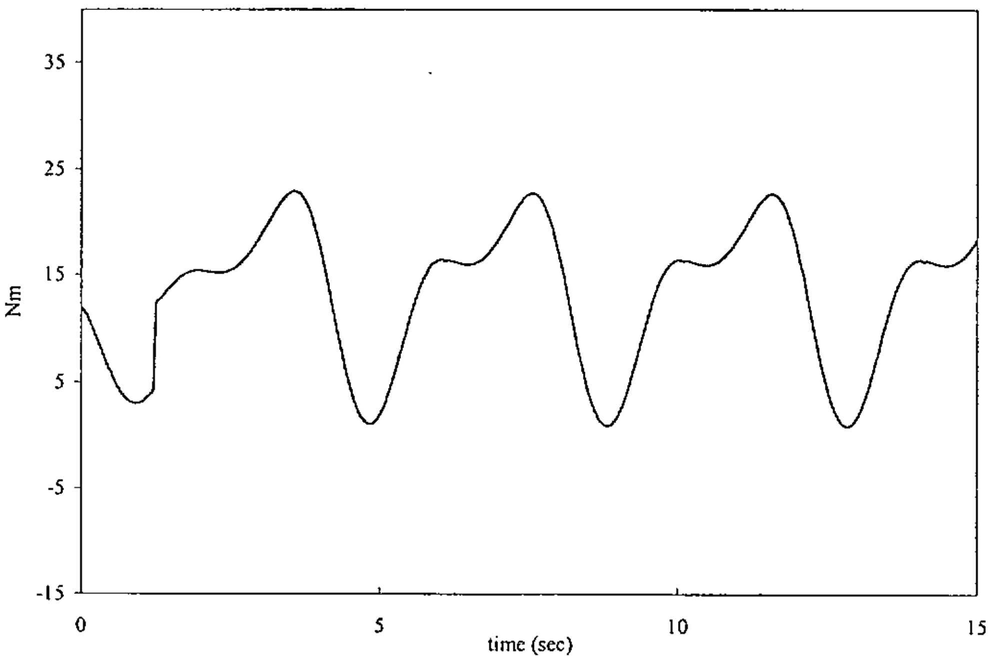
(a)



(b)

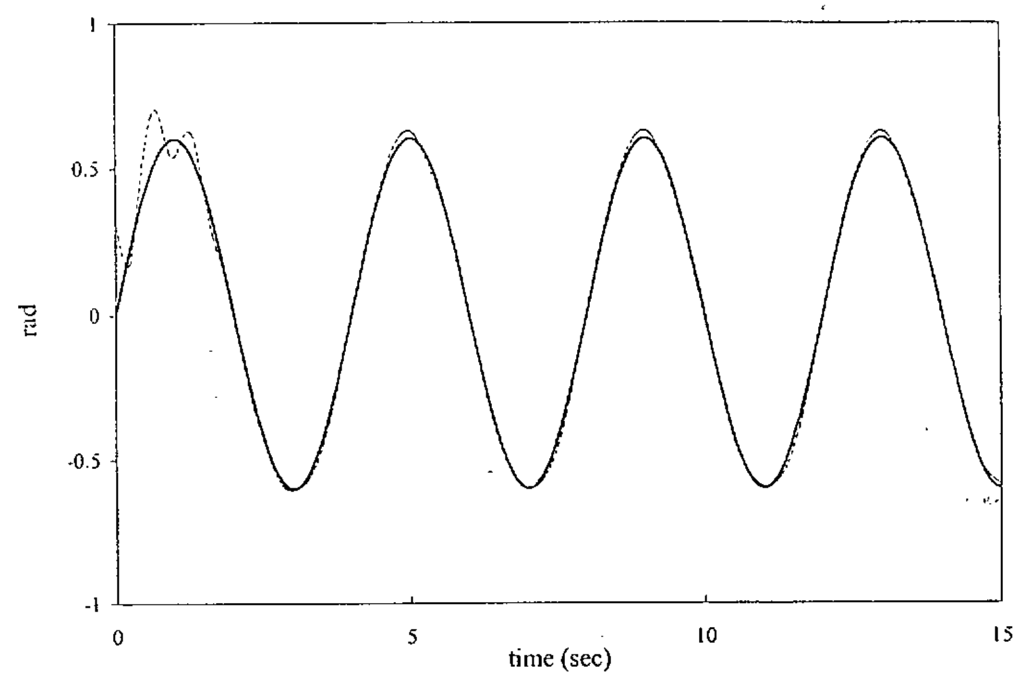


(c)

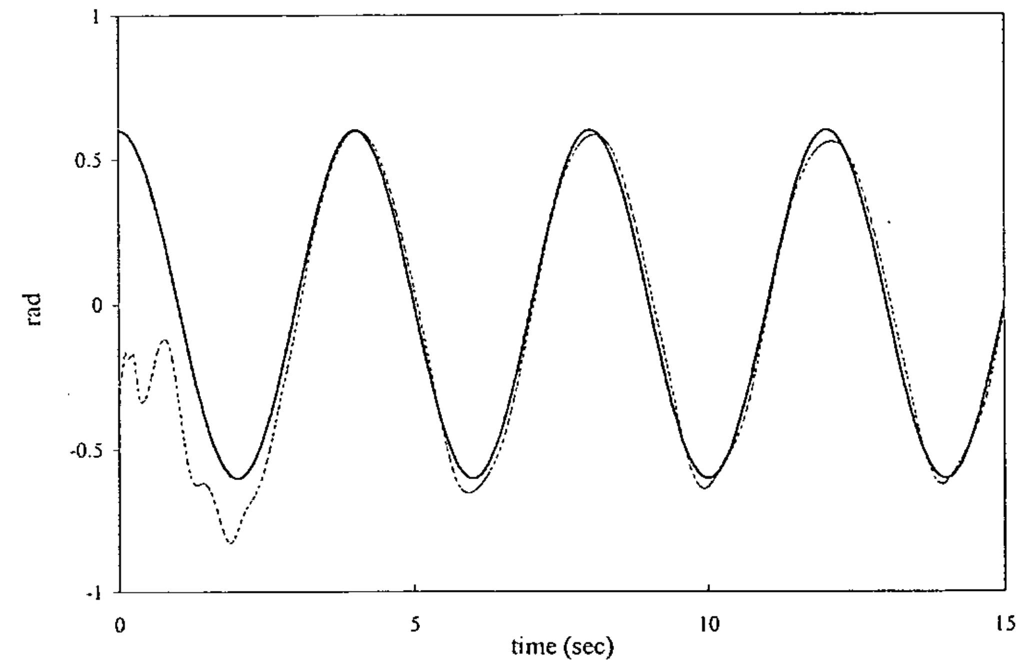


(d)

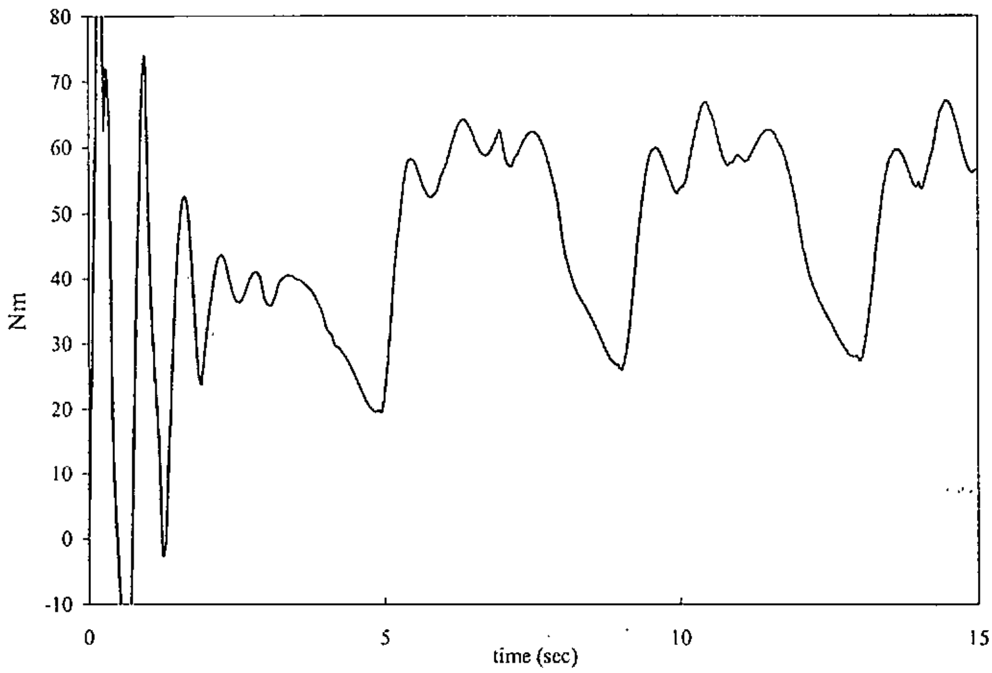
Figure 7.3 Simulation Results of PD-Gravity control (a) desired output q_{d_1} (solid line) and system output q_1 (dash line). (b) desired output q_{d_2} (solid line) and system output q_2. (c) control torque τ_1 .(d) control torque τ_2 .



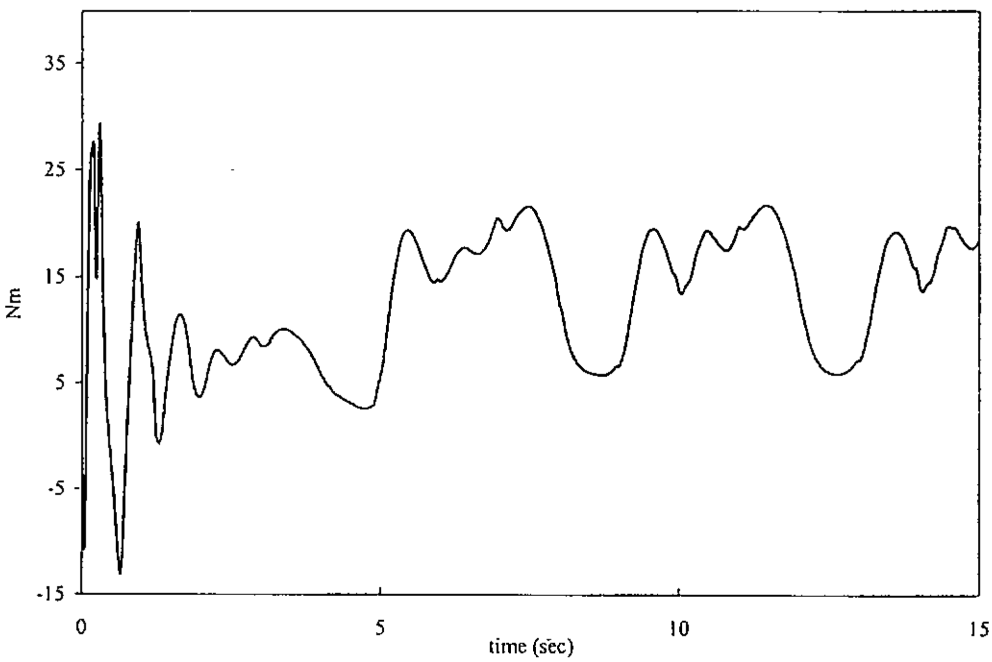
(a)



(b)



(c)



(d)

Figure 7.4 Simulation Results of Adaptive Fuzzy Control (a) desired output q_{d_1} (solid line) and system output q_1 (dash line). (b) desired output q_{d_2} (solid line) and system output q_2 (c) control torque τ_1 .(d) control torque τ_2 .

7.5 Conclusions

In this chapter, we have presented a robust fuzzy control algorithm for robotic manipulators. The method is developed based on the fuzzy modeling technique with integral sliding mode robust control. The control scheme does not require the robot dynamics to be exactly known. Fuzzy logic system has been used to implement an adaptive feedback control strategy with the boundary layer integral sliding control, which compensate for unknown uncertainties with estimated bound. Both chattering and reaching phase problem can be avoided. The design has been proved to guarantee the closed loop stability in the sense of Lyapunov method. Finally, the simulation results show that the proposed control algorithm is appropriate for the practical controller design of robotic manipulator with uncertainties and payload disturbance.

In the next chapters, we will consider the nonlinear MIMO control for two degrees of freedom helicopter by using output feedback fuzzy control scheme.

CHAPTER EIGHT

DIRECT ADAPTIVE FUZZY CONTROL FOR A CLASS OF NONLINEAR MIMO SYSTEM

8.1 Introduction

In this chapter, a direct adaptive fuzzy control scheme is developed for a class of nonlinear multiple-input-multiple-output (MIMO) systems by using Takagi-Sugeno (TS) fuzzy systems. A simple observer is designed to generate an error signal for the adaptive law. The system states of the system are not required to be available for measurement. The overall adaptive scheme guarantees the all the signals involved being uniformly bounded in the Lyapunov sense. Experimental results of a two-degree-freedom helicopter are presented to confirm the usefulness of the proposed control scheme.

The rest of this chapter is organized as follow: The rest of this paper is organized as follows: The observer based control problem of the nonlinear MIMO systems is described in Section 8.2. The direct adaptive fuzzy logic controls derived in Section 8.3. The stability analysis for the proposed adaptive fuzzy control included in Section 8.4. The performance of the proposed algorithm for real time two degree freedom helicopter system is evaluated in Section 8.5. Finally, the chapter is concluded in Section 8.6.

This chapter is based on the papers 7 and 12 on chapter one, pages 11 and 12

8.2 Problem Statement

Consider a MIMO nonlinear system of the form:

$$
\begin{aligned}
y_1^{(n_1)} &= f_1(x) + \sum_{j=1}^{p} g_{1j}(x)u_j \\
y_2^{(n_2)} &= f_2(x) + \sum_{j=1}^{p} g_{2j}(x)u_j \\
&\cdots \\
y_p^{(n_p)} &= f_p(x) + \sum_{j=1}^{p} g_{pj}(x)u_j
\end{aligned}
\tag{8.1}
$$

where $f_i(x)$, $g_{ij}(x)$ $(i, j = 1, \cdots, p)$ are unknown but bounded nonlinear function, $u = [u_1, \cdots, u_p]^T$ is the control input, $y = [y_1, \cdots, y_p]^T$ is the output vector. $x = [x_1, \cdots, x_n]^T$ is the state vector where not all variables are assumed to be available for measurement, only the system output is assumed to be measured. $n = n_1 + \cdots + n_p$. In particular, we consider a MIMO nonlinear system in the following form:

$$
\begin{aligned}
\dot{x} &= Ax + B[F(x) + G(x)u] \\
y &= C^T x
\end{aligned}
\tag{8.2}
$$

where

$F(x) = [f_1(x), \cdots, f_n(x)]^T$ $G(x) = [G_1(x), \cdots, G_p(x)]^T$ with $G_1 = [g_{1i}(x) \cdots g_{pi}(x)]^T$

$A = diag[A_1, \cdots, A_p]$,

$B = diag[B_1, \cdots, B_p]$,

$C^T = diag[C_1, \cdots, C_p]$

and

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{n_i \times n_i}, B_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{n_i \times 1}, C_i = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \end{bmatrix}_{1 \times n_i}$$

The control objective can be stated as follows:

1. Design the fuzzy controller $u_j, j = 1 \cdots p$ with adaptive law such that the states of the plant (1) follow the reference trajectory.
2. All the signal involved in the closed loop are uniformly bounded and the tracking errors tend to zero.

For the given reference signals $y_{1m}, \cdots, y_{pm}$. First, define the tracking error and the estimation errors

$$Y_m = [y_{1m}, \cdots, y_{1m}^{(n_1-1)}, \cdots, y_{pm}, \cdots, y_{pm}^{(n_p-1)}]^T$$

$$X = [x_1, \cdots, x_1^{(n_1-1)}, \cdots, x_p, \cdots, x_p^{(n_p-1)}]^T$$

and

$$\begin{aligned} e &= Y_m - x \\ &= [e_1, \cdots, e_1^{(n_1-1)}, \cdots, e_p, \cdots, e_p^{(n_p-1)}]^T \end{aligned}$$

$$\begin{aligned} \hat{e} &= Y_m - \hat{x} \\ &= [\hat{e}_1, \cdots, \hat{e}_1^{(n_1-1)}, \cdots, \hat{e}_p, \cdots, \hat{e}_p^{(n_p-1)}]^T \end{aligned}$$

where $\hat{x}$ and $\hat{e}$ denote the estimates of x and e, respectively. If $F(x)$ and $G(x)$ are know and the system is free of external disturbance. The control law can be chosen as.

$$U^* = \frac{1}{G(x)}[-F(x) + Y_m^{(n)} + K_c^T e] \tag{8.3}$$

where $K_c = [k_n^c, k_{n-1}^c, \cdots, k_1^c]^T$ is the feedback gain vector to give all roots of the polynomial $A - BK_c^T$ to be Hurwitz. Since in general case $F(x)$ and $G(x)$ are unknown and not all system states variables are available for measurement. We have to design an observer to estimate the system state. In order to achieve the control objective, the direct fuzzy controller is chosen

$$U = U_d + U_s \tag{8.4}$$

where U_d is the TS based fuzzy controller and U_s is additional term to overcome the uncertainties. (described in section 8.3 and 8.4). By replacing the system state x and error e with their estimate $\hat{x}$ and $\hat{e}$, controller (8.3) can be rewritten as

$$U = \frac{1}{G(\hat{x})}[-F(\hat{x}) + Y_m^{(n)} + K_c^T \hat{e}] \tag{8.5}$$

Applying (8.4) and (8.5) to (8.2) and after some simple manipulations, we can obtain the error dynamic equation

$$\begin{aligned} \dot{e} &= Ae - BK_c^T \hat{e} + B[G(U^* - U)] \\ E_1 &= C^T e \end{aligned} \tag{8.6}$$

Design the state observer as follow:

$$\begin{aligned} \dot{\hat{e}} &= A\hat{e} - BK_c^T \hat{e} + K_o(E_1 - \hat{E}_1) \\ \hat{E}_1 &= C^T \hat{e} \end{aligned} \tag{8.7}$$

where $K_o = [k_n^o, k_{n-1}^o, \cdots, k_1^o]^T$ is the observer gain vector to give all roots of polynomial $A - K_o C^T$ to be Hurwitz. We define the observation errors $\tilde{e} = e - \hat{e}$ and $\tilde{E}_1 = E_1 - \hat{E}_1$ subtracting (8.7) from (8.6), we have

$$\begin{aligned} \dot{\tilde{e}} &= A_o \underline{\tilde{e}} + B[G(U^* - U)] \\ \tilde{E}_1 &= C^T \tilde{e} \end{aligned} \tag{8.8}$$

where $A_o = A - K_o C^T$. Since A_o is strictly Hurwitz. There exists a positive definite symmetric matrix P which satisfies the Lyapunov equation $A_o^T P + PA_o = -Q$, where Q is an arbitrary positive definite matrix.

8.3 Direct Adaptive Fuzzy Logic Control

In this section, the TS type direct adaptive fuzzy logic controller to be designed is discussed. The basic configuration includes a fuzzy base and a collection of fuzzy IF-THEN rules as follows:

$$R_k^i : \text{IF} \ \ v^i \ \ \text{is} \ \ V_k^i \ \ \text{Then} \ \ u_{d_i} = a_{k_0}^i + a_{k_1}^i x_1 + \cdots + a_{k_n}^i x_n \tag{8.9}$$

where $k = 1, \cdots, m_i$ denotes the number of the fuzzy IF-THEN rules. v^i is the fuzzy controller input vector and V_k^i are labels of the fuzzy sets. The output value of the ith fuzzy controller is

$$u_{d_i} = \frac{\sum_{k=1}^{m_i} \mu_k^i(v^i)(a_{k_0}^i + \sum_{j=1}^{n} a_{k_j}^i x_j)}{\sum_{k=1}^{m_i} \mu_k^i(v^i)} \tag{8.10}$$

where $\mu_k^i(v^i)$ is the grade of membership function of the linguistic variable v^i. The compact form of equation (8.12) can be rewritten as

$$u_{d_i} = \xi_i \phi_i z_i \tag{8.11}$$

where $z_i = [1 \ \ x]^T$, $\phi_i = \begin{bmatrix} a_{1_0}^i & \cdots & a_{1_n}^i \\ \vdots & \cdots & \vdots \\ a_{m_{i0}}^i & \cdots & a_{m_{in}}^i \end{bmatrix}$

and ξ_i is a regressive vector defined as

$$\xi_i = \frac{1}{\sum_{k=1}^{m_i} \mu_k^i}[\mu_1^i \ \ \mu_2^i \cdots \mu_{m_i}^i] \tag{8.12}$$

We can assume that the ith component of the direct fuzzy logic controller can be described by TS fuzzy system plus a approximation error ω_i. This means that there exists parameter ϕ^* such that

$$u_{d_i}^* = \xi_i \phi_i^* z_i + \omega_i \tag{8.13}$$

since the optimal parameters ϕ_i^* are unknown, the fuzzy logic system approximation of $u_{d_i}^*$ of the estimates ϕ_i is given by

$$u_{d_i} = \xi_i \phi_i z_i \tag{8.14}$$

Define $\widetilde{\phi}_i = \phi_i - \phi_i^*$ as the parameters estimation errors and the parameter adaptation laws are chosen as

$$\dot{\phi}_i = \gamma_{i1} B_i^T P_i \widetilde{e}_i \xi_i^T (\hat{x}) z_i^T \tag{8.15}$$

8.4 Stability Analysis

In order to establish the stability of the proposed fuzzy control system, we need the following assumption.

Assumption 8.1: the approximation errors are bounded by ϖ_i i.e., $\omega_i \le \varpi_i$ where ϖ_i are some known constants.

Assumption 8.2: the input gains are bounded by $0 < \underline{g}_{ii} \le g_{ii} \le \overline{g}_{ii} < \infty$ and their derivatives are bounded by $|\dot{g}_{ii}| \le H_{ii}$, where $\underline{g}_{ii}$ and $\overline{g}_{ii}$ are some known constants and H_{ii} are some known functions.

Theorem 8.1: Consider the control problem of nonlinear systems (8.2) with control law (8.4), u_{di} are given by (8.14), and the parameters ϕ_i are adjusted by adaptation law (8.15). The adaptive control scheme guarantee the feedback system is asymptotically stable and the tracking error converges to zero.

Proof: Consider the Lyapunov function candidate

$$V = \sum_{i=1}^{p} V_i \tag{8.16}$$

with

$$V_i = \frac{1}{2g_{ii}}\tilde{e}_i^T P_i \tilde{e}_i + \frac{1}{2\gamma_{i1}} tr(\tilde{\phi}_i^T \tilde{\phi}_i) \tag{8.17}$$

The time derivative of V_i along the error trajectory of (8.8) is

$$\begin{aligned} \dot{V}_i = & \frac{1}{2g_{ii}}\tilde{e}_i^T P_i \left[A_{io}\tilde{e}_i + B_i[\sum_{j=1}^{p} g_{ij}(u_j - u_j^*)] \right] \\ & + \frac{1}{2g_{ii}} \left[A_{io}\tilde{e}_i + B_i[\sum_{j=1}^{p} g_{ij}(u_j - u_j^*)] \right]^T P_i \tilde{e}_i \quad \ldots \\ & - \frac{\dot{g}_{ii}}{2g_{ii}^2}\tilde{e}_i^T P_i \tilde{e}_i + \frac{1}{\gamma_{i1}} tr(\tilde{\phi}_i^T \dot{\tilde{\phi}}_i) \end{aligned} \tag{8.18}$$

Substitute (8.15) into (8.18), and using the fact that $u_i - u_i^* = u_{s_i} + \xi_i \tilde{\phi}_i z_i - w_i$, we get

$$\begin{aligned} \dot{V}_i = & -\frac{1}{2g_{ii}}\tilde{e}_i^T Q_i \tilde{e}_i - \frac{\dot{g}_{ii}}{2g_{ii}^2}\tilde{e}_i^T P_i \tilde{e}_i \\ & + \frac{\tilde{e}_i^T P_i B_i}{g_{ii}} \cdot \left[-\sum_{j=1}^{p} g_{ij} u_{sj} + \sum_{j=1}^{p} g_{ij} w_j - \sum_{j=1, j\neq i}^{p} g_{ij}\xi_j \tilde{\phi}_j z_j \right]. \end{aligned}$$

$$\begin{aligned} \dot{V}_i \le & -\frac{1}{2g_{ii}}\tilde{e}_i^T Q_i \tilde{e}_i - \frac{\dot{g}_{ii}}{2g_{ii}^2}\tilde{e}_i^T P_i \tilde{e}_i - \tilde{e}_i^T P_i B_i u_{s_i} \\ & + \tilde{e}_i^T P_i B_i \cdot \left[\sum_{j=1}^{p} \frac{\overline{g}_{ij}}{\underline{g}_{ii}} \overline{w}_j + \sum_{j=1, j\neq i}^{p} \frac{\overline{g}_{ij}}{\underline{g}_{ii}} (|u_{sj}| + |\xi_j \tilde{\phi}_j z_j|) \right] \end{aligned} \tag{8.19}$$

Define

$$\eta_i = \sum_{j}^{p} \frac{\overline{g}_{ij}}{\underline{g}_{ii}} \overline{w}_j + \sum_{j=1, j\neq i}^{p} \frac{\overline{g}_{ij}}{\underline{g}_{ii}} \overline{U}_j, \quad \rho_i = \frac{H_{ii}}{2\underline{g}_{ii}^2}\tilde{e}_i^T P_i \tilde{e}_i \tag{8.20}$$

where $|\xi_j \tilde{\phi}_j z_j| \le \overline{U}_j$, and the additional control term u_{si} are chosen as

$$u_{si} = \left(\eta_i + \sum_{j=1, j\neq i}^{p} \frac{\overline{g}_{ij}}{\underline{g}_{ii}} \overline{U}_s \right) \cdot \mathrm{sgn}(\tilde{e}_i^T P_i B_i) + \rho_i \tag{8.21}$$

and $\overline{U}_s$ are chosen as

$$\overline{U}_s \geq \max_{i=1,\cdots,p} \left(\frac{|\eta_i| + |\rho_i|}{1 - \sum_{j=1,j\neq i}^{p} \frac{\overline{g}_{ij}}{\underline{g}_{ii}}} \right) \tag{8.22}$$

such that $|u_{di}| \leq \overline{U}_s$, $i = 1,\cdots,p$.

from equation (8.18), the above inequality is equivalent to the following

$$\dot{V}_i \leq -\frac{1}{2g_{ii}} \widetilde{e}_i^T Q_i \widetilde{e}_i \tag{8.23}$$

It follows immediately from (8.16) and (8.23)

$$\dot{V} \leq -\sum_{i=1}^{p} \frac{1}{2g_{ii}} \widetilde{e}_i^T Q_i \widetilde{e} \tag{8.24}$$

Thus, $\dot{V}$ is always negative if $\widetilde{e}_i \neq 0$, then $V_i \in L_\infty$. Therefore, $\widetilde{e}_i, \widetilde{\phi}_i \in L_\infty$ for $i = 1,\cdots,p$.

Since all variables in the right-hand side of (8.8) are bound, $\dot{\widetilde{e}}_i \in L_\infty$ for $i = 1,\cdots,p$.

Integrating both sides of (8.24) yields

$$\sum_{i=1}^{p} \int_0^\infty \|\widetilde{e}_i\|^2 dt \leq \sum_{i=1}^{p} \frac{2\overline{g}_{ii}}{\lambda_{\min}(Q_i)} V_i(0) \tag{8.25}$$

where $\|\cdot\|$ is the Euclidean norm, $\lambda_{\min}(Q_i)$ is the minimum eigenvalue of Q_i. Since the right hand side of (8.25) is bounded, $\widetilde{e}_i \in L_2$ for $i = 1,\cdots,p$ .Using Barbalat's lemma [Sastry and Bodson 1989], we have that the errors converge asymptotically to zero. $\lim t \to \infty$ $\widetilde{e}_i = 0$ for $i = 1,\cdots,p$.

8.5 Experimental Studies

In this section, a two degrees of freedom helicopter CE150 from Humusoft [Humusoft 1985] was used in this experiment. The model is a multidimensional, nonlinear system with two manipulated input and two measured outputs (horizontal and vertical angles) with significant cross couplings. The system consisted of a massive support, and the main body, carrying two propellers driven by DC motors (R_1, R_2) as shown in Figure 8.1. The model can be described by the nonlinear state equations with four states, two inputs, which are the control value for main and side propeller motors. The two output values are the elevation θ and azimuth angles ϕ. The dynamics of the helicopter are given by:

$$M(\phi,\theta)\begin{bmatrix}\ddot{\phi}\\ \ddot{\theta}\end{bmatrix}+C(\phi,\theta,\dot{\phi},\dot{\theta})\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\end{bmatrix}+G(\phi,\theta)+F_v+F_c=\tau \qquad (8.26)$$

where

$$M(\dot{\phi},\theta)=\begin{bmatrix}\cos^2\phi I_{L2} & 0\\ 0 & I_{L2}\end{bmatrix}$$

$$C(\phi,\theta,\dot{\phi},\dot{\theta})=\begin{bmatrix}-\cos\theta\sin\theta\dot{\theta}I_{L2} & -\cos\theta\sin\theta\dot{\phi}I_{L2}\\ \cos\theta\sin\theta\dot{\phi}I_{L2} & 0\end{bmatrix}$$

$$G(\phi,\theta)=\begin{bmatrix}0\\ mgl_c\cos\theta\end{bmatrix}$$

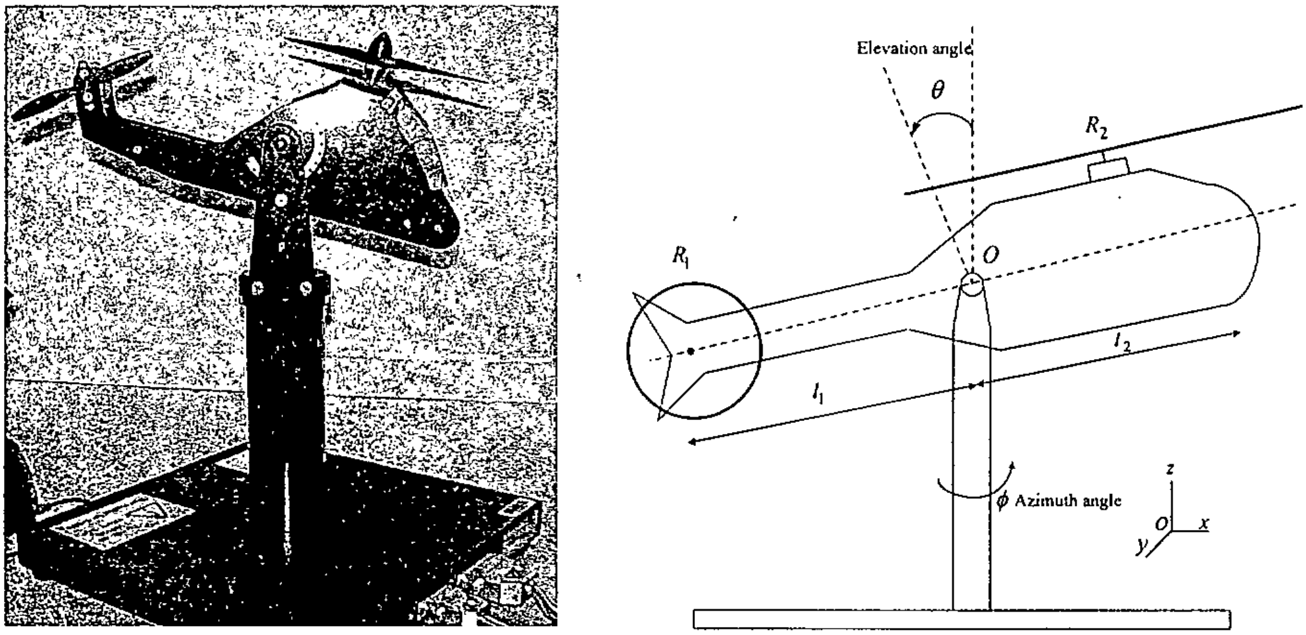
Actually , different friction terms should be added into the dynamics model in a practical implementation, two friction are consider in our model, one is viscous friction (A force that are proportional to the velocity)

$$F_v=\begin{bmatrix}k_{v_\phi}\dot{\phi}\\ k_{v_\theta}\dot{\theta}\end{bmatrix}$$

and coloumb friction (A force with constant amplitude that are acting in the direction in the velocity)

$$F_c=\begin{bmatrix}k_{c_1}\operatorname{sgn}(\dot{\phi})\\ k_{c_2}\operatorname{sgn}(\dot{\theta})\end{bmatrix}$$

The inertia matrix M is a positive definite symmetric matrix and $\dot{M}-2C$ is skew-symmetric matrix. The parameters of the helicopter are: the arm length to rotor 1 $l_1=0.198\text{m}$, the arm length to rotor 2 $l_2=0.174\text{m}$,mass of main body $m_l=0.35\text{kg}$,mass of rotor 1 $m_1=0.42\text{kg}$ mass of rotor 2 $m_2=0.16\text{kg}$ .The total mass of the main body is $m=m_l+m_1+m_2$, the intertia of the amin body is the sum of the moment of inertia for the solid bar and fro the pint masses of the rotors $I_{L2}=m_l/3\cdot(l_1^3+l_2^3)/(l_1+l_2)+m_1l_1^2+m_2l_2^2$. The center of gravity is $l_c=(m_l(l_1-l_2)+m_1l_1-m_2l_2)/m$. .



(a) (b)

Figure 8.1 (a) The 2DOF helicopter apparatus.(b) Sketch of the helicopter model seen from the side.

All the constant numbers are given in the Humusoft [Humusoft 1985]. The control objective is to maintain the elevation and azimuth angles to track the desired angle trajectory $y_{1m}=(\pi/40)\sin(t)$ and $y_{2m}=(\pi/10)\sin(t)$. Assume the motors be able to generate torques without any delay. The model can be described by the nonlinear state equations with four states, two inputs and two outputs nonlinear system. The feedback and observer gain vector are given as

$$K_c = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}, \ K_o^T = \begin{bmatrix} 90 & 180 & 0 & 0 \\ 0 & 0 & 80 & 190 \end{bmatrix}$$

Let the learning rate $\gamma_{11} = 5$, $\gamma_{21} = 8$ and step size 0.05sec. The fuzzy controllers are defined by the following set of rules:

FAC1:

$$\begin{aligned} &\textit{If} \quad x_1 \quad \textit{is} \quad V_k^1 \\ &\textit{Then} \quad u_{d1} = a_{k_0}^1 + a_{k_1}^1 x_1 + a_{k_2}^1 x_2 + a_{k_3}^1 x_3 + a_{k_4}^1 x_4 \end{aligned}$$

FAC2:

$$\begin{aligned} &\textit{If} \quad x_3 \quad \textit{is} \quad V_k^2 \\ &\textit{Then} \quad u_{d_2} = a_{k_0}^2 + a_{k_1}^2 x_1 + a_{k_2}^2 x_2 + a_{k_3}^2 x_3 + a_{k_4}^2 x_4 \end{aligned}$$

V_k^1 and V_k^2 $k = 1,\cdots,3$ are the membership functions for fuzzy controllers are chosen as:

$$\mu_{V_1^1}(\hat{x}_j) = \exp[-((\hat{x}_i + \pi/15)/(\pi/30))^2]$$

$$\mu_{V_2^1}(\hat{x}_j) = \exp[-(\hat{x}_i/(\pi/30))^2]$$

$$\mu_{V_3^1}(\hat{x}_j) = \exp[-((\hat{x}_i - \pi/15)/(\pi/30))^2]$$

$$\mu_{V_1^2}(\hat{x}_j) = \exp[-((\hat{x}_i + \pi/6)/(\pi/12))^2]$$

$$\mu_{V_2^2}(\hat{x}_j) = \exp[-(\hat{x}_i/(\pi/12))^2]$$

$$\mu_{V_3^2}(\hat{x}_j) = \exp[-((\hat{x}_i - \pi/6)/(\pi/12))^2]$$

And initial consequent parameters of fuzzy rules are chosen randomly in the interval [0,1].

Figure 8.2-8.5 shows the results for real time 2DOF helicopter control system. It can be seen that the tracking performance can achieve.

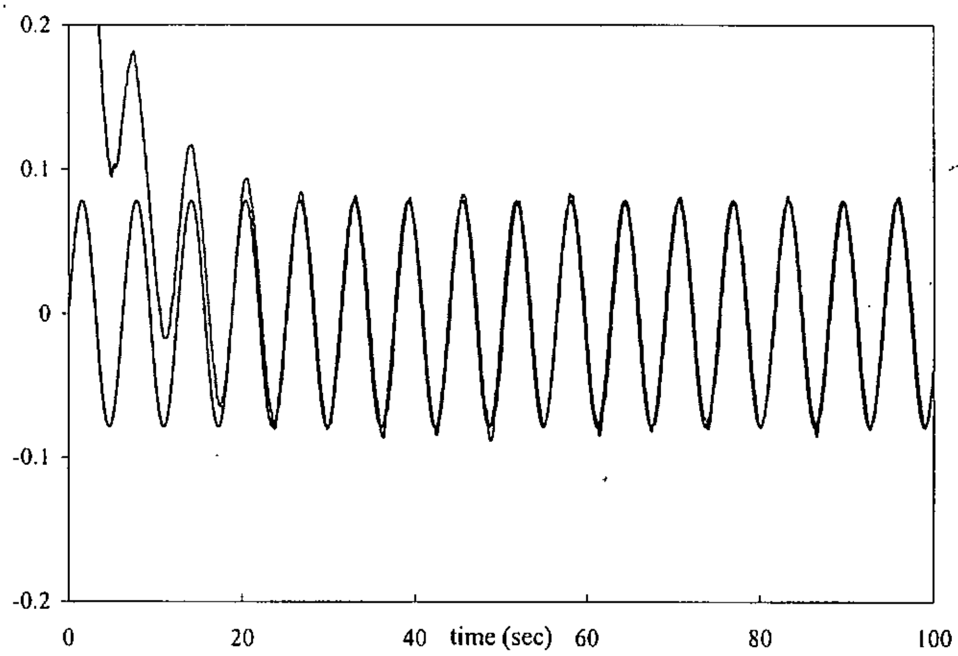


Figure 8.2 Trajectory of the Elevation

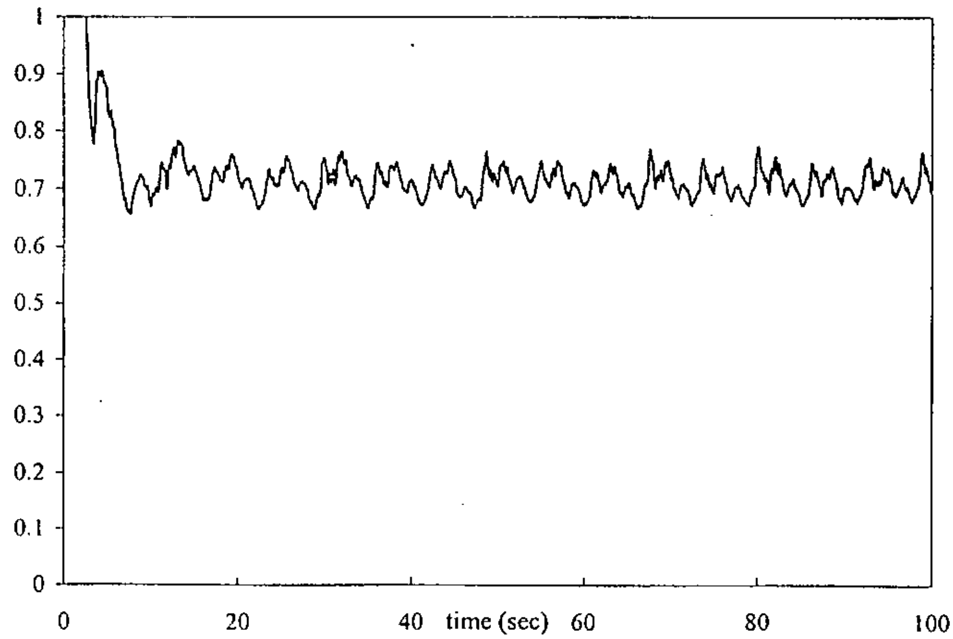


Figure 8.3 Elevation control

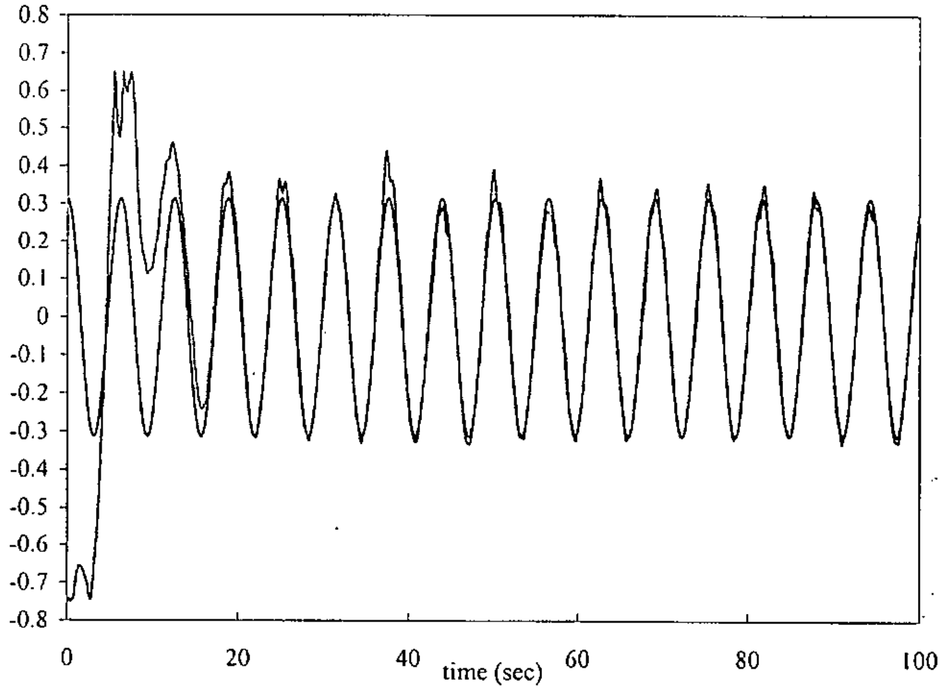


Figure 8.4 Trajectory of the Azimuth

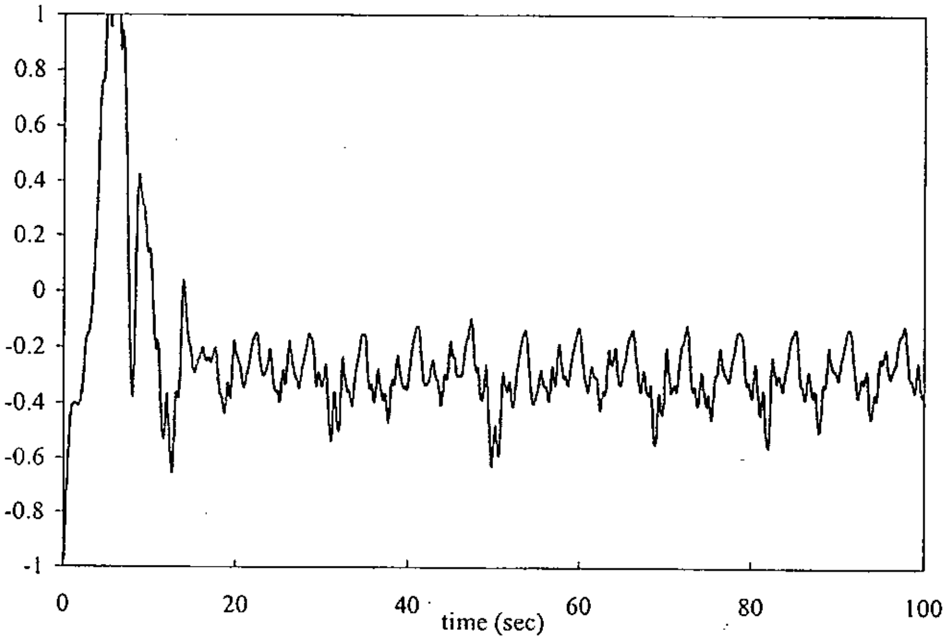


Figure 8.5 Azimuth control

8.6 Conclusion

In this chapter, a new observer-based direct adaptive fuzzy controller for MIMO nonlinear systems has been proposed. Using the state observer, it does not require the system full state to be available for measurement. Based on the TS base fuzzy systems, which show that the adaptive controller used few rules can achieve the control performance. The stability of the adaptive control scheme is proved in Lyapunov sense. Finally, the proposed method has been applied to control a two degree freedom helicopter system to track a reference trajectory. The experimental results show that the adaptive controller can achieve the desired performance, which permits real time applications.

In the next chapter, we will consider the adaptive robust fuzzy control for nonlinear systems by using other modern control technique i.e. Backstepping control with adaptive fuzzy system.

CHAPTER NINE

ROBUST ADAPTIVE FUZZY CONTROL FOR STRICT FEEDBACK NONLINEAR SYSTEM

9.1 Introduction

In Chapters 5-8, stable adaptive fuzzy controllers were proposed for nonlinear systems in a Byrnes-Isidori normal form [Isidori 1995, Sastry 1999]. In this chapter, using the idea of backstepping, adaptive fuzzy backstepping control scheme is presented for nonlinear systems without satisfying matching condition. The control can not only guarantee the boundedness of closed-loop system signals for the case of both unknown nonlinear functions and parameter uncertainty are present in the systems, but also render that the system output follows a desired trajectory. Note that in the literature survey, while the nonlinear strict-feedback systems have been much investigated via backstepping design [Polycarpou and Ioannou 1995, Krstic *et al.* 1995]. Only a few results are available in the literature for the approximator-based adaptive control of the strict-feedback nonlinear systems with parametric uncertainties [Ge *et al.* 1996, Li *et al.* 2004, Yang *et al.* 2004]. In this chapter, a direct adaptive fuzzy control scheme is presented by using the idea of backstepping control. Some of the states variables are considered as virtual control, and the intermediate control laws are designed in constructive design procedures. With the help of fuzzy approximation, to cancel the unknown functions in backstepping design. It is proved that the adaptive fuzzy control scheme achieve semi-global uniform ultimate boundedness of the signal in the closed loop system.

The rest of this chapter is organized as follow: Section 9.2 a description of the system is given. Adaptive fuzzy backstepping design is included in Section 9.3. Computer simulation

This chapter is based on the paper 8 on chapter one, page 11 results for the proposed control algorithm are illustrated in Section 9.4. Finally, the chapter is concluded in Section 9.5.

9.2 System Description

We assume that the system can be transformed into a strict-feedback canonical form (Section 2.3.4) which is re-stated as follows:

$$\begin{aligned} \dot{x}_i &= f_i(\bar{x}_i) + g_i(\bar{x}_i)x_{i+1} + \Delta_i(t,x), \quad 1 \le i \le n-1 \\ \dot{x}_n &= f_n(\bar{x}_n) + g_n(\bar{x}_n)u + \Delta_n(t,x) \\ y &= x_1 \end{aligned} \tag{9.1}$$

where $\bar{x}_i = [x_1, x_2, \cdots, x_i]^T \in R^i$, $i = 1,2,\cdots,n$, $u \in R$, $y \in R$ are state variables, system input and output, respectively. $f_i(\cdot)$ and $g_i(\cdot)$, $i = 1,2,\cdots,n$ are unknown smooth function and Δ_i are unknown Lipschitz continuous functions. The control objective is to design an adaptive robust fuzzy-based controller for nonlinear SISO systems (1) under plant uncertainties and disturbances, which guarantees boundedness of all the variables of the closed system and the output y follow a given desired trajectory y_m. Since $g_i(\cdot)$, $i = 1,2,\cdots,n$ are smooth functions, they have therefore bounded within some compact set. We make the following assumptions on system (9.1) for functions $g_i(\cdot)$..

Assumption 9.1: The signs of $g_i(\cdot)$ are known, and there exist $b_{i1} \ge b_{i0} > 0$ such that $b_{i1} \ge |g_i(\cdot)| \ge b_{i0}$, $\forall x_i \in R^i$.

The above assumption implies that the smooth function $g_i(\cdot)$ are strictly either positive or negative. Hence, without loss of generality we assume that $b_{i1} \ge g_i(\cdot) \ge b_{i0} > 0, \forall x_i \in R^i$.

Assumption 9.2: There exist constants $b_{id} > 0$ such that, $|\dot{g}_i(\cdot)| \le b_{id}$, $\forall x \in R^n$.

Assumption 9.3: For $1 \le i \le n$, there exists unknown positive constant p_i^* such that

$|\Delta_i(t,x)| \le p_i^* \varphi_i(|x_i|), \quad \forall (t,x) \in R_+ \times R^n$

where φ_i is a known nonnegative function

9.3 Adaptive Fuzzy Control

The detailed design procedure is described in the following steps. Step1 and 2 are described with detailed explanations, while Step i and Step n are simplified, with the relevant equations and explanations being omitted.

Step 1: Define $z_1 = x_1 - y_m$. Its derivative is

$$\begin{aligned}\dot{z}_1 &= f_1(x_1) + g_1(x_1)x_2 + \Delta_1 - \dot{y}_m \\ &= g_1(x_1)[g_1^{-1}(x_1)f_1(x_1) + x_2 + g_1^{-1}(\Delta_1 - \dot{y}_m)]\end{aligned} \tag{9.2}$$

by viewing x_2 as a virtual control input for z_1 subsystem in the above equation, and consider the TS fuzzy system to approximate the uncertain term

$$\begin{aligned}\frac{f_1(x_1)}{g_1(x_1)} &= \xi_1(x_1)A_1^0 + \xi_1(x_1)A_1x_1 + \varepsilon_1 \\ &= \xi_1(x_1)A_1^0 + \xi_1(x_1)A_1z_1 + \xi_1(x_1)A_1y_m + \varepsilon_1 \\ &= K_{g1}\xi_1(x_1)w_1 + \xi_1(x_1)A_1^0 + \xi_1(x_1)A_1y_m + \varepsilon_1\end{aligned} \tag{9.3}$$

where $w_1 = A_1^m z_1$ and ε_1 is a approximating error. Define $K_{g1} = \|A_1\| = \lambda_{\max}^{1/2}(A_1^T A_1)$, such that $A_1 = K_{g1}A_1^m$ and $\|A_1^m\| \le 1$ substituting equation (9.3) into equation (9.2)

$$\dot{z}_1 = g_1(x_1)[x_2 + K_{g1}\xi_1(x_1)w_1 + v_1] \tag{9.4}$$

where $v_1 = \xi_1(x_1)A_1^0 + \xi_1(x_1)A_1y_d + \varepsilon_1 + g_1^{-1}(\Delta_1 - \dot{y}_m)$. In the light of assumption 9.3, we can obtain a bound for v_1 as follows

$$\|v_1\| \le \theta_1\phi_1(x_1) \tag{9.5}$$

where $\theta_1 = \max[\|A_1^0\| + \|A_1\|\|y_m\|, \varepsilon_1 + b_{10}^{-1}|\dot{y}_m|, b_{10}^{-1}p_1^*]$ and $\phi_1(x_1) = 1 + \|\xi_1\| + \varphi_1(x_1)$.

Consider the following Lyapunov candidate:

$$V_1 = \frac{1}{2g_1(x_1)} z_1^2 + \frac{1}{2}\Gamma_{11}^{-1}\tilde{\psi}_1^2 + \frac{1}{2}\Gamma_{12}^{-1}\tilde{\theta}_1^2 \tag{9.6}$$

where Γ_{11}, $\Gamma_{12} > 0$. $\tilde{\psi}_1 = (K_{g1}^2 - \psi_1)$ and $\tilde{\theta}_1 = (\theta_1 - \hat{\theta}_1)$. ψ_1 and $\hat{\theta}_1$ are the estimates of K_{g1}^2 and θ_1, respectively. The time derivative of V_1 is

$$\begin{aligned}\dot{V}_1 &= \frac{z_1\dot{z}_1}{g_1(x_1)} - \frac{\dot{g}_1(x_1)}{2g_1^2(x_1)} - \Gamma_{11}^{-1}\tilde{\psi}_1\dot{\psi}_1 - \Gamma_{12}^{-1}\tilde{\theta}_1\dot{\hat{\theta}}_1 \\ &= z_1[x_2 + K_{g1}\xi_1(x_1)w_1 + v_1] - \frac{\dot{g}_1(x_1)}{2g_1^2(x_1)} z_1^2 - \Gamma_{11}^{-1}\tilde{\psi}_1\dot{\psi}_1 - \Gamma_{12}^{-1}\tilde{\theta}_1\dot{\hat{\theta}}_1\end{aligned} \tag{9.7}$$

Define $z_2 = x_2 - \alpha_1$ and let α_1

$$\alpha_1 = -k_1 z_1 - \frac{\psi_1}{4\gamma_1}\xi_1(x_1)\xi_1^T(x_1) - \hat{\theta}_1\phi_1(x_1)\tanh\left(\frac{\hat{\theta}_1\phi_1(x_1)z_1}{\lambda_1}\right) \tag{9.8}$$

where $k_1 > 0$ and $\lambda_1 > 0$ are the design constants. Consider the following adaptation laws:

$$\dot{\psi}_1 = \Gamma_{11}\left[\frac{1}{4\gamma_1^2}\xi_1(x_1)\xi_1^T(x_1)z_1^2 - \delta_{11}(\psi_1 - \psi_1^0)\right] \tag{9.9}$$

$$\dot{\hat{\theta}}_1 = \Gamma_{12}\left[\phi_1(x_1)\|z_1\| - \delta_{12}(\hat{\theta}_1 - \theta_1^0)\right] \tag{9.10}$$

where $\delta_{11}, \delta_{12} > 0$ and $\psi_1^0, \theta_1^0 \geq 0$ are design constants. The adaptive law (9.9) and (9.10) are so-called σ -modification, [Ioannou and Kokotovic 1983], introduced to improve the robustness and avoid the parameters to drift to very large values.

Using (9.6), a direct substitution of $x_2 = z_2 + \alpha_1$

$$\begin{aligned}\dot{V}_1 &= z_1[z_2 + \alpha_1 + K_{g1}\xi_1(x_1)w_1 + v_1] - \frac{\dot{g}_1(x_1)}{2g_1^2(x_1)} z_1^2 - \Gamma_{11}^{-1}\tilde{\psi}_1\dot{\psi}_1 - \Gamma_{12}^{-1}\tilde{\theta}_1\dot{\hat{\theta}}_1 \\ &= z_1 z_2 - \left(k_1 + \frac{\dot{g}_1(x_1)}{2g_1^2(x_1)}\right)z_1^2 - \frac{\psi_1}{4\gamma_1}\xi_1(x_1)\xi_1^T(x_1)z_1 - \hat{\theta}_1\phi_1(x_1)z_1\tanh\left(\frac{\hat{\theta}_1\phi_1(x_1)z_1}{\lambda_1}\right) \\ &\quad + K_{g1}\xi_1(x_1)w_1 z_1 + v_1 z_1 - \Gamma_{11}^{-1}\tilde{\psi}_1\dot{\psi}_1 - \Gamma_{12}^{-1}\tilde{\theta}_1\dot{\hat{\theta}}_1\end{aligned} \tag{9.11}$$

Let $\gamma_1 > 0$, by the completing squares, we have

$$\begin{aligned} K_{g1}\xi_1(x_1)w_1 z_1 &\le \frac{K_{g1}^2}{4\gamma_1^2}\xi_1\xi_1^T z_1^2 + \gamma_1^2 w_1^T w_1 \\ &\le \frac{\psi_1}{4\gamma_1^2}\xi_1\xi_1^T z_1^2 + \frac{\widetilde{\psi}_1}{4\gamma_1^2}\xi_1\xi_1^T z_1^2 + \gamma_1^2 w_1^T w_1 \end{aligned} \tag{9.12}$$

Using equation (9.5) and the relative item of equation (9.11), we get

$$\begin{aligned} z_1[\theta_1\phi_1(x_1) - \hat{\theta}_1\phi_1(x_1)\tanh\left(\frac{\hat{\theta}_1\phi_1(x_1)z_1}{\lambda_1}\right) &\le \hat{\theta}_1\phi_1(x_1)\|z_1\| - \hat{\theta}_1\phi_1(x_1)z_1\tanh\left(\frac{\hat{\theta}_1\phi_1(x_1)z_1}{\lambda_1}\right) \\ &+ (\theta_1 - \hat{\theta}_1)\phi_1(x_1)\|z_1\| \end{aligned} \tag{9.13}$$

and using the following lemma with regard to function $\tanh(\cdot)$ [Polycarpou and Ioannou 1995]

$$\hat{\theta}_1\phi_1(x_1)\|z_1\| - \hat{\theta}_1\phi_1(x_1)z_1\tanh\left(\frac{\hat{\theta}_1\phi_1(x_1)z_1}{\lambda_1}\right) \le \eta_1 \tag{9.14}$$

Substituting equation (9.12)-(9.14) into equation (9.11), such that

$$\dot{V}_1 \le z_1 z_2 - k_1^* z_1^2 + \gamma_1^2 w_1^T w_1 + \Gamma_{11}^{-1}\widetilde{\psi}_1\left(\frac{\Gamma_{11}}{4\gamma_1^2}\xi_1\xi_1^T z_1^T - \dot{\psi}_1\right) + \Gamma_{12}^{-1}\widetilde{\theta}_1(\Gamma_{12}\phi_1(x_1)\|z_1\| - \dot{\hat{\theta}}_1) + \eta_1 \tag{9.15}$$

substituting (9.9) and (9.10) into(9.15),we get

$$\dot{V}_1 \le -k_1^* z_1^2 + z_1 z_2 + \gamma_1^2 w_1^T w_1 + \delta_{11}\widetilde{\psi}_1(\psi_1 - \psi_1^0) + \delta_{12}\widetilde{\theta}_1(\hat{\theta}_1 - \theta_1^0) + \eta_1 \tag{9.16}$$

By completing of square, we have

$$\begin{aligned} \dot{V}_1 \le &-k_1^* z_1^2 + z_1 z_2 + \gamma_1^2 w_1^T w_1 - \frac{1}{2}\delta_{11}\widetilde{\psi}_1^2 - \frac{1}{2}\delta_{12}\widetilde{\theta}_1^2 + \delta_{11}(K_{g1}^2 - \psi_1^0)^2 \\ &+ \delta_{12}(\theta_1 - \theta_1^0)^2 + \eta_1 \end{aligned} \tag{9.17}$$

where the coupling term $z_1 z_2$ will be canceled in next step. Observing from (9.11) that

$-(k_1 + \frac{\dot{g}_1}{2g_1^2})z_1^2 \le -(k_1 - (\frac{b_{1d}}{2b_{10}^2}))z_1^2$. We can choose $k_1^* = k_1 - (\frac{b_{1d}}{2b_{10}^2}) > 0$.

Step 2: Differentiating z_2 gives

$$\begin{aligned} \dot{z}_2 &= \dot{x}_2 - \dot{\alpha}_1 \\ &= f_2(\bar{x}_2) + g_2(\bar{x}_2)x_3 + \Delta_2 - \dot{\alpha}_1 \\ &= g_2(\bar{x}_2)[g_2^{-1}(\bar{x}_2)f_2(\bar{x}_2) + x_3 + g_2^{-1}(\Delta_2 - \dot{\alpha}_1)] \end{aligned} \tag{9.18}$$

The time derivative of α_1 is

$$\begin{aligned}\dot{\alpha}_1 &= \frac{\partial\alpha_1}{\partial x_1}\dot{x}_1 + \frac{\partial\alpha_1}{\partial\psi_1}\dot{\psi}_1 + \frac{\partial\alpha_1}{\partial\hat{\theta}_1}\dot{\hat{\theta}}_1 + \frac{\partial\alpha_1}{\partial y_m}\dot{y}_m \\ &= \frac{\partial\alpha_1}{\partial x_1}(f_1(x_1) + g_1(x_1)x_2 + \Delta_1) + \bar{\phi}_1 \\ &= f_{11} + \frac{\partial\alpha_1}{\partial x_1}\Delta_1 + \bar{\phi}_1\end{aligned} \tag{9.19}$$

where $\bar{\phi}_1 = \frac{\partial\alpha_1}{\partial\psi_1}\dot{\psi}_1 + \frac{\partial\alpha_1}{\partial\hat{\theta}_1}\dot{\hat{\theta}}_1 + \frac{\partial\alpha_1}{\partial y_m}\dot{y}_m$ is introduced as intermediate variable which is computable. Since, $f_1(x_1)$ and $g_1(x_1)$ are unknown function, $\dot{\alpha}_1$ is a scalar unknown function.

Let

$$\begin{aligned}\frac{1}{g_2(\bar{x}_2)}(f_2(\bar{x}_2) - f_{11}) &= \xi_2(\bar{x}_2)A_2^0 + \xi_2(\bar{x}_2)A_2(\bar{x}_2) + \varepsilon_2 \\ &= \xi_2(\bar{x}_2)A_2^0 + \xi_2(\bar{x}_2)A_2(z_1, z_2) + \xi_2(\bar{x}_2)A_2(y_m, \alpha_1) + \varepsilon_2 \\ &= K_{92}\xi_2(\bar{x}_2)w_2 + \xi_2(\bar{x}_2)A_2^0 + \xi_2(\bar{x}_2)A_2(y_m, \alpha_1) + \varepsilon_2\end{aligned} \tag{9.20}$$

and substituting it into equation (9.18), we will have

$$\dot{z}_2 = g_2(x_1)[x_3 + K_{92}\xi_2(\bar{x}_2)w_2 + v_2] \tag{9.21}$$

where

$$v_2 = \xi_2(\bar{x}_2)A_2^0 + \xi_2(\bar{x}_2)A_2(y_d, \alpha_1) + \varepsilon_2 + g_2^{-1}(\Delta_2 - \frac{\partial\alpha_1}{\partial x_1}\Delta_1 - \bar{\phi}_1) \tag{9.22}$$

where $\|v_2\| \le \theta_2\phi_2$, $\phi_2(x_2) = 1 + \|\xi_2\| + \varphi_2 + \left\|\frac{\partial\alpha_1}{\partial x_1}\right\|\varphi_1$ and $\theta_2 = \max[\|A_2^0\| + \|A_2\|\|y_m\| + \|A_2\|\|\alpha_1\|], |\varepsilon_2| + b_{20}^{-1}|\bar{\phi}_1|, b_{20}^{-1}p_2^*, b_{20}^{-1}p_1^*]$.

Consider the Lyapunov function candidate

$$V_2 = V_1 + \frac{1}{2g_2(\bar{x}_2)}z_2^2 + \frac{1}{2}\Gamma_{21}^{-1}\tilde{\psi}_2^2 + \frac{1}{2}\Gamma_{22}^{-1}\tilde{\theta}_2^2 \tag{9.23}$$

The derivative of V_2 is

$$\dot{V}_2 = \dot{V}_1 + z_2[x_3 + k_{g2}\xi_2(\bar{x}_2)w_2 + v_2] - \frac{\dot{g}_2(\bar{x}_2)}{2g_2^2(\bar{x}_2)} - \Gamma_{21}^{-1}\tilde{\psi}_2\dot{\psi}_2 - \Gamma_{22}^{-1}\tilde{\theta}_2\dot{\hat{\theta}}_2 \tag{9.24}$$

Define the error variable $z_3 = x_3 - \alpha_2$, taking the intermediate stabilizing function α_2

$$\alpha_2 = -z_1 - k_2 z_2 - \frac{\psi_2}{4\gamma_2}\xi_2\xi_2^T - \hat{\theta}_2\phi_2 \tanh\left(\frac{\hat{\theta}_2\phi_2 z_2}{\lambda_2}\right) \tag{9.25}$$

where $k_2 > 0$ and $\lambda_2 > 0$ are the design constants. Consider the following adaptation laws:

$$\dot{\psi}_2 = \Gamma_{21}\left[\frac{1}{4\gamma_2^2}\xi_2\xi_2^T z_2^2 - \delta_{21}(\psi_2 - \psi_2^0)\right] \tag{9.26}$$

$$\dot{\hat{\theta}}_2 = \Gamma_{22}\left[\phi_2\|z_2\| - \delta_{22}(\hat{\theta}_2 - \theta_2^0)\right] \tag{9.27}$$

where $\delta_{21}, \delta_{22} > 0$ and $\psi_2^0, \theta_2^0 \geq 0$ are design constants. Substituting equation (9.22) and equation (9.25) into equation (9.24), such that

$$\begin{aligned}\dot{V}_2 \leq & -k_1^* z_1^2 + z_1 z_2 + \gamma_1^2 w_1^T w_1 - \frac{1}{2}\delta_{11}\tilde{\psi}_1^2 - \frac{1}{2}\delta_{21}\tilde{\theta}_1^2 + \delta_{11}(K_{g1}^2 - \psi_1^0)^2 + \delta_{21}(\theta_1 - \theta_1^0) \\ & + \eta_1 + z_2 z_3 + z_2\alpha_2 + z_2 K_{\theta 2}\xi_2(\bar{x}_2)w_2 + z_2 v_2 - \frac{\dot{g}_2(\bar{x}_2)}{2g_2^2(\bar{x}_2)} - \Gamma_{21}^{-1}\tilde{\psi}_2\dot{\psi}_2 - \Gamma_{22}^{-1}\tilde{\theta}_2\dot{\hat{\theta}}_2\end{aligned} \tag{9.28}$$

substituting equation (9.26) and equation (9.27) into equation (9.28),we get

$$\begin{aligned}\dot{V}_2 \leq & z_2 z_3 - \sum_{l=1}^{2} k_l^* z_l^2 - \sum_{l=1}^{2}\left(\frac{\delta_{l1}\tilde{\theta}_l^2}{2} + \frac{\delta_{l2}\tilde{\psi}_l^2}{2}\right) + \sum_{l=1}^{2}\left(\frac{\delta_{l1}(K_{gl}^2 - \psi_l^0)^2}{2} + \frac{\delta_{l2}(\theta_l - \theta_l^0)^2}{2}\right) \\ & + \sum_{l=1}^{2}\eta_l + \sum_{l=1}^{2}\gamma_l^2 w_l^T w_l\end{aligned} \tag{9.29}$$

where k_2^* is chosen such that $k_2^* = k_2 - (\frac{b_{2d}}{2b_{10}^2}) > 0$.

Step i $(3 \leq i \leq n-1)$: In a similar fashion, we can design a virtual controller to make the error z_i as small as possible. Differenting z_i gives

$$\dot{z}_i = g_i(\bar{x}_i)[g_i^{-1}(\bar{x}_i)f_i(\bar{x}_i) + x_{i+1} + g_i^{-1}(\Delta_i - \dot{\alpha}_{i-1})] \tag{9.30}$$

we also use a TS fuzzy system to approximate the unknown function

$$\begin{aligned}\frac{1}{g_i(\bar{x}_i)}(f_i(\bar{x}_i)-f_{(i-1)(i-1)}) &= \xi_i(\bar{x}_i)A_i^0+\xi_i(\bar{x}_i)A_i\bar{x}_i+\varepsilon_i\\ &= \xi_i(\bar{x}_i)A_i^0+\xi_i(\bar{x}_i)A_iz_i+\xi_i(\bar{x}_i)A_i(y_m,\cdots,\alpha_{i-1})+\varepsilon_i\\ &= K_{gi}\xi_i(\bar{x}_i)w_i+\xi_i(\bar{x}_i)A_i^0+\xi_i(\bar{x}_i)A_i(y_m,\cdots,\alpha_{i-1})+\varepsilon_i\end{aligned} \tag{9.31}$$

and substitute it into equation (9.30)

$$\dot{z}_i = g_i(\bar{x}_i)[x_{i+1}+K_{gi}\xi_i(\bar{x}_i)w_i+v_i] \tag{9.32}$$

where $\|v_i\| \le \theta_i\phi_i$, $\phi_i(\bar{x}_i)=1+\|\xi_i\|+\varphi_i+\sum_{l=1}^{i-1}\left\|\frac{\partial\alpha_{i-1}}{\partial x_l}\right\|\varphi_l$.

Similarly, let the virtual controller to be of the form

$$\alpha_i = -z_{i-1}-k_iz_i-\frac{\psi_i}{4\gamma_i^2}\xi_i\xi_i^T-\hat{\theta}_i\phi_i\tanh\left(\frac{\hat{\theta}_i\phi_iz_i}{\lambda_i}\right) \tag{9.33}$$

where $k_i>0$ and $\lambda_i>0$ are the design constants. Consider the following adaptation laws:

$$\dot{\psi}_i = \Gamma_{i1}\left[\frac{1}{4\gamma_i^2}\xi_i\xi_i^Tz_i^2-\delta_{i1}(\psi_i-\psi_i^0)\right] \tag{9.34}$$

$$\dot{\hat{\theta}}_i = \Gamma_{i2}\left[\phi_i\|z_i\|-\delta_{i2}(\hat{\theta}_i-\theta_i^0)\right] \tag{9.35}$$

where $\delta_{i1},\delta_{i2}>0$ and $\psi_i^0,\theta_i^0\ge 0$ are design constants. In a similar procedure, by considering Lyapunov function

$$V_i = V_{i-1}+\frac{1}{2g_i(\bar{x}_i)}z_i^2+\frac{1}{2}\Gamma_{i1}^{-1}\tilde{\psi}_i^2+\frac{1}{2}\Gamma_{i2}^{-1}\tilde{\theta}_i^2 \tag{9.36}$$

By using equation (9.32)-(9.35) and straightforward derivation similar to those employed in the former steps, the derivative of V_i becomes

$$\begin{aligned}\dot{V}_i \le\; & z_iz_{i+1}-\sum_{l=1}^{i}k_l^*z_l^2-\sum_{l=1}^{i}\left(\frac{\delta_{l1}\tilde{\theta}_l^2}{2}+\frac{\delta_{l2}\tilde{\psi}_l^2}{2}\right)+\sum_{l=1}^{i}\left(\frac{\delta_{l1}(K_{gk}^2-\psi_l^0)^2}{2}+\frac{\delta_{l2}(\theta_l-\theta_l^0)^2}{2}\right)\\ &+\sum_{l=1}^{i}\delta_l+\sum_{l=1}^{i}\gamma_l^2w_l^Tw_l\end{aligned} \tag{9.37}$$

we can choose $k_i^* = k_i-(\frac{g_{id}}{2g_{i0}^2})>0$.

Step n: This is the final step. Define the error variable as $z_n = x_n - \alpha_{n-1}$, we will have

$$\dot{z}_n = g_n(\bar{x}_n)[g_n^{-1}(\bar{x}_n)f_n(\bar{x}_n) + u + g_n^{-1}(\Delta_n - \dot{\alpha}_{n-1})] \tag{9.38}$$

Using the similar way, we have

$$\begin{aligned}\dot{\alpha}_{n-1} &= \sum_{l=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_l}(f_l(\bar{x}_l) + g_l(\bar{x}_l)x_{l+1} + \Delta_l) + \sum_{l=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial\psi_l}\dot{\psi}_l + \sum_{l=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial\hat{\theta}_l}\dot{\hat{\theta}}_l + \frac{\partial\alpha_{n-1}}{\partial y_m}\dot{y}_m \\ &= f_{(n-1)(n-1)} + \sum_{l=1}^{n-1}\frac{\partial\alpha_{n-1}}{\partial x_l}\Delta_l + \bar{\phi}_{(n-1)(n-1)}\end{aligned} \tag{9.39}$$

we also use a TS fuzzy system to approximate the unknown function

$$\begin{aligned}\frac{1}{g_n(\bar{x}_n)}(f_n(\bar{x}_n) - f_{(n-1)(n-1)}) &= \xi_n(\bar{x}_n)A_n^0 + \xi_n(\bar{x}_n)A_n\bar{x}_n + \varepsilon_n \\ &= \xi_n(\bar{x}_n)A_n^0 + \xi_n(\bar{x}_n)A_n z_n + \xi_n(\bar{x}_n)A_n(y_m,\cdots,\alpha_{n-1}) + \varepsilon_n \\ &= K_{\vartheta n}\xi_n(\bar{x}_n)w_n + \xi_n(\bar{x}_n)A_n^0 + \xi_i(\bar{x}_n)A_n(y_m,\cdots,\alpha_{n-1}) + \varepsilon_n\end{aligned} \tag{9.40}$$

where $w_n = A_n^m z_n, K_{\vartheta n} = \left\|A_n^m\right\|$ and $A_n = K_{\vartheta n}A_n^m$ and substitute it into equation (9.38)

$$\dot{z}_n = g_n(\bar{x}_n)[u + K_{\vartheta n}\xi_n(\bar{x}_n)w_n + v_n] \tag{9.41}$$

where $\|v_n\| \le \theta_n\phi_n$, $\phi_n = 1 + \|\xi_n\| + \varphi_n + \sum_{l=1}^{n-1}\left\|\frac{\partial\alpha_{n-1}}{\partial x_l}\right\|\varphi_l$

Similarly letting

$$u = -z_{n-1} - k_n z_n - \frac{\psi_n}{4\gamma_n^2}\xi_n\xi_n^T - \hat{\theta}_n\phi_n\tanh\left(\frac{\hat{\theta}_n\phi_n z_n}{\lambda_n}\right) \tag{9.42}$$

and consider the following adaptation law

$$\dot{\psi}_n = \Gamma_{n1}\left[\frac{1}{4\gamma_n^2}\xi_n\xi_n^T z_n^2 - \delta_{n1}(\psi_n - \psi_n^0)\right] \tag{9.43}$$

$$\dot{\hat{\theta}}_n = \Gamma_{n2}\left[\phi_n\|z_n\| - \delta_{n2}(\hat{\theta}_n - \theta_n^0)\right] \tag{9.44}$$

where $\delta_{i1}, \delta_{i2} > 0$ and $\psi_i^0, \theta_i^0 \ge 0$ are design constants. Consider the overall Lyapunov function candidate

$$V_n = V_{n-1} + \frac{1}{2g_n(\bar{x}_n)} z_n^2 + \frac{1}{2}\Gamma_{n1}^{-1}\tilde{\psi}_n^2 + \frac{1}{2}\Gamma_{n2}^{-1}\tilde{\theta}_n^2 \tag{9.45}$$

By using equation (9.41)-(9.44) and straightforward derivation similar to those employed in the former steps, the derivative of V_n becomes

$$\begin{aligned}\dot{V}_n &\le -\sum_{l=1}^{n} k_l^* z_l^2 - \sum_{l=1}^{n}\left(\frac{\delta_{l1}\tilde{\theta}_l^2}{2} + \frac{\delta_{l2}\tilde{\psi}_l^2}{2}\right) + \sum_{i=1}^{n}\left(\frac{\delta_{1l}(K_{gl}^2 - \psi_l^0)^2}{2} + \frac{\delta_{2l}(\theta_l - \theta_l^0)^2}{2}\right) \\ &\quad + \sum_{l=1}^{n}\eta_l + \sum_{l=1}^{n}\gamma_l^2 w_l^T w_l \\ &\le -\sum_{l=1}^{n} k_l^* z_l^2 - \sum_{l=1}^{n}\left(\frac{\delta_{l1}\tilde{\theta}_l^2}{2} + \frac{\delta_{l2}\tilde{\lambda}_l^2}{2}\right) + \gamma^2\|w\|^2 + \bar{\eta}\end{aligned} \tag{9.46}$$

where k_n^* is chosen such that $k_n^* = k_n - (\frac{g_{nd}}{2g_{n0}^2}) > 0$.

where $\bar{\eta} = \sum_{l=1}^{n}(\eta_l + \delta_{1l}(K_{gl}^2 - \psi_l^0)^2/2 + \delta_{2l}(\theta_l - \theta_l^0)^2/2)$, $w = [w_1, w_2, \cdots, w_n]^T$ and $\gamma = (\gamma_1^2 + \gamma_2^2 + \cdots + \gamma_n^2)^{1/2}$.

Theorem 9.1:

Consider the closed-loop system consisting of equation (9.1), the controller equation (9.42) with the intermediate stabilizing functions $\alpha_i, i = l, \cdots, n$, and the update laws $\psi_i, \hat{\theta}_i$. If we choose $\gamma < 1$ and $k_i^* = k_i - \left(\frac{g_{id}}{2g_{id}^2}\right) > 1,\ i = 1,2,\cdots,n$ in equation (9.46). Then for bounded initial conditions, we have all the signals in the closed-loop system remain bounded and the output tracking error converges to a small neighborhood around zero.

The proof is given as follows:

Proof: In order to use the small gain theorem [Jiang *et al.* 1994, Jiang and Mareels 1997], it is necessary to construct two subsystems $\Sigma_{\bar{z}w}$ and $\Sigma_{w\bar{z}}$ in composite feedback form. According to the error variable z_i, $i=1,2,\cdots,n$ and fuzzy system to approximate the unknown function $(f_n(\bar{x}_n)-f_{(n-1)(n-1)})/g_n(\bar{x}_n)$, $i=1,2,\cdots,n$, then the closed loop systems can be given as follows

$$\Sigma_{\tilde{z}w}:\begin{cases}\dot{z}_i=g_i(\bar{x}_i)[\bar{x}_{i+1}+K_{g_i}\xi_i(\bar{x}_i)w_i+v_i], & 1\le i\le n-1\\ \dot{z}_n=g_n(\bar{x}_n)[u+K_{g_n}\xi_n(\bar{x}_n)w_n+v_n]\\ \tilde{z}=H(z)=z\end{cases} \tag{9.47}$$

where $w=[w_1,w_2,\cdots,w_n]^T$ is considered as the virtual input and $\tilde{z}$ as the output of the system $\Sigma_{\tilde{z}w}$. If we pick $k_i^*>1$, $i=1,2,\cdots,n$ from equation (9.46), we get

$$\dot{V}_n\le -z^2+\gamma^2\|w\|^2+\bar{\eta} \tag{9.48}$$

By using ISpS Lyapunov theorem [Jiang *et al.* 1994], we propose the robust fuzzy control scheme such that the requirement of ISpS for system $\Sigma_{\tilde{z}w}$ can be stratified with the functions $\alpha_3(s)=s^2$ and $\alpha_3(s)=\gamma^2 s^2$ of class K_∞. We can get a gain function $\gamma_z(s)$ of system $\Sigma_{\tilde{z}w}$ as follow:

$$\gamma_z(s)=\alpha_1^{-1}\circ\alpha_2\circ\alpha_3^{-1}\circ\alpha_4 \quad \forall s>0$$

where $\alpha_1(z)\le V(x)\le\alpha_2(z)$.

For system $\Sigma_{w\tilde{z}}$, it is static system, we have

$$\Sigma_{w\tilde{z}}:\begin{cases}w_1=A_1^m z_1\\ w_2=A_2^m[z_1,z_2]^T\\ \vdots\\ w_n=A_n^m[z_1,z_2,\cdots,z_n]^T\end{cases} \tag{9.49}$$

Equation (9.49) can be rewrite as

$$w=\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}=K(\dot{z})$$
$$=\begin{bmatrix} A_1^m & 0 & \cdots & 0 \\ A_2^{m1} & A_2^{m2} & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ A_n^{m1} & A_n^{m2} & \cdots & A_n^{mn} \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}=Az \tag{9.50}$$

we have

$$\|w\| \le \|A\|\|z\| = \bar{\gamma}\|z\| \tag{9.51}$$

Then the gain function for system $\Sigma_{w\tilde{z}}$ is $\gamma_w(s)=\bar{\gamma}s$. Under the condition of small gain theorem , if there is a $\gamma_z \circ \gamma_w(s) < s$, Consider the interconnected system $\Sigma_{\tilde{z}w}$ and $\Sigma_{w\tilde{z}}$, in fact $\bar{\gamma}=\|A\|<1$ and we have $\gamma\bar{\gamma}<1$. Indeed, the small gain type condition can be satisfied by picking $\gamma<1$, such that it can prove that the composite closed-loop system is ISpS.

By substituting equation (9.51) into (9.46), the ISpS-Lyapunov function is satisfies

$$\begin{aligned} \dot{V}_n &\le -z^TQ_kz-\frac{1}{2}\tilde{\psi}^TQ_{\delta_1}\tilde{\psi}-\frac{1}{2}\tilde{\theta}^TQ_{\delta_2}\tilde{\theta}+\bar{\gamma}^2\gamma^2\|z\|^2+\bar{\eta} \\ &\le -z^TQ_kz-\frac{1}{2}\tilde{\psi}^TQ_{\delta_1}\tilde{\psi}-\frac{1}{2}\tilde{\theta}^TQ_{\delta_2}\tilde{\theta}+\|z\|^2+\bar{\eta} \\ &\le -cV_n+\bar{\eta} \end{aligned} \tag{9.52}$$

where

$Q_k=\text{diag}[k_1^*,k_2^*,\cdots,k_n^*], Q_{\delta_1}=\text{diag}[\delta_{11},\delta_{21},\cdots,\delta_{n1}]^T, Q_{\delta_2}=\text{diag}[\delta_{12},\delta_{22},\cdots,\delta_{n2}]^T$

$c=\min\{2(\lambda_{\min}(Q_k)-1)/b_0^{-1}), \lambda_{\min}(Q_{\delta_1})/\lambda_{\max}(\Gamma_1^{-1}), \lambda_{\min}(Q_{\delta_2})/\lambda_{\max}(\Gamma_2^{-1})\}$

$\Gamma_1=[\Gamma_{11},\Gamma_{21},\cdots,\Gamma_{n1}]^T$ and $\Gamma_2=[\Gamma_{12},\Gamma_{22},\cdots,\Gamma_{n2}]^T$

Let $\rho:=\bar{\eta}/c$ then equation (9.52) satisfies

$$0\le V_n(t)\le \rho+(V_n(0)-\rho)e^{-\bar{\eta}t} \tag{9.53}$$

(9.53) means that $V_n(t)$ eventually is bounded by ρ. This prove that all signals of the closed-loop system are uniformly ultimately bounded. Thus, the tracking error $z_1 = x_1 - y_m$ is also uniformly ultimately bounded. This concludes the proof.

Remark 9.1

Decreasing δ_{i1} and δ_{i2} will help to reduce the size of ρ. However, if δ_{i1} and δ_{i2} are too small, it may not be enough to prevent the parameter estimates form drifting to very larges values in the presence of fuzzy approximation errors. The small λ_i might result in a variation of a high gain control. Therefore, in practical applications, the design parameters should be adjusted carefully for achieving suitable transient performance and control action.

Remark 9.2

Compared with the works in [Kwan and Lewis 2000, Choi and Farrell 2001], it is assumed that the gain functions are constants or known function. However, this assumption cannot be satisfied in many cases. In [Zhang 2000, Yang 2004 *et al.*], gain functions are assumed to be unknown and a backstepping design is proposed that incorporates adaptive approximator techniques. However, due to the integral-type Lyapunov function introduced, this approach is complicated and difficult to use in practice. The proposed control scheme can cope with unknown gain functions and avoids controller singularity problem completely without the requirement for integral-type Lyapunov functions.

9.3 Simulation Examples

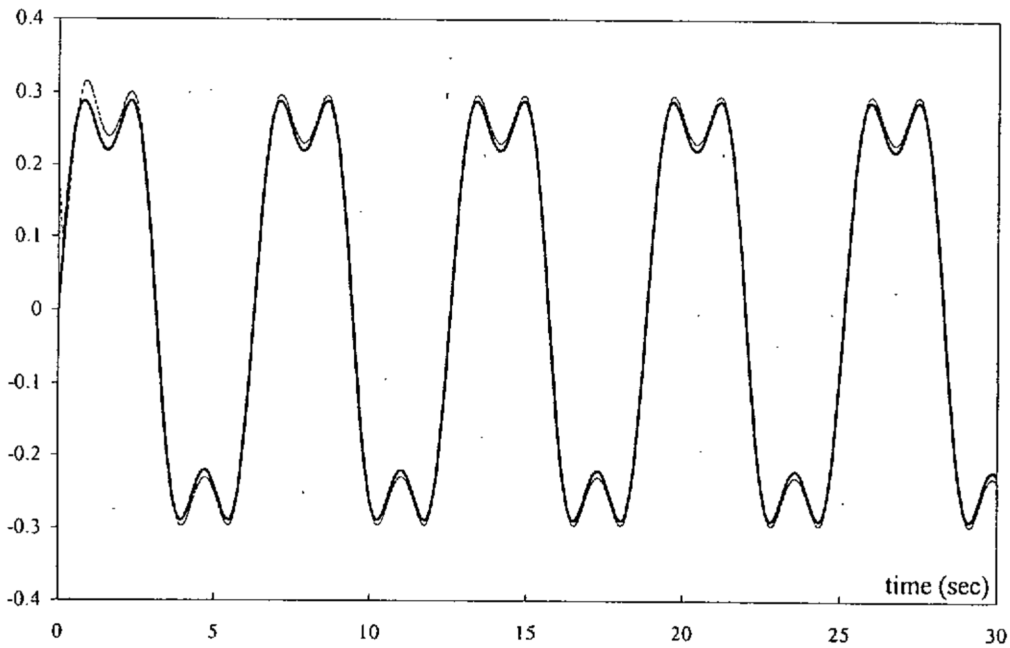
In this section, we apply our proposed adaptive fuzzy controller for three cases. The first example is a second order nonlinear system. The second example is a pendulum system plus driven motor. In the last one, we apply the proposed controller for a one-line robot with the inclusion of motor dynamics.

Example 9.1: **Second order system**

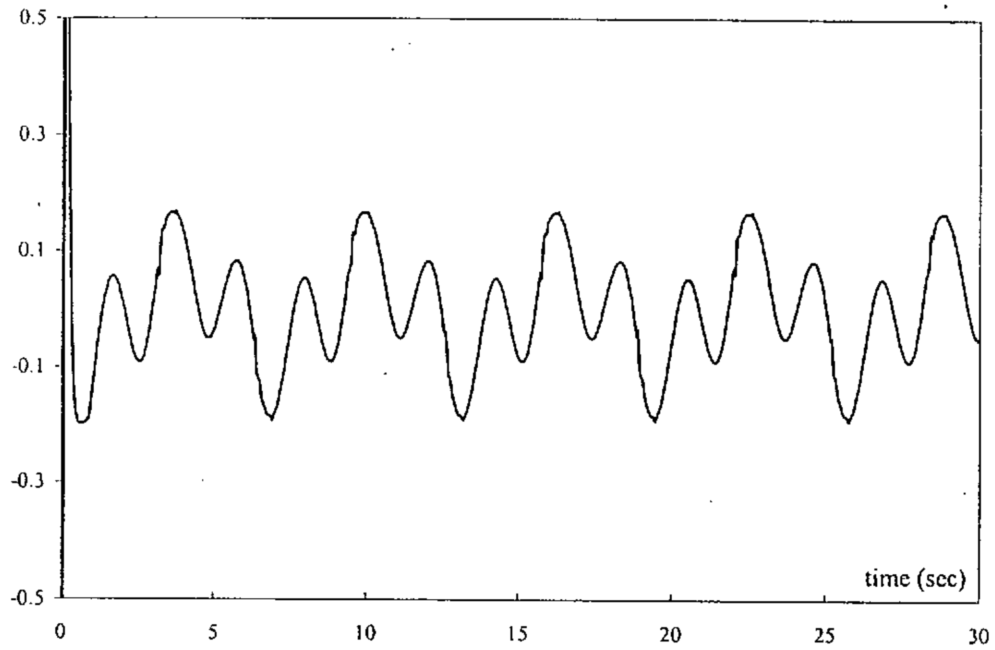
In this example, we verify at the validity of the design approach on the tracking control of a second order system. The dynamic equation of such system is given by

$$\dot{x}_1 = x_1 e^{-0.5x_1} + (1 + x_1^5)x_2 + \Delta_1(t)$$
$$\dot{x}_2 = x_1 x_2^2 + (3 + \cos(x_1 x_2))u + \Delta_2(t)$$

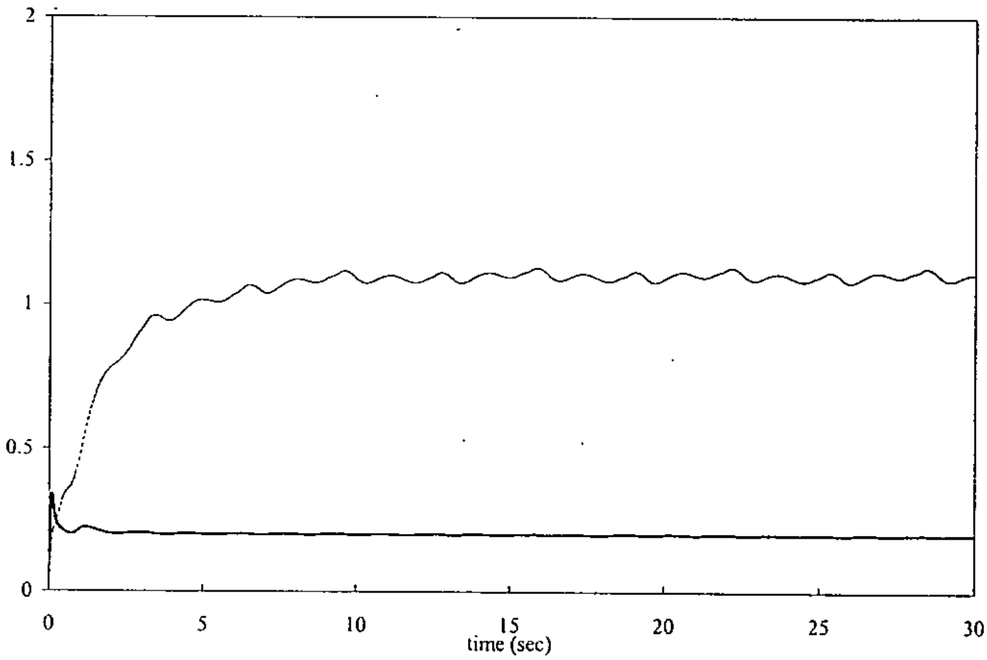
The control objective is to maintain the system to track the desired angle trajectory, $y_m = \pi/10(\sin(t) + 0.3\sin(3t))$ and $\Delta_1 = \sin(x_1)$, $\Delta_2 = 2x_1\sin(t)$, and $\phi_1 = 1$, $|\Delta_2(x,t)| \le p_2^*\phi_2(x), p_2^* = 2$ and $\phi_2 = x_1$ membership functions for system state $x_i, i = 1,2$, are chosen as in Example 5.1. The following initial conditions are controller design parameter are adopted in the simulation: $x(0) = [0.15, 0]^T$,step size 0.1s, $\Gamma_{11} = 70$, $\Gamma_{21} = 10$, $\Gamma_{12} = \Gamma_{22} = 8$, $\delta_{11} = \delta_{21} = 0.03$, $\delta_{12} = \delta_{22} = 0.3$, $\psi_1^0 = \psi_2^0 = 0.2$, $\theta_1^0 = \theta_2^0 = 0.4$ and $\lambda_1 = \lambda_2 = 0.5$. Figure 9.1 shows the simulation results. It can be seen that the tracking performance can achieve



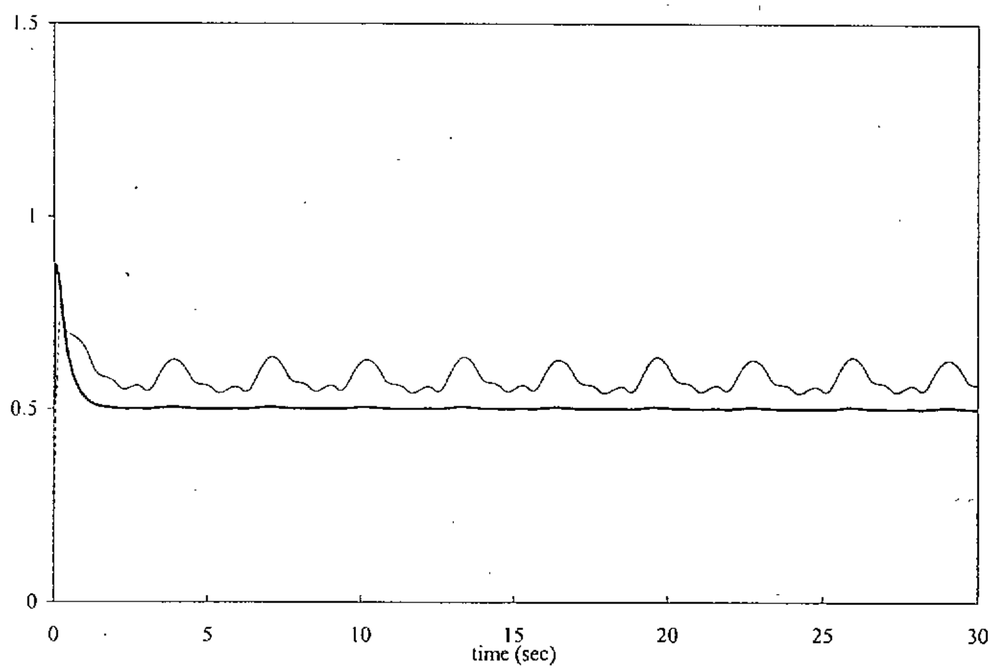
(a)


0.5
0.3
0.1
-0.1
-0.3
-0.5
0
5
10
15
20
25
30
time (sec)

(b)


2
1.5
1
0.5
0
0
5
10
15
20
25
30
time (sec)

(c)



(d)

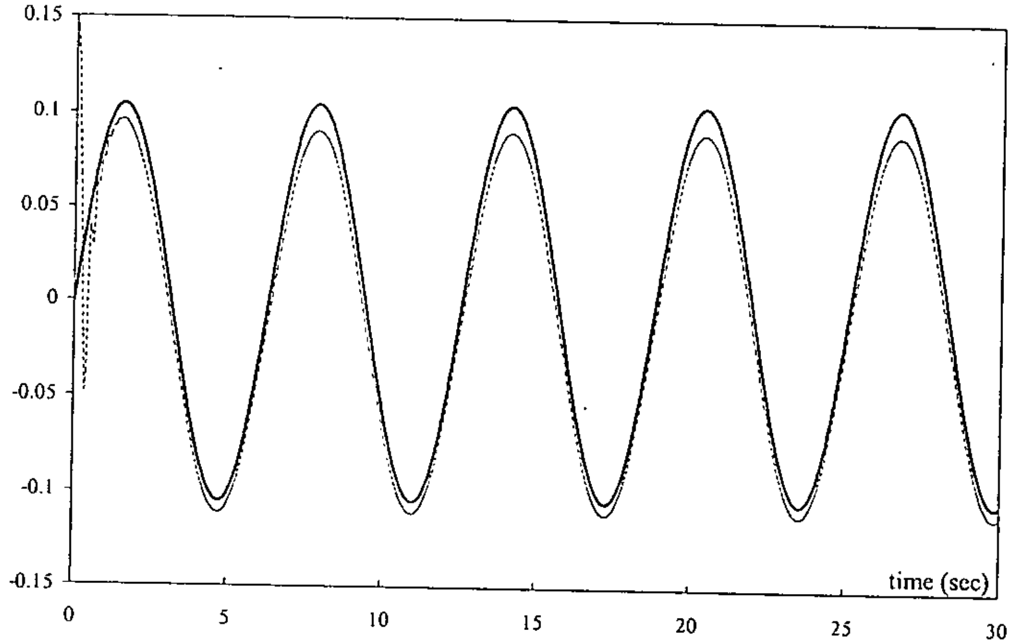
Figure 9.1 Simulation results of robust adaptive fuzzy control: (a) desired output y_m (solid line) and system output x_1 (dash line) (b) control signal u (c) trajectory of gains ψ_1 (solid line) and $\hat{\theta}_1$ (dash line) (d) trajectory of gains ψ_2 (solid line) and $\hat{\theta}_2$ (dash line).

***Example 9.2*: Pendulum system with motor**

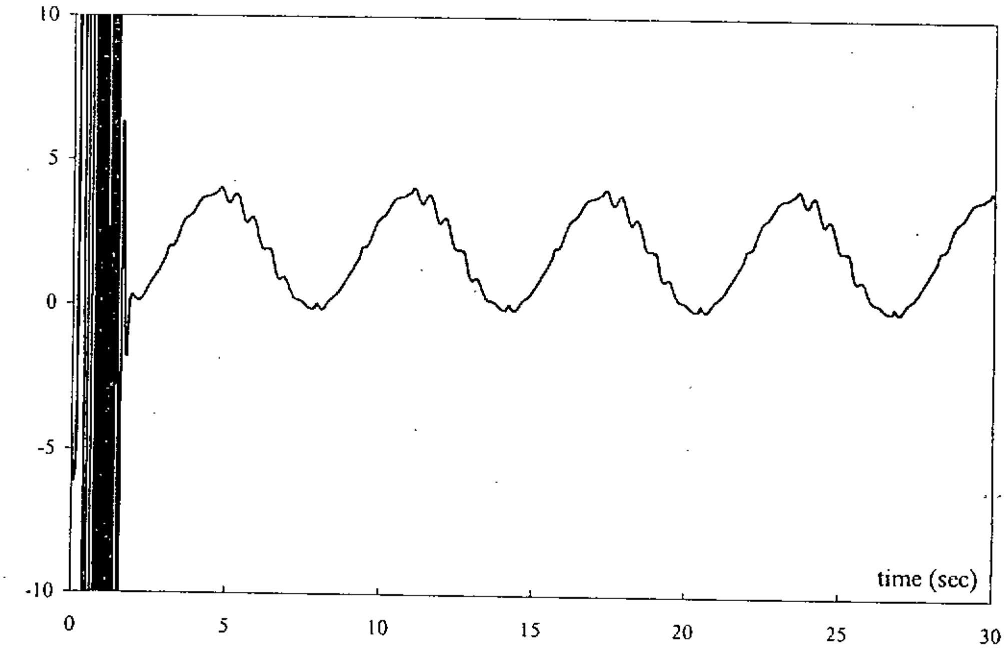
In this example, we verify at the validity of the design approach on the tracking control of a pendulum systems with motor. The dynamic equation of such system is given by [Gutman 1979, Kwan 1995].

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 - 11\sin(x_1) - 2\cos(x_1) + \Delta_1(t) \\ \dot{x}_3 &= -x_3 + u + \Delta_2(t) \end{aligned}$$

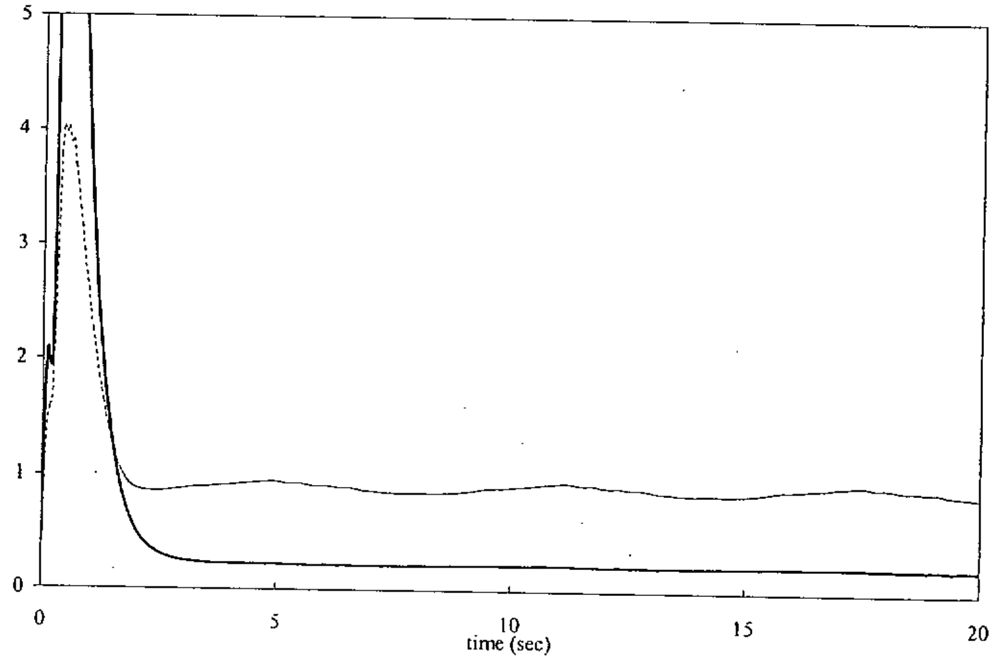
The control objective is to maintain the system to track the desired angle trajectory, $y_m = 0.1\sin(t)$ and $\Delta_1(t) = \sin(t)$, $\Delta_2(t) = 2\sin(x_1)$. The membership functions for system state x_i, $i = 1,2,3$, are chosen as in Example 9.1. The following initial conditions are controller design parameter are adopted in the simulation: $x(0)=[0.15,0]^T$,step size 0.1s, $\Gamma_{21}=70$, $\Gamma_{31}=2$, $\Gamma_{22}=8$, $\Gamma_{32}=2.5$, $\delta_{21}=\delta_{31}=0.01$, $\delta_{22}=\delta_{32}=0.1$, $\psi_2^0=\psi_3^0=0.1$, $\theta_2^0=\theta_3^0=0.3$ and $\lambda_2=\lambda_3=0.2$. Figure 9.2 shows that the system output converges to a small neighborhood around zero.



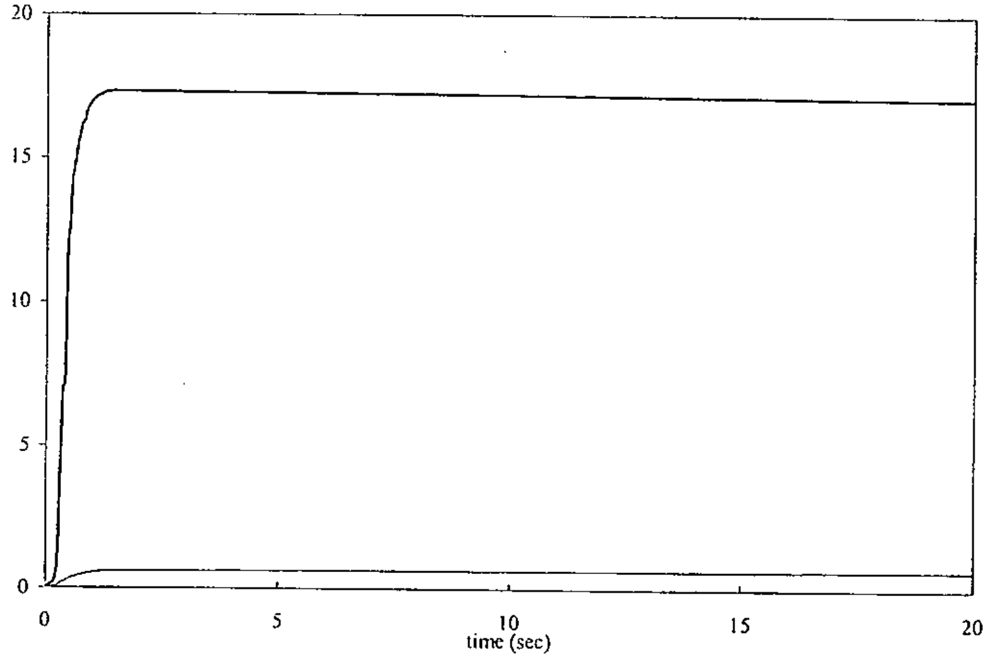
(a)



(b)



(c)



(d)

Figure 9.2 Simulation results of robust adaptive fuzzy control applied to the pendulum systems with motor: (a) desired output y_m (solid line) and system output x_1 (dash line) (b) control signal u (c) trajectory of gains ψ_2 (solid line) and $\hat{\theta}_2$ (dash line) (d) trajectory of gains ψ_3 (solid line) and $\hat{\theta}_3$ (dash line).

Example 9.3**: One-Link Robot Tracking**

In this example, we consider a one-link manipulator with the inclusion of motor dynamics. The dynamic equation of such system is given by

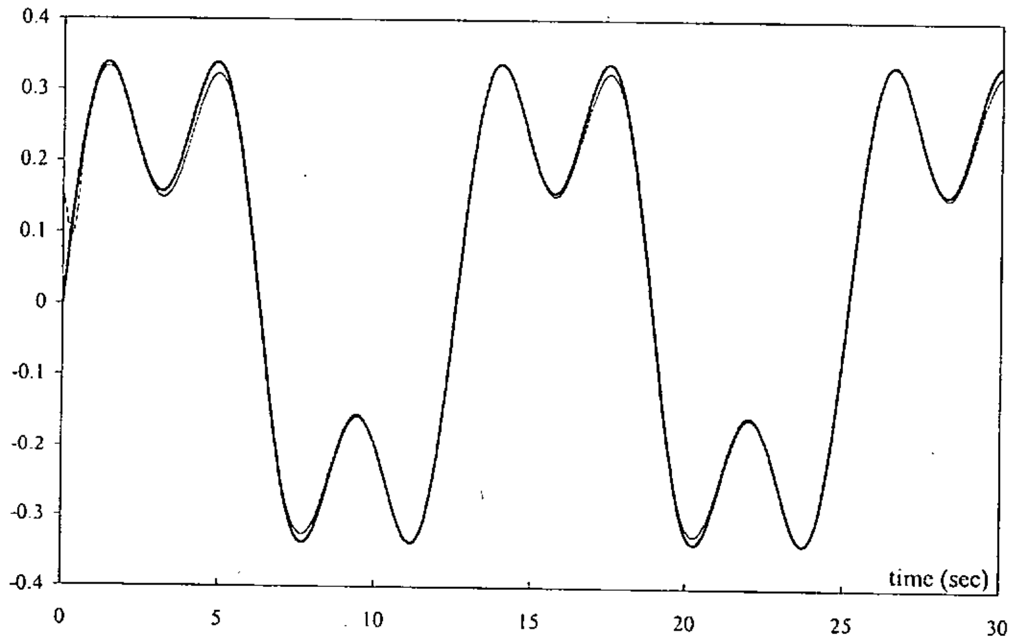
$$\begin{aligned} D\ddot{q} + B\dot{q} + N\sin(q) &= \tau + \tau_d \\ M\dot{\tau} + H\tau &= u - K_m\dot{q} \end{aligned}$$

where $q, \dot{q}, \ddot{q}$ denote the link position, velocity and acceleration, respectively. τ and $\dot{\tau}$ are the motor shaft angle and velocity. τ_d represents the torque disturbance. u is the control input used to represent the motor torque. Above equation can be expressed in the form (9.1) by noting that

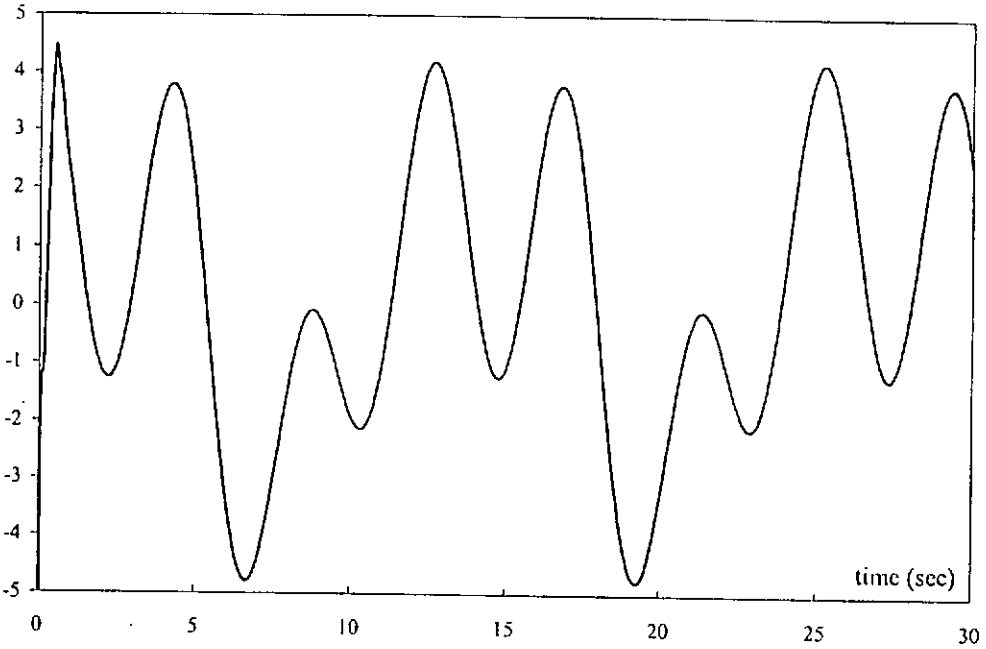
$x_1 = q,\ \ x_2 = \dot{q},\ \ x_3 = \tau$

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -\frac{B}{D}x_2 - \frac{N}{D}\sin x_1 + \frac{1}{D}(x_3 + \tau_d)$$
$$\dot{x}_3 = -\frac{H}{M}x_3 - \frac{K_m}{M}x_2 + \frac{u}{M}$$

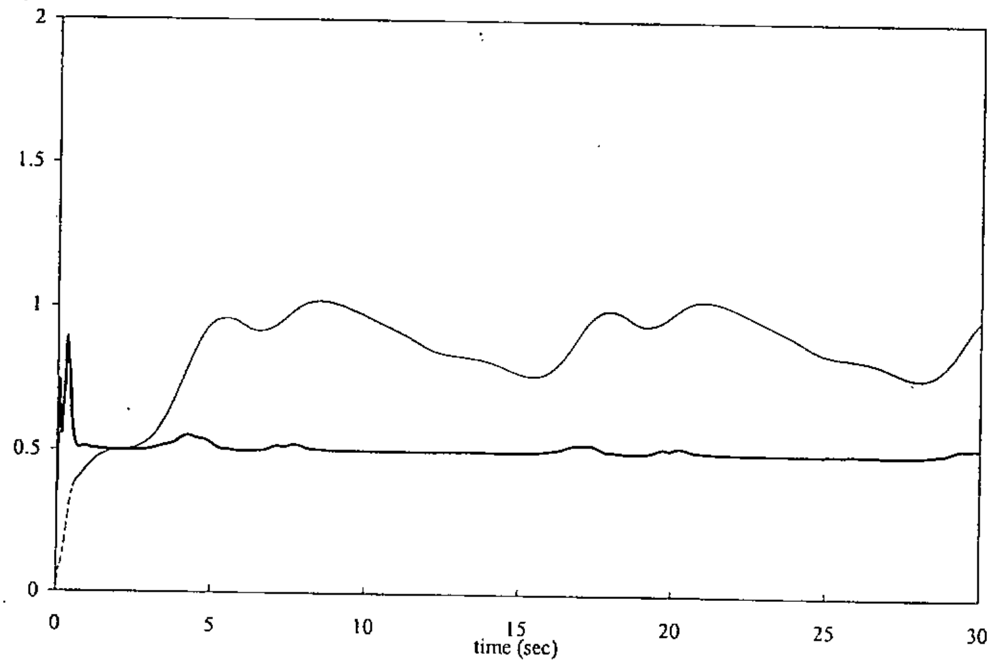
The control objective is to maintain the system to track the desired angle trajectory, $y_m = \pi/10(\sin(0.5t) + 0.5\sin(1.5t))$ and $\tau_d = 2\sin(t)$ .The membership functions for system state $z_i, i = 1,2$, are selected as example 9.1. The following initial conditions are controller design parameter are adopted in the simulation: $x(0) = [0.15,0]^T$,step size 0.1s, $\Gamma_{21} = 70$, $\Gamma_{31} = 2.5$, $\Gamma_{22} = 2$, $\Gamma_{32} = 1.5$, $\delta_{21} = \delta_{31} = 0.3$, $\delta_{22} = \delta_{32} = 0.2$, $\psi_2^0 = \psi_3^0 = 0.1$, $\theta_2^0 = \theta_3^0 = 0.5$ and $\lambda_2 = \lambda_3 = 0.4$. Figure 9.3 shows that the system output converges to a small neighborhood around zero.



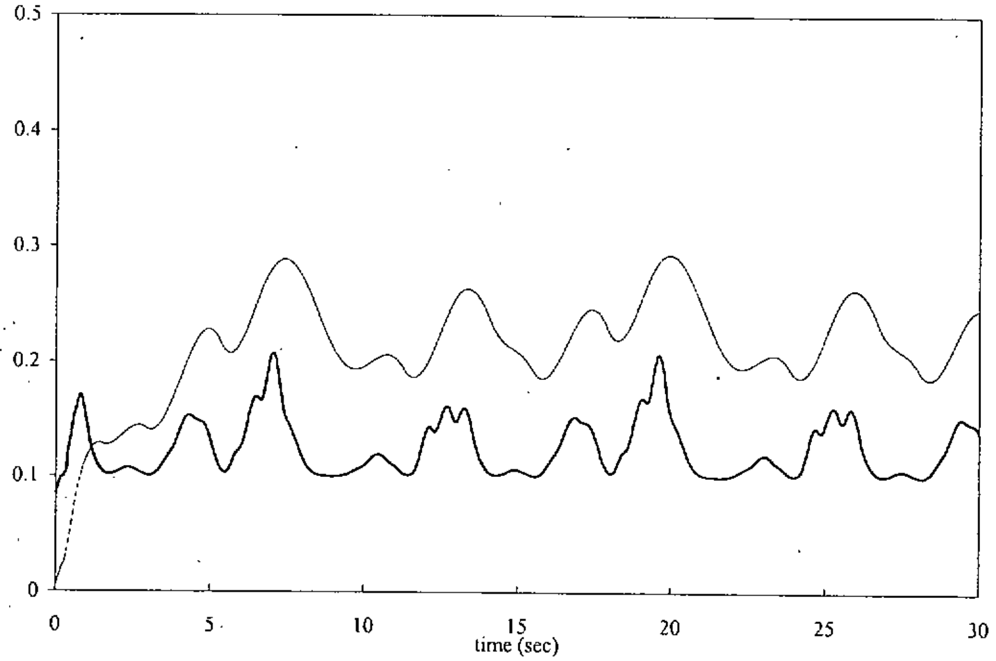
(a)


5
4
3
2
1
0
-1
-2
-3
-4
-5
0
5
10
15
20
25
30
time (sec)

(b)


2
1.5
1
0.5
0
0
5
10
15
20
25
30
time (sec)

(c)



(d)

Figure 9.3 Simulation results of robust adaptive fuzzy control applied to the one-link manipulator with the inclusion of motor dynamics (a) desired output y_m (solid line) and system output x_1 (dash line) (b) control signal u (c) trajectory of gains ψ_2 (solid line) and $\hat{\theta}_2$ (dash line) (d) trajectory of gains ψ_3 (solid line) and $\hat{\theta}_3$ (dash line)

9.5 Conclusions

In this chapter, by combining backstepping technique with small gain theorem and using TS type fuzzy systems to approximate unknown functions, adaptive fuzzy control for a class of uncertain nonlinear systems in strict-feedback form is developed. Moreover, by utilizing a special property of the affine term, the developed scheme avoids the controller singularity problem completely. All signals in the closed-loop system is guaranteed to be uniformly ultimately bounded and the steady state tracking error can be made arbitrarily small. Finally, the proposed method has been applied to control three nonlinear systems example to track a reference trajectory. The simulation results show that the adaptive controller can achieve desired performance.

CHAPTER TEN

CONCLUSIONS AND FUTURE WORK

This thesis has reported the studies undertaken to construct a systematic framework for the design of adaptive soft-computing based control schemes for linear and nonlinear systems. The work reported in the preceding chapters outlined the direction of designing hybrid adaptive controllers by exploiting concepts and ideas from the domains of linear, nonlinear and soft computing methodologies. The new paradigm provides a strong potential for the control of ill-defined systems. By combining PID control with on-line lower modeling algorithm. A procedure for the design of adaptive control with guaranteed performance was proposed in chapter 3 and 4. In chapter 5 to 9, the control schemes were derived for a class of SISO and MIMO nonlinear system, transformable to a Byrnes-Isidori normal form and a parametric strict-feedback form. Nevertheless, without a concise methodology, the advantage or ability of the combined methodology can not be fully exploited and the performance may be degraded.

Much of the previous work has been focused on the formulation of soft-computing algorithms for the nonlinear systems. As such, those remain as heuristic and *ad hoc* techniques. The thesis, on the other hand, devotes to improve the control performance by combining soft-computing techniques and modern control techniques such as sliding mode control (SMC), H^{∞} control and backstepping control. The results are significant on broadening the class of the nonlinear systems being handled, studying the convergence, stability and improving performance and robustness of the adaptive soft-computing based systems. To verify the control schemes of the proposed soft-computing based control schemes. The adaptive control systems have been applied to control some complex plants, namely flow control systems, heating process system, inverted pendulum system, pendulum systems with motor, two-link robot arm, one-link manipulator with motor dynamics, two degree-of-freedom (DOF) helicopter. In this chapter, we provide a brief summary of this research work along with comments we were able to draw from it. We conclude the chapter by providing some suggestions on possible extensions and future developments of this thesis.

10.1 Main Contributions

This thesis has made the following contributions:

- For a class of nonlinear systems that can be linearized around an equilibrium point, a new on-line lower order modeling based on neural network is designed for PID control with parameters adaptation. The neural network has been applied to approximated higher order systems with first order plus time delay model. The relevant parameters of the systems are obtained through neural network by back-propagation (BP) algorithm. An on-line PID tuning method is then applied with guaranteed performance. Then, an adaptive scheme has been developed by combining on line lower modeling with fuzzy system, and a theorem on convergence analysis of this identification method has been studies and proved. An extensive simulation and experimental studies for a typical real-time flow rate system and temperature control system have been conducted to evaluate and compare the performance of the proposed algorithms with PI controller. The results demonstrate that the proposed schemes have the most satisfactory robust property in the presence of neglected dynamics and varying time delay.

- A stable adaptive fuzzy sliding mode control (AFSMC) is presented for a class of SISO nonlinear system in Byrnes- Isidori normal form. By combining the sliding mode control and PI control, with adaptive fuzzy systems. A chattering elimination algorithm has been proposed. The adaptive control algorithm can eliminate chattering in steady-state by removing the discontinuous control signal. Besides, zero steady-state error can be obtained. By applying Lyapunov direct method and parameter projection algorithm, it has been proved that the close-loop system is stable. Simulation results have been elaborated to compare the conventional sliding mode control of proposed control system.

- To tackle those nonlinear systems whereby only the output of the plant is available for measurement, an adaptive observer-based fuzzy control design, variable structure control (VSS) and H^{∞} disturbance attenuation theory are combined together to construct hybrid indirect adaptive observer-based robust tracking control schemes. A robust and H^{∞} control terms are added into the adaptive controller in order to compensate the system model uncertainties and external disturbances. Lyapunov direct method has been applied to give the proof of the closed-loop system is globally stable. Simulation studies have demonstrated that the algorithm can track the desired reference input asymptotically.

- A robust adaptive fuzzy control method has been proposed for a class of MIMO nonlinear system, *i.e.* robot manipulators. The adaptive control scheme is used to release the assumption of linear parameterization (nonlinearities of the robot manipulator in the forms of linear in the parameters). The proposed method combines the adaptive fuzzy algorithm and robust control technique with uncertainties bound estimation to guarantee a robust tracking performance for uncertain robotic system. The design has been proved to guarantee the closed loop stability in the sense of Lyapunov method. Simulated examples have demonstrated and compared the performance of the proposed controller with computed torque control with uncertainties and payload disturbance.

- A direct adaptive fuzzy control approaches has been proposed for a class of nonlinear MIMO systems by using Takagi-Sugeno (TS) fuzzy system. The system states of the system are not required to be available for measurement and a simple observer is designed to generate an error signal for the adaptive law. By introducing a novel kind of Lyapunov function, a new observer based fuzzy control scheme is developed, in which the cancellation of nonlinearity $g(x)$ is not needed, and completely avoid the control singularity problem. A TS fuzzy system is used to reduce the number of fuzzy rules, it was show that the implementation is simple, a few fuzzy rule controller was sufficient to achieve the control of the two degree-of-freedom (DOF) helicopter, which permits real time applications.

- By combining backstepping technique with small gain theorem, a robust adaptive fuzzy controller is developed for strict-feedback nonlinear systems, which completely eliminate the possible controller singularity problem. The Takagi-Sugeno (TS) fuzzy systems are used to approximate unknown function in the systems with few tuning parameters and robust control technique is used to compensate the system uncertainties. The Lyapunov stability method has been used to prove the robust adaptive fuzzy control scheme can achieve semi-global uniform ultimate boundedness of all the signals in the closed-loop system. Moreover, the output of the system was proven to converge to a small neighborhood of the desired trajectory. Simulation examples have demonstrated the effectiveness of the developed control scheme.

10.2 Future Research Direction

As with most research efforts, this thesis has probably triggered as many questions as it has attempted to solve. We believe that the adaptive soft-computing control, many open questions remain for future research. In the following, we present some suggestions for further development.

- *Extending the class of lower order model handled by soft-computing techniques*

  As described in chapters 3 and 4, the proposed on-line lower order modeling identification is suitable to a first-order plus time delay model (FOPTD). The next step will extend the approximating algorithm to second order plus time delay model (SOPTD). Such a model is especially useful for oscillatory systems, which are associated with the second-order-underdamped systems.

- *Output feedback control for nonlinear systems*

  In chapters 5, 7 and 9, the systems are assumed that the full state variables are available for measurement. The future work will be focused on output feedback control. In particular, the development of output feedback stable adaptive control for strict-feedback system is an important and challenging problem in the future.

- *Adaptive control for non-affine nonlinear system*

  In chapters 5 to 8, the systems under studies are affine, *i.e.* model is linear in the input variables, and adaptive control of nonlinear systems can be solved by using feedback linearization techniques. However, many physical systems, *i.e.* PH neutralization and chemical reactors, are inherently nonlinear, whose input variables can not be expressed in an affine nonlinear form. In future, research efforts can be directed to study for non-affine nonlinear systems, $\dot{x} = f(x,u)$ $y = h(x)$. They represent an important class of nonlinear systems that adaptive soft-computing based control should be investigated.

- *Control of discrete time systems*

The adaptive approaches developed for nonlinear systems are based on continuous time representation. As compared with discrete base control, design and analysis of the continuous time adaptive systems are more tractable. In particular, the elegant methods for continuous time systems are not directly applicable to discrete time systems due to noncausal problem in the controller design procedure. However, complex systems are hybrid dynamical systems that contain both discrete and continuous signals. Adaptive soft-computing based control should be investigated these class of systems.

- *Adaptation algorithm for membership functions*

In this thesis, the membership functions used to fuzzify the inputs of the fuzzy systems are static. Thus, these parameters are not adjusted. In order to improve the approximation performance, a possible way would be to derive an adaptation law to automatically tune the parameters of these membership functions as we did for the rule base of the fuzzy systems.

- *Type-2 fuzzy system for nonlinear systems*

In the development of adaptive fuzzy nonlinear control, the ordinary type of fuzzy system or type-1 fuzzy systems has been used. Another future research direction is to apply a type-2 fuzzy system [Karnik *et al.* 1999, Mendel and John 2002] to the control of nonlinear systems. In general, type-2 fuzzy systems are credited to be more insensitive to parametric and modeling imprecision then type-1 fuzzy systems. The research on type-2 fuzzy systems and its effects on adaptive system is important topic in adaptive fuzzy control field.

Finally, the author hopes that his modest efforts in the design of adaptive controllers is a contribution towards designing better controllers for improved productivity in industrial plants with subsequent effect on the quality of life for all people living on earth!

APPENDIX

Norm of vectors and functions [Vidyasagar 1993]

The class of L_p is defined by

$$\|x\|_p = (\sum_{i=1}^{n} | x_i |^p)^{1/p}, \quad p \in [1, \infty)$$
$$\|x\|_\infty = \max_{1 \le i \le n} | x_i | \quad p = \infty$$

the three commonly used norms are $\|x\|_1, \|x\|_2$ (or $\|x\|$ for simplicity) and $\|x\|_\infty$.

For a function $f : R_+ \to R^n$, the L_p norm of f is defined by

$$\|f\|_p = (\int_0^\infty | f(t) |^p \, dt)^{1/p}, \quad p \in [1, \infty)$$
$$\|f\|_\infty = \sup_{t \in [0,\infty)} | f(t) |, \quad p = \infty$$

By letting $p = 1, 2, \infty$, the corresponding normed spaces are called L_1, L_2, L_∞ respectively.

Barbalat's Lemma

Lemma A1 (Barbalat) Let the function $f : R_+ \to R$. If $\lim_{t\to\infty} f(t) = k < \infty$ and $\dot{f}(t)$ is uniformly continuous, then $\lim_{t\to\infty} \dot{f}(t) = 0$.

Corollary A1 Consider the function $f : R_+ \to R$. If $f, \dot{f} \in L_\infty$ and $f \in L_p$ for some $p \in [1, \infty)$, then $\lim_{t\to\infty} f(t) = 0$.

Corollary A2 Consider the function $f : R_+ \to R$. If $f(t)$ is uniformly continuous, such that $\lim_{t\to\infty} \int_0^\infty f(\tau) d\tau$ exists and is finite, then $\lim_{t\to\infty} f(t) = 0$.

Definition A1: Class K Functions [Khalil 2002]:

A continuous function $\alpha:[0,a)\to[0,\infty)$ is said to belong to class K, if it is strictly increasing and $\alpha(0)=0$ .It is said to belong to class K_∞ if $a=\infty$ and $\alpha(r)\to\infty$ as $r\to\infty$.

Definition A2: Class KL Functions [Khalil 2002]:

A continuous function $\beta:[0,a)\times[0,\infty)\to[0,\infty)$ is said to belong to class KL if for each fixed s the mapping $\beta(r,s)$ belongs to class K with respect to r, and for each fixed r, and for each fixed r the mapping $\beta(r,s)$ is decreasing with respect to s and $\beta(r,s)\to 0$ as $s\to 0$.

Definition A3: Stability [Khalil 2002]

Consider an autonomous system

$$\dot{x}=f(x),\quad x\in R^n \tag{A1}$$

with an equilibrium x_e. The point x_e is a stable equilibrium if and only if for all $\varepsilon>0$ there is a $\delta>0$ such that for $\|x(t_0)-x_e\|\le\delta$ it holds that $\|x(t)-x_e\|\le\varepsilon$ for all $t>t_0$.

Theorem A1: Lyapunov Stability Theorem [Khalil 2002]

Consider the dynamical system $\dot{x}=f(x)$ with an equilibrium point at origin and $D\subset R^n$ be a domain containing $x=0$. Let $V:D\to R$ be a continuously differentiable function such that

$$V(0)=0 \text{ and } V(x)>0 \text{ in } D-\{0\} \tag{A2}$$

$$\dot{V}(x)\le 0 \text{ in } D \tag{A3}$$

Then, $x=0$ is stable in the sense of Lyapunov. Moreover, if

$$\dot{V}(x)<0 \text{ in } D-\{0\} \tag{A4}$$

then, $x=0$ is asymptotically stable in D.

Definition A4:

Consider the non-autonomous system

$$\dot{x} = f(x,t), \quad x \in R^n, t \in R \tag{A5}$$

the origin $x = 0$ is the equilibrium point of (A5), $f(0,t) = 0$ for all $t \geq 0$.

Definition A5:

The equilibrium point $x = 0$ of (A5) is

i. globally uniformly stable, if there exists a class K_∞ function $\gamma(\cdot)$ such that

$$\|x(t)\| \leq \gamma(\|x(t_0)\|), \quad \forall t \geq t_0 \geq 0, \ \forall x(t_0) \in R^n \tag{A6}$$

ii. globally uniformly asymptotically stable, if there exists a class KL function $\beta(\cdot,\cdot)$ such that

$$\|x(t)\| \leq \beta(\|x(t_0)\|, t - t_0), \quad \forall t \geq t_0 \geq 0, \quad \forall x(t_0) \in R^n \tag{A7}$$

iii. globally exponentially stable, if (A7) is satisfied with $\beta(r,s) = kre^{-\alpha s}$, $k > 0, \alpha > 0$.

Theorem A2: (LaSalle-Yoshizawa) [Khalil 2002]

Consider (A5) and suppose $V : R^n \times R_+ \to R_+$ is a continuously differentiable function such that

$$\alpha_1(\|x\|) \leq V(x,t) \leq \alpha_2(\|x\|) \tag{A8}$$

$$\dot{V} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x,t) \leq -W(x) \leq 0 \tag{A9}$$

$\forall t \geq 0, \forall x \in R^n$, where α_1 and α_2 are class K_∞ functions and W is a continuous function. Then the equilibrium $x = 0$ is globally uniformly stable and $\lim_{t \to \infty} W(x(t)) = 0$. Moreover, if $W(x)$ is positive definite, then $x = 0$ is globally uniformly asymptotically stable.

Input-to-State Stable (ISS)

Consider the system

$$\dot{x} = f(x,u) \tag{A10}$$

where $\dot{x}$ is the state and u is the input.

Definition A6: system (A10) is said to be input-to-state practically stable (ISpS) if there exist a class K function γ, and class KL function β, such that, for any essentially bounded input $u(t)$ and any x_0 and a nonnegative constant d, the response of $x(t)$ are defined on $[0,\infty)$ and satisfy

$$\|x(t)\| \le \beta(\|x(0)\|,t) + \gamma(\|u_t\|_\infty) + d \tag{A11}$$

where u_t is the truncated function of u at t, when $d = 0$ in (A11), the ISpS property collapses to ISS property introduced in [Sontag 1995, 1989].

Definition A7: A continuou function V is said to be an ISpS-Lyapinov function for (A10) if there exist α_1,α_2 of class K_∞ and α_3,α_4 of class K and constant $d > 0$

$$\alpha_1(\|x\|) \le V(x) \le \alpha_2(\|x\|),\quad \forall x \in R^n \tag{A12}$$

$$\frac{\partial V(x)}{\partial x} f(x,u) \le -\alpha_3(\|x\|) + \alpha_4(\|u\|) + d \tag{A13}$$

when (A13) holds with $d = 0$, V is referred to as ISS-Lyapunov function [Sontag 1995]. Then it hold that the nonlinear L_∞ gain γ in (A11) can be evaluated as

$$\gamma(s) = \alpha_1^{-1} \circ \alpha_2 \circ \alpha_3^{-1} \circ \alpha_4(s),\quad \forall s > 0$$

Input-to-State Small Gain Condition [Jiang *et al.* 1994, 1997]

Consider a system in composite feedback form of two ISpS systems

$$\Sigma_1 : \begin{cases} \dot{x} = f_1(x, w) \\ z = H(x) \end{cases} \tag{A14}$$

$$\Sigma_2 : \begin{cases} \dot{y} = f_2(y, z) \\ w = K(y, z) \end{cases} \tag{A15}$$

there exist two constants $d_1, d_2 > 0$ and β_1, β_2 of class KL, and γ_1, γ_2 of class K such that for each w and z in supremum norm, each $x \in R^n$ and each $y \in R^m$, all the solutions $X(x; w, t)$ and $Y(y; z, t)$ are defined on $[0, \infty)$ and satisfy for all t.

$$\|H(X(x; w, t))\| \le \beta_1(\|x\|, t) + \gamma_1(\|w\|_\infty) + d_1$$

$$\|K(Y(y; z, t))\| \le \beta_2(\|y\|, t) + \gamma_2(\|z\|_\infty) + d_2.$$

under this conditions

$$\gamma_1 \circ \gamma_2 < s, \quad \forall s > 0$$

the solution of the composite system (A14) and (A15) is ISpS.

References

Alligood, K.T., Sauer, T. and Yorke, J.A. *Chaos: an introduction to dynamical systems.* Springer, New York (1997)

Åström, K.J., Hagglund, T., Hang, C.C. and Ho, W.K. Automatic tuning and adaptation for PID controllers -A survey, *IFAC Journal Control Engineering Practice*, 1(4), pp. 699-714 (1993)

Åström, K.J. and Wittenmark, B. *Adaptive Control.* Addison-Wesley Publishing Company, MA (1995)

Battiti, R. "Accelerating Back-propagation Learning, Two Optimization Methods". *Complex system*, 3, pp. 331-342 (1989)

Baik, I.C., Kim, K.H. and Youn, M.J. "Robust nonlinear speed control of PM synchronous motor using boundary layer integral sliding mode control technique". *IEEE Transactions Control System Technology*, 8, pp.47-54 (2000)

Byronic (UK): *Document for the Bytronic Process Control Rig*. Byronic (1994)

Chen, B.S., Lee, C.H. and Chang, Y.C. "H^{∞} tracking design of uncertain nonlinear SISO systems: Adaptive fuzzy approach". *IEEE Transactions on Fuzzy System,* 4, pp.32-43 (1996)

Choi, J.Y. and Farrell, J.A. "Adaptive observer backstepping control using neural networks". *IEEE Transactions on Neural Networks,* 12, pp.1103-1112 (2001)

Craig, J.J. *Introduction to Robotics.* 2nd ed. Reading, MA:Addison-Wisley (1989)

Cybenko, G. "Approximation by superposition of a sigmoidal function". *Mathematic Control Signal System*, 2 , pp. 303-310 (1989)

DeCarlo, R.A., Zak, S.H. and Matthews, G.P. "Variable structure control of nonlinear multivariable systems: A Tutorial". *IEEE Proceeding*, 76(3), pp. 212-232 (1998)

Feedback Inst. Ltd., UK, 'Manual for PT326' (1982)

Francis, B.A. *A Course in H^{∞} Control Theory*. Lecture Notes in Control Information Science. Vol. 8, Springer Verlay, Berlin. (1997)

Francis, B.A. and Zames, G. "On H^{∞} optimal sensitivity theory for SISO feedback systems". *IEEE Transactions on Automatic Control,* 29, pp.9-16 (1984)

Freeman, J.A. and Skapura, D.M. *Neural Networks: Algorithms, Applications and Programming Techniques*, Addison-Wesley Publishing Company, Inc. (1991)

Ge, S.S and Wang, J. "Robust adaptive neural control for a class of perturbed strict feedback nonlinear systems". *IEEE Transactions on Neural Networks,* 13(6), pp.1409-1419 (2002)

Ge, S.S., Hang, C.C. and Woon, L.C. "Adaptive neural network control of robot manipulators in task space". *IEEE Transactions Industrial Electronic*, 44, pp. 746-752 (1997)

Gutman, S. "Uncertain dynamical system-A Lyapunov min max approach". *IEEE Transactions on Automatic Control,* 24, pp.437-443 (1979)

Hagglund, T. "A predictive PI controller for processes with long dead time" *IEEE Control System*,12 (1), pp. 57-60 (1992)

Hang, C.C., Astrom, K.J. and Ho, W.K. "Refinements of the Ziegler-Nichols tuning formula". *Proceeding of IEE – Part D*, 138(2), pp. 111-118 (1991)

Hebb, D.O. *The organization of behavior.* John Wiley, New York (1949)

Ho, W.K., Lee, T.H., X,W., Zhou, J.R. and Tay, E.B. "The direct Nyquist array design of PID controllers". IEEE Transactions on Industrial Electronics, 47(1), pp. 175-185 (2000)

Hoo, K. A., Sinzinger, E. D. and Piovoso, M. J. "Improvements in the predictive capability of neural networks". *Journal of Process Control,* 12(1), pp.193-202 (2002)

Hsu, L., Lizarralda, F. and A.D. De Araujo. "New results on output feedback variable structure model reference adaptive control: Design and Stability Analysis". *IEEE Transactions on Automatic Control,* 42, pp.386-393 (1997)

Humusoft, 'CE150 Helicopter Model User's Manual' (1989)

Ioannou, P.A. and Sun, J. *Robust Adaptive Control.* Prentice Hall, Upper Saddle River, NJ. (1996)

Isidori, A. *Nonlinear Control Systems*, 3rd ed. New York: Springer-Verlag (1995)

Jacobs, O.L.R, Hewkin, P.F. and While, C. "Online computer control of pH in an industrial process". *Proceeding of IEE – Part D*, 127(4), July (1980)

Jamshidi, M., Vadiee, N. and Ross, J. *Fuzzy Logic and Control Software and Hardware Applications.* Prentice Hall Inc (1993)

Jiang, Z.P. and Mareels, I. "A small-gain control method for nonlinear cascaded systems with dynamic uncertainties". *IEEE Transactions on Automatic Control,* 42, pp.292-308 (1997)

Jiang, Z.P., Teel, A.R. and Mareels, I. "Small-gain theorem for ISS systems and applications". *Mathematics of control, signals, and systems*, 7, pp.95-120 (1994)

Kanellakopoulos, I., Kokotovic, P.V. and Morse, A.S. "Systematic design of adaptive controllers for feedback linearizable systems". *IEEE Transactions on Automatic Control,* 36, pp.1241-1253 (1991)

Kang, H.J., Lee, H. and Park, M. "Comments on H^{∞} tracking design of uncertain nonlinear SISO systems: Adaptive fuzzy approach". *IEEE Transactions on Fuzzy System,* 6, pp. 605-606 (1998)

Karnik, N.N., Mendel, J.M. and Liang, Q. "Type-2 fuzzy logic systems" *IEEE Transactions on Fuzzy System,* 7, pp.643-658 (1999)

Khalil, H.K. *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ: Prentice Hall (2002)

Kim, S.W. and Lee, J.J. "Design of a fuzzy controller with fuzzy sliding surface". *Fuzzy Sets and Systems*, 71, pp. 359-367 (1995)

Krstic, M., Kanellakopoulos, I. and Kokotovic, P.V. *Nonlinear and Adaptive Control Design.* New York:Wiley (1995)

Kwan, C.M. "Sliding mode control of linear systems with mismatched uncertainties". *Automatica*, 31, pp. 303-307 (1995)

Kwan, C. and Lewis, F.L. "Robust backstepping control of nonlinear systems using neural networks". *IEEE Transactions on Systems, Man, and Cybernetics-Part A*, 30, pp. 753-765 (2000)

Lee, C.H. and Teng, C.C. "Identification and control of dynamic systems using recurrent fuzzy neural networks". *IEEE Transactions Fuzzy System*, 8(4), pp.349-366 (2000)

Li, Y., Qiang, S. and Kaynak, O. "Robust and adaptive backstepping control for nonlinear systems using RBF neural networks". *IEEE Transactions on Neural Networks*, 15(3), pp.693-701 (2004)

Linkens, D.A.and Nyongeso, H.O. "Learning systems in intelligent control: an appraisal of fuzzy, neural and genetic algorithm control applications". *IEE Proceedings Control Theory Applications*, 143(3), pp. 367-386 (1996)

Mamdani, E.H. "Applications of fuzzy algorithms for control of simple dynamic plant". *Proceedings of the IEE 121*, 12, pp. 1585-1588 (1974)

Marino, R. and Tomei, P. "Global adaptive output-feedback control of nonlinear systems, Part I: linear parameterization; Part II: nonlinear parameterization". *IEEE Transactions on Automatic Control*, 38, pp.17-49 (1993)

Marino, R and Tomei, P. *Nonlinear Control Design: Geometric, Adaptive and Robust.* Prentice Hall. (1995)

McCulloch, W.S. and Pitts, W. "A logical calculus of the ideas immanent in nervous activity" *Bulletin of Mathematical Biophysics*,9, pp. 127-147 (1943)

Mendel, J.M. "Fuzzy logic systems for engineering: A tutorial". *Proceedings of the IEEE,* 83, pp. 345-377 (1995)

Mendel, J.M. and John, R.I. "Type-2 fuzzy sets made simple". *IEEE Transactions on Fuzzy System,* 10, pp.117-127 (2002)

Miller, D.E. and Davison, E.J. "An adaptive controller which provides an arbitrarily good transient and steady state response". *IEEE Transactions on Automatic Control,* 36, pp.68-81 (1991)

Miller, J.A., Lopez, A.M., Smith, C.L. and Murrill, P.W. "A comparison of controller tuning rules" *Control Engineering 14.* December,72 (1967)

Mizumoto, M. "Realization of PID controls by fuzzy control methods". *Fuzzy Sets and Systems,* 70, pp. 171-182 (1995)

Moon, Un-C. and Lee, K.Y. "Hybrid algorithm with fuzzy system and conventional PI control for the temperature control of TV glass furnace". *IEEE Transactions on Control Systems Technology,* 11(4), pp.548-554 (2003)

Narendra, K.S. *Stable adaptive systems*. Prentice Hall, Englewood Cliffs, NJ (1989)

Narendra, K.S. and Balakrishnan, J. "Adaptive control using multiple models". *IEEE Transactions on Automatic Control*, 42(2), pp.171-187 (1997)

Ordonez, R. and Passino, K.M. "Adaptive fuzzy control: experiments and comparative analysis". *IEEE Transactions on Fuzzy System,* 5(2), pp.167-188 (1997)

Ortega, R. and Spong, M.W. "Adaptive motion control of rigid robots: A tutorial". *Automatica*, 25, pp. 877-888 (1989.)

Ordonez, R. and Passino, K.M. "Stable multi-input multi-output adaptive fuzzy/neural control". *IEEE Transactions on Fuzzy System*,7, pp. 345-353 (1999)

Palm, R. "Robust control by fuzzy sliding mode". Automatica, 30(9), pp.1429-1437 (1994)

Polycarpou, M.M. and Ioannou, P.A. "A robust adaptive nonlinear control design". *Automatica*, 32(3), pp. 423-427 (1996)

Procyk, T. and Mamdani, E. "An linguistic self organizing process controller". *Automatica*, 15(1), pp. 15-30 (1979)

Rad, A.B., Lo, W.L. and Tsang, K.M. "Self-tuning PID controller using Netwon Raphson search method". *IEEE Transactions Industrial Electronics*, 44 (5), pp. 717-725 (1997)

Ray, L.R. and Stegenl, R.F. "Monte carlo approach to the analysis of control system robustness". *Automatica*, 29(1), pp. 229-236 (1993)

Rosenblatt, F. "The perceptron: aprobabilistic model for information storage and organization in the brain". *Psychological Review*, 65, pp.386-408 (1958)

Rough, W. "Design and analysis of gain-scheduled control using local controller networks". *International Journal of Control*, 66, pp. 619-651 (1997)

Sastry, S. S. *Nonlinear Systems: Analysis, Stability, and Control*. New York: Springer (1999)

Sastry, S. S. and Bodson, M. *Adaptive Control: Stability, Convergence and Robustness*. Prentice Hall. (1989)

Sastry, S. S. and Isidori, A. "Adaptive control of linearizable systems". *IEEE Transactions on Automatic Control*, 34, pp.1123-1131 (1989)

Schaft, A.J. van der. "L_2-gain analysis of nonlinear systems and nonlinear state feedback H^{∞} control," *IEEE Transactions Automatic Control*, 37, pp. 770-784 (1992)

Slotine, J.J.E. and Li, W.P. *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prectice Hall (1991)

Slotine, J.J.E. "Sliding controller design for nonlinear systems". *International Journal of Control*, 40, pp. 421-434 (1984)

Slotine, J.E. and Li, W. "Composite adaptive control of robot manipulator". *Automatica*, 25, pp. 509-519 (1989)

Smith, C.A. and Corripio, A.B. *Principles and Practice of Automatic Process Control.* John Wiley & Sons, 2nd. (1997)

Smith, C.L. *Digital Computer Process Control.* U.S.A: International Textbook Company. (1972)

Sontag, E.D. "On the input-to-state stability property". *European Journal of Control*, 1, pp. 24-36 (1995)

Sontag, E.D. "Smooth stabilization implies coprime factorization". *IEEE Transactions on Automatic Control,* 34, pp.1411-1428 (1989)

Song, Y.H., Johns, A. and Aggarwal. *Computational Intelligence Applications to Power Systems.* Science Press, Kluwer Academic Publishers (1996)

Spong, M.W. and Vidyasagar, M. *Robot Dynamics and Control.* New York:Wiley (1989)

Stepanenko, Y. and Su, C.Y. "Variable structure control of robust manipulators with nonlinear sliding manifolds". *International Journal of Control*, 58, pp.285-300 (1993)

Su, C.Y. and Leung, T.P. "A sliding mode controller with bound estimation for robot manipulators". *IEEE Transactions Robotics Automation.* 9, pp. 208-214 (1993)

Sugeno, M. "An introductory survey of fuzzy control". *Information Sciences*, 36,pp. 59-83 (1985)

Sun, J. "A modified model reference adaptive control scheme for improved transient performance". *IEEE Transactions on Automatic Control,* 38(8), pp.1255-1259 (1993)

Sun, J. "A hybrid adaptive control scheme using sample data and intersampling compensation". *Automatica,* 33, pp. 203-207 (1997)

Sun, D. and Hoo, K.A "Dynamic transition control structure for a class of SISO nonlinear systems". *IEEE Transactions on Control Systems Technology*, 7(5), pp.622-629 (1999)

Sundaresan, K.R. and Krishnaswamy, P.R. "Estimation of Time Delay Time Constant Parameters in Time, Frequency, and Laplace Domains". *The Chemical Journal of Chemical Engineering*, 56, April (1978)

Takagi, T. and Sugeno, M. "Fuzzy identification of systems and its application to modeling and control". *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 15, pp.116-132 (1985)

Tanaka, K. And Sugeno, M. "Stability analysis and design of fuzzy control systems". *Fuzzy Sets and Systems*, 45, pp. 135-156 (1992)

Tanaka, K., Ikeda, T. and Wang H.O. "Robust stabilization of a class of uncertain nonlinear system via fuzzy control: Quadratic stability, H^{∞} control theory and linear matrix inequalities". *IEEE Transactions on Fuzzy System,* 4(1), pp.1-13 (1996)

Terano, T., Asai, K. and Sugeno, M. *Applied Fuzzy Systems*. Boston:AP Professional (1994)

Tong, S., Li, H.X. and Wang, W. "Observer-based adaptive fuzzy control for SISO nonlinear systems". *Fuzzy Sets and System*, In Press, Corrected Proof.

Tsang, K.M. and Rad, A.B. "A new approach to auto-tuning of PID controllers". *International Journal of Systems Science*, 26(3),pp. 639-658 (1995)

Utkin, V.I. "Variable structure systems with sliding modes: A survey". *IEEE Transactions on Automatic Control,* 22, pp. 212-222 (1977)

Vidyasagar. M. *Nonlinear systems analysis*. Prentice Hall, Englewood Cliffs, NJ (1993)

Wang, L.X. and Mendel, J.M. "Fuzzy basis functions, universal approximation, and orthogonal least-squares learning". *IEEE Transactions on Neural Networks,* 3(5), pp.807-814 (1992)

Wang, L.X. "Stable adaptive fuzzy control of nonlinear systems". *IEEE Transactions on Fuzzy System*, 1, pp.146-155 (1993)

Wang, L.X. *Adaptive Fuzzy Systems and Control*. Prentice Hall, Englewood Cliffs, NJ (1994)

Wang, L.X. *A course in fuzzy systems and control*. Prectice Hall, Englewood Cliffs, NJ. (1997)

Wang, W.Y., Chan, M.L., Lee, T.T. and Liu, C.H. "Adaptive fuzzy control for strict-feedback canonical nonlinear systems with H^{∞} tracking performance". *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 30(6), pp.878-885 (2000)

Yang, Y., Feng, G. and Ren, J. "A combined backstepping and small gain approach to robust adaptive fuzzy control for strict-feedback nonlinear systems". *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 34(3), pp.406-420 (2004)

Yao, B. and Tomizuka, M. "Adaptive robust control of SISO nonlinear systems in a semi-strict feedback form". *Automatica*, 35(5), pp. 893-900 (1997)

Ying, H., Siler, W. and Buckley, J.J. "Fuzzy control theory: A nonlinear case". *Automatica*, 26(3), pp. 513-520 (1990)

Zadeh, L.A. "Fuzzy Sets". *Information and Control*, 8, pp. 338-353 (1965)

Zadeh, L.A. "Fuzzy logic, neural networks and soft-computing". *Communication of ACM*, 37, pp. 77-84 (1994)

Zames, G. "Feedback and optimal sensitivity: model reference transformations, multiplicative seninors, and approximate inverses". *IEEE Transactions on Automatic Control*, 26(1), pp.301-320 (1981)

Zhang, Y., Peng, P.Y. and Jiang, Z.P. "Stable neural controller design for unknown nonlinear systems using backstepping". *IEEE Transactions on Neural Networks*, 11(6), pp.1347-1360 (2000)

Ziegler, J.G. and Nichols, N.B. "Optimum Settings for Automatic controllers". *Transactions ASME* Nov., pp. 759-768 (1942)