INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Control of Nonlinear Systems Using Sugeno Fuzzy Approximators

Mohanad Alata

A Thesis

in

The Department

of

Department of Mechanical Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University

Montréal, Québec, Canada

January 2001

ABSTRACT

Control of Nonlinear Systems Using Sugeno Fuzzy Approximators

Mohanad Alata, Ph.D. Student

Concordia Unviersity, 2001

This thesis deals with the issue of controlling nonlinear systems by integrating available classical as well as modern tools such as fuzzy logic and neural networks. The proposed approaches throughout this thesis are based on the well known first-order Sugeno fuzzy system. To achieve a better understanding of the approximation and interpolation capabilities of Sugeno fuzzy system, the influence of the fuzzy set parameters and the reasoning method on the interpolation function of the fuzzy system is investigated. Control of nonlinear system based on known dynamic is considered first. A fuzzy gain scheduling approach is developed. The proposed approach is based on quasi-linear dynamic models of the plant. Classical optimal controllers for each set of operating conditions were developed. These controllers are used to construct a single fuzzy-logic gain scheduling-like controller. Adaptive-neuro-fuzzy inference system was used to construct the rules for the fuzzy gain schedule. This will guarantee the continuos change in the gains as the system parameters change in time or space. This procedure is systematic and can be used to design controllers for many nonlinear systems. Also, a modeling approach of some known types of static nonlinearities is proposed. Control of nonlinear system with unknown dynamic is also considered. An adaptive feedback control scheme for the tracking of a class of continuous-time plants is presented. A parameterized Sugeno

fuzzy approximator is used to adaptively compensate for the plant nonlinearities. All parameters in the fuzzy approximator are tuned using a Luapunov-based design. In the fuzzy approximator, a first-order Sugeno consequent is used in the IF-THEN rules of the fuzzy system, which has a better approximation capability compared with that of a constant consequent. Global boundedness of the adaptive system is established. Finally, simulation and experimentation are used to demonstrate the effectiveness of the proposed controllers.

I dedicate this thesis to *Mohammad* and *Mohammad Alata*

ACKNOWLEDGEMENTS

Many different people have made this thesis possible, none more so than my supervisors, Dr. Kudret Demirli and Dr. Akif Bulgak; I am deeply grateful for all their support and encouragement during the course of this work. Also, I would like to gratefully acknowledge the help provided by Dr. Mohammad Al Jarrah from Jordan University of Science and Technology and Dr. C.-Y. Su from Concordia University. I feel very fortunate to have worked with such a great team and I look forward to many more years of interaction.

I am indebted to my colleagues in the Fuzzy System Research Laboratory. In particular my best friend Mohammad Molhim who helped me a lot and made sure that I did not dry up when the end was near. I am also grateful for the support of my friends Nasser and Malak who made it easy for me and my family to enjoy life in Montreal.

My special thanks to my parents, brothers and sister for the love and encouragement they have given me all my life and specially to reach this milestone of my life by sacrificing their happiness. Thanks dad, mom, Ahmad, Maysa and Belal.

Last, but not least, never ending thanks to my beautiful wife Dr. Azal Eisa, whose love, invaluable support, patience and understanding made the course of this work possible and successful, as well as taking care of the kids in my absence. Of course, I can not leave out the kids; thanks Mohammad, Mustafa and Ahmad for adding a little spice to life.

The financial support provided by the Canadian International Development Agency (CIDA) and by my supervisors is greatly acknowledged.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A, B, C Arbitrary fuzzy sets

a, b, c Arbitrary fuzzy sets

$\mu_A(x)$ Membership value of fuzzy set A at support point x

$h(A)$ Core of A

supp(A) Support of A

$\bar{A}$ Complement of set A

$\forall$ 'for all'

sup Supremum

AND Linguistic conjunction

OR Linguistic disjunction

$\cap$ Set intersection

$\cup$ Set union

T T-norms

S T-conorms

T_w Drastic-product intersection

S_w Drastic-sum union

T_{bold} Bold intersection

S_{bold} Bold union

$T_{\min}$, $\wedge$, min Minimum-intersection

$S_{\max}$, $\vee$, max Maximum-union

$T_{prod.}$ Algebraic product-intersection

S_{sum} Algebraic sum-union

IF-THEN Linguistic rule

I Implication function

$\rightarrow$ Implication

I_{K-D} Kleene-Diense implication

I_{Re} Reichenbach implication

I_L Lukasiewicz implication

$I_{G\ddot{o}}$ Gödel implication

I_{Go} Goguen implication

I_{za} Zadeh implication

e Error

$\dot{e}$ Change in error

F Force

Q Semidefinite weight matrix

R Semidefinite weight matrix

P_i Potential or density measure

exp Exponential

c_k Cluster center

η Squash factor

r_a Cluster radius

$\bar{\epsilon}$ Accept ratio

$\underline{\epsilon}$ Reject ratio

τ Torque

q_1, q_2 Joint angles

$u(t)$ Control input

θ_i A set of unknown parameters

$Y_j(x)$ A set of known regressors

x_d Desired trajectory

w Functional consequent

σ Variance

ξ Center

$\varepsilon_f(x)$ Approximation error

d_f Residual term

$o(.)$ Sum of higher order terms

$s(t)$ Error metric

Γ Positive definite matrix

M Mass of the cart

m_p Mass of the pendulum

l_p Length of the pendulum

V_{in} Input voltage

K_m Motor torque constant

K_g Internal gear ratio

R_m Armature resistance

r Motor gear radius

Chapter 1

Introduction

In this chapter, motivations and objectives of this work are presented and discussed.

1.1 Motivation

New ideas and techniques for control are being introduced by the researchers in the field of intelligent control. In many cases of more complex and ill-structured problems, the conventional technology is not sufficiently strong to represent and implement the knowledge needed for a powerful solution. Intelligent control technology has been considered seeking hybrid solutions by enhancing control engineering where it is needed and where it makes sense. Dynamical systems are in general complex and nonlinear. Some systems are well defined that can be modelled easily while there are systems which are difficult to model such as a process planing or product

control. Conventional control methods are in general based on mathematical models that describe the system response to its inputs. Even if a relatively accurate model of a dynamic system can be developed, it is often too complex to use in controller development. Thus, model free controllers, specifically PIDs are widely used in practice. The usual application to optimize the system performance controlled by a PID is to tune the PID coefficients. This approach may be acceptable as long as the system parameters are not varying or do not display nonlinearities.

Some robust control methods, such as H_∞, have been developed to deal with parametric uncertainties and disturbances. But they still require a low order of the system and knowledge of the disturbance variations, and they are computationally difficult. An alternate approach to control complex, nonlinear and ill-defined systems is the use of fuzzy control. Fuzzy control provides a formal methodology for representing a human's heuristic knowledge about how to control a system. In this research, modelling and control approach of dynamic systems that can be well defined and modelled using mathematical equations, is developed. Also, modelling of some known simple and hard nonlinearities is discussed. Finally, control of systems with unknown dynamics or are difficult to transform into closed form mathematical model (equations of motion), are considered in this study.

1.1.1 Nonlinear system with known dynamics

The vast majority of control design techniques are based upon a mathematical model of the system to be controlled. These models allow the use of analytical tools to guarantee that performance specifications will be met, but these guarantees only hold as long as the underlying models are valid. Thus, many systems require complex control strategies to perform their designed tasks. Furthermore, dealing with the entire dynamic range of operation can bring a control design technique to its knees. Varying parameters and uncertainty from sensor noise, and disturbances, ensure that the model is never perfect. It seems that the most useful way of dealing with nonlinearity of the model is to linearize it about some point, p, in its operating range. If the model is "smooth", the linearized equation will accurately represent the true system in some "sufficiently small" region about the equilibrium point p in the parameter space. Once linearization is done, we have all available tools for linear analysis, and the solution within this neighbourhood can be obtained by one of linear control synthesis techniques, such as linear quadratic regulators (LQR). However, one must still deal with varying parameters over the entire operating range. Varying the model's parameters may remove the system from within this region of model validity. The controller achieved above may yield an acceptable performance beyond the region for which it was designed, but this must be considered as luck in a specific problem solution. In an attempt to ensure adequate performance over

the entire parameter space. the designer must adequately cover the entire space with a valid region, or regions, upon which to base the design. But, no single controller can increase the volume of such a region to cover the entire parameter space. A common practice is to perform several point-wise control designs, each design performed for a fixed p, that will adequately cover the entire operational range. In this type of approach (controller scheduling), one must devise a mean of smoothly switching between controllers without inducing an objectionable response during the transition.

Gain scheduling is widely used in controlling a nonlinear system. The gain schedule is an interpolator, which takes a set of operating conditions as an input and gives the gains as outputs. There are many applications where gain scheduling is used, especially in aircraft control [1–3]. Linear control designs provide a set of gains for the controller at each operating condition over the entire flight. The gains at the operating points can be designed using any linear robust control strategy. If limited number of operating points and their corresponding gains is generated using a linear controller the performance of the system will be very poor. Therefore, the performance of the system can be improved by using a large database of the gains and the corresponding operating conditions. However, this requires the solution of the algebraic Riccati equation at every sample time which is complicated and expensive. In [4], it is shown that the restriction to slow variations in the scheduling variables is crucial in the gain scheduling practice. If the time variations

in the operating conditions are ignored, instability may result. Also, the main problems of conventional gain scheduling are where to place the point controllers in the operational state space, how to switch between controllers and when to switch.

1.1.2 Nonlinear systems with unknown dynamic

The weakness of traditional quantitative techniques to adequately describe and control complex and ill-defined phenomena was summarized in the well known principle of incompatibility formulated by Zadeh [5]. This principle states that "as the complexity of a system increases, our ability to make precise and yet significant statements about its behaviors diminishes." The idea of fuzzy modelling and control first emerged in Mamdani [6], and has subsequently been pursued by many others. Although fuzzy modelling and control is thought of as an alternative approach compared with traditional control methods, its effectiveness is now well proven. Over the past two decades, engineers have applied fuzzy modelling and control methods very successfully [7–12]. Fuzzy control provides nonlinear controllers, which are well justified by the universal approximation theorem in [13–17]. In other words, these fuzzy controllers are general enough to perform any nonlinear control action. Therefore, by carefully choosing the parameters of the fuzzy controller, it is always possible to design a fuzzy controller that is suitable for the nonlinear system under consideration. Based on this fact, a global stable adaptive fuzzy controller is firstly synthesized from a collection of fuzzy IF-THEN rules [18]. The fuzzy system, used

to approximate an optimal controller. is adjusted by an adaptive law based on Lu-apunov synthesis approach. An adaptive tracking control architecture is proposed in [19] for a class of continuous time nonlinear dynamic systems, where an explicit linear parameterization of the uncertainty in the dynamics is not possible. The architecture employs fuzzy systems, which are expressed as a series expansion of fuzzy basis functions (FBF's), to adaptively compensate for the plant nonlinearities. It is shown in [13] that Gaussian basis functions (GBF's) have the best approximation property. In the GBF expansion, three parameter vectors are used; connection weights (constant consequents), variances and centers. It is obvious that as these parameters change, the shape of the GBF vary accordingly. However, in the developed fuzzy schemes in [18–21] only connection weights are updated in the GBF expansion. In [22], an adaptive controller using a similar approach to the one used in [18] is introduced. Sugeno fuzzy system is used to approximate the controller. In the recent developed adaptive fuzzy controller [23] all three parameters are updated, which result in a better tracking performance. The approach in [23] can still be improved by using a parameterized Sugeno fuzzy approximator with first-order Sugeno consequent in the IF-THEN rules of the fuzzy system, which has a better approximation capability compared with that of a constant consequent, which motivated the work in Chapter 5. It is shown in [24] that higher order Sugeno systems are expected to result in at least the same system performance with fewer rules. This is due to a better approximation capability of higher order Sugeno consequents

[25]. The interpolation properties of Sugeno fuzzy system are discussed in Chapter 3.

1.2 Objectives and contributions

1.2.1 Interpolation behavior of Sugeno fuzzy controllers

Sugeno fuzzy model is used for modelling and control of nonlinear systems in all applications presented in this thesis. The interpolation function of Sugeno fuzzy systems depends on the shape, the distribution of the membership functions, the conjunction operator of the antecedents, and the order of Sugeno functional consequent. A mathematical study of the influence of these parameters on the interpolation function of the fuzzy systems is presented in Chapter 3 which will help achieve a general understanding of these systems. In this chapter the influence of the shape and the distribution of the membership functions and the order of the functional consequent on the interpolation behavior of the Sugeno fuzzy systems is investigated. In general, it can be shown by induction that for a controller with any number of inputs, product conjunction operator, constant consequent and 0.5 complementary triangular membership functions, the output is a linear function with respect to each input. Increasing the order of Sugeno consequent by one will increase the interpolation order by one. For a controller with any number of inputs, product

conjunction operator, constant consequent and noncomplementary triangular membership functions, the output is a rational function, which is a linear function with respect to each input in the numerator as well as in the denominator. Increasing the order of Sugeno consequent will increase the order of the function in the numerator by one with respect to each input, while the denominator will remain a linear function with respect to each input.

1.2.2 Fuzzy control of nonlinear system based on known dynamic: Fuzzy gain scheduling approach

The goal of this research is to derive and explore a technique to design a fuzzy-logic gain scheduling-like controllers for nonlinear plants using point-wise designs that adequately span the parameter space of the plant to be controlled. The ability to base the controller on point-wise designs allows the designer to use all available tools of classical, modern and robust control theories to aid in the solution. The proposed approach is based upon using fuzzy logic to blend the individual point designs such that for any trajectory in the parameter space, the system performs adequately. The ability to systematically design such a dynamic scheduler is a major contribution.

In Chapter 4 and [26], fuzzy-logic gain scheduling-like controller synthesis technique is developed for multiple-input single-output (MISO) nonlinear systems. The resulting controller does not require on line adaptation, estimation, or prediction. Fuzzy logic (FL) is used to smoothly schedule independently designed point

controllers over the operational and parameter spaces of the system's model. These point controllers are synthesized using techniques chosen by the designer, thus allowing unprecedented amount of design freedom. The proposed approach utilizes the advantages of gain scheduling, fuzzy logic control, neural networks and optimal control theory. This approach starts with linearizing the plant dynamics about different equilibrium points. The effect of various parameters can be introduced and studied. This results in a linear model of the system. Using one of the optimal control algorithms such as LQR, a set of gains are obtained for each equilibrium point. Subtractive clustering and ANFIS are used to construct the rules for the fuzzy gain schedule. The fuzzy gain schedule is used to continuously provide the controller with gains as a function of the states of the system. Conventional gain schedule cannot work efficiently when controlling a fast varying system, which is a major drawback. The fuzzy gain schedule provides a smooth gains without any discontinuities. Therefore, there will be no spikes in the response due to jumping from a set of gains to another. The proposed controller is implemented on the unstable inverted pendulum system, as shown in Chapter 6.

1.2.3 Fuzzy control of nonlinear systems with unknown dynamics using parameterized Sugeno fuzzy approximator

In Chapter 5 and [27, 28], we introduce a controller along the lines of [23]. The principal difference is that our controller is designed based on the well known Sugeno first-order fuzzy system. The consequent part of IF-THEN rules is a linear combination of input variables and a constant term, and the final output is the weighted average of each rule's output. This introduces additional parameter vectors to be updated, but improves the tracking performance due to a better approximation ability of a higher order Sugeno consequents model. It is also shown in [24] that a higher order Sugeno consequents model, compared with a lower order, could identify a system with less error for the same number of rules or could achieve the required performance with fewer rules. Global boundedness of the adaptive system is established. Finally, simulation results, in Chapter 5, are used to demonstrate the effectiveness of the proposed controller. A comparison between the implementation of the proposed approach in Chapter 5 and that of the approach in [23] on the unstable inverted pendulum system is shown in Chapter 6. This comparison verifies that a higher order Sugeno consequents model has a better approximation ability compared with a lower order model.

Chapter 2

Fuzzy sets and fuzzy systems

In this chapter, an introduction to fuzzy sets and fuzzy systems is given. Fuzzy connectives and fuzzy implication functions are discussed and examples on the most frequently used ones are presented. This is followed by a literature on fuzzy control. A brief description of a fuzzy controller and its components are introduced through a benchmark example of balancing an inverted pendulum on a cart. Two types of fuzzy models are introduced in this chapter. The main difference between these two models lies in the consequents of their fuzzy rules.

2.1 Introduction

A fuzzy set is a set without a crisp, clearly defined boundary and can contain elements with only a partial degree of membership. Zadeh [29] introduced fuzzy sets in 1965. Set theory was founded in 1874 by G. Cantor, a German mathematician.

The set theory is usually called "classical set theory". A classical set is a container that wholly includes or wholly excludes any given element. The membership function $\mu_A(x)$ of a classical set A, as a subset of the universe X, is defined by:

$$\mu_A(x) = \begin{cases} 1, & \text{iff} \quad x \in A \\ 0, & \text{iff} \quad x \notin A \end{cases}$$

This means that an element x is either a member of set A (with $\mu_A(x) = 1$) or not a member (with $\mu_A(x) = 0$). A fuzzy set, introduced by Zadeh [29], is a set with graded membership in the real interval: $\mu_A(x) \in [0, 1]$. In other words, it allows each element of X to belong to the set with a membership degree characterized by a real number in the closed interval $[0, 1]$. This grade corresponds to the degree to which that individual is similar or compatible with the concept represented by the fuzzy subset. Thus, individuals may belong in the fuzzy set to a greater or lesser degree by a larger or smaller membership grade. Because full membership and full non membership in the fuzzy set can still be indicated by the values of 1 and 0, respectively, we can consider the concept of a crisp set to be a restricted case of the more general concept of a fuzzy set.

2.1.1 Properties of fuzzy sets

1. One of the most important concepts of fuzzy sets is the concept of an α-cut.

   The α-cut of a fuzzy set is defined by:

$$\alpha - \text{cut}(A) = \{x \in X \quad | \, \mu_A(x) \geq \alpha\}$$

A strong α-cut is defined by

$$\bar{\alpha} -\text{cut}(A) = \{x \in X \quad | \mu_A(x) > \alpha\}$$

That is, the α-cut of a fuzzy set A is the crisp set that contains all elements of the universal set X, whose membership grades in A are greater than or equal to the specified value of α.

2. The height ($h(A)$) of a fuzzy set A is the largest membership grade obtained by any element in that set. An $h(A)$ is defined by :

$$h(A) = \sup_{x \in X} \mu_A(x)$$

Fuzzy sets with a height equal to 1 are called normal. They are called subnormal when $h(A) < 1$.

3. The support of a fuzzy set A within a universal set X, is the crisp set that contains all elements of X that have nonzero membership in A. It is defined by:

$$\text{supp}(A) = \{x \in X \quad | \mu_A(x) > 0\}$$

The support of A is the same as the strong 0-cut of A. The 1-cut is often called the core of A. The elements of A, where $\mu_A(x) = 0.5$, are called crossover points. Figure 2.1 shows the support, core and height of a fuzzy set.

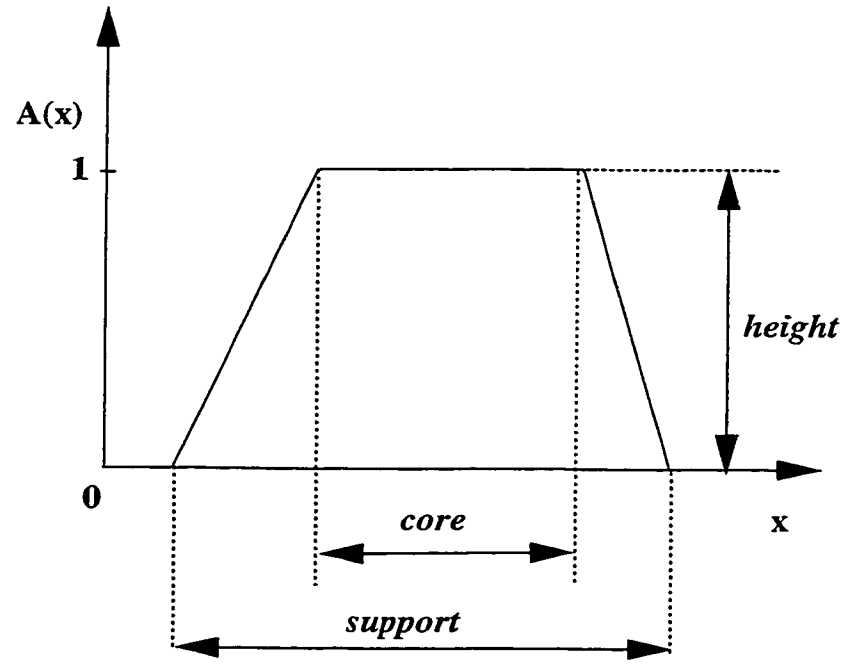

Figure 2.1: The support, core and height of a fuzzy set

2.1.2 Representation of a fuzzy set

1. Functional representation: uses functional description to represent fuzzy sets. An example is the functional description of a triangular-shaped fuzzy set:

$$\mu_A(x) = \begin{cases} 0 & \text{If} \quad x < -a \\ \frac{x+a}{2a} & \text{If} \quad -a \leq x \leq a \\ 1 & \text{If} \quad x > a \end{cases}$$

2. Ordered pairs representation [29]: A fuzzy set A, a fuzzy subset of X, may be represented as a set of ordered pairs of elements x and its grade of membership function:

$$A = \{(x, \mu_A(x)) \quad | \, x \in X\} \quad \text{or}$$
$$A = \sum_{i=1}^{m} \mu_A(x_i)/x_i = \mu_A(x_1)/x_1 + \cdots + \mu_A(x_m)/x_m$$

In this equation the summation sign does not denote summation, it denotes the collection of all points $x \in X$, with associated membership function $\mu_A(x) \in [0, 1]$. The slash in this expression associate the elements in X with their membership grades.

Example: Let A = integer close to 7, then:

$A = .1/4 + .5/5 + .8/6 + 1/7 + .8/8 + .5/9 + .1/10$

Three points to note from A:

(a) The integers not explicitly shown all have memberships functions equal to zero.

(b) The values for the membership functions were chosen by a specific individual, except for the unity membership value when $x = 7$, they can be modified based on our own personal interpretation of the phrase "close."

(c) The membership function is symmetric about $x = 7$, because there is no reason to believe that integers to the left of 7 are close to 7 in a different way than are integers to the right of 7. But again, we are free to make other interpretations.

When X is continuous(e.g., the real numbers), A is commonly written as :

$$A = \int_x \mu_A(x)/x$$

Again, in this equation, the integral sign denotes the collection of all points $x \in X$ with associated $\mu_A(x) \in [0, 1]$, but not integration.

Comparing the above two types of fuzzy set representation, it is obvious that the ordered pairs representation will be chosen for computer implementation. Due to complexity of the operations when using functional representation discretizations are necessary in practical applications.

2.2 Fuzzy Connectives

In classical set theory intersection, union and complement of sets are uniquely defined, due to the fact that *and*, *or* and *not* are two-valued logical operations. Let us start with elementary crisp-set operations. Let A and B be two subsets of U. The union of A and B, denoted $A \cup B$, contains all of the elements in either A or B, i.e., $\mu_{A \cup B}(x) = 1$, if $x \in A$ or $x \in B$, and $\mu_{A \cup B}(x) = 0$, if $x \notin A$ and $x \notin B$. The intersection of A and B, denoted $A \cap B$, contains all the elements that are simultaneously in A and B, i.e., $\mu_{A \cap B} = 1$, if $x \in A$ and $x \in B$, and $\mu_{A \cap B} = 0$, if $x \notin A$ or $x \notin B$. The complement $\bar{A}$ contains all the elements that are not in A, $\mu_{\bar{A}}(x) = 1, x \notin A$, $\mu_{\bar{A}}(x) = 0$, if $x \in A$. In summary, If A and B are crisp sets then

$$A \cap B = \{a \mid a \in A \quad \text{AND} \quad a \in B\}$$

$$A \cup B = \{a \mid a \in A \quad \text{OR} \quad a \in B\}$$

$$\bar{A} = \{a \mid a \notin A\}$$

The extension of standard concepts of classical set theory like union, intersection, etc. to the theory of fuzzy sets is not obvious, due to the extension of the range

of membership function to the interval [0. 1] instead of the restricted set of $\{0, 1\}$. Linguistic connectives *and* and *or* are modelled through the use of fuzzy set theoretic operators, intersection and union, respectively. Intersection and union are also interchangeably used with conjunction, and disjunction, respectively.

An infinite number of possible definitions can be chosen to implement intersection and union. Functions that qualify as fuzzy intersections and fuzzy unions are usually referred to in the literature as triangular norms (T-norms) and triangular conorms (T-conorms or S-norms), respectively. A significant body of literature has appeared concerning the appropriate definitions for intersection and union of fuzzy sets [30–35]. Zadeh [29] has stated that the selection of the connectives should be a local choice depending on the situation. Yager [36] also noted that the desire for some specific properties for a certain application acts as a filter for selecting a particular intersection\union pair.

2.2.1 Triangular Norms (T-norms)

The intersection of two fuzzy sets (T-norm), is specified in general by a two-place function from $[0, 1] \times [0, 1]$ to $[0, 1]$, satisfying the following four axioms [35]:

T1. Commutativity

$$T(a, b) = T(b, a)$$

This means that conjunction does not depend on the order of a, and b. This is expected since there is no reason to assign different truth values to $T(a, b)$, and

$T(b, a)$.

T2. Monotonicity

$$T(a, c) \leq T(b, d) \quad \text{when} \quad a \leq b, \text{ and } \quad c \leq d$$

Which means that the value of the conjunction should not decrease when the value of at least one proposition increased. We should be more willing to accept "b and d" than "a and c."

T3. Associativity

$$T(a, T(b, c)) = T(T(a, b), c)$$

T4. Boundary condition

$$T(a, 1) = a$$

An infinite number of possible definitions can be chosen to implement union, but the desire for some additional properties for certain applications act as a filter for selecting a particular union. Some of these properties are:

T5. The property of idempotency $T(a, a) = a$. This property is useful in a situation in which there exist a possibility of a set to be repeated, and we desire that this set has no double effect.

T6. Archimedean property is defined as follows:

A T-norm is Archimedean if

T is continuous, and

$\forall a \in (0, 1)$, $T(a, a) < a$.

Remarks

1. A many-valued conjunction is a pointwise operation, which means $T(a, b)$ depends only on the values of a and b.

2. Each T-norm satisfies additional boundary conditions:

$$T(0, a) = T(a, 0) = 0$$
$$T(1, a) = a$$

We can conclude the following:

$$T(1, 0) = T(0, 1) = T(0, 0) = 0$$
$$T(1, 1) = 1$$

This property is called consistency with the classical set theory.

3. If for T_1 and T_2, $T_1(a, b) \leq T_2(a, b)$, for all $(a, b) \in [0, 1] \times [0, 1]$, then it can be said that T_1 is weaker than T_2. It can be proven that $T(a, b) \leq \min(a, b)$, accepting "a and b" requires more than accepting a or b alone.

4. A T-norm is said to be strictly monotone if it is strictly increasing on $]0, 1] \times]0, 1]$ as a function from $[0, 1] \times [0, 1]$ into $[0, 1]$:

$$T(a, c) < T(b, d) \quad \text{when} \quad a < b, \quad \text{and} \quad c < d$$

A T-norm is called strict if it is continuous and strictly monotone.

2.2.2 Triangular conorms (T-conorms, S-norms)

The union of two fuzzy sets (T-conorm, S-norms), is specified in general by a two-place function from $[0,1] \times [0,1]$ to $[0,1]$, satisfying the following conditions:

S1. Commutativity

$$S(a,b) = S(b,a)$$

This means that disjunction does not depend on the order of a and b. This is expected since there is no reason to assign different truth values to $S(a,b)$, and $S(b,a)$.

S2. Monotonicity

$$S(a,c) \leq S(b,d) \quad \text{when} \quad a \leq b, \text{and} \quad c \leq d$$

Which means that the value of the disjunction should not decrease when the value of at least one proposition increased. We should be more willing to accept "b or d" than "a or c."

S3. Associativity

$$S(a,S(b,c)) = S(S(a,b),c)$$

S4. Boundary condition

$$S(a,0) = a.$$

Some of additional properties are:

S5. The property of idempotency $S(a,a) = a$. This property is useful in a situation

in which there exist a possibility of a set to be repeated, and we desire that this set has no double effect.

S6. Archimedean property is defined as follows:

A T-conorm is Archimedean if

S is continuous, and

$\forall a \in (0,1),\ S(a,a) > a.$

Remarks

1. A many-valued disjunction is a pointwise operation, which means $S(a,b)$ depends only on the values of a and b.

2. Each S-norm satisfies additional boundary conditions:

$$S(1,a) = S(a,1) = 1$$

$$S(a,0) = a$$

We can conclude the following:

$$S(1,1) = S(0,1) = S(1,0) = 1$$

$$S(0,0) = 0$$

This property is called consistency with the classical set theory.

3. If for S_1 and S_2, $S_1(a,b) \leq S_2(a,b)$, for all $(a,b) \in [0,1] \times [0,1]$, then it can be said that S_1 is weaker than S_2. It can be proven that $S(a,b) \geq \max(a,b)$, accepting "a or b" requires less than accepting a or b alone.

4. The property of idempotency:

$$S(a,a) = a.$$

This property is useful in a situation in which there exist a possibility of a set to be repeated, and we desire that this set has no double effect.

5. An S-norm is said to be strictly monotone if it is strictly increasing on $]0,1] \times]0,1]$ as a function from $[0,1] \times [0,1]$ into $[0,1]$:

$$S(a,c) < S(b,d) \quad \text{when} \quad a < b, \quad \text{and} \quad c < d$$

An S-norm is called strict if it is continuous and strictly monotone.

2.2.3 Fuzzy sets intersection and union: Examples

There are several examples of T-norms and T-conorms. We list here the most frequently used, and important ones.

Example 1: In his classical paper Zadeh [29] proposed to use:

$T_{\min}(a,b) = \min(a,b)$, as a conjunction. (minimum)

$S_{\max}(a,b) = \max(a,b)$, as a disjunction. (maximum)

Min-max are the most popular in fuzzy literature. $T_{\min}$ is the strongest T-norm, while, $S_{\max}$ is the weakest T-conorm. If a modification of A (or B) does not necessarily imply an alteration of $A \cap B$ or $A \cup B$, therefore, $\cap$ and $\cup$ are said to be non interactive. Min and max are non interactive operators. They are said to have a dominance effect, for example, a and $b \in [0,1]$, $a \wedge y = a$, for $y \in [a,1]$, and $x \vee b = b$, for $x \in [0,b]$.

Example 2: One pair of the probabilistic like operators is:

$T_{prod.}(a,b) = ab$, as a conjunction. (algebraic product)

$S_{sum}(a,b) = a + b - ab$, as a disjunction. (algebraic sum)

Prod-sum operators reflect a trade-off between A and B, they are said to be interactive, because the outcome of the combination using one of these operators depends on A as well as B, this means that any modification of A or B will affect the outcome of $A \cap B$ or $A \cup B$. These operators are used to eliminate the dominance effect of min-max operators, wherever undesirable.

Example 3: Bold intersection and union:

$T_{bold}(a,b) = \max(0, a + b - 1)$, as a conjunction. (bold intersection)

$S_{bold}(a,b) = \min(1, a + b)$, as a disjunction. (bold union)

Example 4: Drastic product and sum:

$$T_w(a,b) = \begin{cases} a & \text{if} \quad b = 1 \\ b & \text{if} \quad a = 1 \\ 0 & \text{otherwise} \end{cases}, \text{ as a conjunction. (Drastic product)}$$

$$S_w(a,b) = \begin{cases} a & \text{if} \quad b = 0 \\ b & \text{if} \quad a = 0 \\ 1 & \text{otherwise} \end{cases}, \text{ as a disjunction. (Drastic sum)}$$

T_w is the weakest T-norm, and S_w is the strongest T-conorm.

To compare different T-norms (T-conorms), a graphical representation of these T-norms (T-conorms) is given in [35]. These operators are ordered as follows:

$T_w \leq T_{bold} \leq T_{prod.} \leq T_{\min}$

$S_w \geq S_{bold} \geq S_{sum} \geq S_{\max}$

2.2.4 Operator selection

The variety of operators for conjunction and disjunction of fuzzy sets might make it difficult to decide which one to use in a specific application. Zimmermann [37] stated eight important criteria according to which operators can be classified:

1. Axiomatic strength: An operator is considered better if it satisfies the required axiomatic properties for a given application. These properties have been discussed in section 2.2.

2. Empirical fit: The operators must be appropriate models of real system behavior. This can be proven by empirical testing. Zimmermann and Zysno [38] did some work in this aspect.

3. Adaptability: If we want to use a very small number of operators to model many situations, then these operators have to be adaptable to the specific context. This can be achieved by parametric families of operators.

4. Numerical efficiency: Parametrized operators usually require more computational effort than other operators like min or product. It is important to take this issue into consideration when large problems have to be solved.

5. Compensation: This is the opposite of the dominance property which is possessed by minimum. Compensation effect is observed when the decrease in the membership value of one of the components, intersection, is compensated

Intersection operators T-norms	Union operators T-conorms
minimum	maximum
algebraic product	algebraic sum
bold intersection	bold union
drastic product	drastic sum

Table 2.1: A summary of operators used to model fuzzy intersection and union

by an increase in the value of the other component. The product operator is compensatory.

6. Aggregating behavior: If one combines fuzzy sets by the product operator, each additional fuzzy set will normally decrease the resulting aggregate degrees of membership. This might be desirable feature; it might be inadequate. However, this will not be observed if the combination is done by the minimum operator.

A summary of operators used to model fuzzy intersection and union is given in Table 2.1.

2.3 Implication Functions

The most widely studied type of fuzzy statements involving several variables is the "If.....Then....." rule with fuzzy predicates. We will here write the If...Then statement in the form "If x is A Then y is B." The implication of logic is represented by a truth table which specifies the truth value of the implication when the truth value of the antecedent, $x = A$, and of the consequence, $y = B$, are given. According to

Zadeh [5], a conditional statement "If x is A Then y is B", describes a relation between two fuzzy variables x and y. He, therefore, suggests that the conditional statement should be represented by a fuzzy relation from the universe of the antecedent to the universe of the consequence. This elementary conditional statement can be represented by the following:

$$\mu_{A \to B} = I(\mu_A(x), \mu_B(y))$$

where I is a fuzzy implication. I is a function of the form :

$$I : [0,1] \times [0,1] \to [0,1]$$

Dubois and Prade [39, 40] made a summary of the different types of fuzzy implications, some of them are discussed below:

1. Implication based on the classical view of implication, related to the formalizm of Boolean logic : (where $a \to b$ is defined by $\neg\, a \vee b$), and that of the form:

$$I(a,b) = S(n(a), b) \qquad \text{(S-Implication)}$$

Where S is a T-conorm, and n is a strong negation.

Examples:

By $S(a,b) = \min(1, a+b)$, and $n(a) = 1-a$, the following implication function is generated:

$$I_L(a,b) = \min(1 - a + b, 1) \qquad \text{(Lukasiewicz implication)}$$

By $S(a,b) = a + b - ab$ and $n(a) = 1 - a$, the following implication function is generated:

$$I_{Re}(a,b) = 1 - a + ab \qquad \text{(Reichenbach implication)}$$

By $S(a,b) = \max(a,b)$ and $n(a) = 1 - a$, the following implication function is generated:

$$I_{K-D}(a,b) = \max(1 - a, b) \qquad \text{(Kleene-Dienes implication)}$$

2. Implication related to the residuation concept from intuitionistic:

$$I(a,b) = \sup\{x \in [0,1] \mid T(a,x) \leq b\} \qquad \text{(R-Implication)}$$

Where T is a T-norm. This can be justified by the following classical set theoretic identity:

$$a \to b = (\overline{a \setminus b}) = \cup\{x \mid a \cap x \subseteq b\}$$

Where $\setminus$ denotes the set-difference.

Examples:

By $T(a,b) = \min(a,b)$, the following implication function is generated:

$$I_{G\ddot{o}}(a,b) = \begin{cases} 1, & \text{if } a \leq b \\ b, & \text{otherwise} \end{cases} \qquad \text{(G ö del implication)}$$

By $T(a,b) = ab$, the following implication function is generated:

$$I_{Go}(a,b) = \begin{cases} 1, & \text{if } a = 0 \\ \min(1, \frac{b}{a}), & \text{otherwise} \end{cases} \qquad \text{(Goguen implication)}$$

By $T(a,b) = \max(0, a+b-1)$, the following implication function is generated:

$$I_L(a,b) = \min(1-a+b, 1) \qquad \text{(Lukasiewicz implication)}$$

3. Implications based on the implication in Quantum logic:

$$I(a,b) = S(n(a), T(a,b)) \qquad \text{(QL-implications)}$$

Where T is a T-norm, S is a T-conorm and n is a strong negation.

Examples:

By $T(a,b) = \min(a,b)$, $S(a,b) = \max(a,b)$ and $n(a) = 1-a$, the following implication function is generated:

$$I_{Za} = \max(1-a, \min(a,b)) \qquad \text{(Zadeh implication)}$$

By $T(a,b) = \max(0, a+b-1)$, $S(a,b) = \min(1, a+b)$ and $n(a) = 1-a$, the following implication function is generated:

$$I_{K-D} = \max(1-a, b) \qquad \text{(Kleene-Dienes implication)}$$

In classical logic, "If a Then b" is always receives the same value as its contrapositive, "If not-b Then not-a". We will say that a fuzzy implication operator $\rightarrow$ possesses contrapositive symmetry, iff:

$$(a \rightarrow b) = (1-b) \rightarrow (1-a)$$

It was found that contrapositive symmetry is, in general, in conflict with using a T-norm and its n-dual T-conorm in generating the QL implications. Examples on QL implications generated from non dual T-norms and T-conorms:

By $T(a,b) = \min(a,b)$ and $S(a,b) = \min(1, a+b)$, the following implication function is generated:

$$I_L(a,b) = \min(1-a+b, 1) \qquad \text{(Lukasiewicz implication)}$$

By $T(a,b) = \min(a,b)$ and $S(a,b) = \min(1, a+b)$, the following implication function is generated:

$$I_L(a,b) = \min(1-a+b, 1) \qquad \text{(Lukasiewicz implication)}$$

4. Interpretation of the implication as a conjunction:

$$I(a,b) = T(a,b)$$

This type is clearly not a generalization of the classical implication, but complies with the classical conjunction. Fuzzy implication which are represented by a conjunction are usually used in fuzzy control.

Examples:

$$I(a,b) = \min(a,b) \qquad \text{(Mamdani implication)}$$
$$I(a,b) = ab \qquad \text{(Larsen implication)}$$

2.4 Fuzzy control

2.4.1 Literature

During the past decade, fuzzy control (FC), initiated by the pioneering work of Mamdani and Assillian [41] (Steam Engine Control), which was motivated by Zadeh's

paper [42] (A rationale for fuzzy control), has emerged as one of the most active and fruitful areas of research in the application of fuzzy set theory, fuzzy logic and fuzzy reasoning. In contrast to conventional control techniques, FC is best utilized in complex and ill-defined processes that can be controlled by a skilled human operator without much knowledge of their underlying dynamics. Some of the most important developments in this field is the work of Rutherford and Bloore [43] (Analysis of control algorithms), Ostergaard [44] (Heat exchanger and cement klin control), Willaeys [45] (Optimal fuzzy control], Fukami, Mizumoto and Tanaka [46] (Fuzzy conditional inference), Takagi and Sugeno [47] (Derivation of fuzzy control rules), Yasunobo and Miyamoto [48] (Predictive fuzzy control), Sugeno and Murakami [49] (Parking control of a model car), Kiszka, Gupta and Nikiforuk [50] (Fuzzy system stability), Yamakawa [51] (Fuzzy controller hardware system).

Notable applications of fuzzy logic control include Steam Engine Control by Mamdani and Assillian [41], a warm water process by Kickert and Van Nauta Lemke [52]; Heat exchanger by Ostergaard [44]; activated sludge wastewater treatment by Tong, Beck and latten [53]; Traffic junction control by Pappis and Mamdani [54]; a cement klin by Larsen [55]; aircraft flight control by Larkin [56]; robot control by Uragami Mizumoto and Tanaka [57], Scharf and Mandic [58], Wakileh and Gill [59], Isik [60], Tsay and Huang [61], Hsu and Fu [62], Demirli and Turksen [63]; model car parking and tuning by Sugeno and Murakami [49, 64], Sugeno and Nishida [65]; automobile speed control by Murakami [66].

One of the main directions in the theory of fuzzy systems is the linguistic approach, based on linguistically described models. It was originally initiated by Zadeh [5] and developed further by Bezdek [67], Mamdani and Assillian [41], Tong [68], Pedrycz [69], Takagi and Sugeno [70], Chiu [71], Emami *et al.* [72] and Demirli *et al.* [24, 73].

Stability analysis of fuzzy control systems is gaining much attention. Of various existing methodologies for stability analysis of fuzzy systems Mamdani [74]; Negoita [75]; Braae and Rutherford [76]; Pedrycz [77]; Ray and Majumder [78]; Kiszka *et al* [50]; Langari and Tomizuka [79–82]; Tanaka and Sugeno [83]; Tanaka and Sano [84, 85]; Chen and Ying [86].

Fuzzy logic control systems have found applications in household appliances [87] such as air conditioners (Mitsubishi); washing machines (Matsushita, Hitachi); video recorders (Sanyo, Matsushita); television autocontrast and brightness control cameras (Canon); vacuum cleaners (Matsushita); microwave ovens (Toshiba);

2.4.2 Fuzzy models

We shall introduce two types of fuzzy models that have been widely employed in various applications. The difference between these two models lies in the consequents of their fuzzy rules.

Mamdani fuzzy models

The Mamdani fuzzy model [41] was proposed as the first attempt to control a steam engine and boiler combination by a set of linguistic control rules obtained from experienced human operator. The basic idea of Mamdani fuzzy model is to incorporate the experience of a human operator in the design of a controller for a process whose input-output relationship is described by a collection of fuzzy control rules involving linguistic variables rather than a complicated dynamic model. The typical architecture of Mamdani fuzzy model is shown in Figure 2.2.

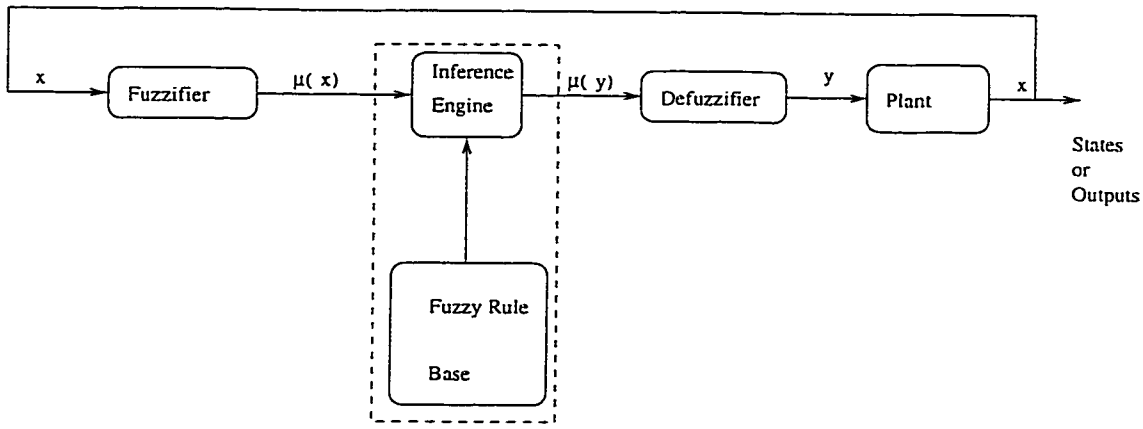

Figure 2.2: The typical architecture of Mamdani fuzzy model

- Fuzzifier: It can be defined as a mapping from an observed input space to labels of fuzzy sets in a specified input universe of discourse. In fuzzy control applications, the observed data are usually crisp. A natural and simple fuzzification approach is to convert a crisp value x_0 into a fuzzy singleton A within the specified universe of discourse. That is, the membership function of A,

Range of error	Normalized segments	Fuzzy set
$e \leq -4$	[-1,-0.5]	NB
$-6 < e \leq -2$	[-0.75,-0.25]	NM
$-4 < e \leq 0$	[-0.5,0]	NS
$-2 < e \leq 2$	[-0.25,0.25]	Z
$0 < e \leq 4$	[0,0.5]	PS
$2 < e \leq 6$	[0.25, 0.75]	PM
$4 \leq e$	[0.5,1]	PB

Table 2.2: Normalization and fuzzy partition for fuzzy input variable e

$\mu_A(x)$, is equal to 1 at the point x_0, and zero at other places. The fuzzification involves the following functions.

1. Measures the values of input variables.
2. Performs a scale mapping that transfers the range of values of input into corresponding universe of discourse.
3. Performs the function of fuzzification that converts input data into suitable linguistic values which may be viewed as labels of fuzzy sets.

The process of fuzzification is explained in Table 2.2. Figure 2.3 shows the Diagrammatic representation of fuzzy sets corresponding to Table 2.2.

- The Knowledge base: It consists of a data base and a linguistic fuzzy control rule base.

1. The data base provides necessary definitions, which are used to define linguistic control rules and fuzzy data manipulation in an FC.

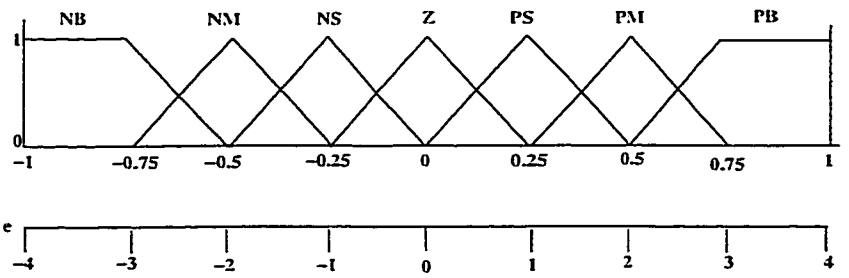

Figure 2.3: Diagrammatic representation of fuzzy sets corresponding to Table 2.2

2. The rule base characterizes the control goals and control policy. Fuzzy control rules are characterized by a collection of fuzzy IF-THEN rules.

- Inference engine: It represents the processing unit. It determines the corresponding output value from the measured input according to the knowledge base. It has the capability of simulating human decision-making based on fuzzy concepts and of inferring fuzzy control actions employing fuzzy implication and the rules of inference in fuzzy logic.

- Defuzzifier: Defuzzification is a mapping from a space of fuzzy control actions defined over an output universe of discourse into a space of nonfuzzy (crisp) control actions.

The components of the Mamdani fuzzy controller is introduced for a benchmark example of balancing an inverted pendulum on a cart. Consider the inverted pendulum mounted on a motor driven cart as shown in Figure 2.4. This is a model of the attitude control of a space booster on take off. The inverted pendulum is unstable in that it may fall over any time in any direction unless a suitable control force is

applied. The objective of this controller is to balance the pendulum in the upright position. Because all linear controllers are designed based on the linearized model of the system, they are valid only for a region a bout a specific point. For this reason, such linear controllers tend to be sensitive to parametric variations, uncertainties and disturbances. To enhance the performance of the balancing control, we will use a nonlinear control scheme, which is a fuzzy controller. The first task is to define the

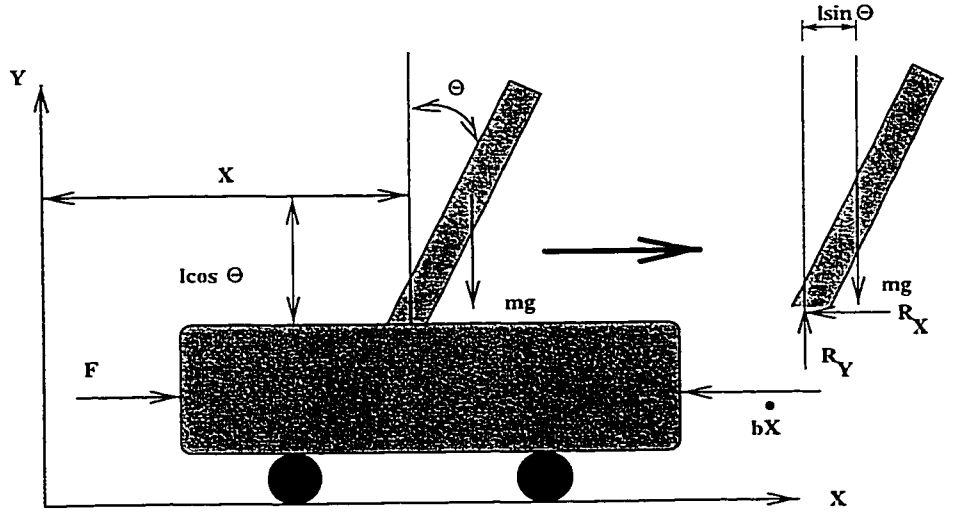

Figure 2.4: The Cart-Pole system

inputs and the outputs, this stage depends on expert decision. Let us say that the expert choses $e = \theta_d - \theta$ and $\dot{e}$ as two inputs and F is the control variable. Certainly there are many other choices for the inputs, for example, the integral of the error. In this system the force is the only choice. The membership function quantifies to which linguistic value does the input value of a variable belong. Many shapes can be used to define the meaning of a linguistic value, triangular, bell , trapezoid and many others. The inputs e and $\dot{e}$ and their mapping into membership functions are

given in Figures 2.5 and 2.6 respectively. Figure 2.7 shows the fuzzification of the output variable F.

In fuzzy control the expert will put the commands (rules) that will be used to control the system in natural language.

- ϵ : error
- $\dot{e}$: change in error
- F : Force

Let the error, change in error and force take the following values:

NL : Negative Large

NM : Negative Medium

NS : Negative Small

ZE : Zero

PS: Positive Small

PM : Positive Medium

PL : Positive Large

To have the balance upright, θ_d (the desired position) must equal to zero, then:

$\epsilon = -\theta$

$\dot{e}$=-$\dot{\theta}$

Note that:

- If θ is negative, this means that pole is to the left of the upright position.

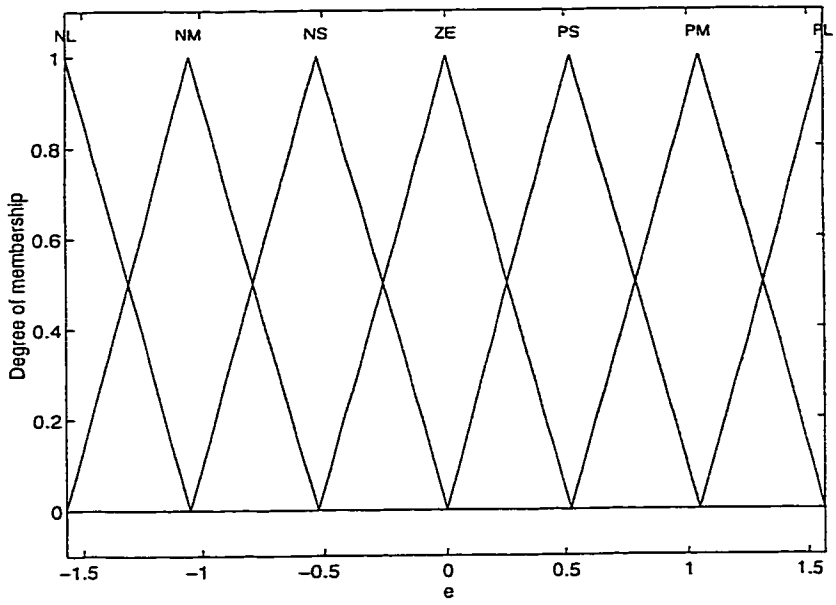

Figure 2.5: The membership functions of the 1st input (e)

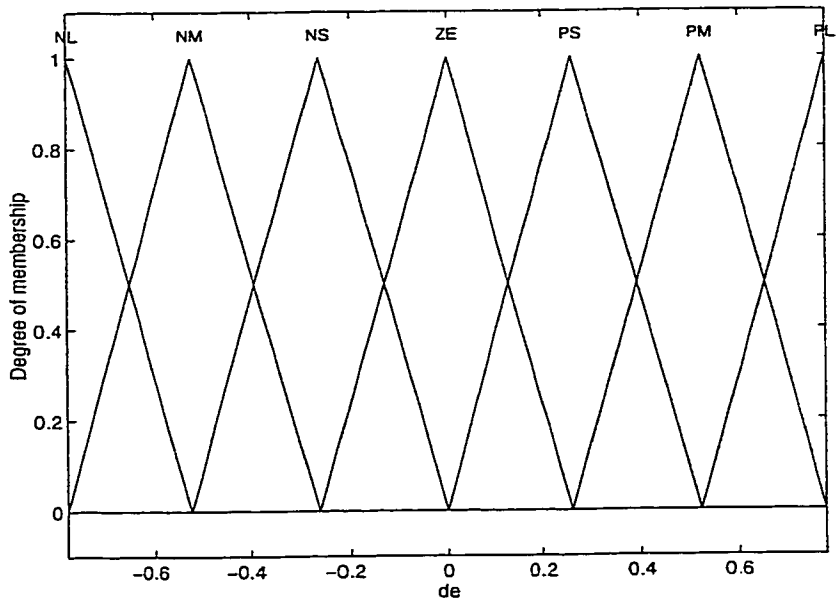

Figure 2.6: The membership functions of the 2nd input ($\dot{e}$)

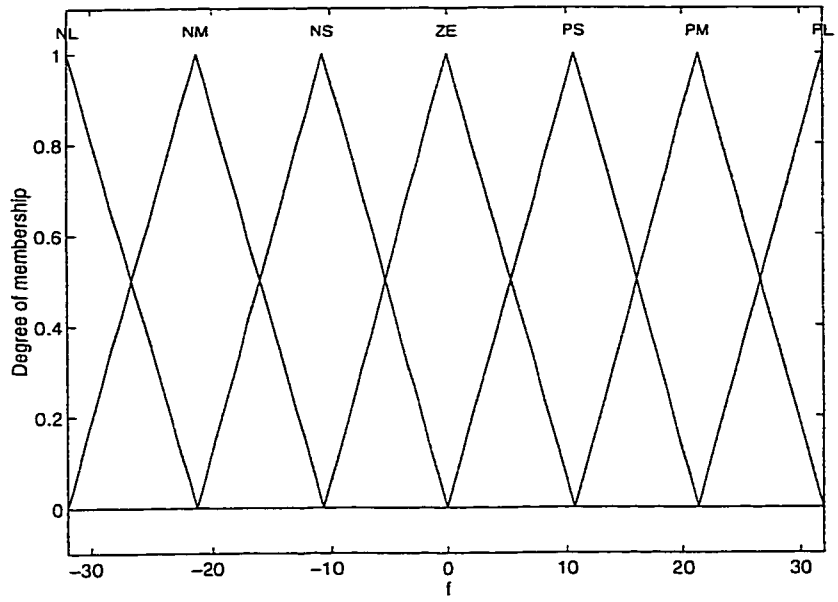

Figure 2.7: The membership functions of the output (F)

- If θ is positive, this means that pole is to the right of the upright position.

Now the expert will specify a set of rules that describes the system behavior, one of the rules is the following:

IF e is NL **AND** $\dot{e}$ is NL **THEN** F is PL

There are $7^2 = 49$ possible rules (all possible combinations of premise linguistic values for two inputs), see Table 2.3.

To perform inference, there are four steps:

1. Matching (determination of the degree of membership of the input in the rule-antecedent). This step is to take the inputs and determine the degree to which they belong to each of the appropriate fuzzy sets via membership function.

		$\dot{e}$						
		NL	NM	NS	ZE	PS	PM	PL
e	NL	PL	PL	PL	PL	PM	PS	ZE
	NM	PL	PL	PL	PM	PS	ZE	NS
	NS	PL	PL	PM	PS	ZE	NS	NM
	ZE	PL	PM	PS	ZE	NS	NM	NL
	PS	PM	PS	ZE	NS	NM	NL	NL
	PM	PS	ZE	NS	NM	NL	NL	NL
	PL	ZE	NS	NM	NL	NL	NL	NL

Table 2.3: All possible combinations of premise linguistic values for two inputs

2. Apply fuzzy conjunction operator. Once the inputs have been matched, we know the degree to which each part of the antecedent has been satisfied for each rule. If the antecedent of a given rule has more than one part, the fuzzy operator *and* or *or* is applied to obtain one number that represents the result of the antecedent for that rule. This number is called a firing strength and will then be applied to the output membership function.

3. Apply implication function (computation of the rule consequences). The inferred output of each rule is a fuzzy set scaled by its firing strength via implication function.

4. Aggregate all outputs(aggregation of rule consequences to the fuzzy set control action). It is the process of taking all the fuzzy sets that represent the output of each rule and combining them into a single fuzzy set.

There are two common applications of inference in fuzzy logic control. If we adopt max and min as our choice for the T-conorm and T-norm operators, respectively,

and use max-min composition, then the resulting reasoning is called max-min-min inference system and is shown in Figure 2.8. On the other hand, if max and algebraic product are chosen for the T-conorm and T-norm operators, respectively, and max-product composition is used, then the resulting reasoning is called max-product-product inference system and is shown in Figure 2.9. Since technical processes

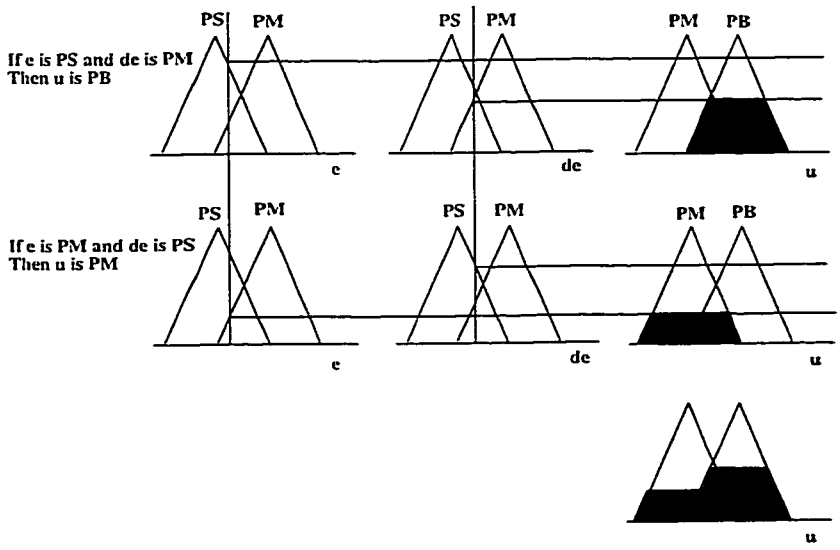

Figure 2.8: Graphical representation of max-min composition

requires crisp control actions, a procedure that generates a crisp value out of the given fuzzy set is required. These defuzzification methods are based on heuristic ideas like take the action that corresponds to the maximum membership. The most commonly used defuzzification methods are: center of area (COA), center of sums (COS), and mean of maxima (MOM).

In this example, the inference step and defuzzification step can be achieved using the fuzzy toolbox of MATLAB. In the fuzzy toolbox, the AND operator is

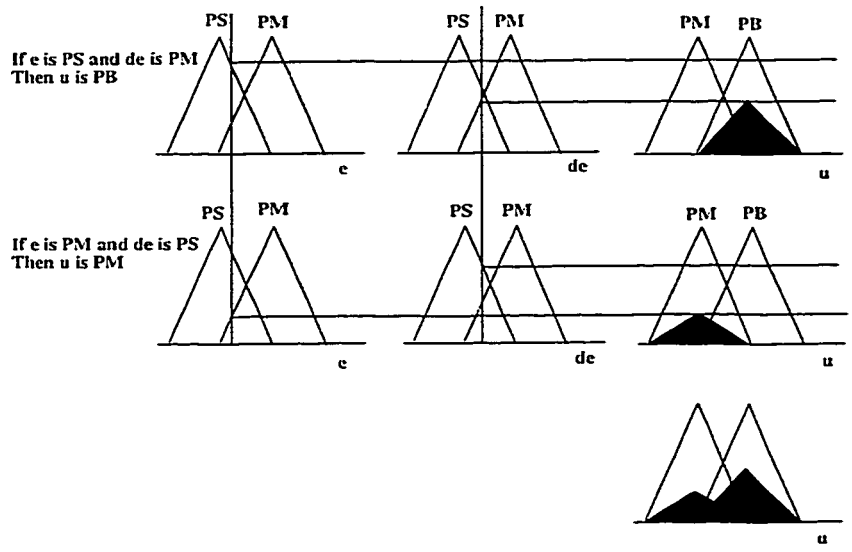

Figure 2.9: Graphical representation of max-product composition

defined by minimum or product, THEN is defined by minimum, and maximum is used as an aggregation operator. This arrangement is called Mamdani type of inference or max-min-min inference system. Simulation tests are carried out with the *Fuzzy Toolbox for MATLAB and Simulink*. The first step is to guantify the meaning of the linguistic values from Table 2.3. This is done by specifying the membership functions for the fuzzy sets. The inputs e and $\dot{e}$ and their mapping into membership functions are given in Figures 2.5 and 2.6. The following step is to edit the rules from Table 2.3 using the rule editor. Finally, we can test the fuzzy system using a set of inputs. The fuzzy system should give a crisp control value. This completes the operation of the fuzzy controller. One of the great advantages of the fuzzy toolbox is the ability to take fuzzy systems directly into Simulink and test them out in a simulation environment. A Simulink block diagram for this system is shown in

Figure 2.10. Mamdani method is able to stabilize the pendulum with a very smooth

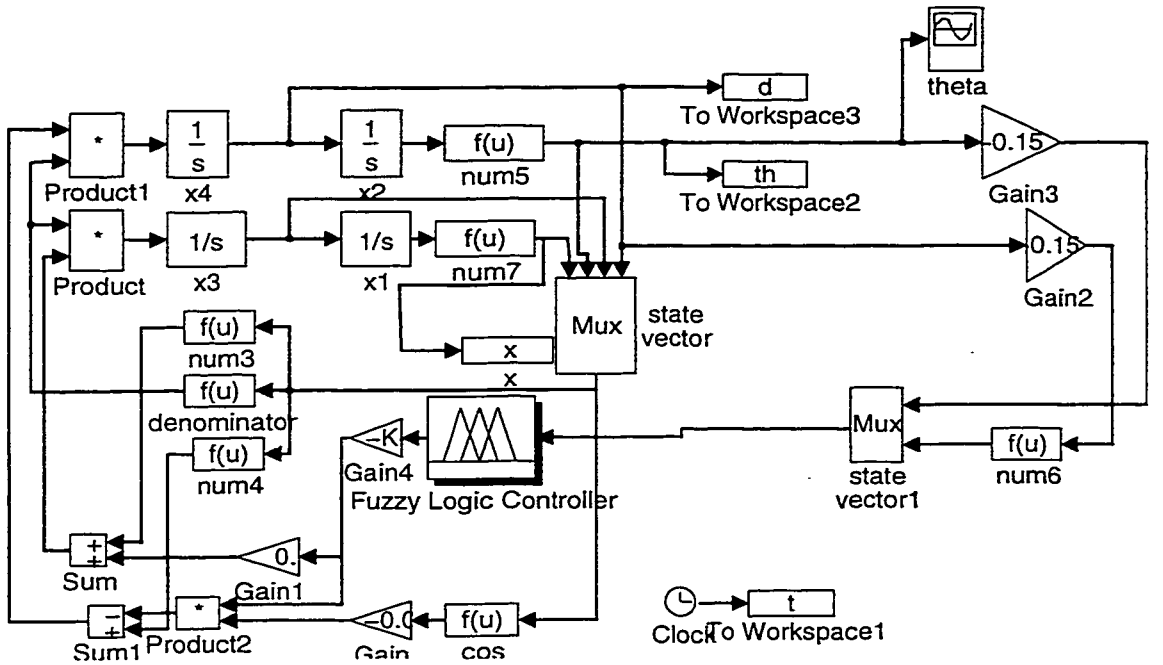

Figure 2.10: Simulation of the inverted pendulum fuzzy control

response and with a steady state error which is equal to zero. Figure 2.11 shows the time diagram of the angle when max-min-min method is used. Position of the cart is given in Figure 2.12.

Sugeno fuzzy model

Sugeno model [70] is first introduced in 1985. The type of rule referred to as a Sugeno rule has the following form:

IF e is NL **AND** $\dot{e}$ is NL **THEN** $F = f(e, \dot{e})$

which shows that the rules have fuzzy antecedents, just like Mamdani controller, and consequences of these rules are functions of the controller inputs rather than

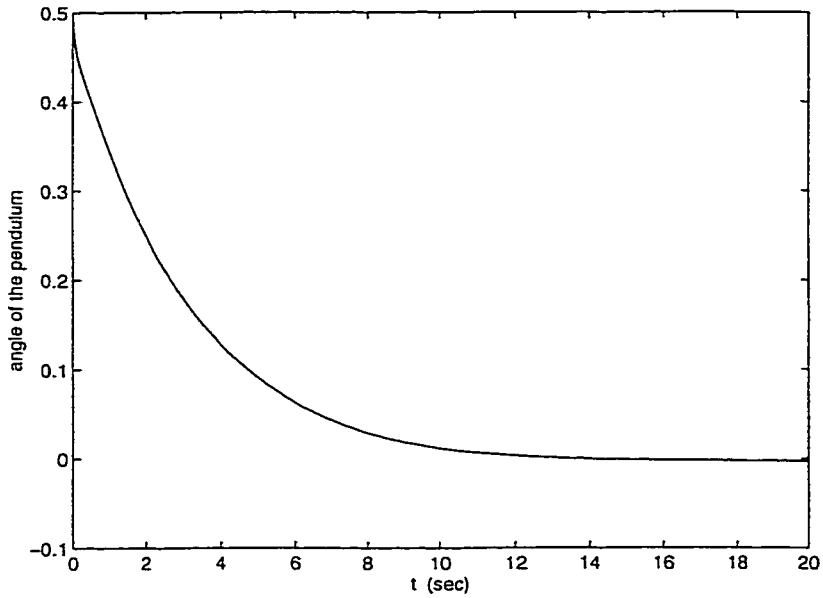

Figure 2.11: The time diagram of θ with max-min-min inference system

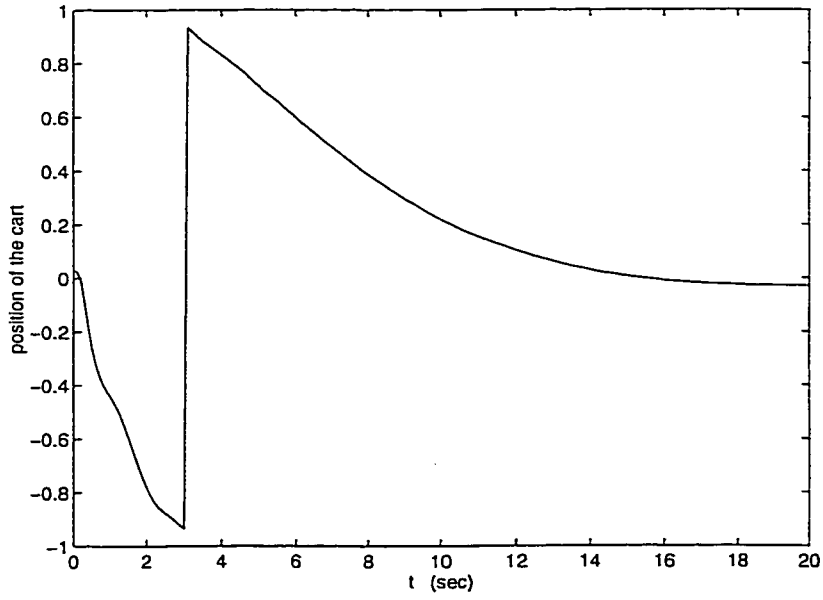

Figure 2.12: The time diagram of x with max-min-min inference system

fuzzy sets. The consequence function is usually a first-order polynomial but other types can be used as long as all rules collectively can appropriately describe the output of the model within the fuzzy region specified by the antecedents. When the consequent is a first-order polynomial, the resulting fuzzy system is called a first-order Sugeno fuzzy model. The rule results are aggregated as weighted sums of the control actions corresponding to each rule. For example, Consider the following set of Sugeno fuzzy rules that describe the behavior of a fuzzy system:

$$R_1: \quad \textbf{IF } x_1 \text{ is } A_1 \textbf{ AND } x_2 \text{ is } B_1 \textbf{ THEN} \quad w_1 = a_1 x_1 + b_1 x_2$$

$$R_2: \quad \textbf{IF } x_1 \text{ is } A_2 \textbf{ AND } x_2 \text{ is } B_1 \textbf{ THEN} \quad w_2 = a_2 x_1 + b_2 x_2$$

- Using Sugeno approach
- with the fuzzy inputs x_{10} and x_{20}
- the conclusions w_1^* and w_2^* are obtained as shown in Figure 2.13.

The firing strength of the rules, denoted $\alpha_k, k = 1, 2$ are computed by

$$\alpha_1 = T(A_1(x_{10}), B_1(x_{20})),$$

$$\alpha_2 = T(A_2(x_{10}), B_2(x_{20}))$$

A T_{min} is used as a conjunction operator in Figure 2.13. Each rule conclusion is obtained by:

$$w_1^* = a_1 x_{10} + b_1 x_{20}$$

$$w_2^* = a_2 x_{10} + b_2 x_{20}$$

The the over all system conclusion is obtained by

$$w^* = \frac{\alpha_1 * w_1^* + \alpha_2 * w_2^*}{\alpha_1 + \alpha_2}$$

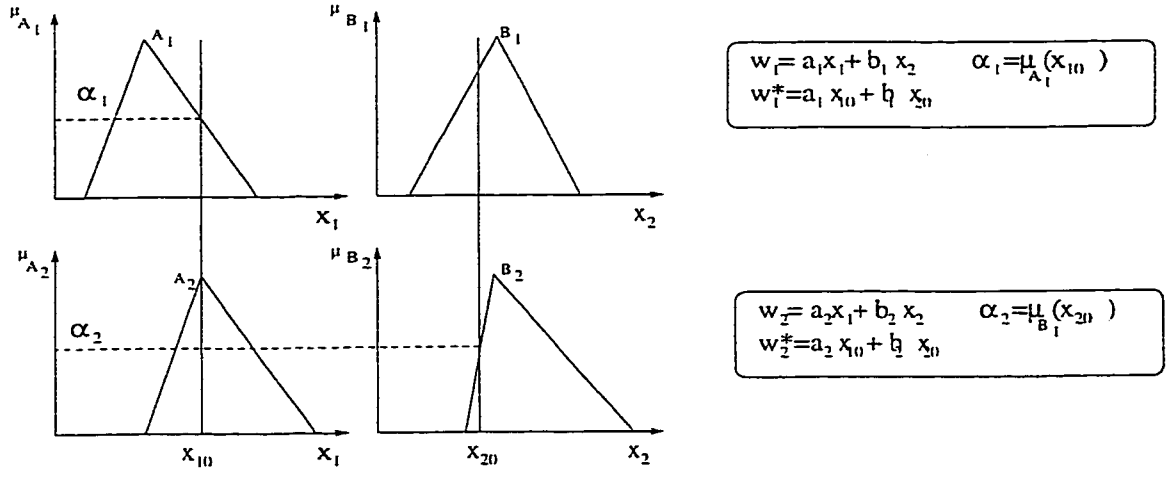

Figure 2.13: Fuzzy inference with two antecedent using Sugeno approach

In practical applications the simplest case of Sugeno rules is that when the consequences are constants, this means that all output membership functions are fuzzy numbers. When the consequent is a constant, we then have a constant consequent Sugeno fuzzy model. Sugeno system has several advantages. From a control engineering prespective the use of local models bridges the gap between fuzzy control and conventional control. Many existing tools and theories in linear systems theory can be partially applied to Sugeno fuzzy models and controllers. The relationship to gain scheduling is evident, as discussed in Chapter 4. Sugeno system is computationally efficient, and better suited to mathematical analysis. However, Mamdani

system has a widespread acceptance and better suited for human inputs. Sugeno fuzzy model is used for modeling and control of nonlinear systems in all applications presented in this thesis.

Chapter 3

Interpolation behavior of Sugeno fuzzy controllers

In recent publications Sugeno reasoning is considered as a powerful tool for system modelling and control. Therefore, it is important to analyze what type of interpolation function does this reasoning type result in. In this chapter, the influence of the shape, the distribution of the membership functions and the order of the functional consequent (in case of Sugeno controller) on the interpolation function of the fuzzy system is investigated. A linear membership function, product conjunction operator, and a general functional consequent are used. Different interpolation behavior can be observed by changing the conjunction operator to minimum, or the membership function to Gaussian.

3.1 Introduction

Many investigations [17, 88, 89] proved that fuzzy systems can approximate a sufficiently regular mapping from inputs to outputs to any degree of accuracy. To achieve a better understanding of the approximation and interpolation capability of fuzzy systems, the influence of the fuzzy set parameters and the reasoning method on the interpolation function of the fuzzy system is needed. In this chapter, we investigate the influence of the shape and the distribution of the membership functions and the order of the functional consequent (in case of Sugeno controller [70]) on the interpolation function of the fuzzy system.

Each rule form a node in the interpolation net of a fuzzy system. The fuzzy system interpolates between these nodes using an interpolation function, which depends on the shape, the distribution of the membership functions, the conjunction operator of the antecedents, and the order of Sugeno functional consequent. A mathematical study of the influence of these parameters on the interpolation function of the fuzzy systems will help achieve a general understanding of these systems.

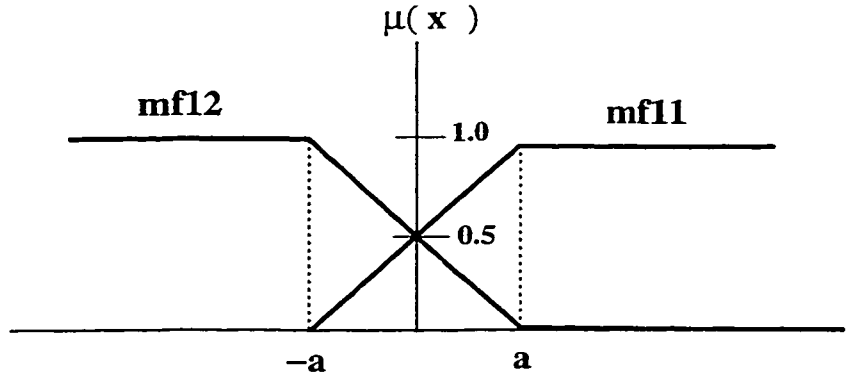

Figure 3.1: Membership functions for input x

3.2 Sugeno controller with single input, 0.5 complementary triangular membership functions and functional consequent

The interpolation function of Sugeno controller with single input, 0.5 complementary triangular membership functions [11], and functional consequent is considered. Consider a two-rule system, where the membership functions are as given below and shown in Figure 3.1. The Sugeno rules have the following format:

IF x is mf_{11} THEN $f(x) = a_{01} + a_{11}x + a_{21}x^2 + ... + a_{n1}x^n$

IF x is mf_{12} THEN $f(x) = a_{02} + a_{12}x + a_{22}x^2 + ... + a_{n2}x^n$

where

$$\mu_{mf11}(x) = \begin{cases} 0 & \text{If} \quad x < -a \\ \frac{x+a}{2a} & \text{If} \quad -a \leq x \leq a \\ 1 & \text{If} \quad x > a \end{cases}$$

$$\mu_{mf12}(x) = \begin{cases} 1 & \text{If} \quad x < -a \\ \frac{-x+a}{2a} & \text{If} \quad -a \leq x \leq a \\ 0 & \text{If} \quad x > a \end{cases}$$

The overall output can be calculated as a weighted sum of the outputs of all rules ($i = 1, 2$). Degree of fulfilment has to be calculated for each rule. The output of the controller is:

$$f(x) = \frac{\sum_{i=1}^{2} \alpha_i (a_{0i} + a_{1i}x + a_{2i}x^2 + ... + a_{ni}x^n)}{\sum_{i=1}^{2} \alpha_i} \tag{3.1}$$

where the degrees of fulfilment, α_1 and α_2, are $\mu_{mf11}(x)$ and $\mu_{mf12}(x)$, respectively. For the considered membership functions, $\mu_{mf11}(x) + \mu_{mf12}(x) = 1$. The overall output can be rewritten as:

$$y = \frac{a_{01}+a_{02}}{2} + \frac{(a_{01}-a_{02})+(a_{11}a+a_{12}a)}{2a}x + ... + \frac{a_{n1}-a_{n2}}{2a}x^{n+1} \tag{3.2}$$

PROPOSITION 1: *The interpolation function of Sugeno controller with single input, 0.5 complementary triangular membership functions, and functional consequent is a polynomial function of the input. The order of this polynomial is one degree higher than that of the functional consequent. Therefore, increasing the order of Sugeno consequent by one will increase the interpolation order by one.*

Only two special cases are proven below, but one can prove it in general.

Special case 1: Sugeno controller with single input and constant consequent

Consider the Sugeno controller with single input, 0.5 complementary membership functions and constant consequent, i.e., $n = 0$ in Equations (3.1) and (3.2). The

overall output can be calculated from the general equation above as:

$$y = \frac{a_{01}-a_{02}}{2a}x + \frac{a_{01}+a_{02}}{2}$$

As this equation shows, the output for a triangular membership function is a linear function of x. This relation is determined by the shape of the membership function, since the consequent of the rules is constant. The above equation shows that only two constants are required to determine the appropriate action for any input. These constants, $\frac{a_{01}-a_{02}}{2a}$ and $\frac{a_{01}+a_{02}}{2}$, are completely determined by the rule base and the midpoints of the triangular membership functions of the input domain. A similar but more complicated expression can be derived for Gaussian membership functions which shows a smoother interpolation behavior, as it is demonstrated in Example 1.

EXAMPLE 1: This example shows that we can achieve a smoother interpolation by changing the shape of the membership function from triangular to Gaussian.

For $x = 0, ..., 10$;

$$y = \frac{\sin(2x)}{e^{(x/5)}} + \frac{rand(\ size(x))}{30}$$

where *rand* is random number. The Sugeno fuzzy model of the above equation generated by ANFIS is shown in Table 3.1. The root mean squared errors (RMSE) is 0.082638. Triangular membership functions are used to partition the input space as shown in Figure 3.2. Each row in this table represents a rule. For example, the first rule is:

If x is in1mf1 THEN $y = 3.83x - 0.04608$

R_i	x	a_{i1}	a_{i0}
R_1	in1mf1	3.83	-0.04608
R_2	in1mf2	2.252	-6.215
R_3	in1mf3	0.8179	-4.535
R_4	in1mf4	0.8164	-5.75
R_5	in1mf5	1.455	-14.34

Table 3.1: The fuzzy model of 5 rules using triangular membership functions

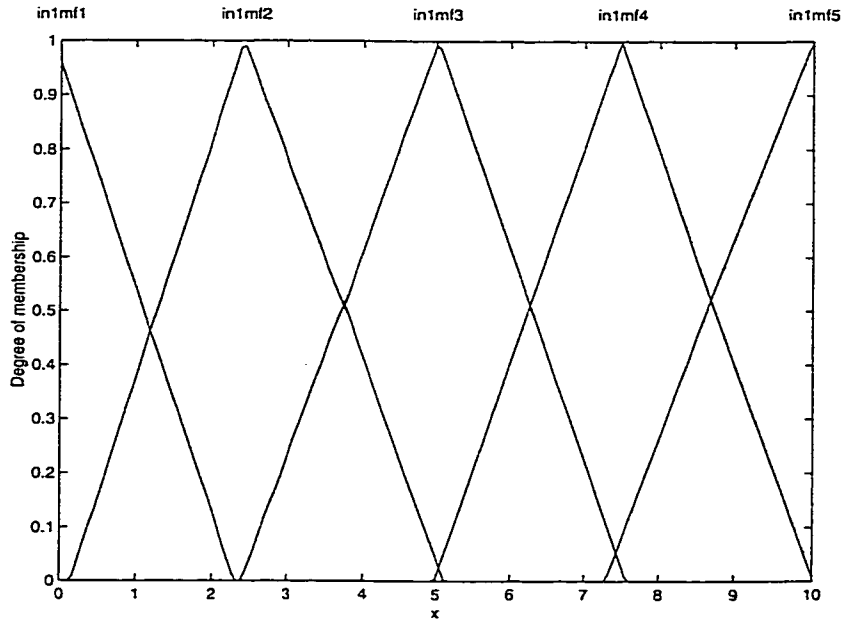

Figure 3.2: Triangular membership functions for the input (x)

where in1mf1,..., in1mf5 are the triangular membership functions for the input (x).

Sugeno output is shown in Figure 3.3. By changing the membership function to Gaussian type the RMSE is reduced to 0.026449. The Sugeno fuzzy model is also generated by ANFIS and shown in Table 3.2. Gaussian membership functions are used to partition the input space as shown in Figure 3.4. Figure 3.5 shows the output of the Sugeno fuzzy model with Gaussian membership functions.

Special case 2: Sugeno controller with single input and linear consequent

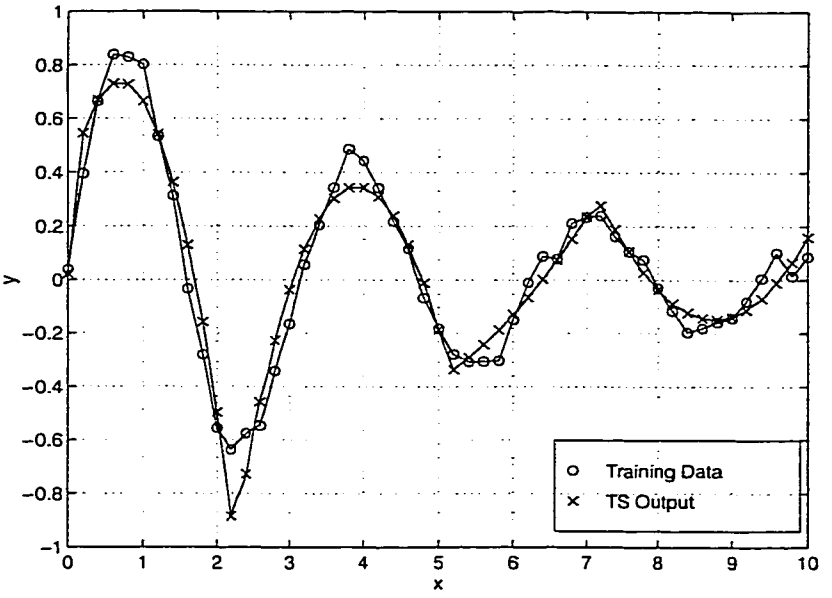

Figure 3.3: Sugeno with 5 rules, linear consequent, and triangular membership function

R_i	x	a_{i1}	a_{i0}
R_1	in1mf1	3.158	0.5444
R_2	in1mf2	1.721	-6.08
R_3	in1mf3	-0.9669	3.979
R_4	in1mf4	-0.8758	6.834
R_5	in1mf5	-0.1023	1.276

Table 3.2: The fuzzy model of 5 rules using Gaussian membership functions

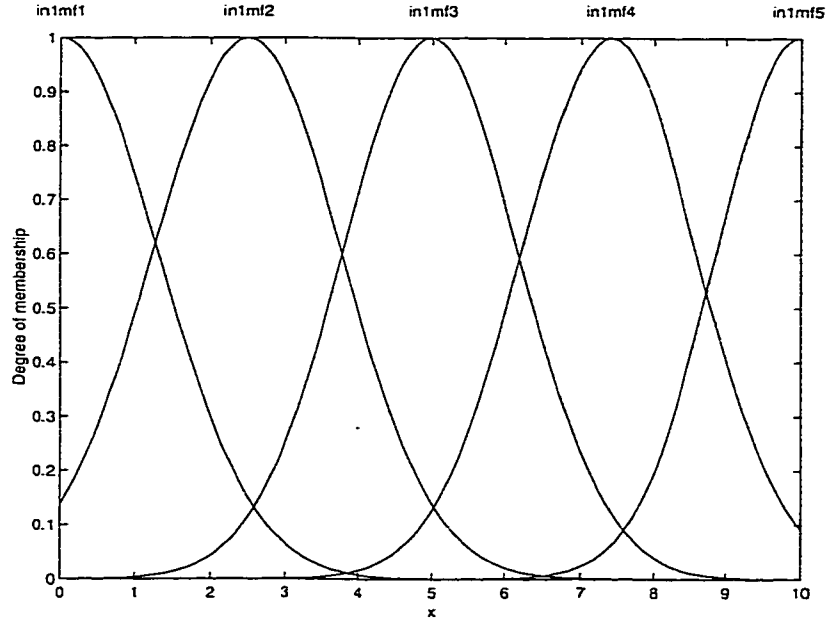

Figure 3.4: Gaussian membership functions for the input (x)

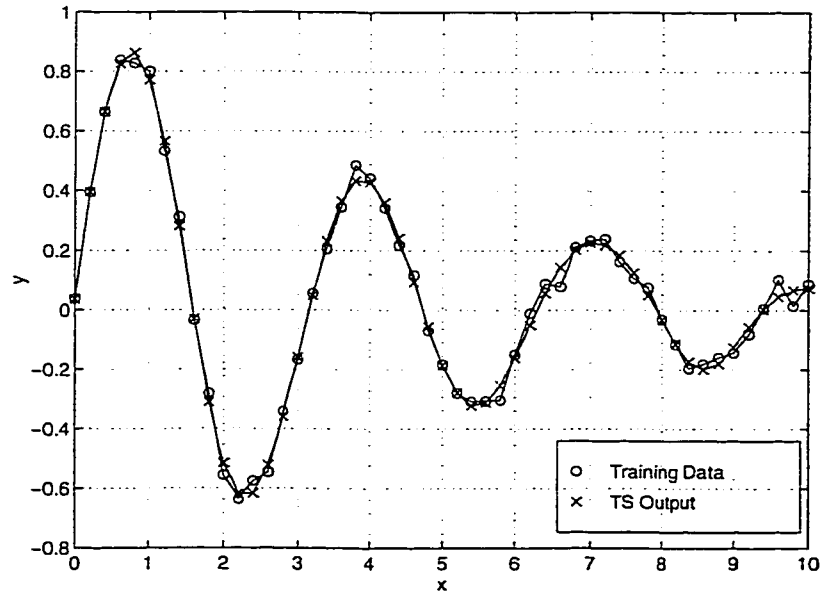

Figure 3.5: Sugeno with 5 rules, linear consequent, and Gaussian membership function

Consider the Sugeno controller with single input and linear consequent, i.e., $n = 1$ in Equations (3.1) and (3.2). The overall output of the controller is:

$$y = \left(\frac{a_{11}-a_{12}}{2a}\right) x^2 + \left(\frac{a_{01}-a_{02}+a_{11}a+a_{12}a}{2a}\right) x + \frac{a_{01}+a_{02}}{2}$$

The overall output is a quadratic function of x. This relation is determined by the shape of the membership function, and the consequent of the rules. This means that the interpolation has a higher degree than that of a constant consequent. Increasing the order of Sugeno consequent by one will increase the interpolation order by one. Higher order interpolation for this kind of Sugeno controller can be achieved by higher order consequent. For example, it can be shown that a second order consequent, and a triangular membership functions for the antecedent will give a third order interpolation.

3.3 Sugeno with single antecedent, noncomplementary membership functions, and functional consequents

The interpolation function of Sugeno controller with single input, noncomplementary triangular membership functions, and functional consequent is considered. Consider the two-rule system in Equation (3.1) with the following membership functions:

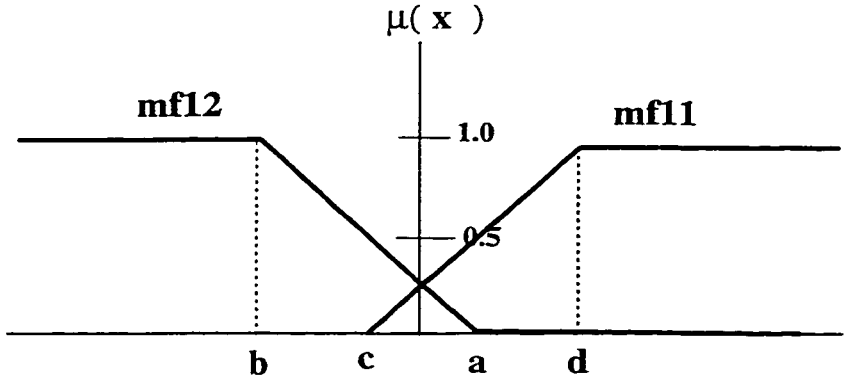

Figure 3.6: Membership functions for input x

$$\mu_{mf11}(x) = \begin{cases} 0 & \text{If} \quad x < c \\ \frac{x-c}{d-c} & \text{If} \quad c \leq x \leq d \\ 1 & \text{If} \quad x > d \end{cases}$$

$$\mu_{mf12}(x) = \begin{cases} 1 & \text{If} \quad x < b \\ \frac{-x+a}{a-b} & \text{If} \quad b \leq x \leq a \\ 0 & \text{If} \quad x > a \end{cases}$$

These membership functions are shown in Figure 3.6. The output of the controller is:

$$f(x) = \frac{\sum_{i=1}^{2} \alpha_i (a_{0i} + a_{1i}x + a_{2i}x^2 + ... + a_{ni}x^n)}{\sum_{i=1}^{2} \alpha_i} \tag{3.3}$$

where the degrees of fulfilment, α_1 and α_2, are $\mu_{mf11}(x)$ and $\mu_{mf12}(x)$, respectively. Then the output can be rewritten as

$$y = \frac{k_0 + k_1 x + k_2 x^2 + ... + k_{n+1} x^{n+1}}{g_0 + g_1 x} \tag{3.4}$$

where,

$k_0 = aa_{02}(d-c) - ca_{01}(a-b)$,

$k_1 = (a_{01} - ca_{11})(a-b) + (aa_{12} - a_{02})(d-c)$,

$k_2 = (a_{11} - ca_{21})(a - b) + (aa_{22} - a_{12})(d - c)$,

$k_{n+1} = a_{n1}(a - b) - a_{n2}(d - c)$,

$g_0 = ad + cb - 2ac$,

$g_1 = a + c - b - d$.

PROPOSITION 2: *The interpolation function of Sugeno controller with single input, noncomplementary triangular membership functions, and functional consequent is a rational function. The numerator is a polynomial. The order of this polynomial depends on the order of the consequent. The denominator is a linear function of the input.*

Two special cases are proven below.

Special case 1: Sugeno with single antecedent, noncomplementary membership functions, and constant consequent

The overall output can be calculated from the general equation above, for $n = 0$ in Equations (3.3) and (3.4). The overall output of the controller is

$$y = \frac{(a_{01}(a-b)-a_{02}(d-c))x+(aa_{02}(d-c)-ca_{01}(a-b))}{(a+c-b-d)x+(ad+cb-2ac)}$$

The output is not a linear function of x. This relation is determined by the shape and the distribution of the membership functions. In this case the fuzzy system performs a nonlinear interpolation. The output is a rational function. The numerator and the denominator are of the same order, since the consequent is constant.

Special case 2: Sugeno with single antecedent, noncomplementary membership functions, and linear consequent

When the consequent is a linear function of the input, i.e., $n = 1$ in Equations (3.3) and (3.4). The overall output of the controller is:

$$y = \frac{\alpha x^2 + \beta x + \gamma}{(a+c-b-d)x + (ad+cb-2ac)}$$

where,

$\alpha = (a_{11}(a-b) - a_{12}(d-c))$,

$\beta = (a_{01} - ca_{11})(a-b) + (aa_{12} - a_{02})(d-c)$,

$\gamma = aa_{02}(d-c) - ca_{01}(a-b)$.

The output is a second order function of x in the numerator, which is determined by the shape of the membership function, and the consequent of the rules. The denominator is a linear function of x, which is determined by the distribution and the shape of the membership functions. Increasing the order of consequents will increase the order of the numerator, but will not change the order of the denominator.

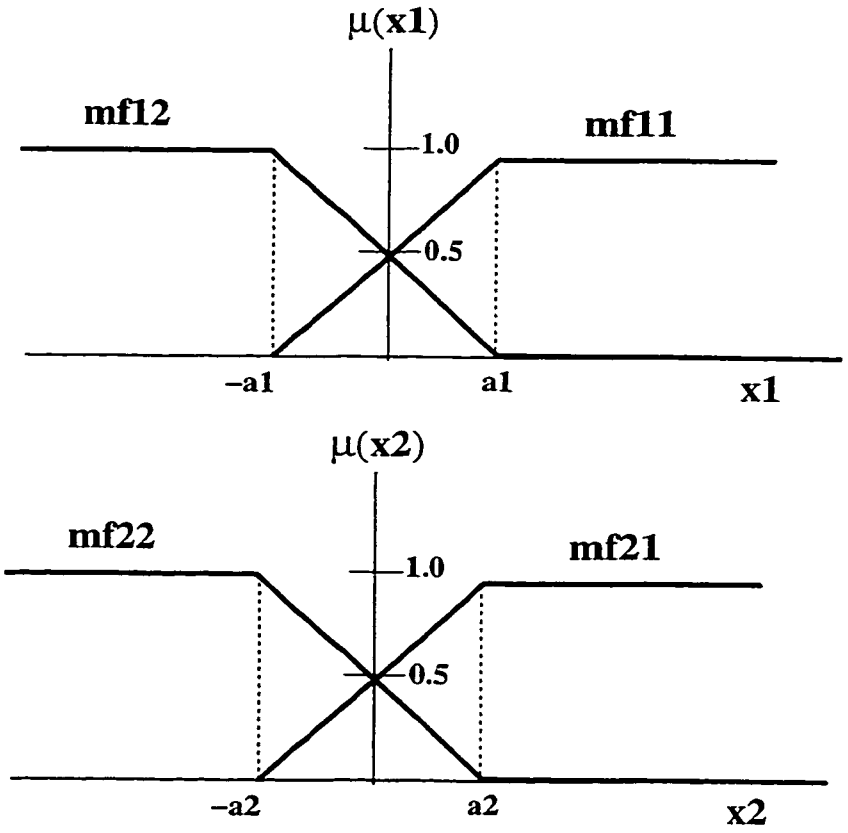

Figure 3.7: Membership functions for input x_1, and x_2

3.4 Sugeno controller with two inputs, 0.5 complementary membership functions and functional consequents

The interpolation function of Sugeno controller with two inputs, 0.5 complementary triangular membership functions, product conjunction operator, and functional consequents is considered in this section. Consider the case of two inputs, x_1 and x_2, with two fuzzy partitions in each input space, as shown in Figure 3.7.

Sugeno rules with two inputs and functional consequent are:

IF x_1 is mf_{11} AND x_2 is mf_{21} THEN $y = f_1(x_1, x_2)$

IF x_1 is mf_{12} AND x_2 is mf_{21} THEN $y = f_2(x_1, x_2)$

IF x_1 is mf_{11} AND x_2 is mf_{22} THEN $y = f_3(x_1, x_2)$

IF x_1 is mf_{12} AND x_2 is mf_{22} THEN $y = f_4(x_1, x_2)$

Another factor is added in these rules that will affect the interpolation function of Sugeno controller. This factor is the conjunction operator of the antecedents. Degree of fulfilment for each rule is:

$\alpha_1 = T(\mu_{mf11}(x_1), \mu_{mf21}(x_2))$

$\alpha_2 = T(\mu_{mf12}(x_1), \mu_{mf21}(x_2))$

$\alpha_3 = T(\mu_{mf11}(x_1), \mu_{mf22}(x_2))$

$\alpha_4 = T(\mu_{mf12}(x_1), \mu_{mf22}(x_2))$

where T is a t-norm, which will be chosen as product in this chapter. For $T =$ *product*

$\alpha_1 = \mu_{mf11}(x_1)\mu_{mf21}(x_2) = \frac{x_1x_2+a_2x_1+a_1x_2+a_1a_2}{4a_1a_2}$

$\alpha_2 = \mu_{mf12}(x_1)\mu_{mf21}(x_2) = \frac{-x_1x_2-a_2x_1+a_1x_2+a_1a_2}{4a_1a_2}$

$\alpha_3 = \mu_{mf11}(x_1)\mu_{mf22}(x_2) = \frac{-x_1x_2+a_2x_1-a_1x_2+a_1a_2}{4a_1a_2}$

$\alpha_4 = \mu_{mf12}(x_1)\mu_{mf22}(x_2) = \frac{x_1x_2-a_2x_1-a_1x_2+a_1a_2}{4a_1a_2}$

The overall output of the controller is:

$$y = \sum_{i=1}^{m} \alpha_i ABC \tag{3.5}$$

where

$$A = \begin{bmatrix} 1 & x_2 & \cdots & x_2^n \end{bmatrix}$$

$$B = \begin{bmatrix} a_{0i} & a_{1i} & \cdots & a_{ni} \\ a_{(n+1)i} & a_{(n+2)i} & \cdots & a_{(2n+1)i} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n(n+1)i} & a_{(n(n+1)+1)i} & \cdots & a_{(n(n+1)+n)i} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_1^n \end{bmatrix}$$

PROPOSITION 3: *The interpolation function of Sugeno controller with two inputs, 0.5 complementary triangular membership functions, product conjunction operator, and a functional consequent is a polynomial function of all inputs. The order of this polynomial depends on the order of the functional consequent.*

The following are two special cases.

Special case 1: Sugeno controller with two inputs and a constant consequent

Consider the Sugeno controller with two inputs and a constant consequent, i.e., $n = 0$ in Equation (3.5). The overall output of the controller is

$$y = \alpha_1 [a_{01}] + \alpha_2 [a_{02}] + \alpha_3 [a_{03}] + \alpha_4 [a_{04}] .$$

Let us rewrite this as:

$$y = k_{01} + k_{02}x_1 + k_{03}x_2 + k_{04}x_1x_2$$

where

$k_{01} = \frac{a_{01}+a_{02}+a_{03}+a_{04}}{4}$,

$k_{02} = \frac{a_{01}-a_{02}+a_{03}-a_{04}}{4a_1}$,

$k_{03} = \frac{a_{01}+a_{02}-a_{03}-a_{04}}{4a_2}$,

$k_{04} = \frac{a_{01}-a_{02}-a_{03}+a_{04}}{4a_1a_2}$.

The output is a linear interpolation with respect to each input (this includes the interaction term x_1x_2).

Example 2: For $x_1 = 0, \ldots, 4$, and for $x_2 = 0, \ldots, 4$

$$z = x_1^2 + x_2^2$$

The fuzzy model of the above equation is shown in Table 3.3. Each row in this table represents a rule. For example, the first rule is:

If x_1 is mf1 AND x_2 is mf1 THEN $z = 0$

where mf1,..., mf3 in the second column of Table 3.3 are membership functions for the first input (x_1) and mf1,..., mf3 in the third column of Table 3.3 are membership functions for the second input (x_2).

The output of a fuzzy system with 9 rules, triangular membership functions, product conjunction operator and constant consequent is shown in Figure 3.8. The membership functions for x_1 and x_2 are shown in Figures 3.9, and 3.10. The interpolation between the nodes is linear. Different interpolation behavior between the nodes can be observed by changing the conjunction operator to minimum, Figure

R_i	x_1	x_2	z
R_1	mf1	mf1	0
R_2	mf1	mf2	4
R_3	mf1	mf3	16
R_4	mf2	mf1	4
R_5	mf2	mf2	8
R_6	mf2	mf3	20
R_7	mf3	mf1	16
R_8	mf3	mf2	20
R_9	mf3	mf3	32

Table 3.3: The fuzzy model of 9 rules

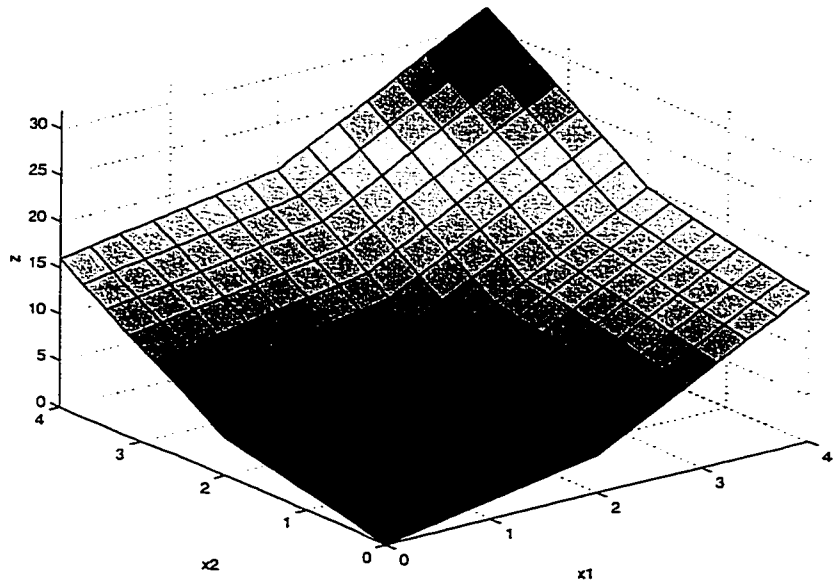

Figure 3.8: Sugeno FIS with 9 rules, constant consequent, triangular membership function and product conjunction

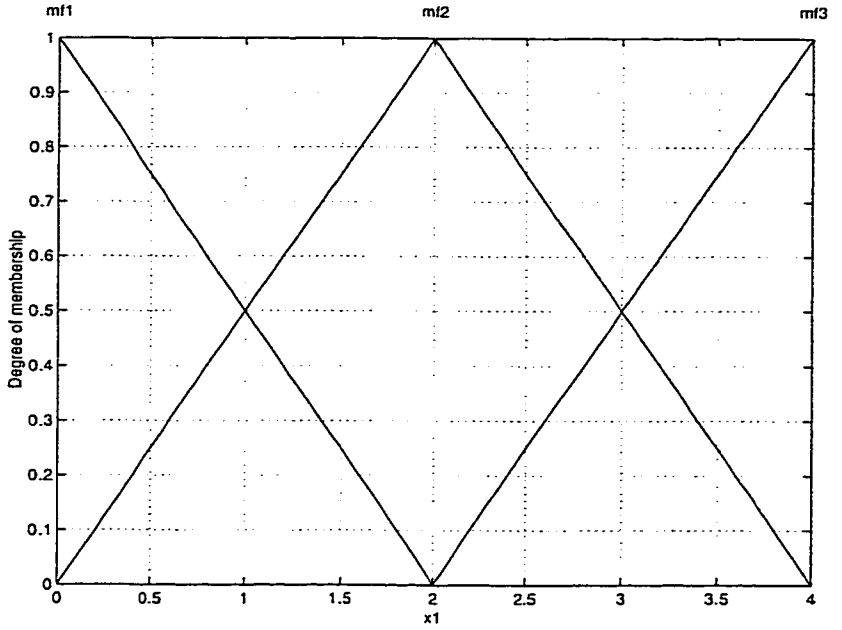

Figure 3.9: Triangular membership functions for input x_1

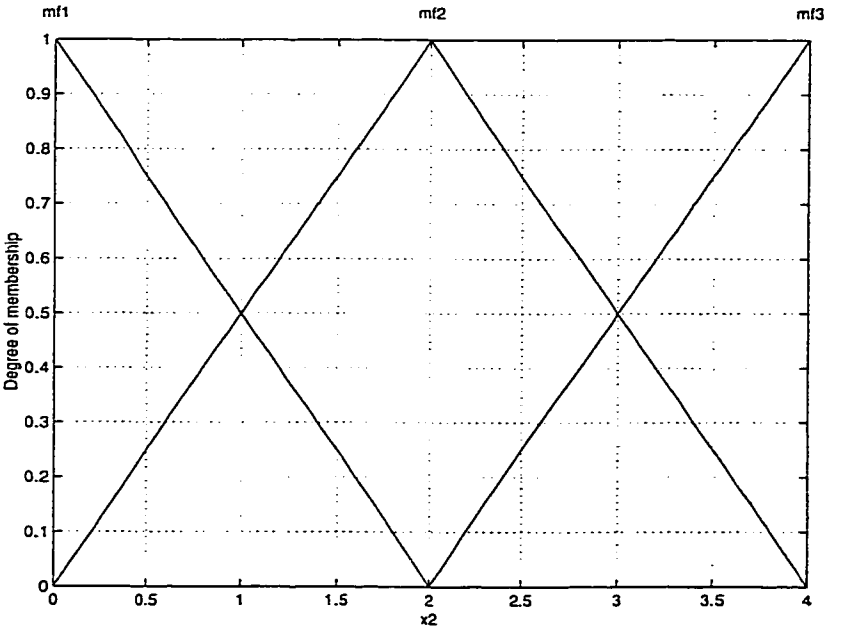

Figure 3.10: Triangular membership functions for input x_2

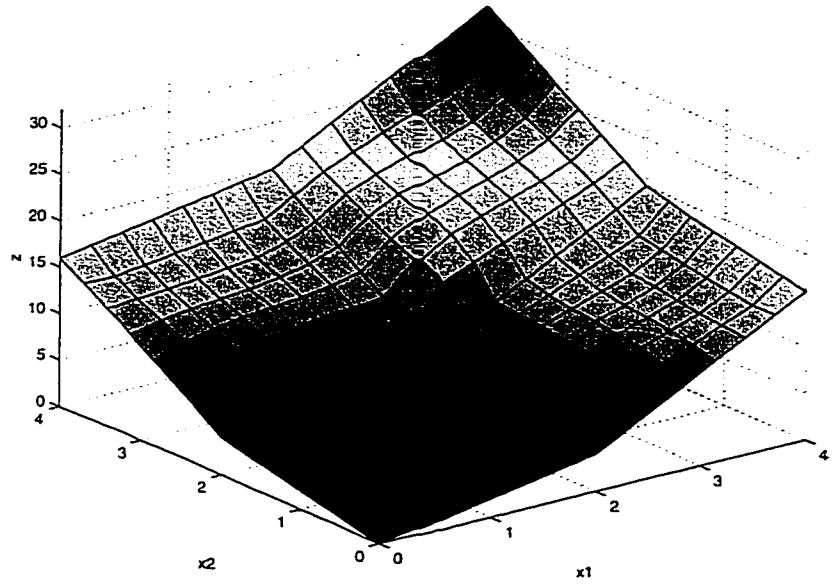

Figure 3.11: Sugeno FIS with 9 rules, constant consequent, triangular membership function and minimum conjunction

3.11, or by changing the shape of the membership function to Gaussian, Figure 3.12. The Gaussian membership functions for x_1 and x_2 are shown in Figures 3.13, and 3.14.

Special case 2: Sugeno controller with two inputs and bilinear consequent

When the consequent is a linear function with respect to each input (called bilinear consequent), i.e., $n = 1$ in Equation (3.5), the overall output of the controller is:

$$y = g_0 + g_1 x_1 + g_2 x_2 + g_3 x_1 x_2$$

where

$g_0 = k_{01} + k_{02} x_1 + k_{03} x_2 + k_{04} x_1 x_2,$

$g_1 = k_{11} + k_{12} x_1 + k_{13} x_2 + k_{14} x_1 x_2,$

$g_2 = k_{21} + k_{22} x_1 + k_{23} x_2 + k_{24} x_1 x_2,$

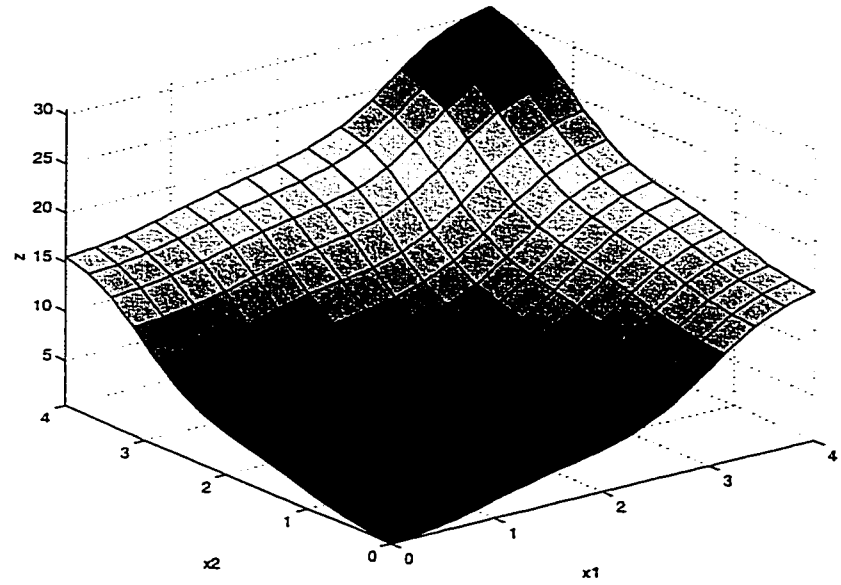

Figure 3.12: Sugeno FIS with 9 rules, constant consequent, Gaussian membership function and product conjunction

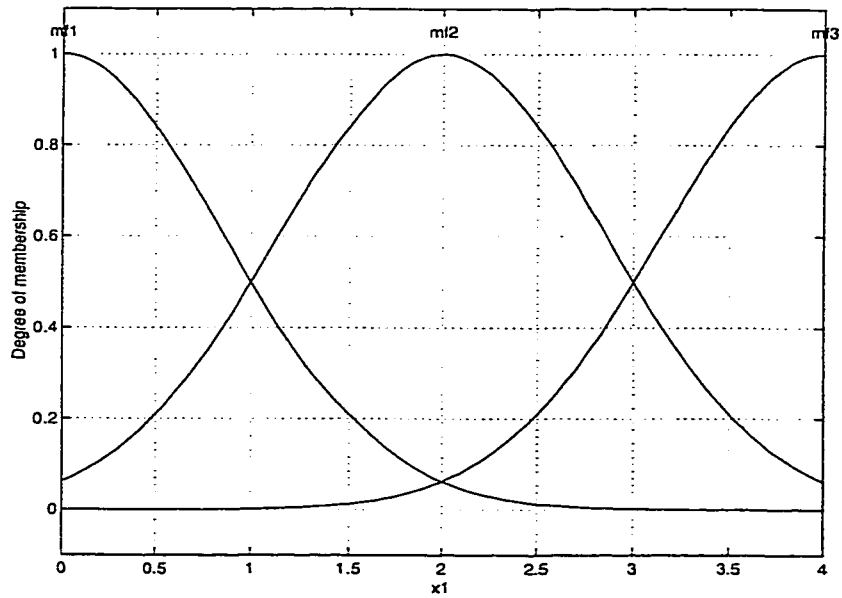

Figure 3.13: Gaussian membership functions for input x_1

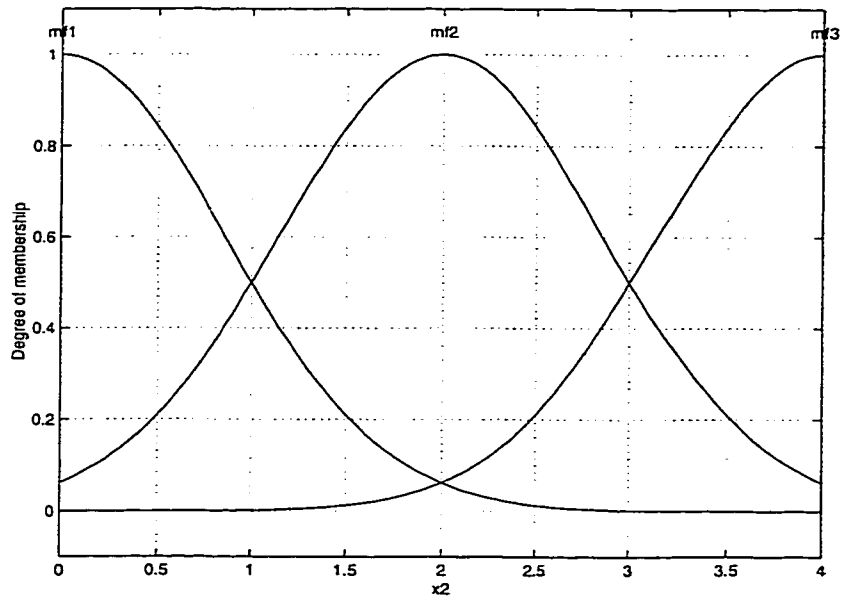

Figure 3.14: Gaussian membership functions for input x_2

$g_3 = k_{31} + k_{32}x_1 + k_{33}x_2 + k_{34}x_1x_2,$

$k_{i1} = \frac{a_{i1}+a_{i2}+a_{i3}+a_{i4}}{4},$

$k_{i2} = \frac{a_{i1}-a_{i2}+a_{i3}-a_{i4}}{4a_1},$

$k_{i3} = \frac{a_{i1}+a_{i2}-a_{i3}-a_{i4}}{4a_2},$

$k_{i4} = \frac{a_{i1}-a_{i2}-a_{i3}+a_{i4}}{4a_1a_2}.$

The output of the controller is a second order interpolator with respect to each input (this includes all interaction terms, such as $x_1^2x_2^2$). In the above case, a more general consequent (includes all interaction terms) than that of Sugeno [70] is used. If the interaction terms between the input variables are not considered, the output of the controller will still be a second order interpolator with respect to each input without the interaction terms.

In general, it can be shown by induction that for a controller with any number of inputs, product conjunction operator, constant consequent and 0.5 complementary triangular membership functions, the output is a linear function with respect to each input. Increasing the order of TS consequent by one will increase the interpolation order by one with respect to each input.

3.5 Sugeno with two inputs, noncomplementary membership functions, and a functional consequent

The interpolation function of Sugeno controller with two inputs, noncomplementary membership functions, product conjunction operator and a functional consequent is considered in this section. Consider the case of two inputs, x_1 and x_2, with two fuzzy partitions of each input space, as shown in Figure 3.15.

The overall output of the controller is:

$$y = \frac{\sum_{i=1}^{m} \alpha_i ABC}{\sum_{i=1}^{m} \alpha_i} \tag{3.6}$$

where A, B, C and α_i are defined in Section 3.4.

PROPOSITION 4: *The interpolation function of Sugeno controller with two inputs, noncomplementary triangular membership functions, and functional consequent is a polynomial function of each input in the numerator. The order of this*

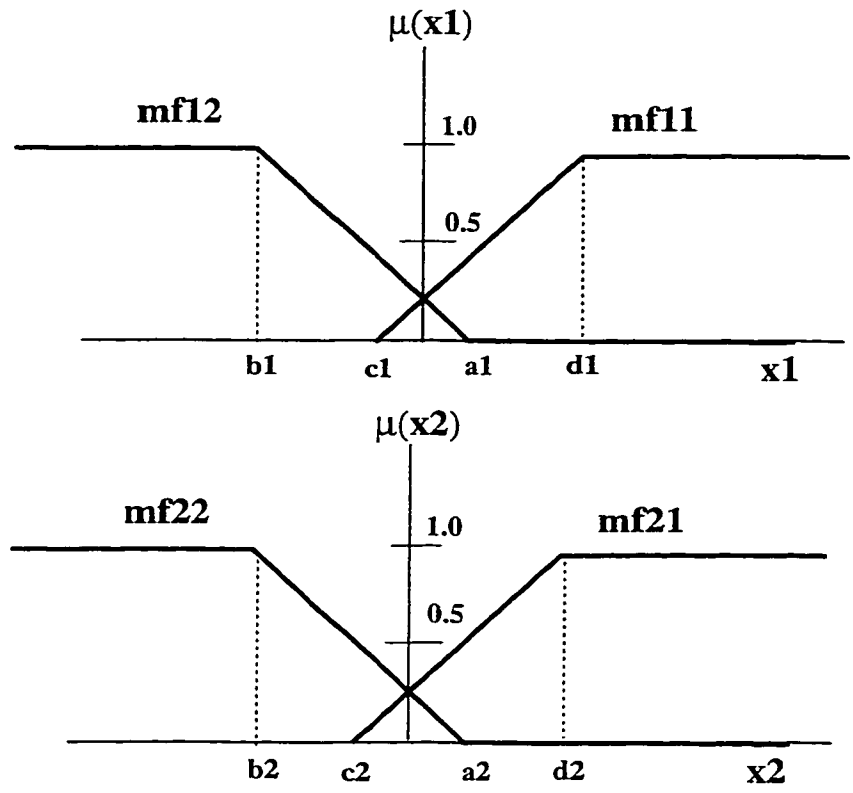

Figure 3.15: Membership functions for input x_1 and x_2

polynomial depends on the order of the rule consequents. The denominator is a linear function of all inputs.

A special case is considered and proven below:

Special case: Sugeno with two inputs, noncomplementary triangular membership functions, product conjunction operator and constant consequent

When the consequent is a constant, i.e., $n = 0$ in Equation (3.6), the overall output of the controller is:

$$y = \frac{k_{04}x_1x_2 + k_{03}x_1 + k_{02}x_2 + k_{01}}{k_{08}x_1x_2 + k_{07}x_1 + k_{06}x_2 + k_{05}}$$

where

$k_{04} = r_1a_{01} - r_2a_{02} - r_3a_{03} + r_4a_{04}$,

$k_{03} = -r_1a_{01}c_2 + r_2a_{02}c_2 + r_3a_{03}a_2 - r_3a_{04}a_2$,

$k_{02} = -r_1a_{01}c_1 + r_2a_{02}a_1 + r_3a_{03}c_1 - r_4a_{04}a_1$,

$k_{01} = r_1c_1c_2 - r_2a_1c_2 - r_3a_2c_1 + r_4a_1a_2$,

$k_{08} = r_1 - r_2 - r_3 + r_4$,

$k_{07} = -r_1c_2 + r_2c_2 + r_3a_2 - r_4a_2$,

$k_{06} = -r_1c_1 + r_2a_1 + r_3c_1 - r_4a_1$,

$k_{05} = r_1c_1c_2 - r_2a_1c_2 - r_3a_2c_1 + r_4a_1a_2$,

$r_1 = \frac{1}{(d_1-c_1)(d_2-c_2)}$,

$r_2 = \frac{1}{(a_1-b_1)(d_2-c_2)}$,

$r_3 = \frac{1}{(d_1-c_1)(a_2-b_2)}$,

$r_4 = \frac{1}{(a_1-b_1)(a_2-b_2)}$.

In general, one can show that for a controller with any number of inputs, product conjunction operator, constant consequent and noncomplementary triangular membership functions, the output is a rational function, which is a linear function with respect to each input in the numerator as well as in the denominator. Increasing the order of Sugeno consequent by one will increase the order of the function in the numerator by one with respect to each input, while the denominator will remain a linear function with respect to each input.

Chapter 4

Fuzzy control of nonlinear system based on known dynamic: Fuzzy gain scheduling approach

In this chapter, an interactive procedure is presented for controller design of nonlinear systems by integrating available classical as well as modern tools such as fuzzy logic, and neural networks. The proposed approach is based on quasi-linear dynamic models of the plant. Classical optimal controllers for each set of operating conditions were developed. These controllers are used to construct a single fuzzy-logic gain scheduling-like controller. Adaptive-neuro-fuzzy inference system was used to construct the rules for the fuzzy gain schedule. This will guarantee the continuous change in the gains as the system parameters change in time or space. This procedure is systematic and can be used to design controllers for many nonlinear systems.

Two degrees of freedom planar manipulator was chosen to show the effectiveness of the proposed approach. A robot manipulator is inherently unstable and displays a strong nonlinearity. The resulting system is stable for different reference trajectories. The system is robust for wide range of driving frequencies of the input. Also, the proposed approach is applied on a well known bench mark system which is the inverted pendulum.

4.1 Introduction

The cooperative use of fuzzy logic and neural networks to control a nonlinear dynamic systems found many publications [11, 12, 45, 47, 49, 51, 57, 70, 90–94] due to their ability to deal with complex systems and effective in nonlinear mappings. In spite of the learning ability of neural networks, it has some limitations. Its architecture depends on expert's decision, and there is no guidelines to determine the number of layers. Fuzzy logic can control the system with partial knowledge of the systems dynamics, and provides a compact structure of rule representation, but it lacks the learning ability. On the other hand, in many cases of more complex and highly nonlinear systems, the conventional technology is not sufficiently strong to represent and implement the knowledge needed for powerful solutions. Some of these approaches have complementing strengths that may overcome certain limitations. Recent research trend indicates a use of combined approach.

The proposed approach is based upon using fuzzy logic to blend the individual point designs such that for any trajectory in the parameter space, the system performs adequately. The ability to systematically design such a dynamic scheduler is a major contribution. The goal of this research is to derive and explore a technique to design a fuzzy-logic gain scheduling-like controllers for nonlinear plants using point-wise designs that adequately span the parameter space of the plant to be controlled. The ability to base the controller on point-wise designs allows the designer to use all available tools of classical, modern and robust control theory to aid in the solution.

In this chapter, the proposed approach utilizes the advantages of gain scheduling, fuzzy logic control, neural networks, and optimal control theory. This approach starts with linearizing the plant dynamics about different equilibrium points. The effect of various parameters can be introduced and studied. This results in a linear model of the system. Using one of the optimal control algorithms such as LQR, a set of gains are obtained for each equilibrium point. Subtractive clustering and ANFIS are used to construct the rules for the fuzzy gain schedule. The fuzzy gain schedule is used to continuously provide the controller with gains as a function of the states of the system. Conventional gain schedule cannot work efficiently when controlling a fast varying system, which is a major drawback. The fuzzy gain schedule provides a smooth gains without any discontinuities. Therefore, there will be no spikes in the response due to jumping from a set of gains to another.

4.2 Linear quadratic regulator

LQR can be used to design an optimal controller. The LQR cost function is the sum of the steady state mean square weighted state x, and the steady state mean square weighted control effort u. The linearized system can be expressed in the state space as follows

$$\dot{x} = Ax + Bu \tag{4.1}$$

The LQR cost function is

$$J_{lqr} = \frac{1}{2}\int_{t_0}^{\infty} (x(t)^T Q x(t) + u(t)^T R u(t))dt \tag{4.2}$$

where Q and R are positive semidefinite weight matrices; the first term penalizes deviation of x from zero, and the second term represents the cost of using the actuator effort. Standard assumptions are considered, (Q, A) are observable, and (A, B) are controllable. With these assumptions, there exists a controller (K_{lqr}) that achieves the smallest LQR cost, which is simply a constant state feedback. This controller

$$K_{lqr} = -R^{-1}B^T X_{lqr} \tag{4.3}$$

where X_{lqr} denotes the unique positive definite solution of the algebraic Riccati equation

$$A^T X_{lqr} + X_{lqr} A - X_{lqr} B R^{-1} B^T X_{lqr} = 0 \tag{4.4}$$

then.

$$u = -K_{lqr}x \tag{4.5}$$

$$\dot{x} = (A - BK)x \tag{4.6}$$

4.3 Subtractive Clustering

Subtractive clustering proposed by Chiu [71], is considered as an alternative to the mountain clustering algorithm. In this algorithm all data points are considered as candidates for cluster centers, this will solve the problem of computational complexity in mountain clustering when the dimension of the problem under consideration is increased. In subtractive clustering the computational complexity is proportional to the number of data point and has nothing to do with the dimension of the problem. The density measure at any point x_i is equal to

$$P_i = \sum_{j=1}^{N} \exp(-\frac{\gamma \left\| x_i - x_j \right\|^2}{r_a^2}) \tag{4.7}$$

where x_i is the i'th data point and N is the total number of data points, γ is a positive constant and is selected as 4, r_a is a constant that define the neighbourhood (data points outside this radius will contribute less to the density measure). The data point with the highest potential (density measure) is selected as the first cluster center. To find the next cluster center, we reduce the effect of the previous identified cluster center and the data points near this center by revising the density measure,

this is done by subtraction as shown in the following equation:

$$P_i = P_i - P_k^* \exp(-\frac{\gamma \left\| x_i - c_k \right\|^2}{r_b^2}) \tag{4.8}$$

$$r_b = \eta * r_a \tag{4.9}$$

where P_k^* is the potential of the k'th cluster center and c_k is the k'th cluster center, r_b is a constant defines the efficient subtractive range, η is a positive constant called squash factor, r_b is a positive constant greater than r_a which helps avoiding closely spaced cluster centers.

After subtractions the second cluster center is selected based on its new potential in relation to an upper acceptance threshold $\bar{\epsilon}$ called accept ratio, lower rejection threshold $\underline{\epsilon}$ called reject ratio, and a relative distance criterion. This process is kept repeated until a sufficient number of cluster centers is identified in the input and output space (see [24, 73] for more details). Subtractive clustering has four significant parameters, accept ratio $\bar{\epsilon}$, reject ratio $\underline{\epsilon}$, cluster radius r_a and squash factor η (or r_b). These parameters have influence on the number of rules and the error performance measures. For example, a large value of r_a generally results in fewer clusters that lead to a coarse model. On the other hand, a small value of r_a can produce excessive number of rules that may result in an over-defined system. The optimal parameters suggested by Chiu are $1.25 \leq \eta \leq 1.5$, $0.15 \leq r_a \leq 0.30$, $\bar{\epsilon}$=0.5 and $\underline{\epsilon}$=0.15. The membership functions of all data points in each input space are assigned exponentially as proposed by Chiu [71] with respect to all cluster centers

as follows:

$$\mu_{ij} = \exp(\frac{-\gamma \left\| x_i - c_k \right\|^2}{r_a^2}) \tag{4.10}$$

where $\|x_i - c_k\|$ is the distance measure between the i'th data point and k'th cluster center.

4.4 Adaptive network-based fuzzy inference system

Adaptive network-based fuzzy inference system (ANFIS) [95, 96] combines the structure of fuzzy logic controllers with the learning aspects from neural networks. If human expertise is not available, we can still set up intuitively reasonable initial membership functions and start the learning process to generate a set of fuzzy if-then rules to approximate a desired data set. ANFIS uses a highly efficient training method that combines gradient descent and least-squares optimization to improve training speed compared to standard backpropagation. Summarizing from [95], ANFIS implements a first order Sugeno fuzzy system. ANFIS structure is shown in Figure 4.1. Layer 1 consists of membership functions described by generalized bell function. Layer 2 implements the fuzzy AND operator, while layer 3 acts to scale or normalize the firing strengths. The output of the forth layer is a linear combination of the inputs multiplied by the normalized firing strength. Layer 5 is a simple

summation of the outputs of layer 4. Layers 1 contains premise modifiable parameters, and layer 4 contains consequent parameters. The consequent parameters are identified by a least squares estimator in the forward pass. In the backward pass, the consequent parameters are held fixed, and the premise parameters are modified using gradient descent. The user specified information is the number of membership functions for each input, the membership type, and the input-output training information.

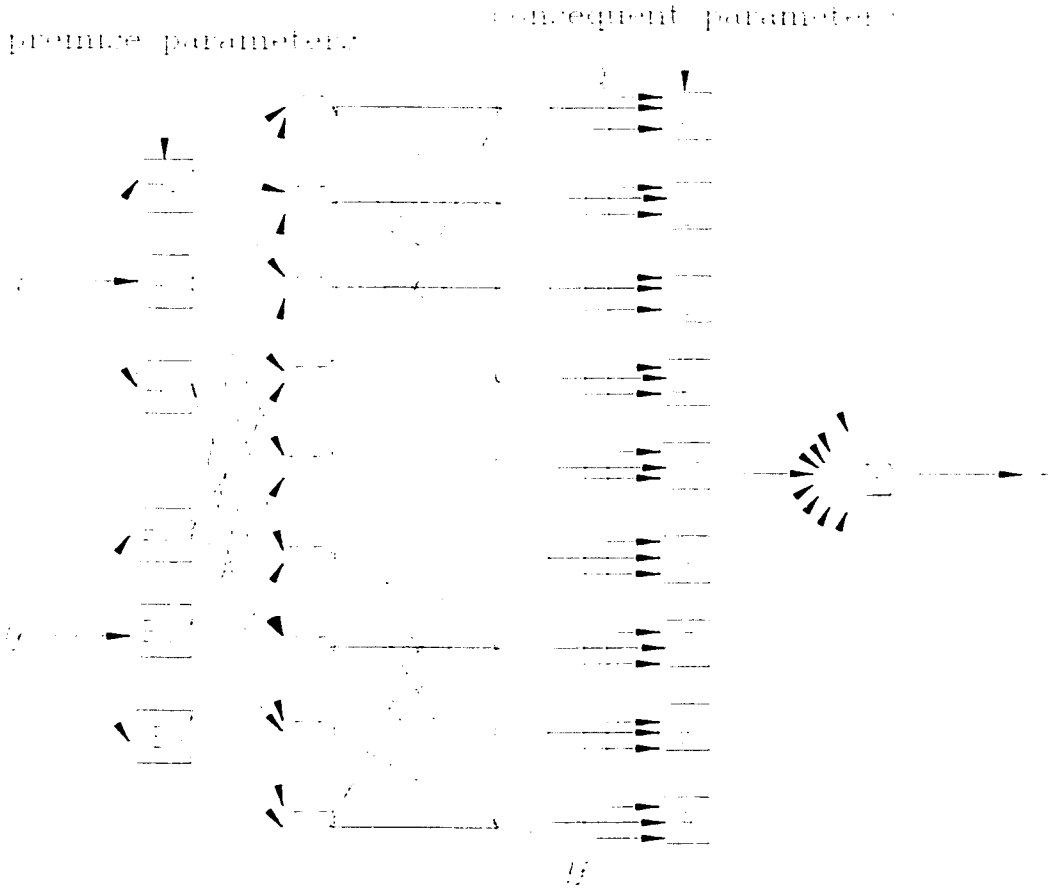

Figure 4.1: ANFIS structure

4.5 Continuous gain scheduling

The basic idea of gain scheduling is to linearize the system at different operating regions. Using a linear control design approach, a gain matrix K is associated with every region. A supervisor controller continuously monitors the state of the system and identifies the operating region. Once a region is identified, the controller uses the associated gain K as in (4.6) or interpolates between the gains. Similar to conventional gain scheduling, continuous gain scheduling gains are obtained from the corresponding linear models of the plant under consideration. However, the continuous gains are obtained from finely spaced points over the operating domain of the plant with the corresponding optimal gains used as points to be tabulated as functions of the state of the system. This table of input-output data is used to construct the rules for the fuzzy gain scheduler. This procedure will be repeated for all elements of the gain matrix. Therefore, we will have $n \times m$ fuzzy systems that approximate the gain matrix (n is the number of the states and m is the number of inputs).

Note that this will effectively approximate the solution of the algebraic Riccati equation over the state space. The solution of the algebraic Riccati equation over the state space is complicated and expensive because to achieve optimality, the equation has to be solved at every sample time. On the other hand, our approach

approximates this behaviour by solving the equation a number of times which corresponds to the number of operating regions and interpolates between these points using a fuzzy approximator.

The stability of the proposed approach is better than that of both coarse and fine gain scheduler. One can argue about stability of the system from the improvements over the classical gain scheduling. Conceptually, when the system moves from one operating region to another, the gains will evolve in a smooth manner regardless of the speed of variation or motion. This overcomes the problem of switching between gains in the classical controller, especially, with fast varying systems. Consequently, if there exists a stable classical gain scheduling controller, the corresponding continuous fuzzy one is also stable with smoother response.

4.6 applications

The proposed approach of fuzzy gain scheduling is used to control a two-link robot manipulator. The system is tested for different reference trajectories and for a wide range of driving frequencies of the inputs. Also, the proposed approach is implemented on the unstable inverted pendulum, as shown in Chapter 6.

4.6.1 LQR-Neuro-Fuzzy controller of a two-link robot manipulator

Control of robotic manipulators is a difficult task, due to the complexity of the dynamic model of the arm. The equation of motion of a robotic arm is of the following general form:

$$M(\theta)\ \ddot{\theta} + c(\theta, \dot{\theta}) + h(\theta) = \tau \tag{4.11}$$

where $M(\theta)\ \ddot{\theta}$ represents the inertia associated with the distribution of mass, $c(\theta, \ddot{\theta})$ is the velocity coupling due to the centrifugal and Coriolis forces, $h(\theta)$ is the loading due to gravity, and τ is the applied torque. For an n-axis robot, the manipulator is modelled as a simultaneous system of n highly coupled nonlinear second order differential equations. A two link robot manipulator is shown in Figure 4.2. Almost all of the advanced robot control methods are difficult to implement in reality. This is due to the fact that obtaining an accurate mathematical model of the robot is not an easy task. Also, implementation of these controllers is difficult because it requires a lot of computations and needs a very fast microprocessors. Fuzzy controller does not require the model of the system. It can control a robot without solving complicated equations. In recent years, some papers have appeared in the literature using a combination of fuzzy control and well-known control techniques, This combination takes advantage of both controllers. One of the earliest studies

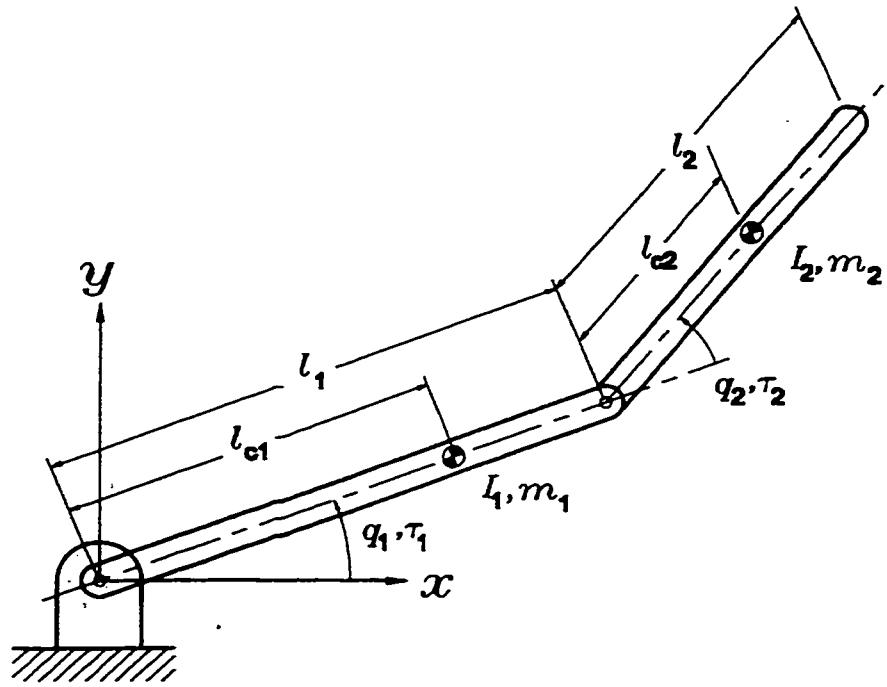

Figure 4.2: Two dof planar manipulator

was conducted by Scharf and Mandic [97] in 1985. Also, a significant work was done by Wakileh and Gill [59]. Lim and Hiyama [98] proposed a method which is based on a conventional PI and a Fuzzy control in 1991. Two other methods which are based on a combination of sliding mode and fuzzy control were presented by Tsay and huang [61], and Hsu and Fu [62]. Most of the industrial robots are designed to work in a well-structured environments. But they may still face parametric changes such as friction and disturbance torques. For simplicity, it is assumed that the links masses are concentrated either at the ends or at the centers of the links, which is not the case in reality. As a result of these uncertainties, it is very difficult to obtain the exact mathematical model of the robot. The closed form dynamic equations in

terms of θ_1 and θ_2 are given below:

$$d_{11}\ \ddot{\theta}_1 + d_{12}\ \ddot{\theta}_2 + h_1 + \phi_1 = \tau_1 \tag{4.12}$$

$$d_{12}\ \ddot{\theta}_1 + d_{22}\ \ddot{\theta}_2 + h_2 + \phi_2 = \tau_2 \tag{4.13}$$

From the above equations:

$$\ddot{\theta}_1 = \frac{1}{\Delta}\left[d_{22}(\tau_1 - h_1 - \phi_1) - d_{12}(\tau_2 - h_2 - \phi_2)\right] \tag{4.14}$$

$$\ddot{\theta}_2 = \frac{1}{\Delta}\left[d_{11}(\tau_2 - h_2 - \phi_2) - d_{12}(\tau_1 - h_1 - \phi_1)\right] \tag{4.15}$$

where

$\Delta = d_{22}d_{11} - d_{12}^2$

$d_{11} = m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos\theta_2) + I_1 + I_2$

$d_{22} = m_2 l_{c2}^2 + I_2$

$d_{12} = m_2(l_{c2}^2 + l_1 l_{c2}\cos\theta_2) + I_2$

$h_1 = -m_2 l_1 l_{c2}\ \dot{\theta}_2^2 \sin\theta_2 - 2m_2 l_1 l_{c2}\ \dot{\theta}_1\dot{\theta}_2 \sin\theta_2$

$h_2 = m_2 l_1 l_{c2}\ \dot{\theta}_1^2 \sin\theta_2$

$\phi_1 = (m_1 l_{c1} + m_2 l_1)g\cos\theta_1 + m_2 l_{c2} g\cos(\theta_1 + \theta_2)$

$\phi_2 = m_2 l_{c2} g\cos(\theta_1 + \theta_2)$

At equilibrium point

$\dot{\theta}_1 = \dot{\theta}_2 = 0$

$h_1 = h_2 = 0$

This results in

$d_{22}(\tau_1 - \phi_1) - d_{12}(\tau_2 - \phi_2) = 0 \quad ;$

$d_{11}(\tau_2 - \phi_2) - d_{12}(\tau_1 - \phi_1) = 0 \quad ;$

$\frac{(\tau_1 - \phi_1)}{(\tau_2 - \phi_2)} = \frac{d_{12}}{d_{22}} \quad ;$

$\frac{(\tau_1 - \phi_1)}{(\tau_2 - \phi_2)} = \frac{d_{11}}{d_{12}} \quad ;$

Redefine the joints angles and their derivatives as follows

$\theta_1 = f(x_1, x_2, x_3, x_4, \tau_1, \tau_2)$

$x_1 = \theta_1$

$x_2 = \dot{\theta_1}$

$x_3 = \theta_2$

$x_4 = \dot{\theta_2}$

$\frac{\partial f}{\partial x_1} = \frac{1}{\Delta}\left[d_{22}(-\frac{\partial h_1}{\partial x_1} - \frac{\partial \phi_1}{\partial x_1}) - \frac{d_{12}}{\partial x_1}(\tau_2 - h_2 - \phi_2) - d_{12}(-\frac{\partial h_2}{\partial x_1} - \frac{\partial \phi_2}{\partial x_1})\right] - \frac{1}{\Delta^2}[d_{22}(\tau_1 - h_1 -$

$\phi_1) - d_{12}(\tau_2 - h_2 - \phi_2)]\frac{\partial \Delta}{\partial x_1}$

$\frac{\partial h_1}{\partial x_1} = \frac{d_{12}}{\partial x_1} = \frac{\partial h_2}{\partial x_1} = \frac{\partial \Delta}{\partial x_1} = 0$

$\frac{\partial f}{\partial x_1} = \frac{1}{\Delta}\left[-d_{22}\frac{\partial \phi_1}{\partial x_1} - d_{12}\frac{\partial \phi_2}{\partial x_1}\right]$

$\frac{\partial f}{\partial x_2} = 0$

$\frac{\partial f}{\partial x_3} = \frac{1}{\Delta}\left[d_{22}(-\frac{\partial h_1}{\partial x_3} - \frac{\partial \phi_1}{\partial x_3}) - \frac{d_{12}}{\partial x_3}(\tau_2 - h_2 - \phi_2) - d_{12}(-\frac{\partial h_2}{\partial x_3} - \frac{\partial \phi_2}{\partial x_3})\right] - \frac{1}{\Delta^2}[d_{22}(\tau_1 - h_1 -$

$\phi_1) - d_{12}(\tau_2 - h_2 - \phi_2)]\frac{\partial \Delta}{\partial x_3}$

$\frac{\partial h_1}{\partial x_1} = \frac{\partial h_2}{\partial x_1} = 0$

$\frac{\partial f}{\partial x_3} = -\frac{1}{\Delta}\left[d_{22}\frac{\partial \phi_1}{\partial x_3} + \frac{d_{12}}{\partial x_3}(\tau_2 - h_2 - \phi_2) - d_{12}\frac{\partial \phi_2}{\partial x_3}\right] - \frac{1}{\Delta^2}[d_{22}(\tau_1 - h_1 - \phi_1) - d_{12}(\tau_2 - h_2 -$

$\phi_2)\frac{\partial \Delta}{\partial x_3}$

$\frac{\partial f}{\partial x_4} = 0$

$$\theta_2 = g(x_1, x_2, x_3, x_4, \tau_1, \tau_2)$$

$$\frac{\partial g}{\partial x_1} = \frac{1}{\Delta}\left[-d_{11}\frac{\partial \phi_2}{\partial x_1} + d_{12}\frac{\partial \phi_1}{\partial x_1}\right]$$

$$\frac{\partial g}{\partial x_2} = 0$$

$$\frac{\partial g}{\partial x_3} = \frac{1}{\Delta}\left[-d_{11}\frac{\partial \phi_1}{\partial x_3} + \frac{d_{11}}{\partial x_3}(\tau_2 - h_2 - \phi_2) + d_{12}\frac{\partial \phi_1}{\partial x_3} - \frac{d_{12}}{\partial x_3}(\tau_1 - h_1 - \phi_1)\right] - \frac{1}{\Delta^2}[d_{11}(\tau_2 -$$

$$h_2 - \phi_2) - d_{12}(\tau_1 - h_1 - \phi_1)]\frac{\partial \Delta}{\partial x_3}$$

$$\frac{\partial g}{\partial x_4} = 0$$

where

$$\frac{\partial f}{\partial \tau_1} = \frac{1}{\Delta}d_{22}$$

$$\frac{\partial f}{\partial \tau_2} = -\frac{1}{\Delta}d_{12}$$

$$\frac{\partial g}{\partial \tau_1} = -\frac{1}{\Delta}d_{12}$$

$$\frac{\partial g}{\partial \tau_2} = \frac{1}{\Delta}d_{11}$$

and

$$\frac{\partial d_{11}}{\partial x_3} = -2m_2 l_1 l_{c2}\sin\theta_2$$

$$\frac{\partial d_{12}}{\partial x_3} = -m_2 l_1 l_{c2}\sin\theta_2$$

$$\frac{\partial d_{22}}{\partial x_3} = 0$$

$$\frac{\partial h_1}{\partial x_3} = -m_2 l_1 l_{c2}\,\dot{\theta}_2^{\,2}\cos\theta_2 - 2m_2 l_1 l_{c2}\,\dot{\theta}_1\dot{\theta}_2\cos\theta_2 = 0$$

$$\frac{\partial h_2}{\partial x_3} = m_2 l_1 l_{c2}\,\dot{\theta}_1^{\,2}\cos\theta_2 = 0$$

$$\frac{\partial \phi_1}{\partial x_3} = -m_2 l_{c2} g\sin(\theta_1 + \theta_2)$$

$$\frac{\partial \phi_2}{\partial x_3} = -m_2 l_{c2} g\sin(\theta_1 + \theta_2)$$

$$\frac{\partial \Delta}{\partial x_3} = d_{22}\frac{\partial d_{11}}{\partial x_3} - 2d_{12}\frac{\partial d_{12}}{\partial x_3}$$

$$\frac{\partial d_{11}}{\partial x_1} = \frac{\partial d_{12}}{\partial x_1} = \frac{\partial d_{22}}{\partial x_1} = 0$$

$\frac{\partial h_1}{\partial x_1} = \frac{\partial h_2}{\partial x_1} = 0$

$\frac{\partial \phi_1}{\partial x_1} = (-m_1 l_{c1} + m_2 l_1) g \sin(\theta_1) - m_2 l_{c2} g \sin(\theta_1 + \theta_2)$

$\frac{\partial \phi_2}{\partial x_1} = -m_2 l_{c2} g \sin(\theta_1 + \theta_2)$

At equilibrium points

$\phi_1 = \tau_{10}$

$\phi_2 = \tau_{20}$

$\phi_1 = (m_1 l_{c1} + m_2 l_1) g \cos(\theta_1) + m_2 l_{c2} g \cos(\theta_1 + \theta_2) = \tau_{10}$

$\phi_2 = m_2 l_{c2} g \cos(\theta_1 + \theta_2) = \tau_{20}$

Thus

$\cos(\theta_1 + \theta_2) = \frac{\tau_{20}}{m_2 l_{c2} g}$

$\theta_{20} = \pm \arccos(\frac{\tau_{20}}{m_2 l_{c2} g}) - \theta_{10}$

$(m_1 l_{c1} + m_2 l_1) g \cos(\theta_1) + \tau_{20} = \tau_{10}$

$\cos(\theta_1) = \frac{\tau_{10} - \tau_{20}}{(m_1 l_{c1} + m_2 l_1) g}$

$\theta_{10} = \pm \arccos(\frac{\tau_{10} - \tau_{20}}{(m_1 l_{c1} + m_2 l_1) g})$

We have the following system specifications:

$m_1 = 1.9008$ kg;

$m_2 = 0.7175$ kg;

$l_1 = l_2 = 0.2$ m;

$l_{c1} = 0.18522$ m;

$l_{c2} = 0.062052$ m;

$I_1 = 0.0043399$ kgm^2;

$I_2 = 0.0052285$ kgm^2.

$x_{10} = pi/2; \quad x_{20} = 0.0; \quad x_{30} = pi; \quad x_{40} = 0;$

We chose, for each link, nine equilibrium points as follows

for $\quad i = 1, 2, ..., 9;$

$\theta_1(i) = 10 * i;$

$\theta_2(i) = -90 + 20 * i;$

Then we used LQR to find the set of K's at each equilibrium point, then we tabulated the results to get an input-output representation, which will be used later in ANFIS to generate the fuzzy inference system. The above information has been arranged in the proper matrix format to be used in LQR :

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \frac{\partial f}{\partial x_3} & \frac{\partial f}{\partial x_4} \\ 0 & 0 & 0 & 1 \\ \frac{\partial g}{\partial x_1} & \frac{\partial g}{\partial x_2} & \frac{\partial g}{\partial x_3} & \frac{\partial g}{\partial x_4} \end{bmatrix}; \quad B = \begin{bmatrix} 0 & 0 \\ T * d_{22} & -T * d_{12} \\ 0 & 0 \\ -T * d_{12} & T * d_{11} \end{bmatrix};$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}; \quad D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

where $T = \frac{1}{((d_{22}d_{11} - d_{12}^2) + \varepsilon)}$. The result of running LQR for each equilibrium point is shown in Appendix A. The Riccati equation was solved at 81 operating points. In our simulation we used an output equation given by:

$$y = Cq + D\tau$$

where $Q = 100C^TC$ and $R = \begin{bmatrix} \frac{1}{100} & 0 \\ 0 & \frac{1}{25} \end{bmatrix}$. With these values for Q and R, the discrete gains at each operating point were obtained. The model for each gain is

identified by using the joints angles as input data and the gain k as the output data. The model building process is performed by using subtractive clustering in both input and output spaces. Each cluster represent certain region of the system state space. Then, the clusters are projected into each dimension in the input space, where each projection forms an antecedent of a rule. Thus, the premise parameters of the model are identified. The model are completed by LSE method which identifies the optimal consequent parameters in Sugeno system. The identified first order Sugeno fuzzy model is as follows:

R_1 :IF q_1 is A_1^1 AND q_2 is A_2^1 THEN $y_1 = a_{10} + a_{11}q_1 + a_{12}q_2$

$\vdots$

R_k :IF q_1 is A_1^k AND q_2 is A_2^k THEN $y_k = a_{k0} + a_{k1}q_1 + a_{k2}q_2$

where the two input variables q_1 and q_2 are the joints angles while the output variable y_i for $i = 1, 2, ..., K$ is the gain for rule k, and a_{i0}, a_{i1}, a_{i2} (for $i = 1, 2, ..., k$) are regression parameters which are identified using the LSE algorithm. The least squares estimation ensures the global optimization of the regression parameters for the given set of clusters. ANFIS can modify the parameters of the membership functions of the fuzzy control rules. ANFIS contains a hybrid learning rule which combines gradient descent and the LSE for fast identification of parameters. In the forward pass of the hybrid learning algorthim, the consequent parameters are identified by LSE. In the backward pass, the error signals propagate backward and the premise parameters are updated by gradient descent. By updating the membership

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	39.6, 10	0.285, 1.016	-0.003164	6.055	2.944
R_2	39.6, 10	0.3536, 0.3554	0.0114	9.345	0.5328
R_3	39.6, 70	0.3843, 1.403	-0.009241	0.9642	8.872
R_4	39.6, -50	0.3685, 1.401	0.006041	2.615	6.228
R_5	39.6, 90	0.3789, 0.5397	-0.0006359	7.364	1.521
R_6	39.6, -70	0.36, 0.7102	0.006813	6.68	0.8995

Table 4.1: 6 rules for k_{11} fuzzy subcontroller

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	39.6, 10	0.3893, 1.029	-0.001588	0.113	1.427
R_2	39.6, 30	0.3761, 0.3825	-0.00109	0.8861	0.6491
R_3	39.6, -50	0.4012, 0.525	0.0003032	0.9738	0.3897
R_4	39.6, 90	0.3535, 1.223	-0.003694	0.1764	1.492
R_5	39.6, -70	0.3746, 1.383	0.002209	0.5145	0.8219

Table 4.2: 5 rules for k_{12} fuzzy subcontroller

functions, we are actually tuning the fuzzy controller for better performance. The rule bases of the models built by using subtractive clustering and ANFIS for all eight gains are shown in Tables 4.1-4.8. Each row in the tables represents a rule. For example, there are 5 rules identified in k_{12} model. The two antecedents of a rule i are identified by cluster centers c_1^i and c_2^i in the table. Given the cluster centers one can easily identify the corresponding exponential membership function from Figures 4.3-4.6. The fuzzy model of the gain as a function of the system parameters were considered continuous gain and were used in the design of the fuzzy gain schedule.

Figure 4.7 shows the *simulink* diagram generated using Matlab. A detailed diagram of the fuzzy gain schedule controller is given in Figure 4.8. Figure 4.9 shows the fuzzy subcontrollers for all gains. The simulation results of the gains show smooth control surfaces, which is due to the combined effects of fuzzy controller and

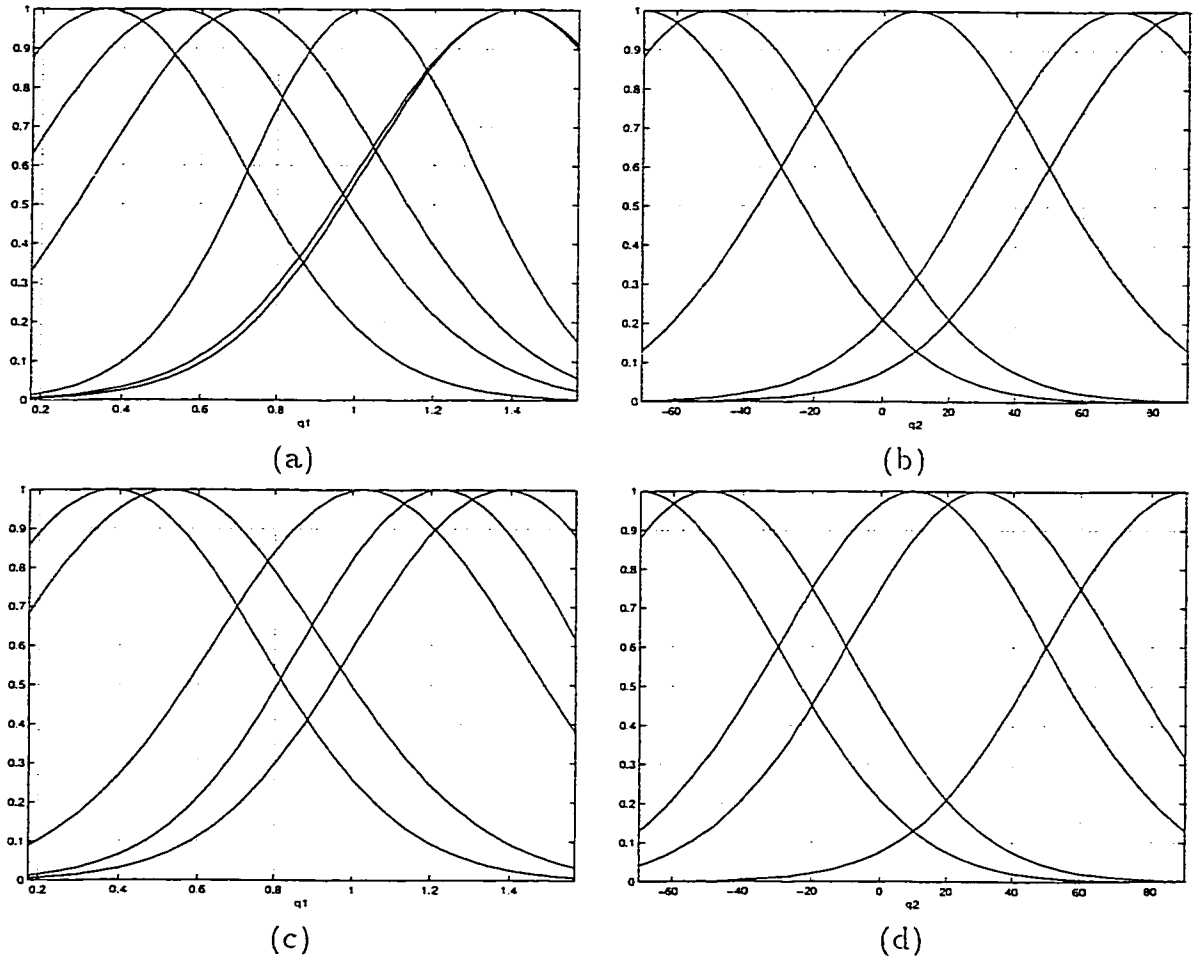

Figure 4.3: (a) and (b) Membership functions of joint angles in k_{11} fuzzy subcontroller. (c) and (d) Membership functions of joint angles in k_{12} fuzzy subcontroller.

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	45.25, 10	0.4209, 0.9098	0.003053	-0.3982	1.588
R_2	45.26, -30	0.4347, 0.339	-0.0008713	-0.03413	-0.2402
R_3	45.26, 90	0.4009, 0.5124	0.001816	0.1438	0.1351
R_4	45.26, 50	0.4341, 1.526	-0.01159	0.02804	1.208
R_5	45.26, -70	0.4242, 1.397	0.007449	0.6488	0.01376

Table 4.3: 5 rules for k_{13} fuzzy subcontroller

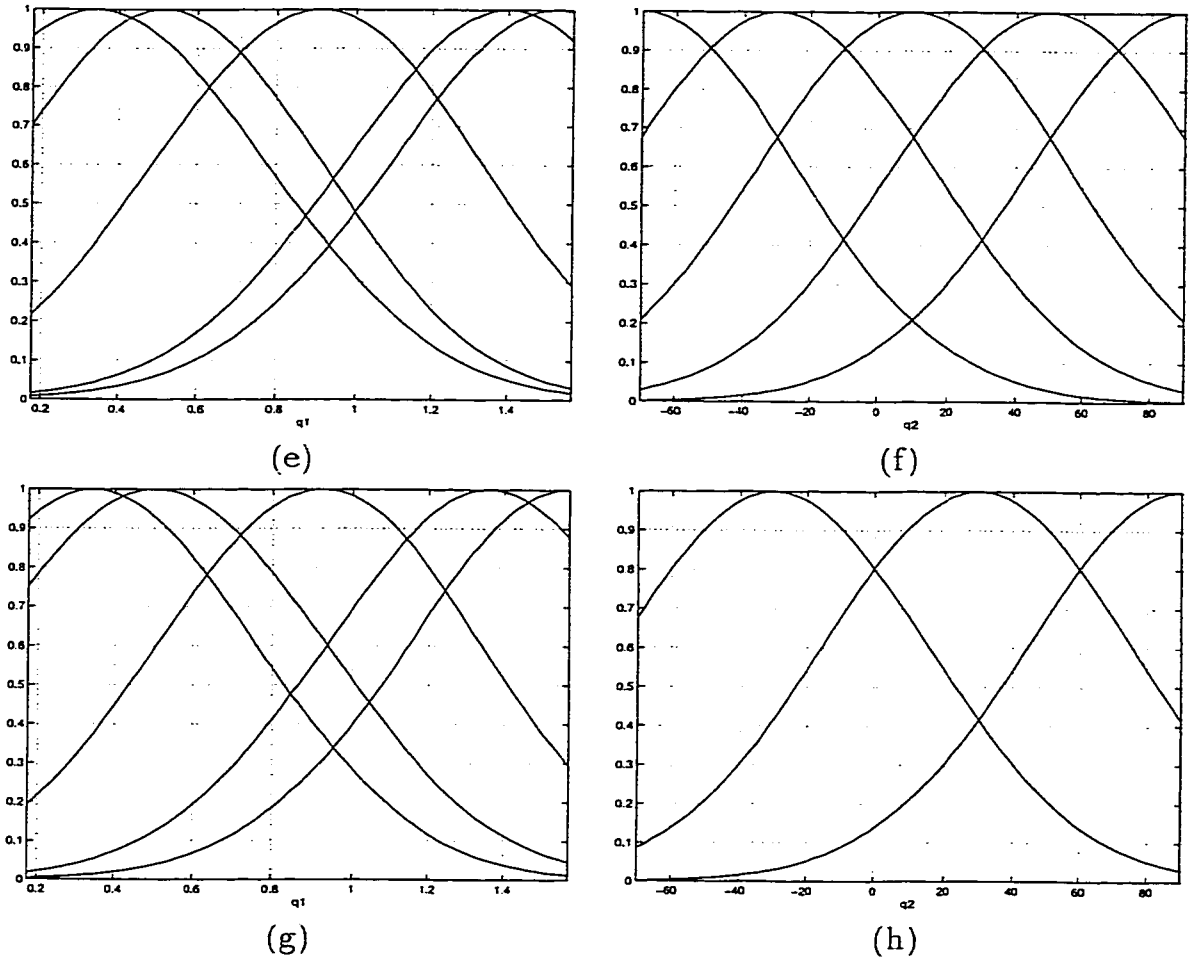

Figure 4.4: (e) and (f) Membership functions of joint angles in k_{13} fuzzy subcontroller. (g) and (h) Membership functions of joint angles in k_{14} fuzzy subcontroller.

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	45.25, 30	0.4138, 0.9242	-0.0003676	-0.06789	0.2907
R_2	45.26, -30	0.4151, 0.3418	0.0004668	0.06597	0.05353
R_3	45.26, -30	0.4181, 1.357	0.001467	0.06665	0.124
R_4	45.26, 90	0.4316, 0.5013	0.0002443	0.01776	0.01751
R_5	45.26, 90	0.4168, 1.566	-0.0012	0.03793	0.1244

Table 4.4: 5 rules for k_{14} fuzzy subcontroller

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	45.26, 10	0.4286, 1.038	-0.01598	-3.16	4.261
R_2	45.25, 50	0.3943, 0.3564	0.006917	0.6988	0.3813
R_3	45.25, -50	0.3759, 0.6896	-0.01884	-2.396	-2.076
R_4	45.26, 7	0.4553, 1.552	-0.02313	-2.024	3.905

Table 4.5: 4 rules for k_{21} fuzzy subcontroller

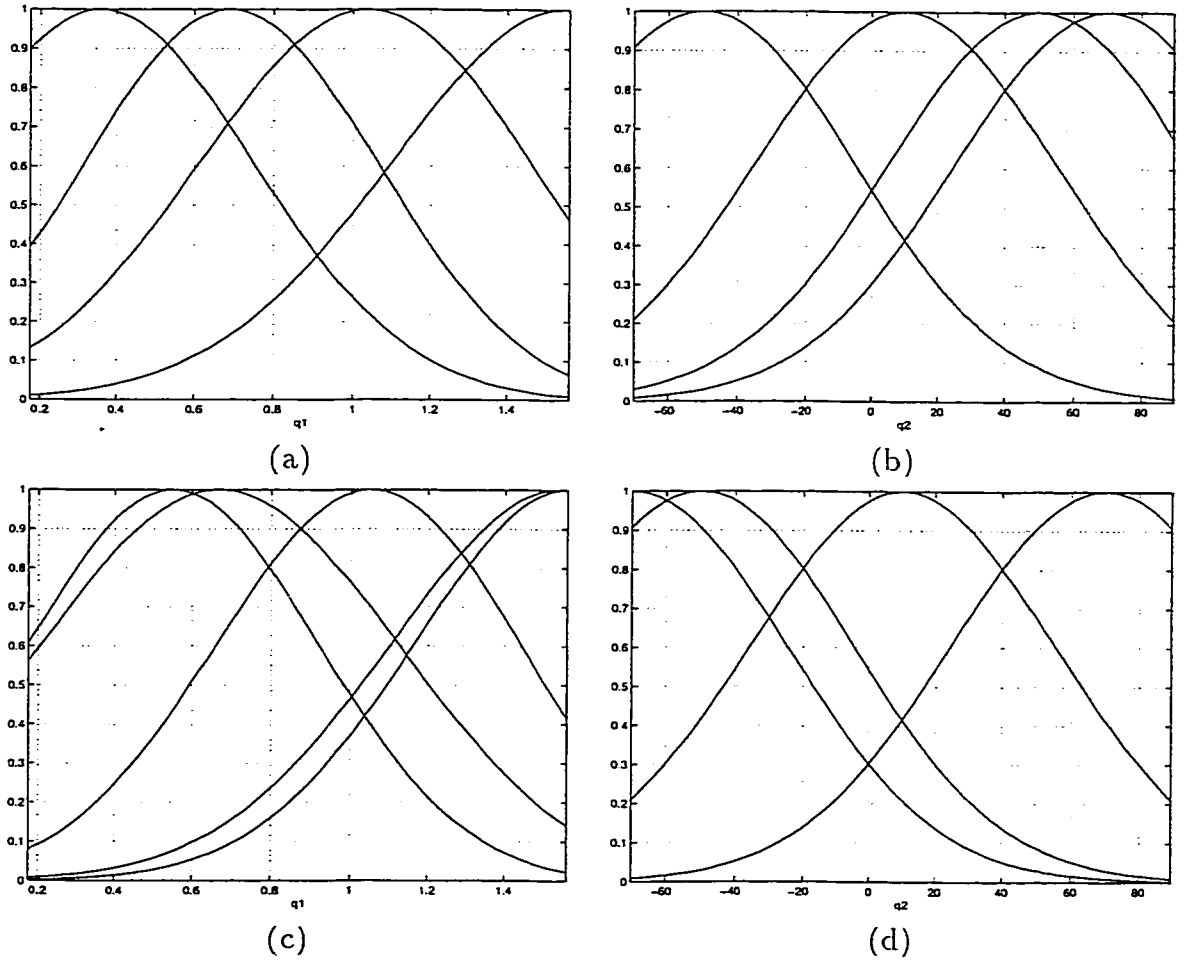

Figure 4.5: (a) and (b) Membership functions of joint angles in k_{21} fuzzy subcontroller. (c) and (d) Membership functions of joint angles in k_{22} fuzzy subcontroller.

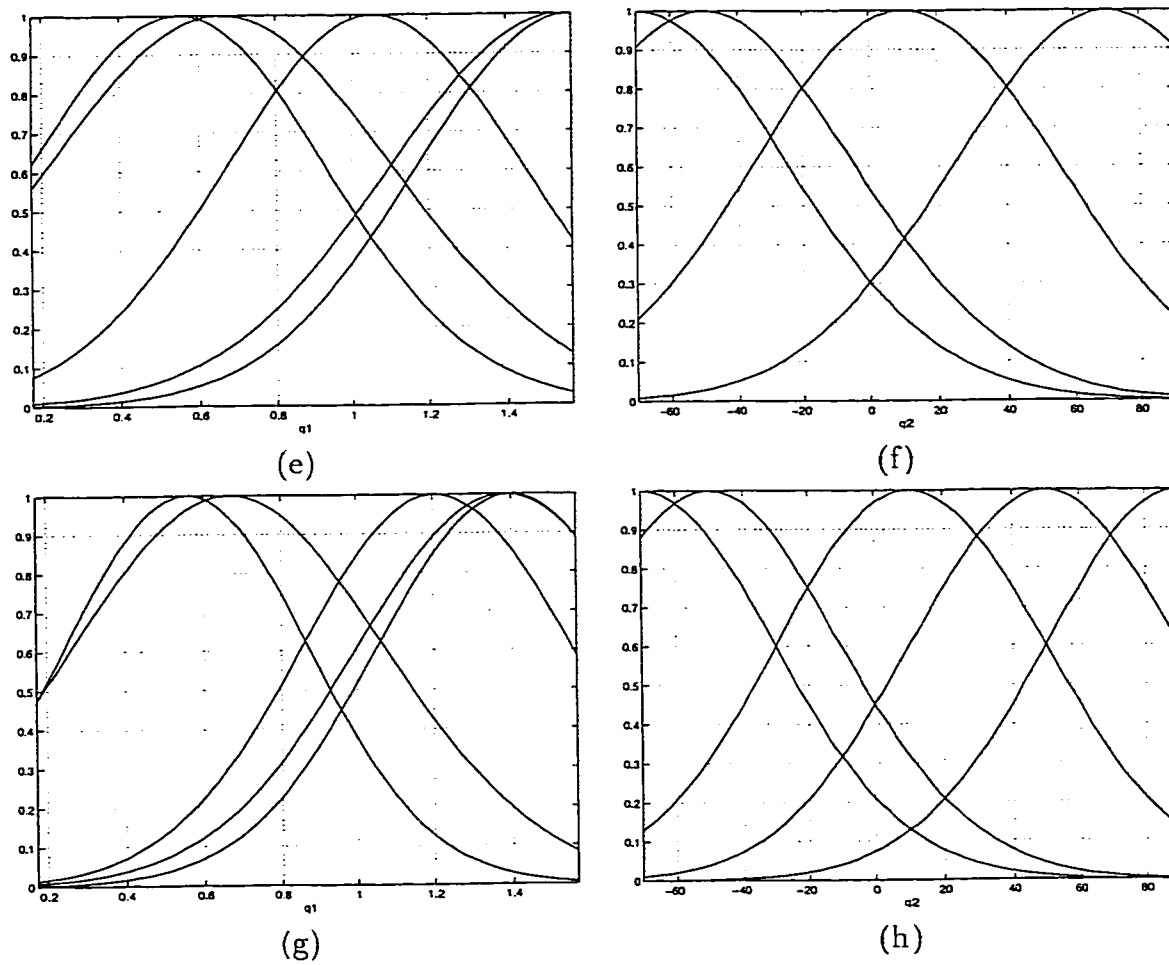

Figure 4.6: (e) and (f) Membership functions of joint angles in k_{23} fuzzy subcontroller. (g) and (h) Membership functions of joint angles in k_{24} fuzzy subcontroller.

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	45.25, 10	0.3905, 1.052	0.0003359	-0.2745	0.4384
R_2	45.25, 70	0.372, 0.5476	-0.0005072	-0.11	0.2956
R_3	45.26, -50	0.4564, 0.666	0.0003307	-0.3071	-0.0795
R_4	45.25, 70	0.4386, 1.544	-0.002087	-0.1154	0.1892
R_5	45.25, -70	0.4007, 1.566	0.003254	0.09419	0.01254

Table 4.6: 5 rules for k_{22} fuzzy subcontroller

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	45.25, 10	0.3894, 1.058	0.008044	-0.3255	2.065
R_2	45.26, -50	0.4503, 0.6596	-0.0006734	0.01165	0.654
R_3	45.25, 70	0.3841, 0.55	0.00145	-0.2393	1.513
R_4	45.26, 70	0.4438, 1.542	-0.003867	0.3952	0.5647
R_5	45.25, -70	0.398, 1.566	0.009414	0.3324	1.272

Table 4.7: 5 rules for k_{23} fuzzy subcontroller

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	39.6, 10	0.3512, 1.206	0.0002743	-0.03133	0.1747
R_2	39.6, 50	0.315, 0.5599	0.000126	-0.006454	0.1574
R_3	39.6, -50	0.4053, 0.6688	-7.987e-05	-0.002551	0.08297
R_4	39.6, 90	0.3849, 1.384	-5.107e-05	-0.0259	0.1546
R_5	39.6, -70	0.3483, 1.4	0.0004706	0.04371	0.08376

Table 4.8: 5 rules for k_{24} fuzzy subcontroller

the neural networks in the ANFIS architecture. In the simulation example eight gains were used assuming full state feedback. These gains are associated with two control torques at the joints and the state of the system represented by the angle of each joint and its corresponding angular velocity. These gains are shown in Figure 4.10 and 4.11. The variation of the gain surface is very smooth. These gains are functions of the state of the system. In this example, the gains are functions of the joint angles of the robot arm. The gain surfaces are approximate solution of the algebraic Riccati equation over the state space.

First the joint angles are required to track the following reference trajectories:

$$\theta_{r1} = \frac{\pi}{4}\left[2 + 6\exp(-\frac{t}{0.3}) - 7\exp(-\frac{t}{0.4})\right]$$

$$\theta_{r2} = \frac{\pi}{4}\left[1 + 6\exp(-\frac{t}{0.3}) - 7\exp(-\frac{t}{0.4})\right]$$

where θ_{r1} and θ_{r2} are the reference trajectories. The robot configuration changes

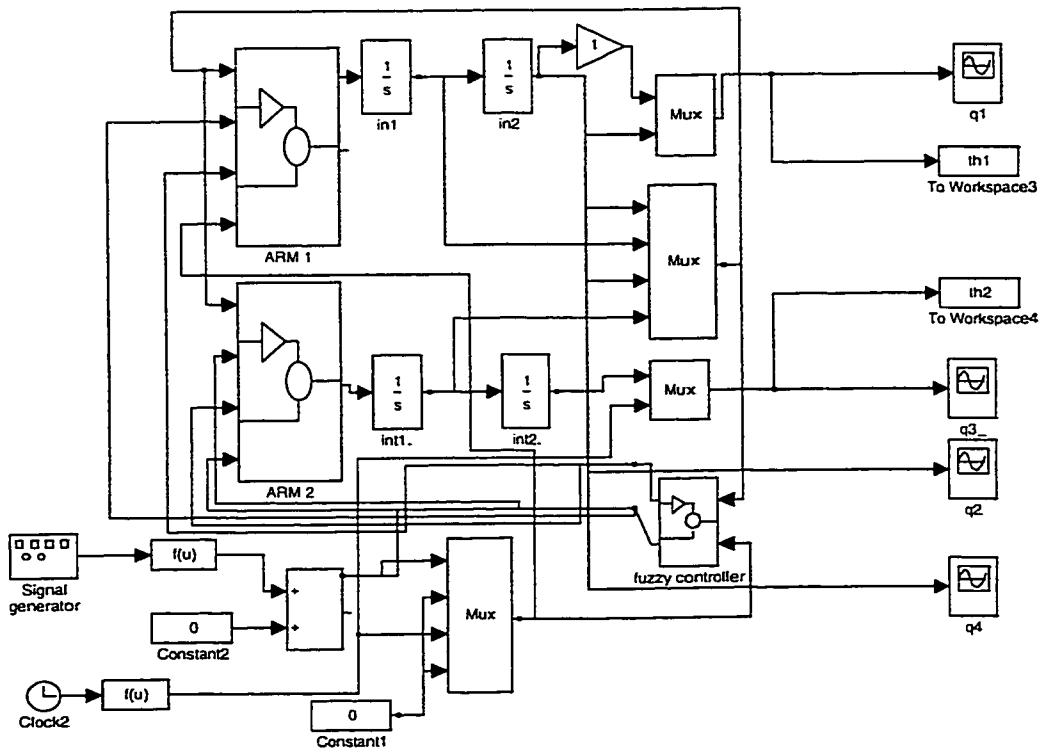

Figure 4.7: *Simulink* diagram for the robot manipulator fuzzy gain scheduling control

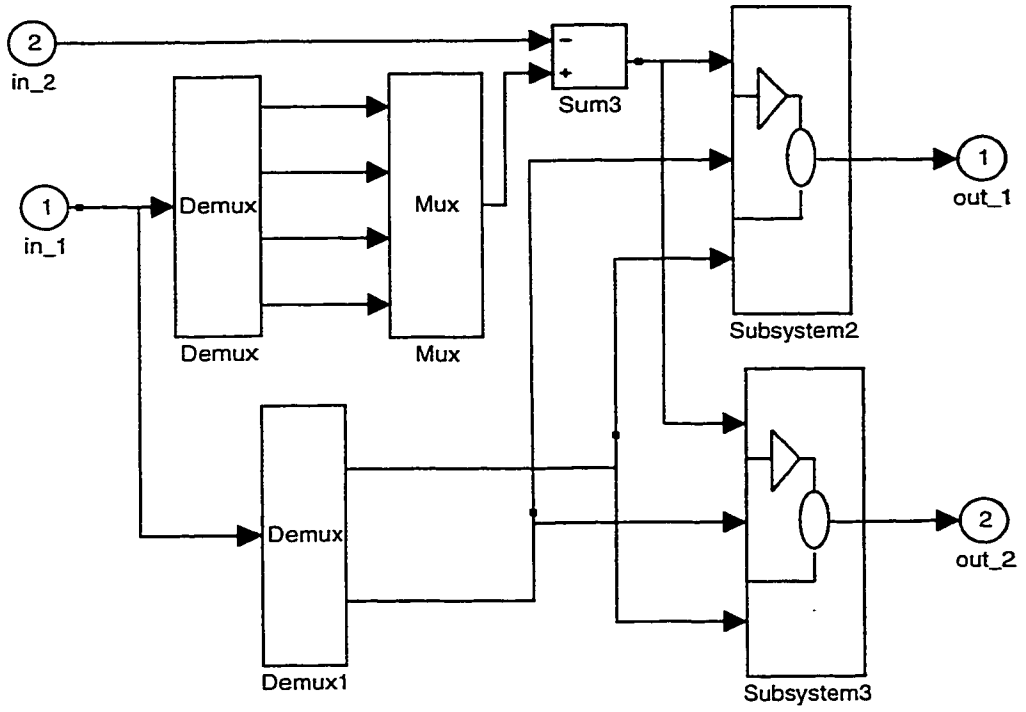

Figure 4.8: Fuzzy gain schedule block diagram

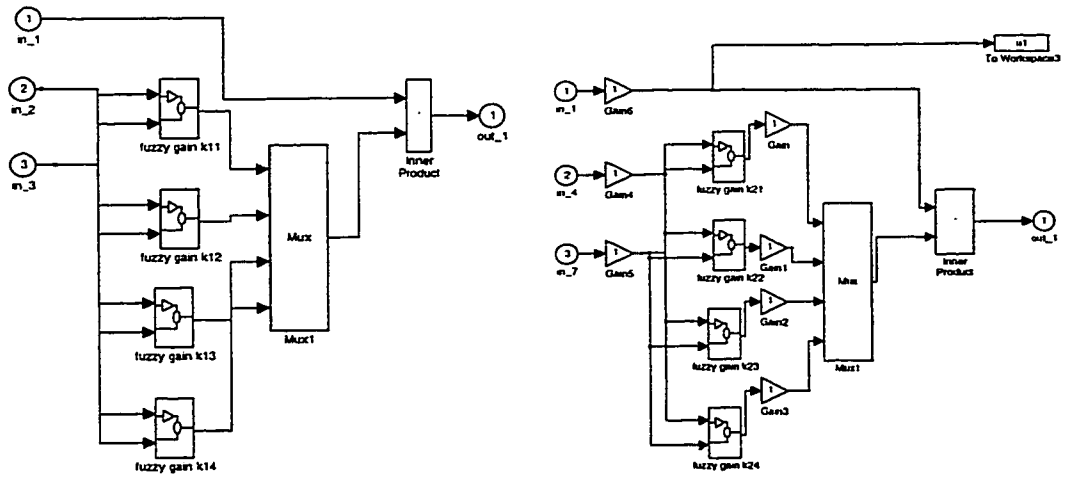

Figure 4.9: Fuzzy subcontrollers of all gains

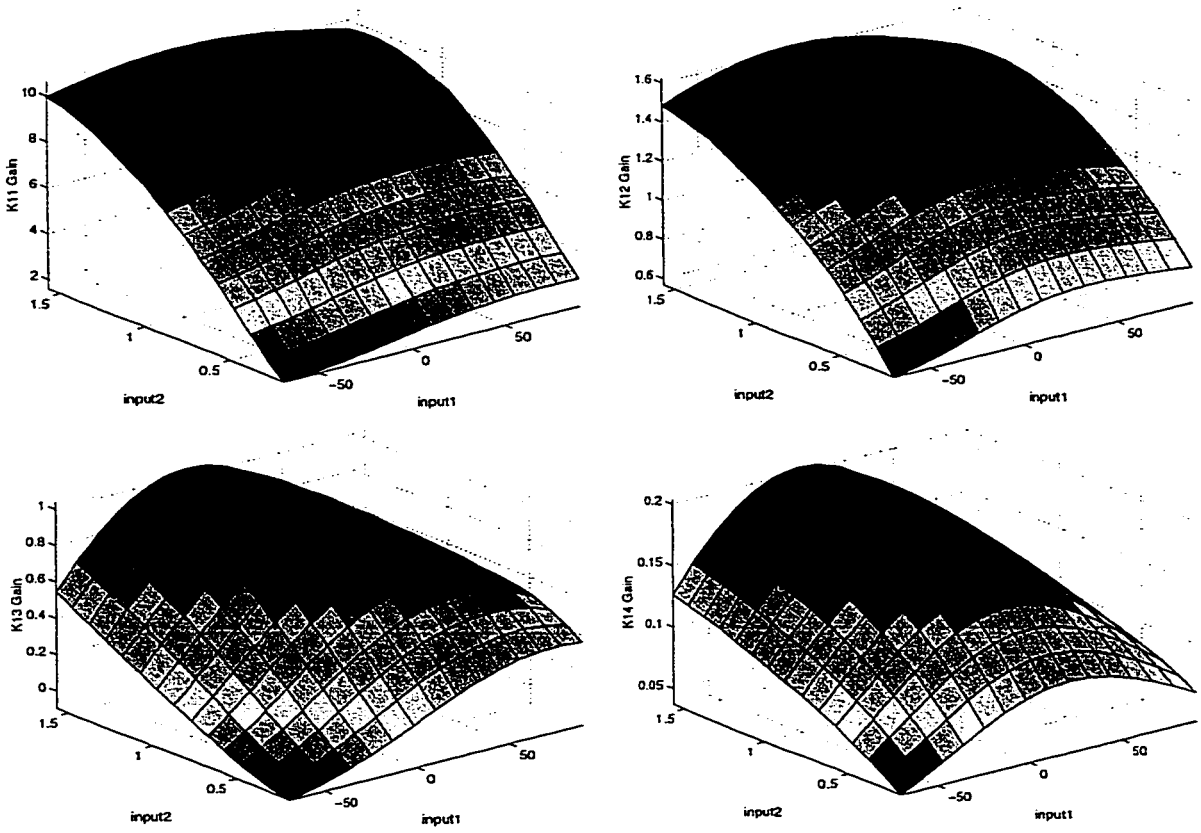

Figure 4.10: The first row of gain matrix as a function of the joint angles

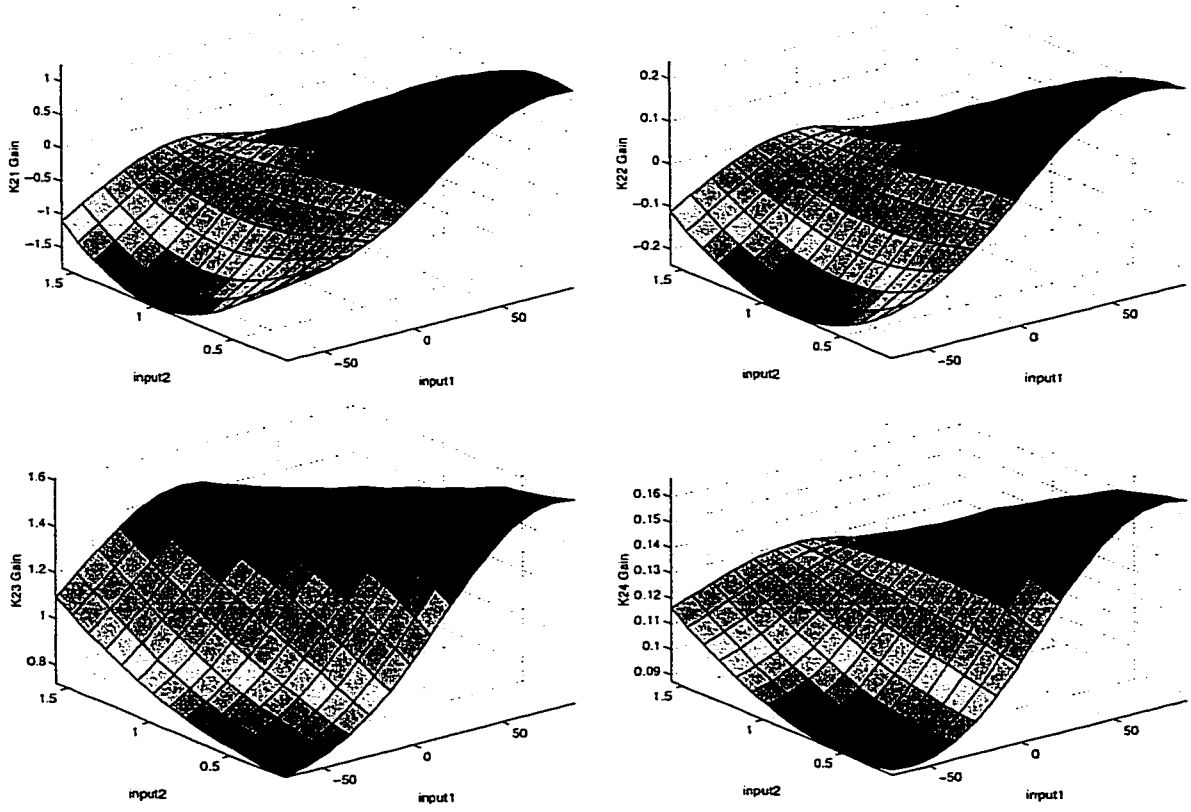

Figure 4.11: The second row of gain matrix as a function of the joint angles

from the initial conditions ($\theta_{10} = \frac{\pi}{4}, \theta_{20} = \dot{\theta_{10}} = \dot{\theta_{20}} = 0$) to the final condition ($\theta_{10} = \theta_{20} = \frac{\pi}{2}, \dot{\theta_{10}} = \dot{\theta_{20}} = 0$). The results of the simulation are shown in Figure 4.12 and indicate that the joint angles track their corresponding reference trajectories very closely. To test the performance of the controller with a different trajectories, let us

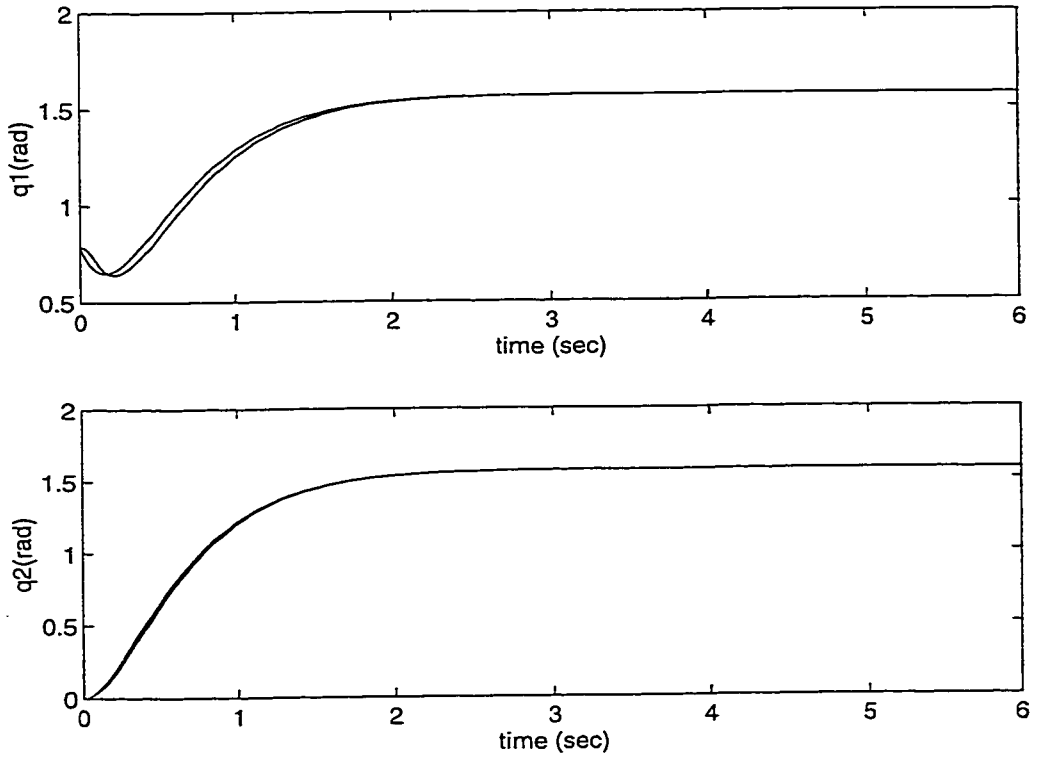

Figure 4.12: Response of the joint angles to a reference trajectory

double the time constants of the reference trajectories:

$$\theta_{r1} = 45 \times \left[2 + 6\exp(-\tfrac{t}{0.6}) - 7\exp(-\tfrac{t}{0.8})\right]$$

$$\theta_{r2} = 45 \times \left[1 + 6\exp(-\tfrac{t}{0.6}) - 7\exp(-\tfrac{t}{0.8})\right]$$

It is seen in Figure 4.13 that the new reference trajectories are tracked very closely. Now let us require that θ_1 follows a sinusoidal input and θ_2 follows the following

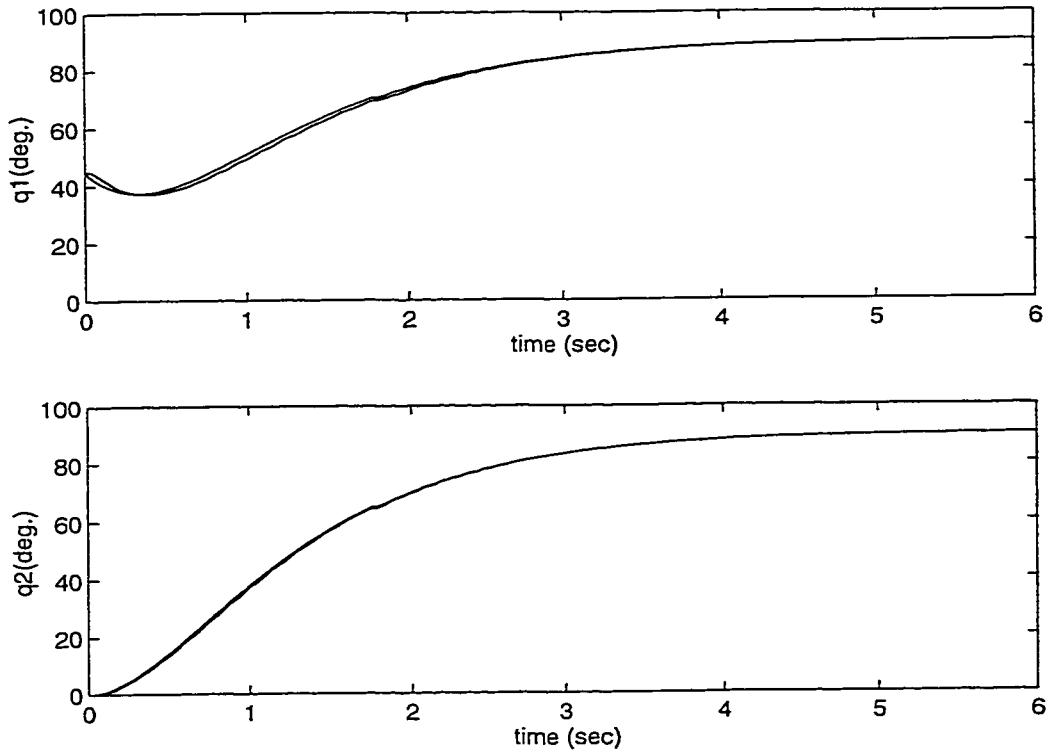

Figure 4.13: Response of the joint angles to a reference trajectory

trajectory:

$$\theta_{r2} = 45 \times \left[1 + 6\exp(-\tfrac{t}{0.6}) - 7\exp(-\tfrac{t}{0.8})\right]$$

Figure 4.14 shows that the new reference trajectories are tracked very closely. Figure 4.15 shows a satisfactory tracking performance of the joints angles to step inputs. We conclude that this controller is robust against reference trajectory changes.

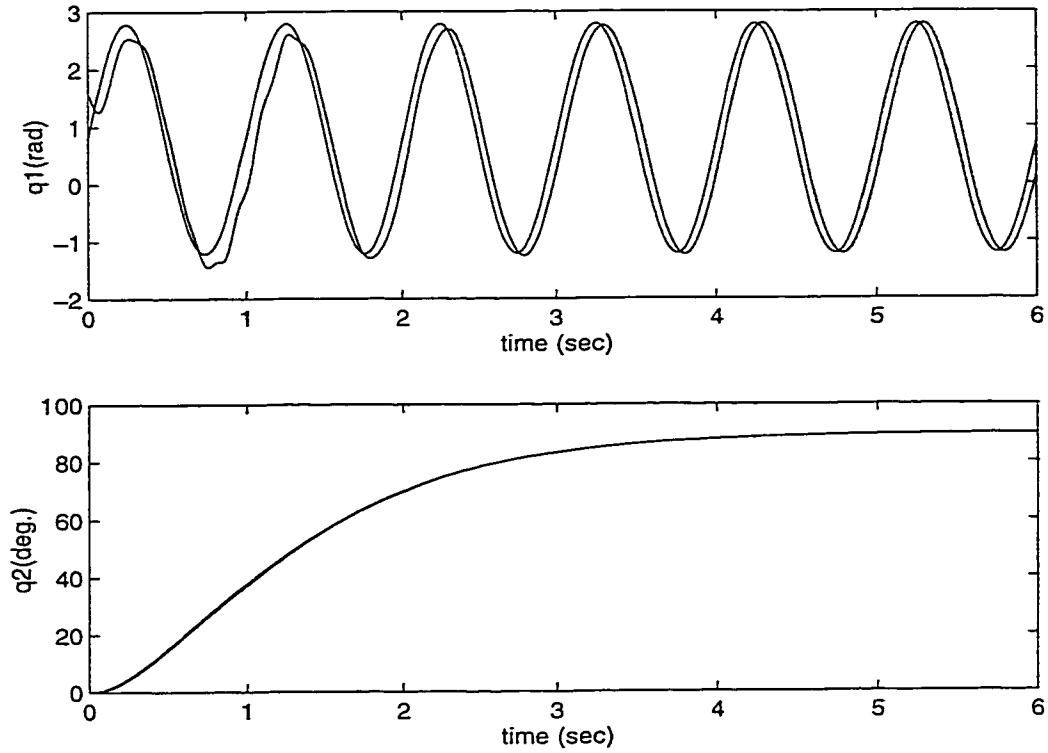

Figure 4.14: Response of the joint angles to a reference trajectory

Figures 4.16 and 4.17 show the simulation results using two different inputs. These inputs are sinusoidal inputs and square wave inputs. To demonstrate the stability of the system, the frequency of the inputs was varied. In Figure 4.16(a), and Figure 4.16(b) we can see that the tracking performance is good for the two joints. The

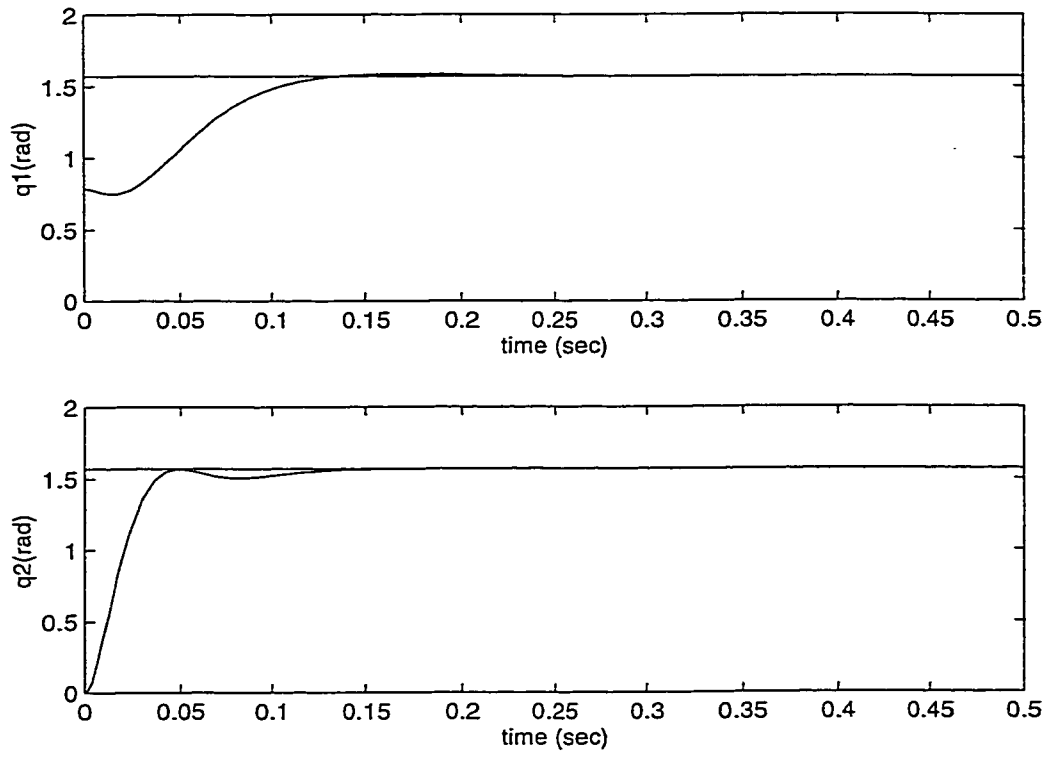

Figure 4.15: Response of the joint angles to a reference trajectory

kinks in Figure 4.16(a) is due to the rapid switch in position of the second input. At frequency of 1 Hz, Figures 4.16(c) and 4.16(d), the tracking is good. In Figure 4.17(a) and Figure 4.17(b) where frequency of the inputs is 2 Hz, we observe a phase lag in addition to very small amplification of the output angles of the first and the second arm. Another observation is a slight overshoot of the output of the first arm, but the motion is very well damped. Figure 4.17(c) and Figure 4.17(d) show the results of the simulation with 4 Hz inputs frequency. The results of the simulation shows a satisfactory tracking performance with different driving frequencies, noting that the natural frequency of the system is close to 1 Hz.

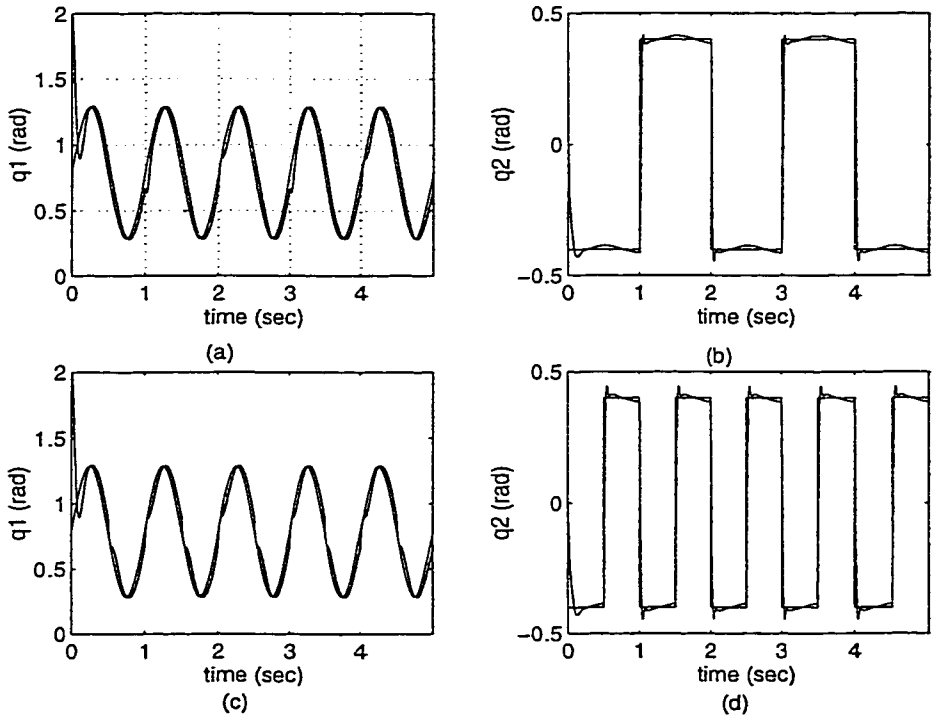

Figure 4.16: Joint angles (a) first arm with 0.5 Hz (b) second arm with 0.5 Hz (c) first arm with 1 Hz (d) second arm with 1 Hz.

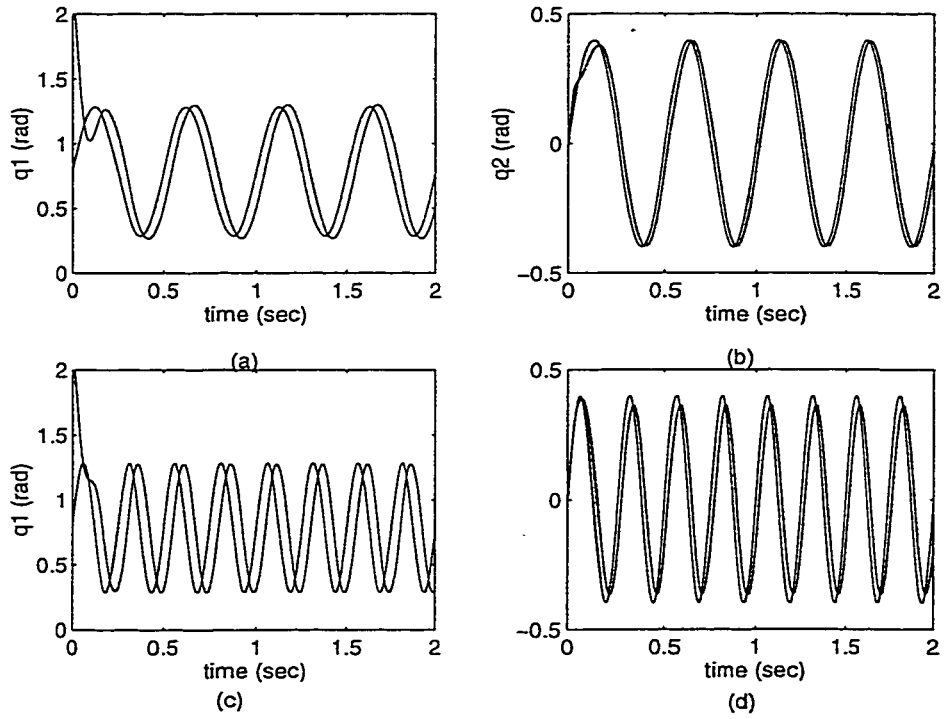

Figure 4.17: Joint angles (a) first arm with 2 Hz (b) second arm with 2 Hz (c) first arm with 4 Hz (d) second arm with 4 Hz.

4.7 Fuzzy modelling of hard nonlinearities in physical systems

Linearization is commonly used to analyze a nonlinear system for small departures from an operating point. When the perturbations are small enough, their products and powers are negligible, so discarding them reduces the system dynamics to linear equations in the perturbed variables. The nature of nonlinearity plays an important role in determining if there is a neighbourhood of validity. Continuous single valued functions are good applications of small perturbation assumption. Discontinuous functions which are not single valued are never the objects of small perturbation analysis. In general, a function expandable in a Taylor series is a candidate for small perturbation analysis, although success requires the truncation error after two terms to be acceptably small. As long as the perturbation is small, we can argue that higher powers in Taylor series are infinitesimal and it can be truncated. Linearization works well for continuous single valued functions. If the function is multi-valued or contains a discontinuity in the neighbourhood of the operating point, no suitable linearization is possible. In the following section we will investigate some types of nonlinearities, and their modelling using a neuro-fuzzy approach.

System nonlinearities are often classified as inherent or intentional nonlinearities. Inherent nonlinearities are those which exist in the components selected to perform a function other than a nonlinear one within the system, often the designer

would be happier if they did not exist. Typically such characteristics are saturation in amplifiers and motors, dead zone in valves, nonlinear friction, and backlash in gears. On the other hand, intentional nonlinearities are those which are deliberately introduced into the system in order to compensate for the effects of other undesirable nonlinearities, or to obtain a better performance. A simple example of an intentional nonlinearity is the use of nonlinear damping to optimize response as a function of the error. Limiters to restrict the rang of a variable is another case of an intentionally introduced nonlinearities.

4.7.1 Simple nonlinearities

Saturation is probably the most common of all nonlinear phenomena. It is a continuous single valued function (simple nonlinearity), as illustrated in Figure 4.18. The output is not proportional to the input, and for sufficiently large input, the output remains constant and is independent of input. Over a limited range, the output is proportional to the input and the device is linear. The shape of the saturation curve is common to many physical devices, though the input and output quantities may be quite different. All amplifiers have a saturation limit, a two-phase motor has a maximum shaft speed and a hydraulic transmission used as a motor device has a maximum output speed. Saturation type nonlinearity can be modelled easily using the following set of fuzzy rules:

R_1: IF x is small THEN y is small

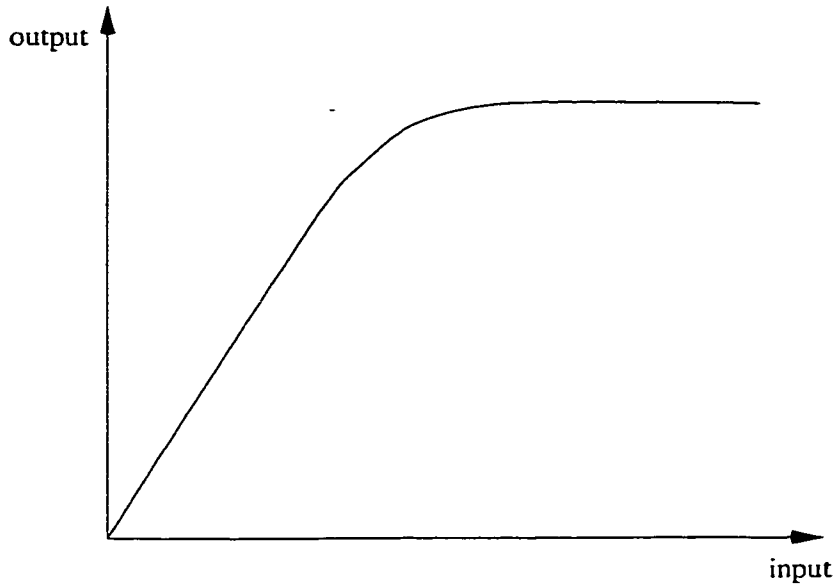

Figure 4.18: Input-output characteristic of saturation nonlinearity

R_2: IF x is medium THEN y is medium

R_3: IF x is high THEN y is medium

where universes of the input and output are partitioned by using the linguistic terms *small*, *medium* and *high*. The first two rules represent the monotonic increasing output for small and medium values of the input variable; the third rule is related to the saturation of the system output for high values of the input as shown in Figure 4.18.

The structure of Sugeno fuzzy logic models with the learning aspects from neural networks combined in ANFIS will be considered to model selected types of nonlinearities. ANFIS uses a highly efficient training method that combines gradient descent and least-squares optimization to improve training speed compared to standard backpropagation.

4.7.2 Hard nonlinearities

Discontinuous multi-valued functions are considered hard nonlinearities, such as hysteresis, gear train backlash and relays. They are complex to model and considered failures to linear analysis. Hysteresis is a nonlinear phenomenon that is most usually associated with magnetization curves or backlash of gear trains. A conventional magnetization curve is the one whose path depends on whether the magnetizing force is increasing or decreasing. An extremely complex form of nonlinear behaviour to model is the backlash which exists in gears. Simply, backlash in gears is the free space between adjacent teeth, Figure 4.19. It permits one gear to move through a finite angle with respect to another gear, without being in contact with the second gear. The nonlinear effects of backlash are not easy to describe mathematically, and the analysis of system performance when backlash is present may be quite difficult. These types of nonlinearities are not easy to be modelled using clustering and ANFIS approach. We present two examples on hard nonlinearities in this section and show how this approach fails to model these nonlinearities. Also, modification to this approach have been suggested to be able to model such types of nonlinearities.

Consider a system of one input (angle of attack (α) and one output (normal force coefficient (C_n)). The input-output experimental data pairs are given in Appendix A and shown in Figure 4.20. Figure 4.20 shows a hysteresis nonlinearity which means that the path of the normal force coefficient depends on whether the

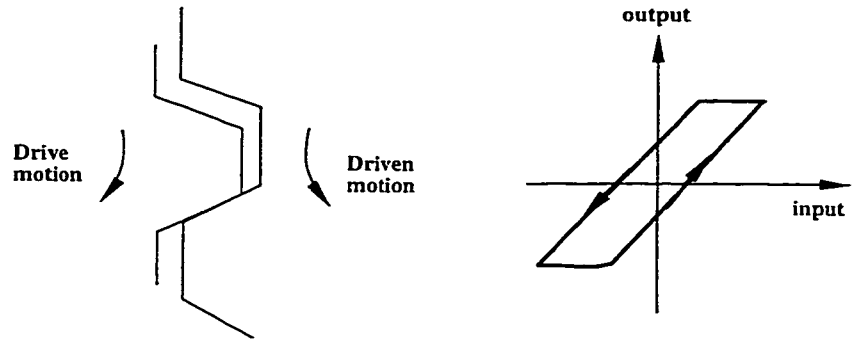

Figure 4.19: Illustration of backlash

angle of attack is increasing or decreasing. Grid partition and ANFIS algorithms of

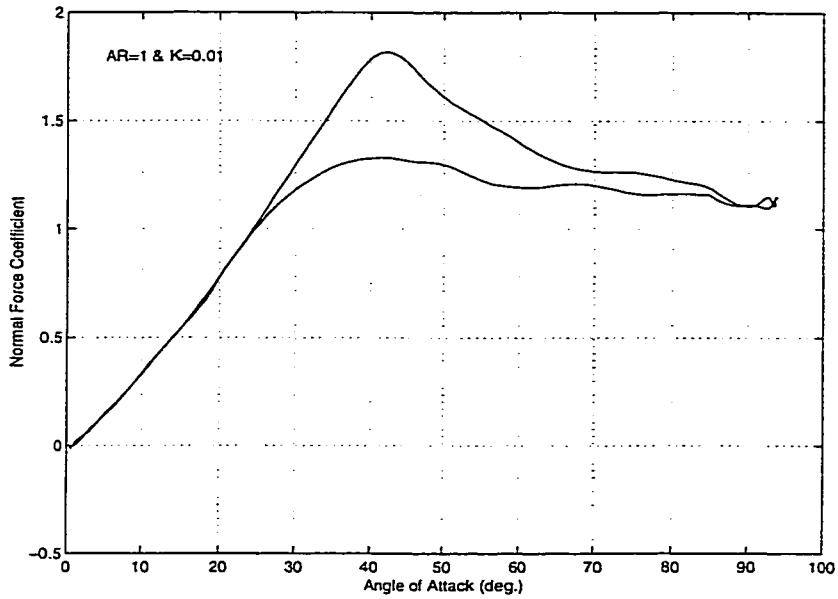

Figure 4.20: Input-output experimental data plot

MATLAB are used to construct a fuzzy model of this system. All data in Appendix A is organized and partitioned into 80% training and 20% checking data as shown in Figure 4.21. Figure 4.22 shows a comparison of experimental to ANFIS output. Grid partition and ANFIS could not model the hysteresis in the system, instead it interpolated between the two output values in the increasing and the decreasing

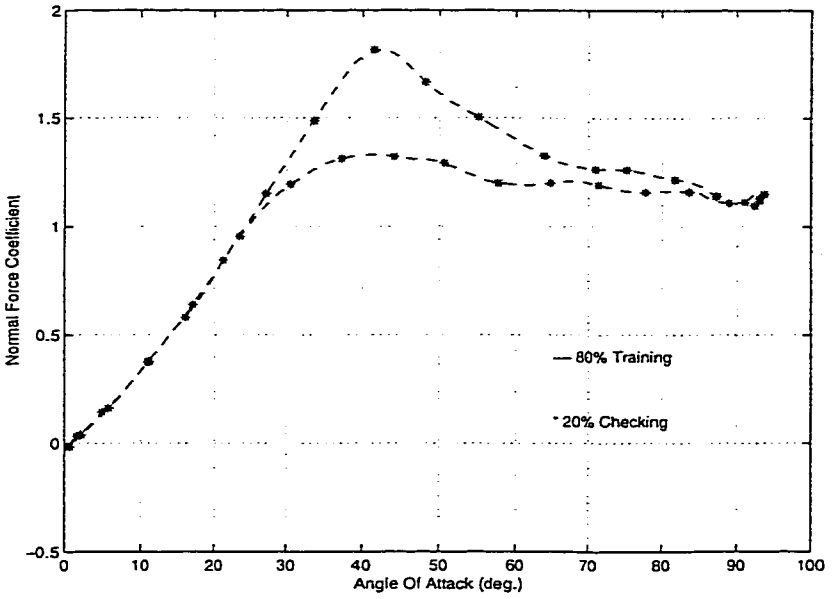

Figure 4.21: Input-output experimental data partitioned into 80% training and 20% checking data

path. One way to solve this problem is to divide the system into two submodels one for the increasing α and the other for the decreasing α. Another way to deal with it is by introducing another input to the fuzzy system, which is the time derivative of α. A plot of the time derivative of α and the normal force coefficient is shown in Figure 4.23. ANFIS is applied on a system of two inputs (α and its derivative ($\dot{\alpha}$)) and one output (C_n). A comparison of experimental to ANFIS output is given in Figure 4.24. The fuzzy model of C_n is given in Appendix A.

Another common nonlinearity is dead zone, as shown in Figure 4.25. It arises in mechanisms which are spring loaded to minimize backlash and in many other devices which are insensitive to small signals. Another modelling problem can be seen when the system has hysteresis and deadzone at the same time. For example,

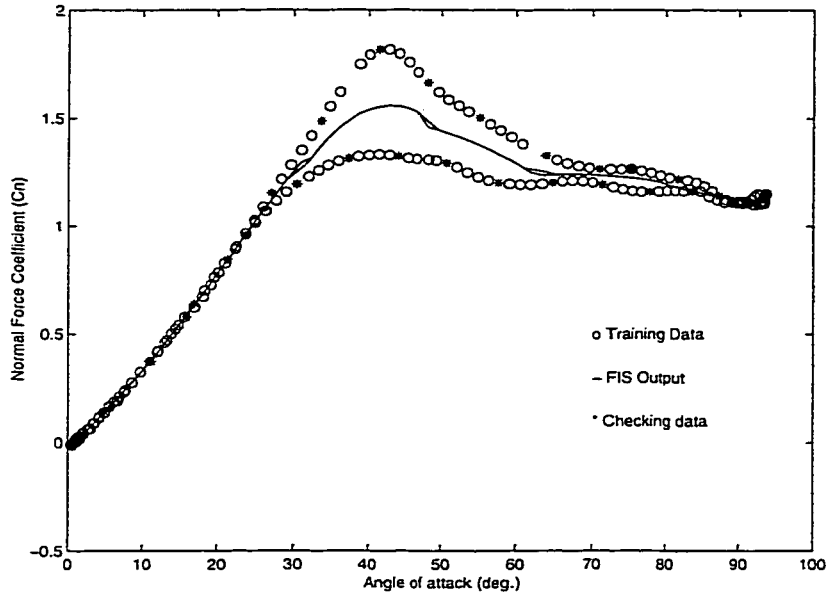

Figure 4.22: ANFIS model of the normal force coefficient as a function of α

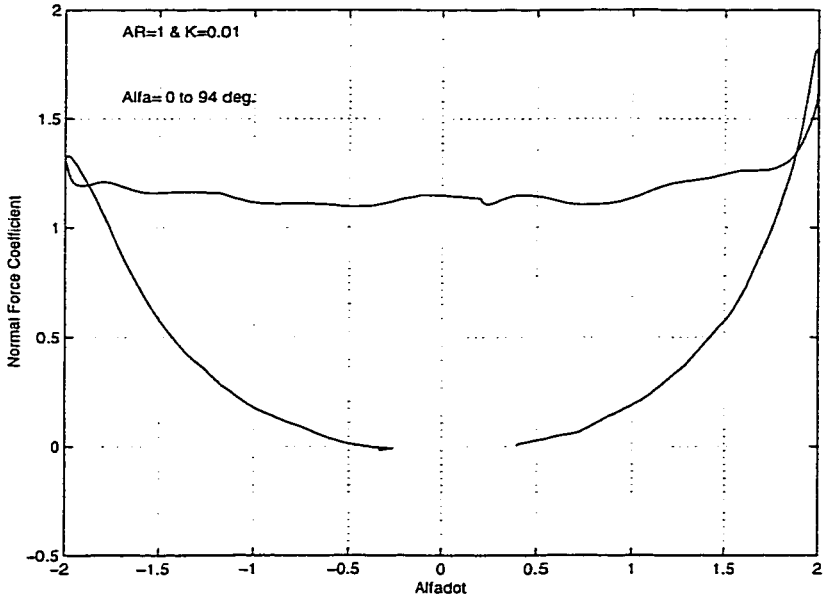

Figure 4.23: A plot of the time derivative of α and the normal force coefficient

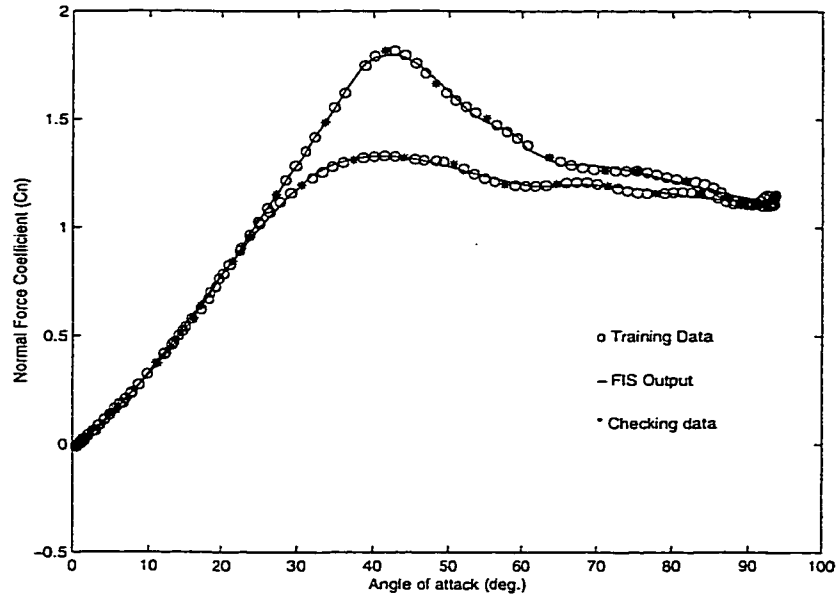

Figure 4.24: ANFIS model of the normal force coefficient as a function of α and $\dot{\alpha}$

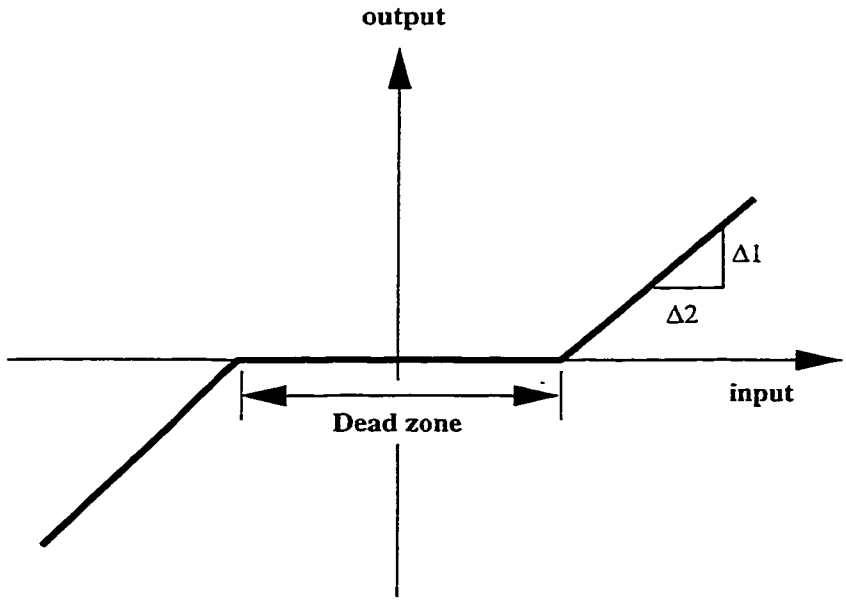

Figure 4.25: Input-output characteristic of deadzone nonlinearity

consider the input-output relationship given in Figure 4.26. We can see a hysteresis loop as well as a dead zone. Following the same procedure as in the previous example, ANFIS is capable of modelling the hysteresis loop but with a poor performance in the deadzone region as shown in Figure 4.27. The time derivative of x is not zero when y is zero in the deadzone region. This problem can be solved by forcing the time derivative of x to be zero in this region, this modification will give a better modelling performance as shown in Figure 4.28.

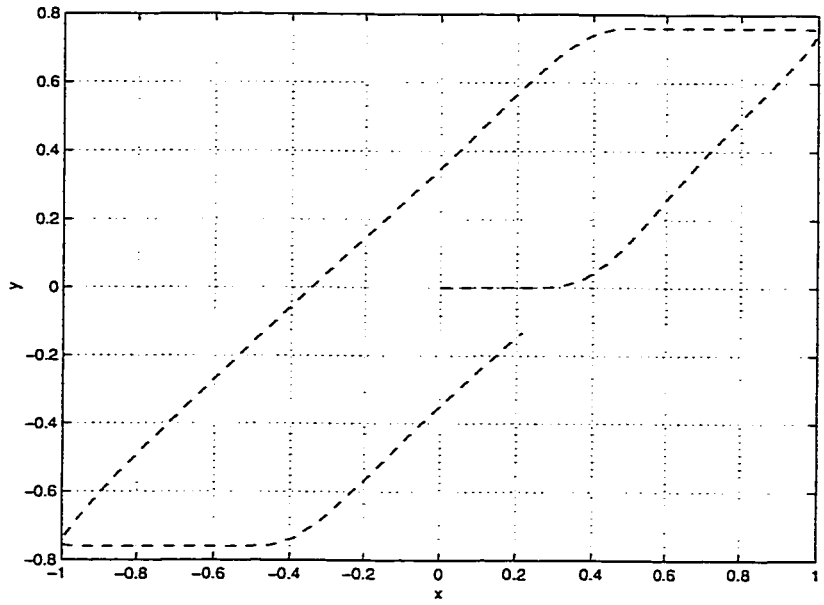

Figure 4.26: A hysteresis loop and a dead zone

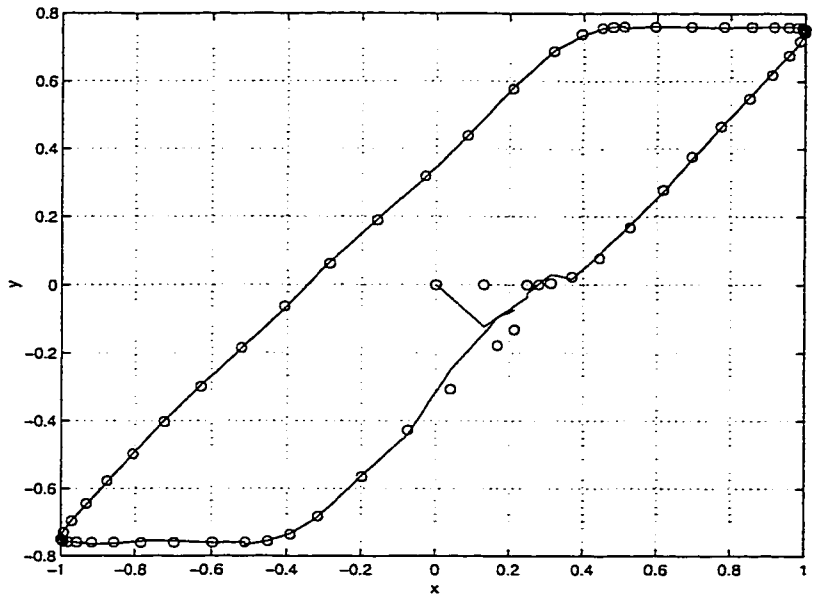

Figure 4.27: A poor modelling performance in the deadzone region

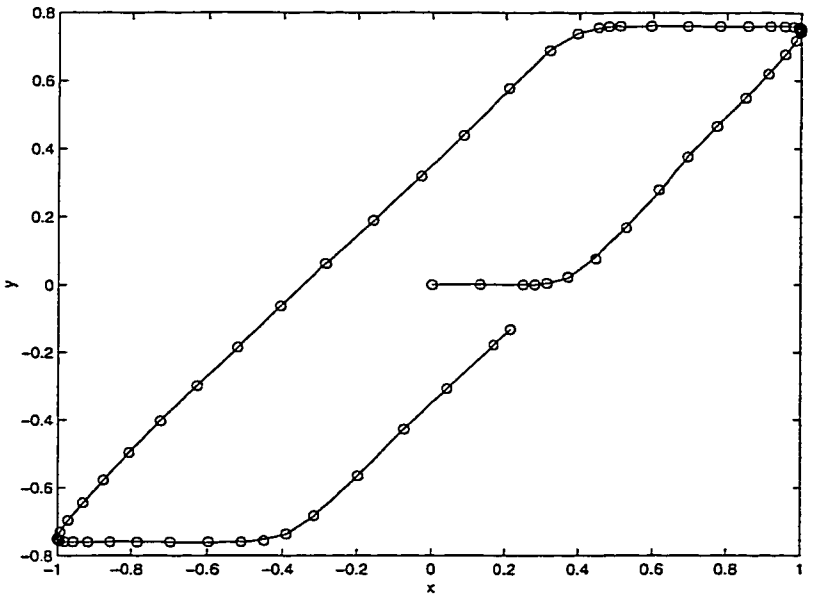

Figure 4.28: A better modelling performance in the deadzone region

Chapter 5

Fuzzy control of nonlinear systems with unknown dynamics using parameterized Sugeno fuzzy approximators

In this chapter, an adaptive feedback control scheme for the tracking of a class of continuous-time plants is presented. A parameterized Sugeno fuzzy approximator is used to adaptively compensate for the plant nonlinearities. All parameters in the fuzzy approximator are tuned using a Luapunov-based design. In the fuzzy approximator, a first-order Sugeno consequent is used in the IF-THEN rules of the fuzzy system, which is observed to have a better approximation capability compared

with that of a constant consequent. Global boundedness of the adaptive system is established. Finally, simulation is used to demonstrate the effectiveness of the proposed controller.

5.1 Fuzzy approximators

5.1.1 Problem statement

In this Chapter, an adaptive control algorithm for a class of dynamic systems is to be developed. The considered systems have the following equation of motion:

$$x^{(n)}(t) + f(x(t), \dot{x}\ (t), \ldots, x^{(n-1)}(t)) = b(x(t), \dot{x}\ (t), \ldots, x^{(n-2)}(t))u(t) \qquad (5.1)$$

where $u(t)$ is the control input, f is an unknown linear or nonlinear function and b is the control gain. The control objective is to force the state $x = [x, \dot{x}, \ldots, x^{(n-1)}]^T$ to follow a specified desired trajectory $x_d = [x_d, \dot{x}_d, \ldots, x_d^{(n-1)}]^T$. In the case considered, an explicit linear parameterization of the function $f(x)$ is unknown or not possible, i.e., $f(x)$ can not be expressed as $f(x) = \sum_{j=1}^{N} \theta_j Y_j(x)$, where θ_j is a set of unknown parameters which appear linearly, and $Y_j(x)$ is a set of known regressors or basis functions. Therefore, the unknown function $f(x)$ will be approximated by a parameterized fuzzy approximator. The required tracking control is achieved by tuning the parameters of the fuzzy approximator.

5.1.2 Fuzzy model

In the Sugeno model, a multi-input and single-output system with n antecedents can be represented as a set of N rules of the following format:

R_j: IF x_1 is A_1^j AND x_2 is A_2^j, ..., x_n is A_n^j THEN $w_j = b_j + a_j^1 x_1 + ... + a_j^n x_n$ for $j = 1, ..., N$

where R_j denotes the jth fuzzy rule, x_i $(i = 1, 2, ..., n)$ is the input, w_j is functional consequent of the fuzzy rule R_j and $b_j, a_j^1, ..., a_j^n$; $(j = 1, 2, ..., N)$ are adjustable design parameters. In this thesis, for simplicity, a MISO system is assumed. In case of a multi output system, several output variables such as w_j^k $(k = 1, 2, ..., m)$ are used, where m represents the number of output variables. A_1^j, A_2^j,..., A_n^j are fuzzy labels of the membership functions. To combine the membership values of the input fuzzy sets in the rule antecedent, any type of fuzzy conjunction operator (AND operator) may be used.

The output of a fuzzy system with Gaussian membership function, product conjunction operator and functional consequent can be written as

$$C(x) = \sum_{j=1}^{N} w_j \left(\prod_{i=1}^{n} \mu_{A_j^i}(x_i) \right)$$

where $C : R^n \rightarrow R$, $x = (x_1, x_2, \ldots, x_n) \in R^n$; w_j is a functional consequent, and $\mu_{A_j^i}(x_i)$ is the membership value, when Gaussian membership function is evaluated at x_i, defined by

$$\mu_{A_j^i}(x_i) = \exp\left[-(\sigma_j^i(x_i - \xi_j^i))^2\right] \tag{5.2}$$

where σ_j^i, ξ_j^i are real-valued parameters. The FBF is defined as

$$g_j(\sigma_j \|X - \xi_j\|) = \prod_{i=1}^{n} \mu_{A_j^i}(x_i), \qquad j = 1, 2, \ldots, N$$

where $\sigma_j = (\sigma_j^1, \sigma_j^2, \ldots, \sigma_j^n) \in U$ and $\xi_j = (\xi_j^1, \xi_j^2, \ldots, \xi_j^n) \in U$, and w_j is defined as

$$w_j = \sum_{i=1}^{n} a_j^i x_i + b_j \tag{5.3}$$

Then, the fuzzy system is equivalent to an FBF expansion:

$$C(x) = \sum_{j=1}^{N} \sum_{i=1}^{n} (a_j^i x_i + b_j) g_j(\sigma_j \|X - \xi_j\|)$$
$$= \sum_{j=1}^{N} \sum_{i=1}^{n} (a_j^i x_i) g_j(\sigma_j \|X - \xi_j\|) + \sum_{j=1}^{N} b_j g_j(\sigma_j \|X - \xi_j\|)$$

Define

$$C(x) = A^T X G(X, \xi, \sigma) + B^T G(X, \xi, \sigma)$$
$$C(x) = A^T L(X, \xi, \sigma) + B^T G(X, \xi, \sigma)$$

where $A = \left[A_1^T, A_2^T, \ldots, A_N^T\right]^T$, $A_j^T = \left[a_1^j, a_2^j, \ldots, a_N^j\right]^T$, $X = [x_1, x_2, \ldots, x_n]^T$, $B = [b_1, b_2, \ldots, b_N]^T$, $\sigma_j = [\sigma_1, \sigma_2, \ldots, \sigma_N]^T$, $\xi_j = [\xi_1, \xi_2, \ldots, \xi_N]^T$, $G(X, \xi, \sigma) = [g_1(\sigma_1 \|X - \xi_1\|), g_2(\sigma_2 \|X - \xi_2\|), \ldots, g_N(\sigma_N \|X - \xi_N\|)]]^T$ and $L(X, \xi, \sigma) = [Xg_1(\sigma_1 \|X - \xi_1\|), Xg_2(\sigma_2 \|X - \xi_2\|), \ldots, Xg_N(\sigma_N \|X - \xi_N\|)^T$.

Remark

A Sugeno first-order consequent model is expected to result in at least the same system performance with fewer rules when compared with a Sugeno constant consequent model. This is due to a better approximation capability of higher order Sugeno consequents [24]. From Equation (5.3) we can see that a Sugeno first-order consequent model is reduced to a constant consequent model when $a_j^i = 0$. Therefore, a Sugeno constant consequent model is a special case of the Sugeno first-order

consequent model, which means that the approximation ability of a first-order consequent rule is at least as good as that of a constant consequent one. This will be demonstrated in an experimental comparison in Chapter 6.

5.1.3 Fuzzy systems as universal approximators

An important property to look for in the Sugeno fuzzy systems, when used as controllers, is the universal approximation property. That is, can a Sugeno model always be constructed to approximate any continuous and nonlinear control solution with any arbitrary accuracy? The issue of fuzzy systems as universal approximators is very important and many significant work has been done in this area. Many studies in literature are for Mamdani fuzzy systems [13–17]. Recently, many researchers have studied the universal approximation property of Sugeno. It is proven in [17] that fuzzy systems with non-fuzzy consequents are universal approximators. Also, it has been constructively proven in [99], in a two step approach using polynomials as the bridge, that Sugeno first order fuzzy systems are universal approximators. The Sugeno systems in [99] are general because they use any type of continuous fuzzy sets, any type of fuzzy conjunction operator, and fuzzy rules with linear consequent. The following theorem states that Sugeno fuzzy systems are universal approximators.

Theorem 1: [13, 19] For any real continuous function f on a compact set and

arbitrary $\varepsilon > 0$, there exist $\overset{*}{C}(x) = \overset{*}{A}^T L(X, \overset{*}{\xi}, \overset{*}{\sigma}) + \overset{*}{B}^T G(X, \overset{*}{\xi}, \overset{*}{\sigma})$ such that

$$\sup \left| f(x) - \overset{*}{C}(x) \right| < \varepsilon \tag{5.4}$$

This theorem states that $\overset{*}{C}(x)$ is a universal approximator, i.e., $\overset{*}{C}(x)$ can approximate the unknown function $f(x)$ with the required accuracy. This theorem is proven in [17] for a general fuzzy system with a non-fuzzy consequent and proven in [99] for a first-order Sugeno fuzzy approximator. The universal approximation property of $\overset{*}{C}(x)$ is characterized by the parameters of the fuzzy sets (ξ, σ) and the parameters of the linear consequents $(b_j, a_j^1, ..., a_j^n)$. $\overset{*}{C}(x)$ can be called as a nonlinearly parameterized fuzzy approximator since ξ and σ appear nonlinearly in the fuzzy system as shown in (5.2).

5.2 Adaptive control using nonlinearly parameterized fuzzy approximators

Based on Theorem 1, the best possible fuzzy approximator is

$$\overset{*}{f}(x) = \overset{*}{A}^T L(X, \overset{*}{\xi}, \overset{*}{\sigma}) + \overset{*}{B}^T G(X, \overset{*}{\xi}, \overset{*}{\sigma})$$

The approximation error on the entire state space can be expressed as:

$$f(x) - \overset{*}{f}(x) = \varepsilon_f(x)$$

Due to Theorem 1, it can be assumed that there exists a constant $\overset{*}{\varepsilon} \geq 0$ such that:

$$|\varepsilon_f(x)| \leq \overset{*}{\varepsilon}$$

To construct $\overset{*}{f}(x)$ the values of the parameters $\overset{*}{A}$, $\overset{*}{B}$, $\overset{*}{\xi}$ and $\overset{*}{\sigma}$ are required. These parameters are replaced with their estimates $\hat{A}$, $\hat{B}$, $\hat{\xi}$ and $\hat{\sigma}$, respectively, so $\hat{f}(x) = \hat{A}^T L(X, \hat{\xi}, \hat{\sigma}) + \hat{B}^T G(X, \hat{\xi}, \hat{\sigma})$ is used to approximate the unknown function f. In this chapter, all parameters in the estimate $\hat{f}(x)$ are tuned. This should provide a better performance than tuning just the consequent parameters A and B and fix the other parameters before controller design. The consequent parameters are easy to tune because they appear linearly in the fuzzy approximator or in the approximation error. It will be possible to update the parameters that appears nonlinearly in the approximation error if it is possible to express the approximation error in a linear parameterized form with respect to each parameter. The approximation capability of the fuzzy approximator can be improved by using a first order Sugeno consequent in the IF-THEN rules of the fuzzy approximator instead of a constant consequent used in [23]. Also higher order consequents will usually minimize the number of rules to be constructed to describe and control the system under consideration [24]. Because $\overset{*}{A}$, $\overset{*}{B}$, $\overset{*}{\xi}$ and $\overset{*}{\sigma}$ are unknown, the approximation function $\overset{*}{f}$ can not be used to directly construct the control law. Using the estimation function $\hat{f}(x)$ of $\overset{*}{f}$, the estimation error $\varepsilon(x)$ need to be formulated.

Theorem 2: The function approximation error between f and $\hat{f}(x)$, written as

$$\varepsilon(x) = f(x) - \hat{f}(x)$$

is equivalent to

$$\varepsilon(x) = \tilde{A}^T(\hat{L} - L'_\xi \hat{\xi} - L'_\sigma \hat{\sigma}) + \hat{A}^T(L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma}) + \tilde{B}^T(\hat{G} - G'_\xi \hat{\xi} - G'_\sigma \hat{\sigma}) + \hat{B}^T(G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma}) + d_f \tag{5.5}$$

where the estimation errors of the parameter vectors are defined as

$$\tilde{A} = \overset{*}{A} - \hat{A}$$

$$\tilde{B} = \overset{*}{B} - \hat{B}$$

$$\tilde{\xi} = \overset{*}{\xi} - \hat{\xi}$$

$$\tilde{\sigma} = \overset{*}{\sigma} - \hat{\sigma}$$

G'_ξ and L'_ξ are derivatives of $G(x, \overset{*}{\xi}, \overset{*}{\sigma})$ and $L(x, \overset{*}{\xi}, \overset{*}{\sigma})$ with respect to $\overset{*}{\xi}$ at $\hat{\xi}$, respectively, also G'_σ and L'_σ are derivatives of $G(x, \overset{*}{\xi}, \overset{*}{\sigma})$ and $L(x, \overset{*}{\xi}, \overset{*}{\sigma})$ with respect to $\overset{*}{\sigma}$ at $\hat{\sigma}$, respectively; d_f is a residual term that satisfies

$$|d_f| = \overset{*}{\theta}_f^T \cdot Y_f$$

where $\overset{*}{\theta}_f$ is an unknown constant vector of optimal weights and bounded constants; and $Y_f = \left[1, \left\|\hat{A}\right\|, \left\|\hat{B}\right\|, \left|\left\|\hat{\xi}\right\|, \left\|\hat{\sigma}\right\|\right]^T$ is a known function vector.

Proof: The approximation error between f and $\overset{*}{f}$ is denoted by $\varepsilon_f(x)$. The approximation error $\varepsilon(x) = f(x) - \hat{f}(x)$ can be written as

$$\varepsilon(x) = f(x) - \hat{A}^T \hat{L} - \hat{B}^T \hat{G}$$

$$\varepsilon(x) = \overset{*}{f}(x) - \hat{A}^T \hat{L} - \hat{B}^T \hat{G} + \varepsilon_f(x)$$

$$\varepsilon(x) = \overset{*}{A}^T \overset{*}{L} - \overset{*}{B}^T \overset{*}{G} - \hat{A}^T \hat{L} - \hat{B}^T \hat{G} + \varepsilon_f(x)$$

$$\varepsilon(x) = \overset{*}{A}^T \overset{*}{L} - \overset{*}{A}^T \hat{L} + \overset{*}{A}^T \hat{L} + \overset{*}{B}^T \overset{*}{G} - \overset{*}{B}^T \hat{G} + \overset{*}{B}^T \hat{G} - \hat{A}^T \hat{L} - \hat{B}^T \hat{G} + \varepsilon_f(x)$$

$$\varepsilon(x) = \tilde{A}^T \tilde{L} + \hat{A}^T \tilde{L} + \tilde{B}^T \tilde{G} + \hat{B}^T \tilde{G} + \tilde{A}^T \hat{L} + \tilde{B}^T \hat{G} + \varepsilon_f(x)$$

Taking the Taylor's series expansion of $\overset{*}{G}$ and $\overset{*}{L}$ at $\overset{*}{\xi}=\hat{\xi}$ and $\overset{*}{\sigma}=\hat{\sigma}$, $\tilde{G}$ and $\tilde{L}$ can be expressed as

$$\tilde{G} = G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma} + o(x, \tilde{\xi}, \tilde{\sigma})$$

$$\tilde{L} = L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma} + h(x, \tilde{\xi}, \tilde{\sigma})$$

where $o(.)$ is the sum of the high order terms in Taylor's series expansion, G'_ξ and L'_ξ are derivatives of $G(x, \overset{*}{\xi}, \overset{*}{\sigma})$ and $L(x, \overset{*}{\xi}, \overset{*}{\sigma})$ with respect to $\overset{*}{\xi}$ at $\hat{\xi}$ and expressed as

$$G'_\xi = G'_\xi(x, \hat{\xi}, \hat{\sigma}) = \left. \frac{\partial G(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\xi}} \right| \overset{*}{\xi}=\hat{\xi}, \overset{*}{\sigma}=\hat{\sigma}$$

$$L'_\xi = L'_\xi(x, \hat{\xi}, \hat{\sigma}) = \left. \frac{\partial L(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\xi}} \right| \overset{*}{\xi}=\hat{\xi}, \overset{*}{\sigma}=\hat{\sigma}$$

also, G'_σ and L'_σ are derivatives of $G(x, \overset{*}{\xi}, \overset{*}{\sigma})$ and $L(x, \overset{*}{\xi}, \overset{*}{\sigma})$ with respect to $\overset{*}{\sigma}$ at $\hat{\sigma}$ and expressed as

$$G'_\sigma = G'_\sigma(x, \hat{\xi}, \hat{\sigma}) = \left. \frac{\partial G(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\sigma}} \right| \overset{*}{\xi}=\hat{\xi}, \overset{*}{\sigma}=\hat{\sigma}$$

$$L'_\sigma = L'_\sigma(x, \hat{\xi}, \hat{\sigma}) = \left. \frac{\partial L(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\sigma}} \right| \overset{*}{\xi}=\hat{\xi}, \overset{*}{\sigma}=\hat{\sigma}$$

Therefore,

$$\varepsilon(x) = \tilde{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma} + h(x, \tilde{\xi}, \tilde{\sigma})) + \hat{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma} + h(x, \tilde{\xi}, \tilde{\sigma})) +$$

$$\tilde{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma} + o(x, \tilde{\xi}, \tilde{\sigma})) - \hat{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma} + o(x, \tilde{\xi}, \tilde{\sigma})) + \tilde{A}^T \hat{L} + \tilde{B}^T \hat{G}$$

$$+ \varepsilon_f(x)$$

$$\varepsilon(x) = \tilde{A}^T L'_\xi(\overset{*}{\xi} - \hat{\xi}) + \tilde{A}^T L'_\sigma(\overset{*}{\sigma} - \hat{\sigma}) + \tilde{A}^T h(x, \tilde{\xi}, \tilde{\sigma}) + \hat{A}^T L'_\xi \tilde{\xi} + \hat{A}^T L'_\sigma \tilde{\sigma}$$

$$+\hat{A}^T h(x,\tilde{\xi},\tilde{\sigma})+\tilde{B}^T G'_{\xi}(\overset{*}{\xi}-\hat{\xi})+\tilde{B}^T G'_{\sigma}(\overset{*}{\sigma}-\hat{\sigma})+\tilde{B}^T o(x,\tilde{\xi},\tilde{\sigma})+\hat{B}^T G'_{\xi}\,\tilde{\xi}+\hat{B}^T$$

$$G'_{\sigma}\,\tilde{\sigma}+\hat{B}^T o(x,\tilde{\xi},\tilde{\sigma})+\tilde{A}^T\hat{G}+\tilde{B}^T\hat{G}+\varepsilon_f(x)$$

$$\varepsilon(x)=\tilde{A}^T(\hat{L}-L'_{\xi}\hat{\xi}-L'_{\sigma}\hat{\sigma})+\hat{A}^T(L'_{\xi}\tilde{\xi}+L'_{\sigma}\tilde{\sigma})+\tilde{B}^T(\hat{G}-G'_{\xi}\hat{\xi}-G'_{\sigma}\hat{\sigma})+$$

$$\hat{B}^T(G'_{\xi}\tilde{\xi}+G'_{\sigma}\tilde{\sigma})+d_f$$

$$d_f=\tilde{A}^T(L'_{\xi}\overset{*}{\xi}+L'_{\sigma}\overset{*}{\sigma})+\overset{*}{A}^T h(x,\tilde{\xi},\tilde{\sigma})+\tilde{B}^T(G'_{\xi}\overset{*}{\xi}+G'_{\sigma}\overset{*}{\sigma})+\overset{*}{B}^T o(x,\tilde{\xi},\tilde{\sigma})+\varepsilon_f(x)$$

It was proven in [23] that higher order terms are bounded by

$$\left\|o(x,\tilde{\xi},\tilde{\sigma})\right\|=\left\|\tilde{G}-G'_{\xi}\tilde{\xi}-G'_{\sigma}\tilde{\sigma}\right\|\le\left\|\tilde{G}\right\|-\left\|G'_{\xi}\right\|\left\|\tilde{\xi}\right\|-\left\|G'_{\sigma}\right\|\left\|\tilde{\sigma}\right\|\le c_1+c_2\left\|\tilde{\xi}\right\|+c_3\left\|\tilde{\sigma}\right\|$$

$$\left\|h(x,\tilde{\xi},\tilde{\sigma})\right\|=\left\|\tilde{L}-L'_{\xi}\tilde{\xi}-L'_{\sigma}\tilde{\sigma}\right\|\le\left\|\tilde{L}\right\|-\left\|L'_{\xi}\right\|\left\|\tilde{\xi}\right\|-\left\|L'_{\sigma}\right\|\left\|\tilde{\sigma}\right\|\le c_4+c_5\left\|\tilde{\xi}\right\|+c_6\left\|\tilde{\sigma}\right\|$$

where c_1, c_2, c_3, c_4, c_5 and c_6 are some bounded constants due to the fact that the FBF and its derivative are always bounded by constants.

There should exist constants $\bar{\xi},\bar{\sigma}$, $\bar{A}$ and $\bar{B}$ satisfying $\left\|\overset{*}{\xi}\right\|\le\bar{\xi}$, $\left\|\overset{*}{\sigma}\right\|\le\bar{\sigma}$, $\left\|\overset{*}{A}\right\|\le\bar{A}$ and $\left\|\overset{*}{B}\right\|\le\bar{B}$

Based on the following facts

$$\left\|\tilde{\xi}\right\|\le\left\|\overset{*}{\xi}\right\|+\left\|\hat{\xi}\right\|\le\bar{\xi}+\left\|\hat{\xi}\right\|$$

$$\left\|\tilde{\sigma}\right\|\le\left\|\overset{*}{\sigma}\right\|+\left\|\hat{\sigma}\right\|\le\bar{\sigma}+\left\|\hat{\sigma}\right\|$$

$$\left\|\tilde{B}\right\|\le\left\|\overset{*}{B}\right\|+\left\|\hat{B}\right\|\le\bar{B}+\left\|\hat{B}\right\|$$

$$\left\|\tilde{A}\right\|\le\left\|\overset{*}{A}\right\|+\left\|\hat{A}\right\|\le\bar{A}+\left\|\hat{A}\right\|$$

The term d_f can be bounded as

$$|d_f|=\left\|\tilde{A}\,L'_{\xi}\overset{*}{\xi}+\tilde{A}^T L'_{\sigma}\overset{*}{\sigma}+\overset{*}{A}^T h(x,\tilde{\xi},\tilde{\sigma})+\tilde{B}^T G'_{\xi}\overset{*}{\xi}+\tilde{B}^T G'_{\sigma}\overset{*}{\sigma}+\overset{*}{B}^T o(x,\tilde{\xi},\tilde{\sigma})+\varepsilon_f(x)\right\|$$

$$|d_f|\le\left\|\tilde{A}\right\|\left\|L'_{\xi}\right\|\left\|\overset{*}{\xi}\right\|+\left\|\tilde{A}\right\|\left\|L'_{\sigma}\right\|\left\|\overset{*}{\sigma}\right\|+\left\|\overset{*}{A}\right\|(c_4+c_5\left\|\tilde{\xi}\right\|+c_6\left\|\tilde{\sigma}\right\|)+\left\|\tilde{B}\right\|\left\|G'_{\xi}\right\|\left\|\overset{*}{\xi}\right\|$$

$$+\left\|\tilde{B}^T\right\|\left\|G'_\sigma\right\|\left\|\overset{*}{\sigma}\right\|+\left\|\overset{*}{B}^T\right\|(c_1+c_2\left\|\tilde{\xi}\right\|+c_3\left\|\tilde{\sigma}\right\|)+\overset{*}{\varepsilon}$$

$$|d_f|\leq(\bar{A}+\left\|\hat{A}\right\|)c_5\,\bar{\xi}+(\bar{A}+\left\|\hat{A}\right\|)c_6\,\bar{\sigma}+\bar{A}\,c_4+\bar{A}\,c_5(\bar{\xi}+\left\|\hat{\xi}\right\|)+\bar{A}\,c_6(\bar{\sigma}+\left\|\hat{\sigma}\right\|)+$$

$$(\bar{B}+\left\|\tilde{B}\right\|)c_2\,\bar{\xi}+(\bar{B}+\left\|\hat{B}\right\|)c_3\,\bar{\sigma}+\bar{B}\;c_1+\bar{B}\,c_2(\bar{\xi}+\left\|\hat{\xi}\right\|)+\bar{B}\,c_3(\bar{\sigma}+\left\|\hat{\sigma}\right\|)+\overset{*}{\varepsilon}$$

$$|d_f|\leq 2c_5\,\bar{A}\bar{\xi}+2c_6\,\bar{A}\bar{\sigma}+c_4\,\bar{A}+(c_5\,\bar{\xi}+c_6\,\bar{\sigma})\left\|\hat{A}\right\|+c_5\,\bar{A}\left\|\hat{\xi}\right\|+c_6\,\bar{A}\left\|\hat{\sigma}\right\|+$$

$$2c_2\,\bar{B}\bar{\xi}+2c_3\,\bar{B}\bar{\sigma}+c_1\,\bar{B}+(c_2\,\bar{\xi}+c_3\,\bar{\sigma})\left\|\hat{B}\right\|+c_2\,\bar{B}\left\|\hat{\xi}\right\|+c_3\,\bar{B}\left\|\hat{\sigma}\right\|+\overset{*}{\varepsilon}$$

$$=\overset{*}{\theta}_f^T\cdot Y_f$$

$$=\left[\overset{*}{\theta}_{f1},\overset{*}{\theta}_{f2},\overset{*}{\theta}_{f3},\overset{*}{\theta}_{f4},\overset{*}{\theta}_{f5}\right]\cdot\left[1,\left\|\hat{A}\right\|,\left\|\hat{B}\right\|,\left\|\hat{\xi}\right\|,\left\|\hat{\sigma}\right\|\right]^T$$

$$\overset{*}{\theta}_{f1}=2c_5\,\bar{A}\bar{\xi}+2c_6\,\bar{A}\bar{\sigma}+c_4\,\bar{A}+2c_2\,\bar{B}\bar{\xi}+2c_3\,\bar{B}\bar{\sigma}+c_1\,\bar{B}+\overset{*}{\varepsilon}$$

$$\overset{*}{\theta}_{f2}=(c_5\,\bar{\xi}+c_6\,\bar{\sigma})$$

$$\overset{*}{\theta}_{f3}=(c_2\,\bar{\xi}+c_3\,\bar{\sigma})$$

$$\overset{*}{\theta}_{f4}=c_5\,\bar{A}+c_2\,\bar{B}$$

$$\overset{*}{\theta}_{f5}=c_6\,\bar{A}+c_3\,\bar{B}$$

The approximation error is expressed in a linearly parameterized form with respect to $\tilde{A}$, $\tilde{B}$, $\tilde{\xi}$ and $\tilde{\sigma}$, which makes the updates of $\hat{A}$, $\hat{B}$, $\hat{\xi}$ and $\hat{\sigma}$ possible. Also, note that the term d_f is not a constant and the assumption on the constant bound will not be imposed in the developed control method. This will make the developed controller more general and more applicable.

5.3 Controller design

Before the introduction of the control law, we defined a filtered tracking error as:

$$s(t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} \tilde{X}(t) \quad \text{with} \quad \lambda > 0 \tag{5.6}$$

where $s(t)$ is an error metric, which can be rewritten as $s(t) = \Lambda^T \tilde{X}(t)$ with $\Lambda^T = [\lambda^{n-1}, (n-1)\lambda^{n-2}, \ldots, 1]$, and $\tilde{X} = X - X_d$. The tracking error vector exponentially approaches zero when $s(t) = 0$, therefore the objective is to design a controller which is able to drive $s(t)$ to zero. It can be easily proven that

$$\dot{s}(t) = -X_d^{(n)}(t) + \Lambda_v^T \tilde{X}(t) + bu - f(X) \tag{5.7}$$

where $\Lambda_v^T = [0, \lambda^{n-1}, (n-1)\lambda^{n-2}, \ldots, (n-1)\lambda]$. Using

$$\hat{f}(X) = \hat{A}^T L(X, \hat{\xi}, \hat{\sigma}) + \hat{B}^T G(X, \hat{\xi}, \hat{\sigma})$$

The Equation in (5.7) can be rewritten as:

$$\dot{s}(t) = -X_d^{(n)}(t) + \Lambda_v^T \tilde{X}(t) + bu - \hat{f}(X) - \varepsilon(X)$$

where $\varepsilon(X) = f(X) - \hat{f}(X)$, which is the fuzzy reconstruction error. The adaptive control law, for a unity control gain case, is defined as

$$u(t) = -k_d s(t) + u_{fd}(t) + u_{fu}(t) \tag{5.8}$$

$$u_{fd}(t) = X_d^{(n)}(t) - \Lambda_v^T \tilde{X}(t) \tag{5.9}$$

$$u_{fu}(t) = \hat{A}^T L(X, \hat{\xi}, \hat{\sigma}) + \hat{B}^T G(X, \hat{\xi}, \hat{\sigma}) - \hat{\theta}_f Y_f \operatorname{sgn}(s) \tag{5.10}$$

$$\dot{\hat{A}} = -s(t)\Gamma_1(\hat{L} - L'_\xi\,\hat{\xi} - L'_\sigma\,\hat{\sigma}) \tag{5.11}$$

$$\dot{\hat{\xi}} = -s(t)\Gamma_2(\hat{A}^T\,L'_\xi + \hat{B}^T\,G'_\xi)^T \tag{5.12}$$

$$\dot{\hat{\sigma}} = -s(t)\Gamma_3(\hat{A}^T\,L'_\sigma + \hat{B}^T\,G'_\sigma)^T \tag{5.13}$$

$$\dot{\hat{B}} = -s(t)\Gamma_4(\hat{G} - G'_\xi\,\hat{\xi} - G'_\sigma\,\hat{\sigma}) \tag{5.14}$$

$$\dot{\hat{\theta}}_f = |s(t)|\,\Gamma_5 Y_f \tag{5.15}$$

where $\Gamma_1, \ldots, \Gamma_5$ are symmetric positive definite matrices which determine the rates of adaptation.

Remarks:

1. Compared with [23], the controller given in this chapter has an additional vector for each input that needs to be tuned. This will require more effort to tune the parameters, but it is expected to enhance the tracking performance due to a better approximation capabilities of the fuzzy approximator with a first order Sugeno consequent than that of a constant consequent.

2. Compared with [22], which also uses a Sugeno approximator, the controller design approach is quite different. In [22], an optimal controller is firstly designed and the fuzzy approximator was used to approximate the designed optimal controller, while our approach just approximates the unknown plant by a fuzzy model and uses this plant approximator for the controller design.

In this case, no assumptions are required and control performance is superior (see simulation example).

5.3.1 Stability analysis

The stability of the closed-loop system with the developed adaptive control law is shown by the following theorem.

Theorem 3: If the control law in (5.8-5.15) is applied to a plant with a unity control gain, then all states in the adaptive system will remain bounded and $x(t) \rightarrow 0$ as $t \rightarrow \infty$.

Proof: Consider the Luapunov function candidate:

$$V(t) = \frac{1}{2}\left(s^2(t) + \tilde{A}^T \Gamma_1^{-1} \tilde{A} + \tilde{\xi}^T \Gamma_2^{-1} \tilde{\xi} + \tilde{\sigma}^T \Gamma_3^{-1} \tilde{\sigma} + \tilde{B}^T \Gamma_4^{-1} \tilde{B} + \tilde{\theta}_f^T \Gamma_5^{-1} \tilde{\theta}_f\right) \tag{5.16}$$

Taking the derivative of both sides:

$$\dot{V}(t) = s(t)\,\dot{s}(t) - \tilde{A}^T \Gamma_1^{-1} \dot{\hat{A}} - \dot{\hat{\xi}}^T \Gamma_2^{-1} \tilde{\xi} - \dot{\hat{\sigma}}^T \Gamma_3^{-1} \tilde{\sigma} - \tilde{B}^T \Gamma_4^{-1} \dot{\hat{B}} - \tilde{\theta}_f^T \Gamma_5^{-1} \dot{\hat{\theta}}_f \tag{5.17}$$

Equation (5.10) can be rewritten as:

$$u_{fu}(t) = f(x) - \varepsilon(x) - \hat{\theta}_f^T Y_f \,\mathrm{sgn}(s) \tag{5.18}$$

Recall that

$$\varepsilon(x) = \tilde{A}^T (\hat{L} - L'_\xi \hat{\xi} - L'_\sigma \hat{\sigma}) + \hat{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma}) + \tilde{B}^T (\hat{G} - G'_\xi \hat{\xi} - G'_\sigma \hat{\sigma}) + \hat{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma}) + d_f \tag{5.19}$$

Then

$$u_{fu}(t) = f(x) + \tilde{A}^T (L'_\xi \hat{\xi} + L'_\sigma \hat{\sigma} - \hat{L}) - \hat{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma}) + \tilde{B}^T (G'_\xi \hat{\xi} + G'_\sigma \hat{\sigma} - \hat{G}) - \hat{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma}) - d_f - \hat{\theta}_f^T Y_f \operatorname{sgn}(s) \tag{5.20}$$

$$u(t) = -k_d s(t) + X_d^{(n)}(t) - \Lambda_v^T \tilde{X}(t) + f(x) + \tilde{A}^T (L'_\xi \hat{\xi} + L'_\sigma \hat{\sigma} - \hat{L}) - \hat{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma}) + \tilde{B}^T (G'_\xi \hat{\xi} + G'_\sigma \hat{\sigma} - \hat{G}) - \hat{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma}) - d_f - \hat{\theta} f^T Y_f \operatorname{sgn}(s) \tag{5.21}$$

From (5.7) and (5.21)

$$\dot{s}(t) = -k_d s(t) + \tilde{A}^T (L'_\xi \hat{\xi} + L'_\sigma \hat{\sigma} - \hat{L}) - \hat{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma}) + \tilde{B}^T (G'_\xi \hat{\xi} + G'_\sigma \hat{\sigma} - \hat{G}) - \hat{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma}) - d_f - \hat{\theta}_f^T Y_f \operatorname{sgn}(s) \tag{5.22}$$

Then

$$\dot{V}(t) = -k_d s^2(t) + s(t) \tilde{A}^T (L'_\xi \hat{\xi} + L'_\sigma \hat{\sigma} - \hat{L}) - s(t) \hat{A}^T (L'_\xi \tilde{\xi} + L'_\sigma \tilde{\sigma}) + s(t) \tilde{B}^T (G'_\xi \hat{\xi} + G'_\sigma \hat{\sigma} - \hat{G}) - s(t) \hat{B}^T (G'_\xi \tilde{\xi} + G'_\sigma \tilde{\sigma}) - s(t) \left[d_f + \hat{\theta}_f^T Y_f \operatorname{sgn}(s) \right] + \tilde{A}^T \Gamma_1^{-1} [s(t) \Gamma_1 (\hat{L} - L'_\xi \hat{\xi} - L'_\sigma \hat{\sigma})] + s(t) (\hat{A}^T L'_\xi + \hat{B}^T G'_\xi)^T \Gamma_2 \Gamma_2^{-1} \tilde{\xi} + s(t) (\hat{A}^T L'_\sigma + \hat{B}^T G'_\sigma)^T \Gamma_3 \Gamma_3^{-1} \tilde{\sigma} + \tilde{B}^T \Gamma_4^{-1} \left[s(t) \Gamma_4 (\hat{G} - G'_\xi \hat{\xi} - G'_\sigma \hat{\sigma}) \right] - |s(t)| \tilde{\theta}_f^T \Gamma_5^{-1} \Gamma_5 Y_f$$

$$\begin{aligned}
\dot{V}(t) &= -k_d s^2(t) - s(t) d_f - s(t) \hat{\theta}_f^T Y_f \operatorname{sgn}(s) - |s(t)| \tilde{\theta}_f^T Y_f \\
&= -k_d s^2(t) - s(t) d_f - |s(t)| \hat{\theta}_f^T Y_f - \left[|s(t)| \overset{*}{\theta}_f^T Y_f - |s(t)| \hat{\theta}_f^T Y_f \right] \\
&= -k_d s^2(t) - s(t) d_f - |s(t)| \overset{*}{\theta}_f^T Y_f \\
&\leq -k_d s^2(t)
\end{aligned}$$

where the facts $|d_f| < \overset{*}{\theta}_f^T Y_f$ and $s(t) \operatorname{sgn}(s) = |s(t)|$ have been used. Therefor, all

signals in the system are bounded. It is important to note that $s(t) \rightarrow 0$ as $t \rightarrow \infty$ has been established in [23], which completes the proof and establish asymptotic convergence of the tracking error.

5.3.2 Nonunity control gain

We extend the result to plants with nonunity control gain. The following assumptions should be stated first:

1. The control gain is finite and non zero.

2. The functions $h(X) = \frac{f(X)}{b(X)}$ and $g(x) = \frac{1}{b(X)}$ are bounded by known positive functions $M_0(X)$ and $M_1(X)$.

3. There exist a known positive function $M_2(X)$ such that

$$\left|\frac{d}{dt}g(X)\right| \leq M_2(X)\,\|X\|$$

Let $\hat{h}\ (X) = \hat{A}_h^T\ L(X, \hat{\xi}_h, \hat{\sigma}_h) + \hat{B}_h^T\ G(X, \hat{\xi}_h, \hat{\sigma}_h)$ and $\hat{g}\ (X) = \hat{A}_g^T\ L(X, \hat{\xi}_g, \hat{\sigma}_g) + \hat{B}_g^T\ G(X, \hat{\xi}_g, \hat{\sigma}_g)$ be the estimates of the optimal fuzzy approximators $\overset{*}{h}\ (X) = \overset{*}{A}_h^T\ L(X, \overset{*}{\xi}_h, \overset{*}{\sigma}_h) + \overset{*}{B}_h^T\ G(X, \overset{*}{\xi}_h, \overset{*}{\sigma}_h)$ and $\overset{*}{g}\ (X) = \overset{*}{A}_g^T\ L(X, \overset{*}{\xi}_g, \overset{*}{\sigma}_g) + \overset{*}{B}_g^T\ G(X, \overset{*}{\xi}_g, \overset{*}{\sigma}_g)$, respectively. We can still get the following approximation error properties

$\tilde{h} = h - \hat{h} = \tilde{A}_h^T\ (\hat{L}_h - L'_{h\xi}\ \hat{\xi}_h - L'_{h\sigma}\ \hat{\sigma}_h) + \hat{A}_h^T\ (L'_{h\xi}\ \tilde{\xi}_h + L'_{h\sigma}\ \tilde{\sigma}_h) + \tilde{B}_h^T\ (\hat{G}_h - G'_{h\xi}\ \hat{\xi}_h - G'_{h\sigma}\ \hat{\sigma}_h) + \hat{B}_h^T\ (G'_{h\xi}\ \tilde{\xi}_h + G'_{h\sigma}\ \tilde{\sigma}_h) + d_h$

also

$\tilde{g}= g- \hat{g}=\tilde{A}_g^T\ (\hat{L}_g\ -L'_{g\xi}\ \hat{\xi}_g\ -L'_{g\sigma}\ \hat{\sigma}_g)+\ \hat{A}_g^T\ (L'_{g\xi}\ \tilde{\xi}_g\ +L'_{g\sigma}\ \tilde{\sigma}_g)+\ \tilde{B}_g^T\ (\hat{G}_g\ -G'_{g\xi}\ \hat{\xi}_g$
$-G'_{g\sigma}\ \hat{\sigma}_g)+\ \hat{B}_g^T\ (G'_{g\xi}\ \tilde{\xi}_g\ +G'_{g\sigma}\ \tilde{\sigma}_g)+d_g$

Furthermore, $|d_h| <\theta_h^{*T}\ Y_h$ and $|d_g| <\theta_g^{*T}\ Y_g$. The robust adaptive control law for the case of the nonunity control gain is:

$$u(t) = -k_d s(t) - \frac{1}{2} M_2(X)\, \|X\|\, s(t) + u_{fu}(t) \tag{5.23}$$

$$\begin{aligned} u_{fu}(t) =& \hat{A}_h^T\ L(X,\hat{\xi}_h,\hat{\sigma}_h)+\ \hat{B}_h^T\ G(X,\hat{\xi}_h,\hat{\sigma}_h)+\ \hat{A}_g^T\ L(X,\hat{\xi}_g,\hat{\sigma}_g)a_r \\ &+\ \hat{B}_g^T\ G(X,\hat{\xi}_g,\hat{\sigma}_g)a_r - \left(\hat{\theta}_h^T\ Y_h+\ \hat{\theta}_g^T\ Y_g\right)\ \mathrm{sgn}(s(t)) \end{aligned} \tag{5.24}$$

$$\dot{\hat{A}}_h= -s(t)\Gamma_{h1}(\hat{L}_h\ -L'_{h\xi}\ \hat{\xi}_h\ -L'_{h\sigma}\ \hat{\sigma}_h) \tag{5.25}$$

$$\dot{\hat{\xi}}_h= -s(t)\Gamma_{h2}(\hat{A}_h^T\ L'_{h\xi}+\ \hat{B}_h^T\ G'_{h\xi})^T \tag{5.26}$$

$$\dot{\hat{\sigma}}_h= -s(t)\Gamma_{h3}(\hat{A}_h^T\ L'_{h\sigma}+\ \hat{B}_h^T\ G'_{h\sigma})^T \tag{5.27}$$

$$\dot{\hat{B}}_h= -s(t)\Gamma_{h4}(\hat{G}_h\ -G'_{h\xi}\ \hat{\xi}_h\ -G'_{h\sigma}\ \hat{\sigma}_h) \tag{5.28}$$

$$\dot{\hat{\theta}}_h= |s(t)|\,\Gamma_{h5}Y_h \tag{5.29}$$

$$\dot{\hat{A}}_g= -s(t)\Gamma_{g1}(\hat{L}_g\ -L'_{g\xi}\ \hat{\xi}_g\ -L'_{g\sigma}\ \hat{\sigma}_g)a_r \tag{5.30}$$

$$\dot{\hat{\xi}}_g= -s(t)\Gamma_{g2}(\hat{A}_g^T\ L'_{g\xi}+\ \hat{B}_g^T\ G'_{g\xi})^T a_r \tag{5.31}$$

$$\dot{\hat{\sigma}}_g= -s(t)\Gamma_{g3}(\hat{A}_g^T\ L'_{g\sigma}+\ \hat{B}_g^T\ G'_{g\sigma})^T a_r \tag{5.32}$$

$$\dot{\hat{B}}_g= -s(t)\Gamma_{g4}(\hat{G}_g\ -G'_{g\xi}\ \hat{\xi}_g\ -G'_{g\sigma}\ \hat{\sigma}_g)a_r \tag{5.33}$$

$$\dot{\hat{\theta}}_g= |s(t)|\,\Gamma_{g5}Y_g\,|a_r| \tag{5.34}$$

where $\hat{A}_h$, $\hat{\xi}_h$, $\hat{\sigma}_h$, $\hat{B}_h$, $\hat{A}_g$, $\hat{\xi}_g$, $\hat{\sigma}_g$, $\hat{B}_g$ are the estimates of $\overset{*}{A}_h$, $\overset{*}{\xi}_h$, $\overset{*}{\sigma}_h$, $\overset{*}{B}_h$, $\overset{*}{A}_g$, $\overset{*}{\xi}_g$, $\overset{*}{\sigma}_g$, $\overset{*}{B}_g$, $a_r = X_d^{(n)}(t) - \Lambda_v^T \tilde{X}$ (t) with $\Lambda_v^T = [0, \lambda^{n-1}, (n-1)\lambda^{n-2}, \ldots, (n-1)\lambda]$, $\Gamma_{h1}, \ldots, \Gamma_{h5}$ and $\Gamma_{g1}, \ldots, \Gamma_{g5}$ are symmetric positive definite matrices which determine the rates of adaptation. The stability of the closed loop system with nonunity control gain is established in the following theorem.

Theorem 4: If the control law in (5.23-5.34) is applied to a plant with a nonunity control gain, then all states in the adaptive system will remain bounded and $x(t) \to 0$ as $t \to \infty$.

Proof: Consider the following Luapunov function candidate:

$$V(t) = \tfrac{1}{2}[g(x)s^2(t) + \tilde{A}_h^T \Gamma_{h1}^{-1} \tilde{A}_h + \tilde{\xi}_h^T \Gamma_{h2}^{-1} \tilde{\xi}_h + \tilde{\sigma}_h^T \Gamma_{h3}^{-1} \tilde{\sigma}_h + \tilde{B}_h^T \Gamma_{h4}^{-1} \tilde{B}_h + \tilde{\theta}_h^T \Gamma_{h5}^{-1} \tilde{\theta}_h + \tilde{A}_g^T \Gamma_{g1}^{-1} \tilde{A}_g + \tilde{\xi}_g^T \Gamma_{g2}^{-1} \tilde{\xi}_g + \tilde{\sigma}_g^T \Gamma_{g3}^{-1} \tilde{\sigma}_g + \tilde{B}_g^T \Gamma_{g4}^{-1} \tilde{B}_g + \tilde{\theta}_g^T \Gamma_{g5}^{-1} \tilde{\theta}_g] \tag{5.35}$$

Taking the derivative of both sides:

$$\dot{V}(t) = \tfrac{1}{2} \dot{g}(x)s^2(t) + s(t)g(x)\dot{s}(t) - \tilde{A}_h^T \Gamma_{h1}^{-1} \dot{\hat{A}}_h - \dot{\hat{\xi}}_h^T \Gamma_{h2}^{-1} \tilde{\xi}_h - \dot{\hat{\sigma}}_h^T \Gamma_{h3}^{-1} \tilde{\sigma}_h - \tilde{B}_h^T \Gamma_{h4}^{-1} \dot{\hat{B}}_h - \tilde{\theta}_h^T \Gamma_{h5}^{-1} \dot{\hat{\theta}}_h - \tilde{A}_g^T \Gamma_{g1}^{-1} \dot{\hat{A}}_g - \dot{\hat{\xi}}_g^T \Gamma_{g2}^{-1} \tilde{\xi}_g - \dot{\hat{\sigma}}_g^T \Gamma_{g3}^{-1} \tilde{\sigma}_g - \tilde{B}_g^T \Gamma_{g4}^{-1} \dot{\hat{B}}_g - \tilde{\theta}_g^T \Gamma_{g5}^{-1} \dot{\hat{\theta}}_g \tag{5.36}$$

Equation (5.7) can be rewritten as:

$$g(x)\dot{s}(t) = -h(x) + u(t) - g(x)a_r \tag{5.37}$$

From (5.23) and (5.37)

$$g(x)\dot{s}(t) = -k_d s(t) - \frac{1}{2} M_2(X) \|X\| s(t) + u_{fu}(t) + u_{su}(t) - h(x) - g(x)a_r \tag{5.38}$$

$$u_{fu}(t) = \hat{h}(x) - \hat{\theta}_h^T Y_h \operatorname{sgn}(s(t)) + \hat{g}(x) a_r - \hat{\theta}_g^T Y_g |a_r| \operatorname{sgn}(s(t)) \tag{5.39}$$

$$u_{fu}(t) = h(x) - \tilde{h}(x) - \hat{\theta}_h^T Y_h \operatorname{sgn}(s(t)) + (g(x) - \tilde{g}(x)) a_r - \hat{\theta}_g^T Y_g |a_r| \operatorname{sgn}(s(t)) \tag{5.40}$$

$$\begin{aligned} u_{fu}(t) = h(x) &- \tilde{A}_h^T (\hat{L}_h - L'_{h\xi} \hat{\xi}_h - L'_{h\sigma} \hat{\sigma}_h) - \hat{A}_h^T (L'_{h\xi} \tilde{\xi}_h + L'_{h\sigma} \tilde{\sigma}_h) - \tilde{B}_h^T (\hat{G}_h - \\ & G'_{h\xi} \hat{\xi}_h - G'_{h\sigma} \hat{\sigma}_h) - \hat{B}_h^T (G'_{h\xi} \tilde{\xi}_h + G'_{h\sigma} \tilde{\sigma}_h) - d_h - \hat{\theta}_h^T Y_h \operatorname{sgn}(s(t)) + [g(x) - \tilde{A}_g^T (\hat{L}_g - \\ & L'_{g\xi} \hat{\xi}_g - L'_{g\sigma} \hat{\sigma}_g) - \hat{A}_g^T (L'_{g\xi} \tilde{\xi}_g + L'_{g\sigma} \tilde{\sigma}_g) - \tilde{B}_g^T (\hat{G}_g - G'_{g\xi} \hat{\xi}_g - G'_{g\sigma} \hat{\sigma}_g) - \hat{B}_g^T (G'_{g\xi} \tilde{\xi}_g + \\ & G'_{g\sigma} \tilde{\sigma}_g) - d_g] a_r - \hat{\theta}_h^T Y_h \operatorname{sgn}(s(t)) \end{aligned} \tag{5.41}$$

From (5.35), (5.38) and (5.41)

$$\begin{aligned} \dot{V}(t) = -k_d s^2(t) + \tfrac{1}{2} \left(\dot{g}(x) - M_2(X) \|X\| \right) s^2(t) - s(t) d_h - |s(t)| \overset{*T}{\theta}_h Y_h + u_{su}(t) s(t) \\ - s(t) d_g a_r - |s(t)| \overset{*T}{\theta}_g Y_g |a_r| < -k_d s^2(t) \end{aligned}$$

where the facts $|d_h| < \overset{*T}{\theta}_h Y_h$ and $|d_g| < \overset{*T}{\theta}_g Y_g$, $s(t) \operatorname{sgn}(s) = |s(t)|$ have been used.

5.4 Simulation

The effectiveness of the proposed approach is shown by applying the developed adaptive fuzzy controller to control the unstable system used in [22, 23], . The system is:

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

The adaptive fuzzy controller is used to drive the system state $x(t)$ to the origin. First, we define seven membership functions over the state space which is chosen to

be [-3, 3]. The simulation is carried out with Sugeno first order fuzzy rules. The initial values $\hat{A}$, $\hat{B}$, $\hat{\xi}$, $\hat{\sigma}$ and $\hat{\theta}_f$ are selected to be:

$$\hat{A}= \begin{bmatrix} -2.04 & -4.08 & -1.02 & 0 & 0.36 & 1.53 & 2.04 \end{bmatrix}^T$$

$$\hat{B}= \begin{bmatrix} -0.8 & -0.6 & -0.4 & 0 & 0.4 & 0.6 & 0.8 \end{bmatrix}^T$$

$$\hat{\xi}= \begin{bmatrix} -3 & -2 & -1 & 0 & 1 & 2 & 3 \end{bmatrix}^T$$

$$\hat{\sigma}= \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \end{bmatrix}^T$$

$$\hat{\theta}_f= \begin{bmatrix} 4 & 1 & 1 & 1 & 1 \end{bmatrix}^T$$

We chose the initial state $x(0) = 2$. In Equation (5.8) $k_d = 10$. Figures 5.1 and 5.2 show $x(t)$ and $u(t)$. We can observe an improvement in the tracking performance compared with the results in [23], for the same number of rules using the same initial conditions. Also, we have a superior transient performance compared with [22].

Figures 5.3 and 5.4 show $x(t)$ and $u(t)$ using five rules instead of seven. The initial values are:

$$\hat{A}= \begin{bmatrix} -4.32 & -1.08 & 0 & 0.37 & 1.62 \end{bmatrix}^T$$

$$\hat{B}= \begin{bmatrix} -0.6 & -0.4 & 0 & 0.4 & 0.6 \end{bmatrix}^T$$

$$\hat{\xi}= \begin{bmatrix} -2 & -1 & 0 & 0.5 & 1 \end{bmatrix}^T$$

$$\hat{\sigma}= \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \end{bmatrix}^T$$

$$\hat{\theta}_f= \begin{bmatrix} 4 & 1 & 1 & 1 & 1 \end{bmatrix}^T$$

Also, we can observe an improvement on the system performance comparing it with that in [23] for different initial conditions with less rules. This improvement was

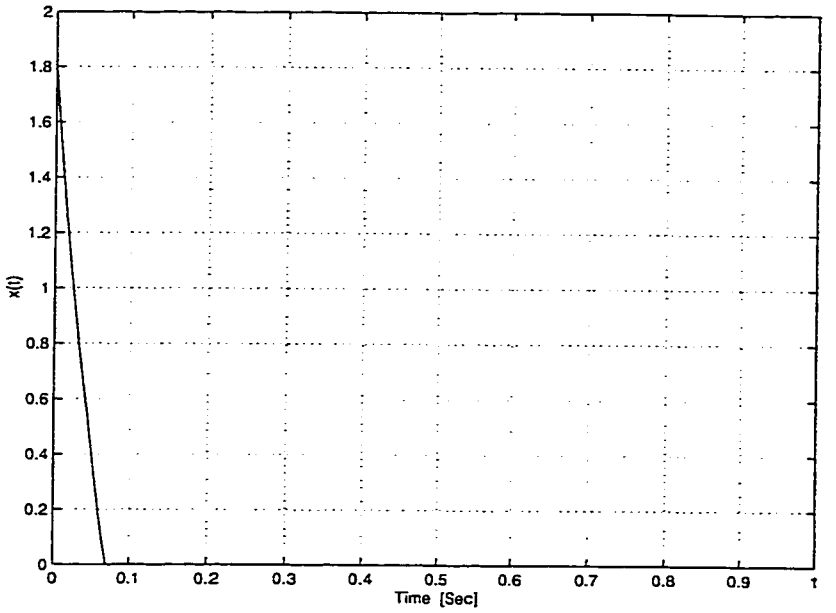

Figure 5.1: Closed-loop $x(t)$ using the developed controller with 7 rules

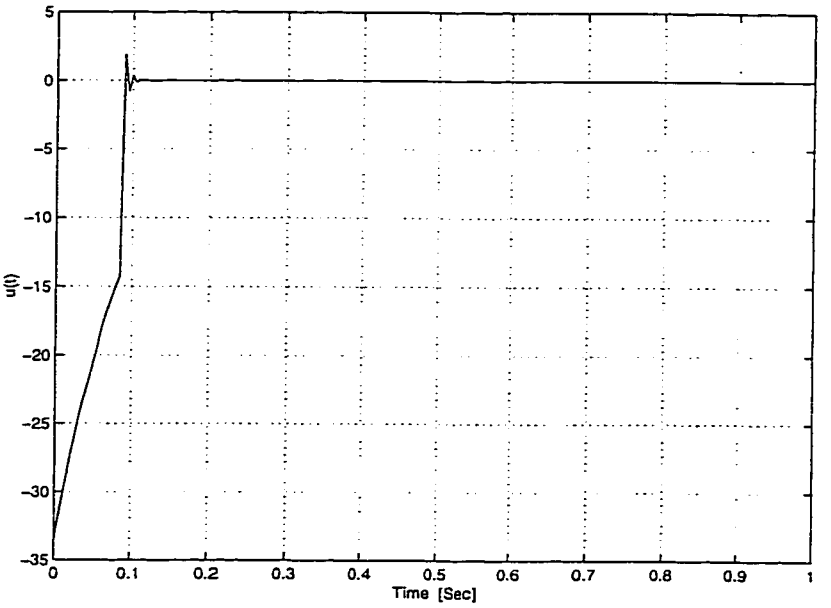

Figure 5.2: Control signal $u(t)$ using the developed controller with 7 rules

expected since a Sugeno first-order consequent model has a better approximation capability than a Sugeno constant consequent model that is used in [23]. Figures 5.5 and 5.6 show the initial and the final membership functions for the system of five rules.

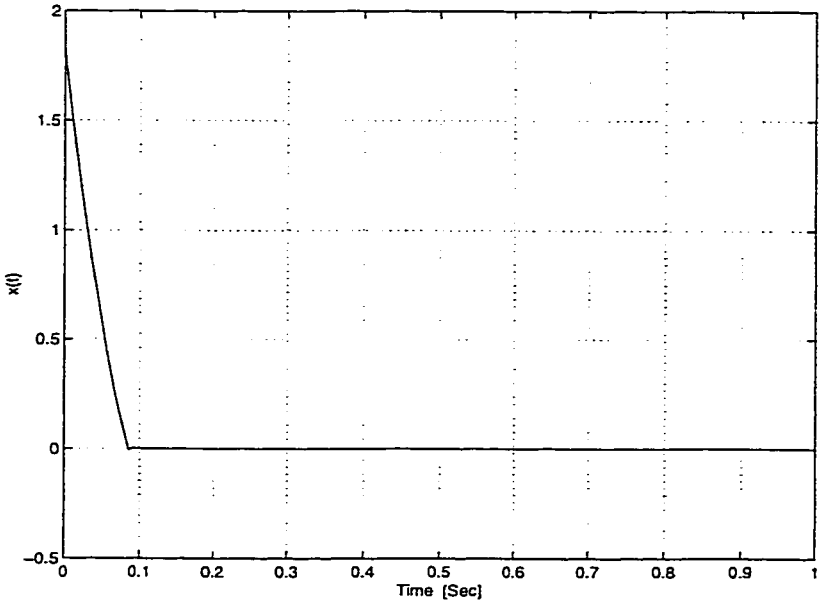

Figure 5.3: Closed-loop $x(t)$ using the developed controller with 5 rules

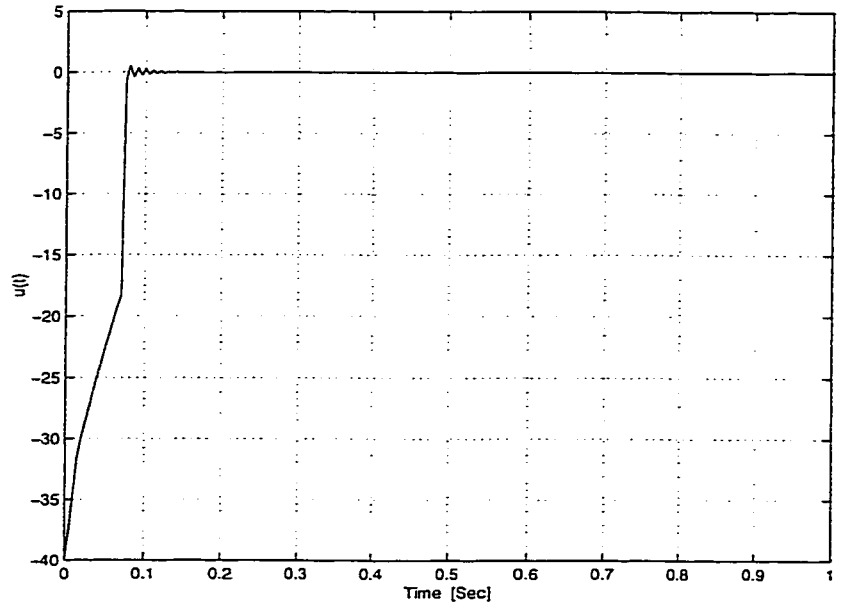

Figure 5.4: Control signal $u(t)$ using the developed controller with 5 rules

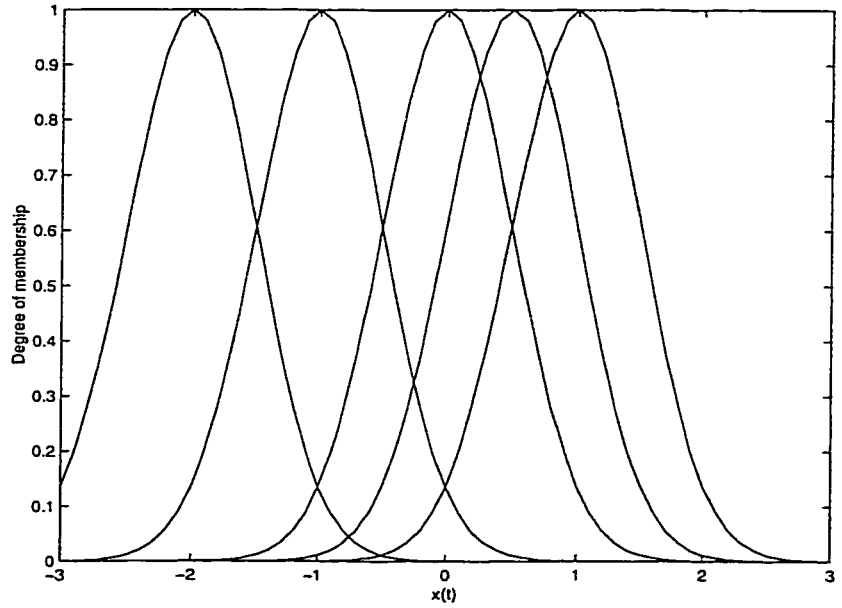

Figure 5.5: Initial membership functions

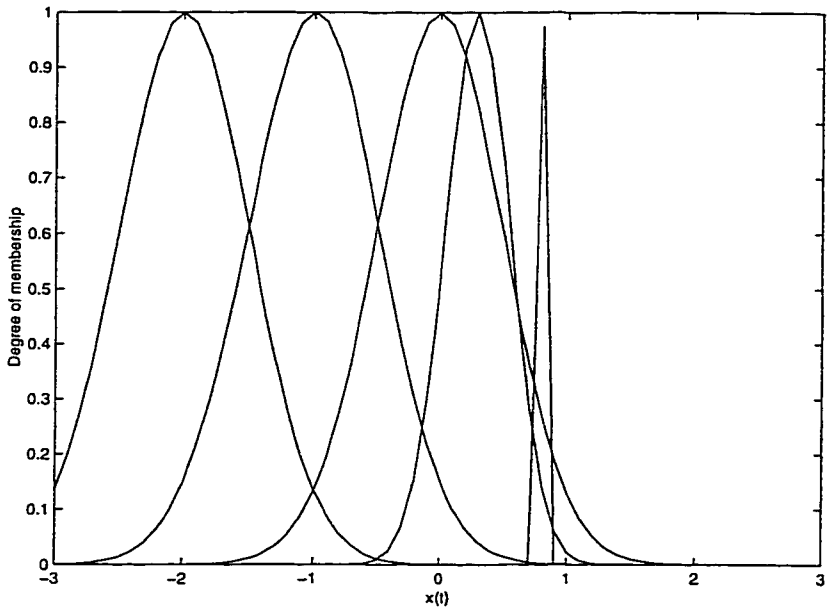

Figure 5.6: Final membership functions

Chapter 6

Experiments

In this chapter, the LQR-Neuro-Fuzzy controller will be implemented on the unstable inverted pendulum. The experiment is carried out in the Fuzzy System Research Laboratory at Concordia university. Also, an experimental comparison between the proposed approach in Chapter 5 and that proposed in [23] is presented. This comparison shows that the performance of a Sugeno fuzzy approximator can be improved by using a first order consequent instead of the constant consequent used in [23].

6.1 LQR-Neuro-Fuzzy controller of the unstable inverted pendulum system

The inverted pendulum system consists of a cart and a rod mounted on the cart. The cart slides on a ground stainless steel shaft. The cart is equipped with a motor

and a potentiometer. These are coupled to a rack and pinion mechanism to input the driving force to the system and to measure cart position, respectively. The motor shaft is connected to a 0.5 inch diameter gear while the potentiometer shaft is connected to a 1.16 inch diameter gear. Both these gears mesh with the toothed rack. When the motor turns, the torque created at the output shaft is translated to a linear force which results in the cart's motion. When the cart moves, the potentiometer shaft turns and the voltage measured from the potentiometer can be calibrated to obtain the track position. The rod's axis of rotation is perpendicular to the direction of motion of the cart. A potentiometer mounted on the axis of rotation allows us to measure the angle of the rod with the vertical axis. A picture of the inverted pendulum setup is shown in Figure 6.1. The state equations for this system are:

$$(m_p + M)\,\ddot{x} + m_p\,\ddot{\theta}\,l_p\cos\theta - m_p\,\dot{\theta}^2\,l_p\sin\theta = F \tag{6.1}$$

$$m_p l_p\cos\theta - m_p l_p\sin\theta\,\dot{x}\dot{\theta} + m_p\,\ddot{\theta}\,l_p^2 - m_p g l_p\sin\theta = 0 \tag{6.2}$$

where

M (Mass of the cart)=0.455 kg

m_p (Mass of the pendulum)=0.210 kg

l_p (Length of the pendulum's center of mass)=0.305 m

F (Impulse force applied to cart) in N

θ (Angle from upright position) in degrees

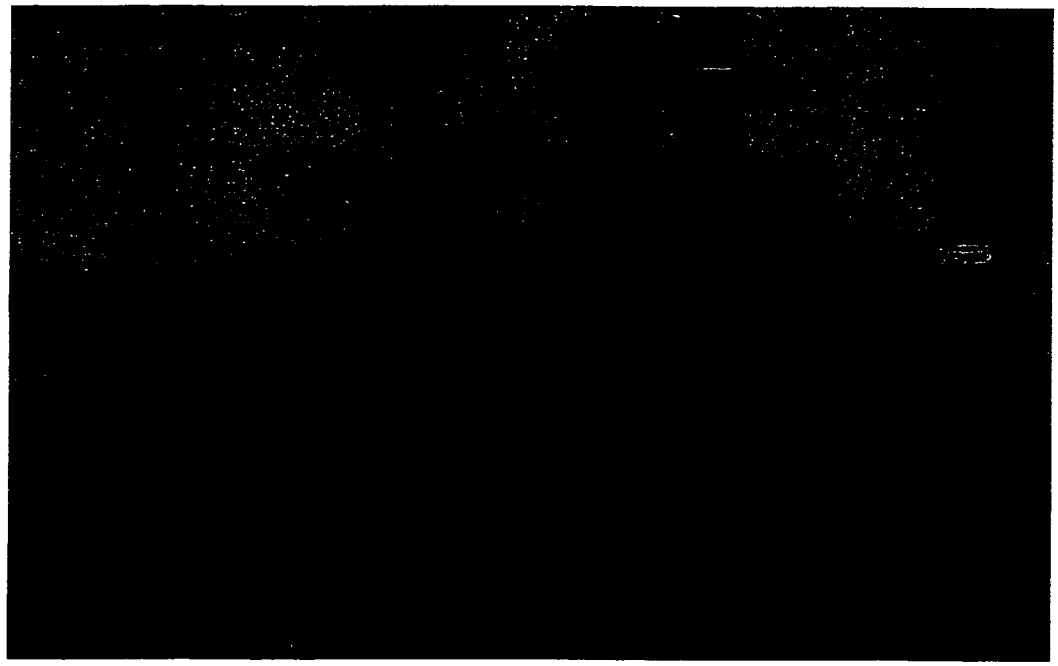

Figure 6.1: A picture of the inverted pendulum setup in the Fuzzy System Research Laboratory at Concordia university

x (Position of the cart) in meters

g (Acceleration due to gravity)=9.81 m^2/s

The above information has been arranged in the proper matrix format to be used in LQR :

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_3} \\ \dot{x_2} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_3} & \frac{\partial f}{\partial x_2} & \frac{\partial f}{\partial x_4} \\ \frac{\partial g}{\partial x_1} & \frac{\partial g}{\partial x_3} & \frac{\partial g}{\partial x_2} & \frac{\partial g}{\partial x_4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{n1} \\ \frac{1}{n2} \end{bmatrix} F \qquad (6.3)$$

where

$x = f(x_1, x_2, x_3, x_4, F)$

$x_1 = x$

$x_2 = \dot{x}$

$x_3 = \theta$

$x_4 = \dot{\theta}$

$n_1 = (M + m_p) - m_p(\cos x_3)^2 + eps$

$n_2 = -(M + m_p)l_p + m_p l_p \cos x_3 + eps$

$\frac{\partial n_1}{\partial x_3} = 2m_p \cos x_3 \sin x_3$

$\frac{\partial n_2}{\partial x_3} = -m_p l_p \sin x_3$

at equilibrium points $\dot{\theta}$=$\dot{x}$= 0;

$\frac{\partial f}{\partial x_1} = 0$

$\frac{\partial f}{\partial x_2} = 0$

$\frac{\partial f}{\partial x_3} = \frac{1}{n_1^2}[-n_1 m_p g((\cos x_3)^2 - (\sin x_3)^2) + \frac{\partial n_1}{\partial x_3} m_p g \cos x_3 \sin x_3 + \frac{\partial n_1}{\partial x_3} F]$ $\frac{\partial f}{\partial x_4} = 0$

$\theta = g(x_1, x_2, x_3, x_4, F)$

$\frac{\partial g}{\partial x_1} = 0$

$\frac{\partial g}{\partial x_2} = 0$

$\frac{\partial g}{\partial x_3} = \frac{1}{n_2^2}[-g(M + m_p)n_2 + \frac{\partial n_2}{\partial x_3} F + (M + m_p)g(\frac{\sin x_3}{\cos x_3})\frac{\partial n_2}{\partial x_3} - (M + m_p)g\frac{(\sin x_3)^2}{(\cos x_3)^2}]$

$\frac{\partial g}{\partial x_4} = 0$

To convert to voltage input, the following relationship is derived in [100]:

$$F = \frac{K_m K_g}{R_m r} V_{in} - \frac{K_m^2 K_g^2}{R_m r^2} \dot{x}$$

where

K_m (Motor torque constant)=0.00767 Nm/amp

K_g (Internal gear ratio)=3.7:1

R_m (Armature resistance)=2.6 ()

r (Motor gear radius)=0.635 cm

V_{in} (Input voltage) in volts.

Substituting this into the matrix equation we have:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_3} \\ \dot{x_2} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_3} & \frac{-K_m^2 K_g^2}{R_m r^2 n_1} & \frac{\partial f}{\partial x_4} \\ \frac{\partial g}{\partial x_1} & \frac{\partial g}{\partial x_3} & \frac{-K_m^2 K_g^2}{R_m r^2 n_2} & \frac{\partial g}{\partial x_4} \end{bmatrix} \begin{bmatrix} x_1 \\ x_3 \\ x_2 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{K_m K_g}{R_m r n_1} \\ \frac{K_m K_g}{R_m r n_2} \end{bmatrix} V_{in} \qquad (6.4)$$

which is the desired representation. Using MATLAB (LQR design) we obtain the optimal feedback gains for the feedback law:

$V_{in} = -(k_1(x_1 - x_{1d}) + k_2 x_3 + k_3 x_2 + k_4 x_4)$

such that the closed loop system:

$A_c = A - BK$

minimizes the quadratic performance index:

$J = \int(\dot{X}\ Q X + R V_{in}^2)dt$

with

$Q = \begin{bmatrix} 0.25 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$ and $R = 0.0003$.

These linearized equations have been used to design LQR at 115 operating points as shown in Table A.9-A.12 (in appendix A). Then, input (θ) has been tabulated with outputs (gains). Subtractive clustering and ANFIS have been applied

R_i	σ^i	c^i	a_{i1}	a_{i0}
R_1	0.06325	-7.315e-13	-5.086e-08	-28.87
R_2	0.05579	-0.1735	0.7406	-28.8
R_3	0.05579	0.1735	-0.7406	-28.8
R_4	0.05665	-0.4397	440.1	214.4
R_5	0.05665	0.4397	-440.1	214.4
R_6	0.06092	0.5392	-281.7	193.1
R_7	0.06092	-0.5392	281.7	193.1
R_8	0.03051	0.3584	-456.6	181.9
R_9	0.03051	-0.3584	456.6	181.9
R_{10}	0.01914	0.2775	-215.4	23.84
R_{11}	0.01914	-0.2775	215.4	23.84

Table 6.1: 11 rules for k_1 fuzzy subcontroller

R_i	σ^i	c^i	a_{i1}	a_{i0}
R_1	0.06497	-8.071e-08	-0.1573	-106.6
R_2	0.06113	-0.4681	4.911	-79.43
R_3	0.06113	0.4681	-4.606	-79.59
R_4	0.06147	-0.1554	-27.12	-108.2
R_5	0.06147	0.1554	26.81	-108.1
R_6	0.05209	0.366	-19.93	-71.79
R_7	0.05209	-0.366	21.19	-71.33
R_8	0.01487	0.2781	1409	-441.8
R_9	0.01492	-0.278	-1425	-447.3
R_{10}	0.01484	-0.292	-2548	-886.8
R_{11}	0.01492	0.292	2580	-896.7

Table 6.2: 11 rules for k_2 fuzzy subcontroller

on these data. As a result we obtained four different fuzzy inference systems for the input (θ) and the four outputs (gains) (see Tables 6.1-6.4). These fuzzy systems are used to stabilize and control the inverted pendulum.

In Figures 6.2 and 6.3, a disturbance force is applied after about 3 seconds in each trial. We can observe that the controller is able to drive the system states back to zero. Note that the two traces in Figures 6.2 and 6.3 are not obtained at the

R_i	σ^i	c^i	a_{i1}	a_{i0}
R_1	0.0403	-7.403e-14	-3.818e-06	-269.3
R_2	0.04021	0.08011	1619	-385.3
R_3	0.04021	-0.08011	-1619	-385.3
R_4	0.03935	0.1404	-8652	902.8
R_5	0.03935	-0.1404	8652	902.8
R_6	0.03899	0.4795	-4217	2231
R_7	0.03899	-0.4795	4217	2231
R_8	0.03829	0.4293	-8525	3592
R_9	0.03829	-0.4293	8525	3592
R_{10}	0.03655	-0.1897	9399	2040
R_{11}	0.03655	0.1897	-9399	2040
R_{12}	0.04018	0.53	-318.5	216
R_{13}	0.04018	-0.53	318.5	216
R_{14}	0.04027	-0.04001	29.62	241.7
R_{15}	0.04027	0.04001	-29.62	241.7
R_{16}	0.02557	0.3912	-2108	661.5
R_{17}	0.02557	-0.3912	2108	661.5
R_{18}	0.02272	0.2277	812.1	-90.64
R_{19}	0.02272	-0.2277	-812.1	-90.64
R_{20}	0.01543	0.3489	-1378	467.4
R_{21}	0.01543	-0.3489	1378	467.4
R_{22}	0.01165	-0.2614	-2737	-762.8
R_{23}	0.01165	0.2614	2737	-762.8

Table 6.3: 23 rules for k_3 fuzzy subcontroller

same time and are not from the same trial, this is due to hardware limitations of the inverted pendulum setup (please see [100] for more details). Figure 6.4 shows that the fuzzy controller is able to stabilize the pendulum. In this trial the pendulum is tapped to about 16 degrees.

R_i	σ^i	c^i	a_{i1}	a_{i0}
R_1	0.0611	-1.02e-13	1.326e-07	-11.66
R_2	0.06377	0.4622	-38.25	20.43
R_3	0.06377	-0.4622	38.25	20.43
R_4	0.05382	-0.1689	-9.959	-14.01
R_5	0.05382	0.1689	9.959	-14.01
R_6	0.01266	0.3349	-1413	499.5
R_7	0.01266	-0.3349	1413	499.5
R_8	0.04269	0.2504	-84.55	0.3567
R_9	0.04269	-0.2504	84.55	0.3567

Table 6.4: 9 rules for k_4 fuzzy subcontroller

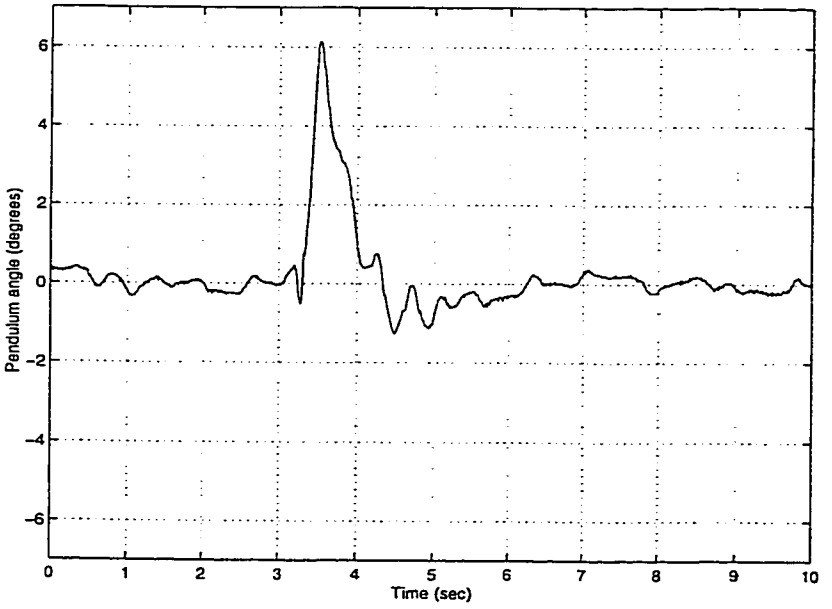

Figure 6.2: Disturbance response of pendulum angle

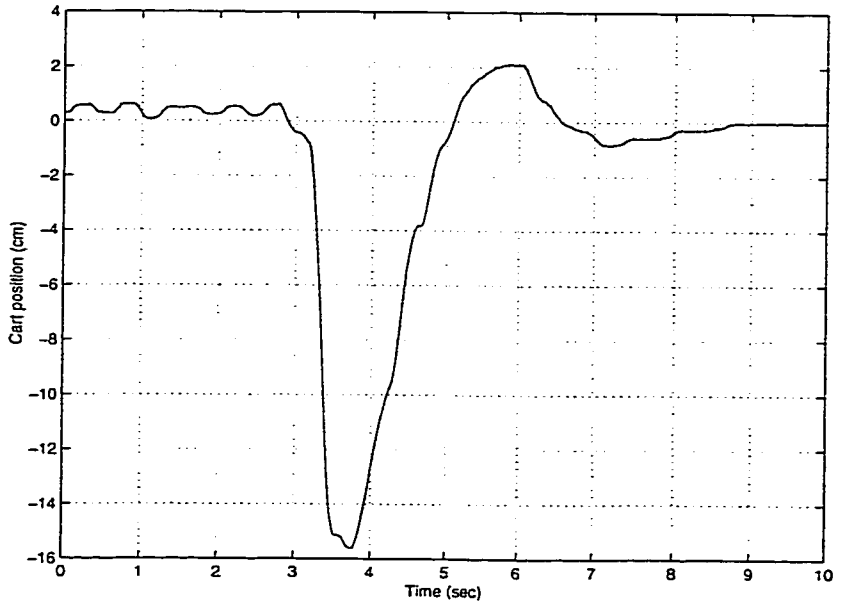

Figure 6.3: Disturbance response of cart position

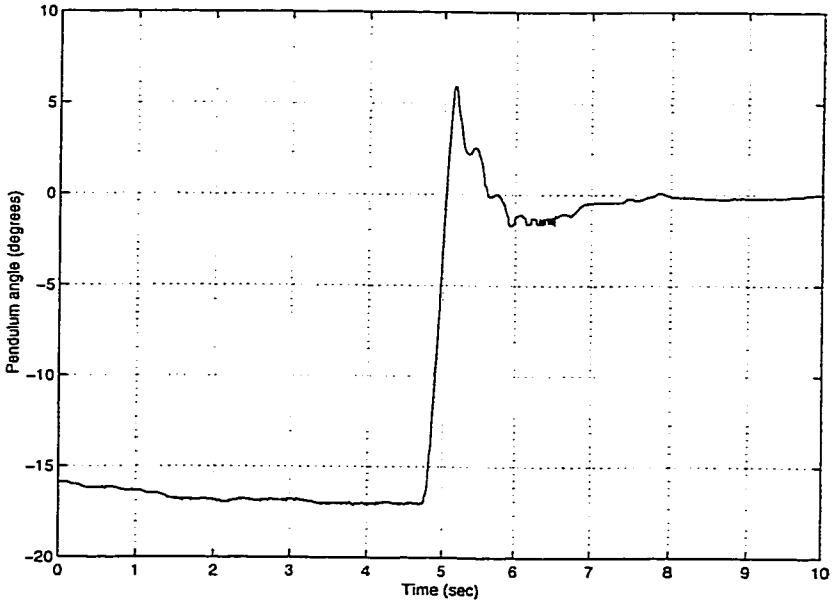

Figure 6.4: Disturbance response of pendulum angle

6.2 Adaptive fuzzy control of the unstable inverted pendulum system

The effectiveness of the proposed approach in Chapter 5 is shown by applying the developed adaptive fuzzy controller to control the unstable inverted pendulum system. Also, the approach presented in [23] is applied on the inverted pendulum system. The objective of this implementation is to compare it with that of the proposed approach in Chapter 5. This comparison is expected to demonstrate the improvement in approximation ability of a first order Sugeno approximator when compared with constant consequent Sugeno approximator.

The adaptive fuzzy controller is used to drive the system states $x(t)$ (cart position) and $\theta(t)$ (pendulum angle) to the origin. First, we define seven initial membership functions over the state space of θ which is chosen to be [-30, 30]. The initial values of the Sugeno fuzzy consequent ($\hat{A}$ and $\hat{B}$), the parameters of the Gaussian membership functions ($\hat{\xi}$, $\hat{\sigma}$ and $\hat{\theta}_f$) are selected to be

$$\hat{A}= \begin{bmatrix} -0.6 & -0.4 & -0.2 & 0 & 0.2 & 0.4 & 0.6 \end{bmatrix}^T,$$

$$\hat{B}= \begin{bmatrix} -0.6 & -0.4 & -0.2 & 0 & 0.2 & 0.4 & 0.6 \end{bmatrix}^T,$$

$$\hat{\xi}= \begin{bmatrix} -30 & -20 & -10 & 0 & 10 & 20 & 30 \end{bmatrix}^T,$$

$$\hat{\sigma}= \begin{bmatrix} 5 & 5 & 5 & 5 & 5 & 5 & 5 \end{bmatrix}^T,$$

$$\hat{\theta}_f= \begin{bmatrix} 4 & 1 & 1 & 1 & 1 \end{bmatrix}^T.$$

In Equation (5.18) $k_d = 10$. A disturbance force is applied on the pendulum angle

after about 7 seconds. Figure 6.5 shows the disturbance response of pendulum angle using the controller in Chapter 5. We can observe that the controller is able to drive the system states back to zero. The same trial is repeated using the approach presented in [23]. The initial values of the Sugeno constant consequent ($\hat{B}$) are chosen to be the same as those shown above. Also, the same initial values for the parameters of the Gaussian membership functions ($\hat{\xi}$, $\hat{\sigma}$ and $\hat{\theta}_f$) are selected. Figure 6.6 shows the disturbance response of pendulum angle using the controller in [23]. We can observe an overshoot when the controller is strated and it took a longer time for the system to settle down after a disturbance force which is applied at about 7 seconds.

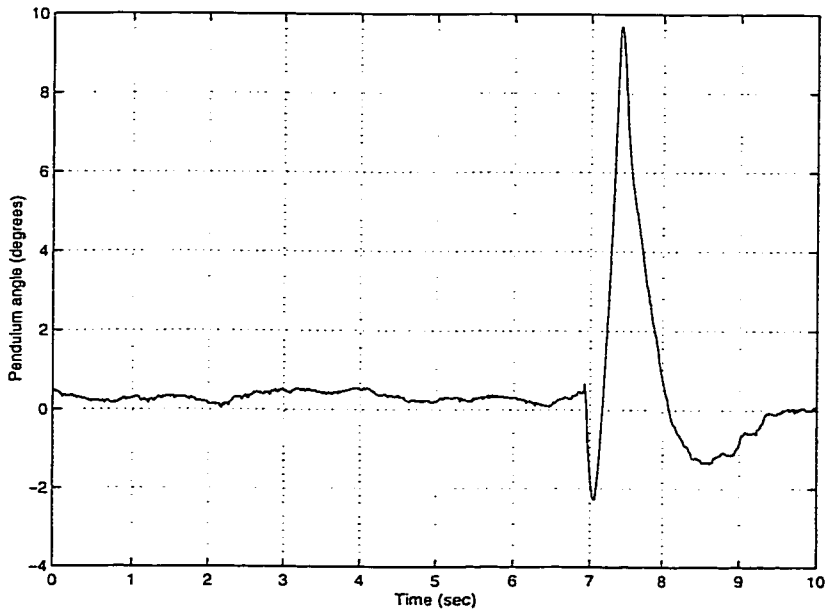

Figure 6.5: Disturbance response of pendulum angle using the proposed controller in Chapter 5

Another trial is presented in Figures 6.7 where the controller in Chapter 5 is

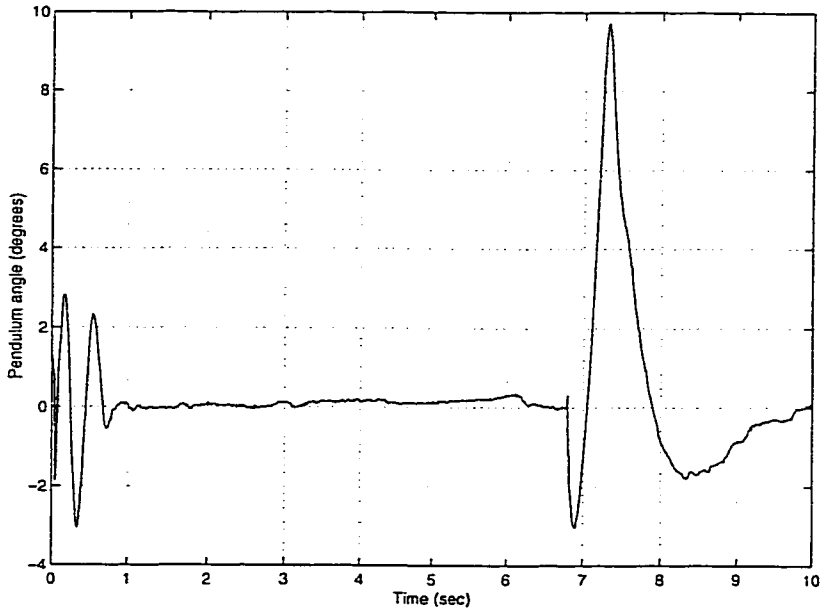

Figure 6.6: Disturbance response of pendulum angle using the controller in [23]

applied and in Figure 6.8 where the controller in [23] is applied. In each trial, the cart is placed at 5 cm distance from the origin and the pendulum is held in upwright position. This trial shows the effect of the initial cart position on the response of the pendulum angle. We observe that the overshoot in Figure 6.8 is three times that in Figure 6.7. Also, the settling time is longer in Figure 6.8 than the one observed in Figure 6.7.

Next, the pendulum is tapped to about 15° and the adaptive controller in Chapter 5 is applied to stabilize the pendulum. Figure 6.9 shows that the controller is able to drive the angle back to zero in less than 1 second. The same trial is repeated using the controller in [23], where the controller is able to stabilize the pendulum in about 3 seconds as shown in Figure 6.10.

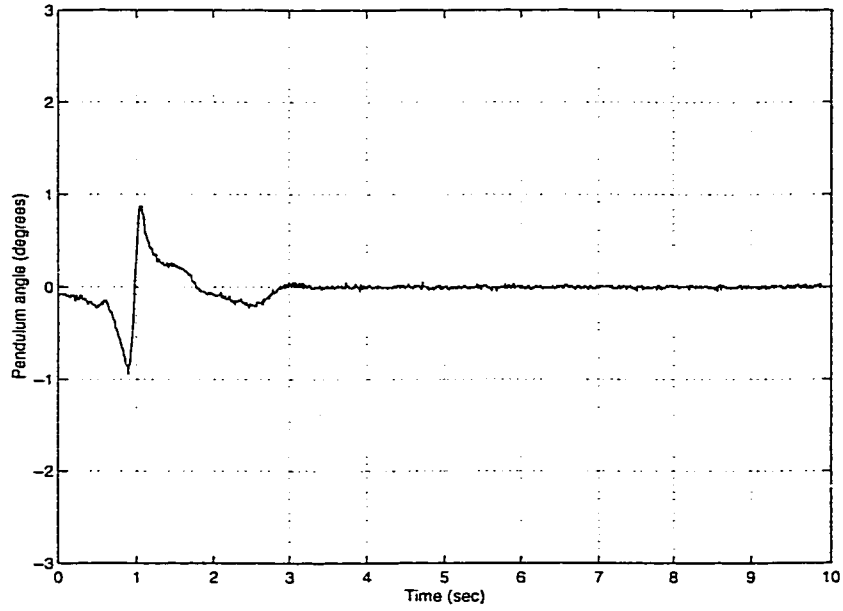

Figure 6.7: Response of pendulum angle for $x = 5$ cm using the proposed controller in Chapter 5

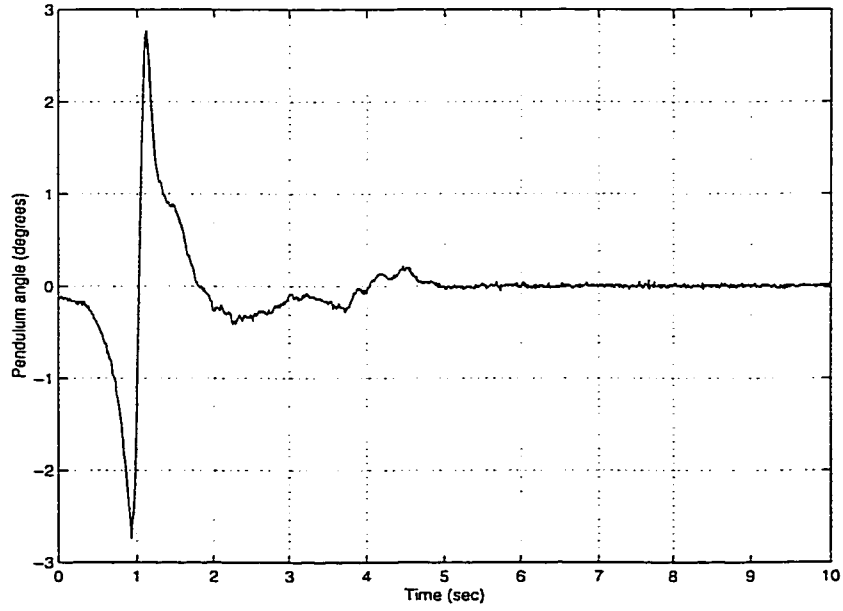

Figure 6.8: Response of pendulum angle for $x = 5$ cm using the controller in [23]

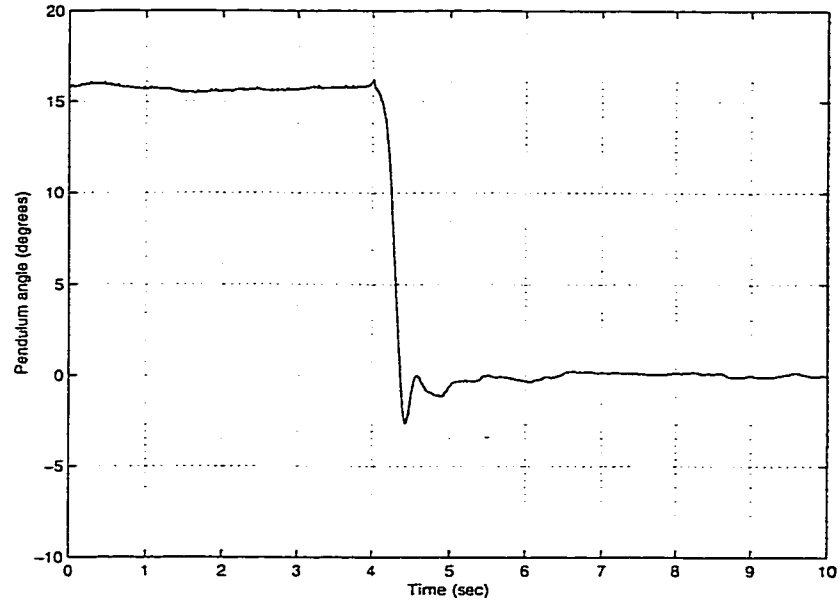

Figure 6.9: Response of pendulum angle for $\theta_0 = 15^o$ using the proposed controller in Chapter 5

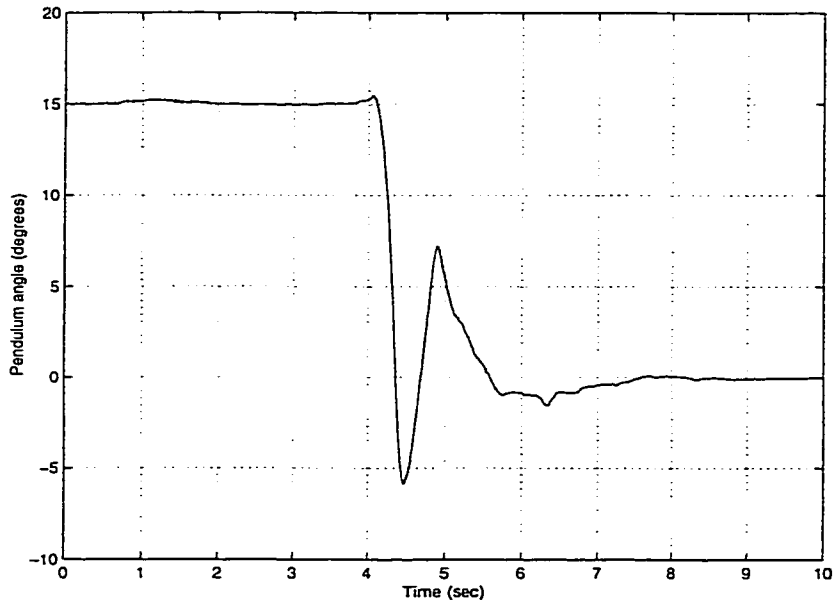

Figure 6.10: Response of pendulum angle for $\theta_0 = 15^o$ using the controller in [23]

Finally. the pendulum is tapped to about 7^{o} and the cart position is being recorded. This trial shows the effect of the initial pendulum angle on the response of the cart position. Figure 6.11 shows the response of the cart position when the controller in Chapter 5 is used, which has a better settling time and less overshoot when compared with that in Figure 6.12 (which shows the response of the cart position when the controller in [23] is used). As expected in Chapter 5, using Sugeno system with a first order consequent instead of a constant consequent enhanced the controller performace , as oserved in all trials presented above, which can be attributed to a better approximation capability of the first order Sugeno systems.

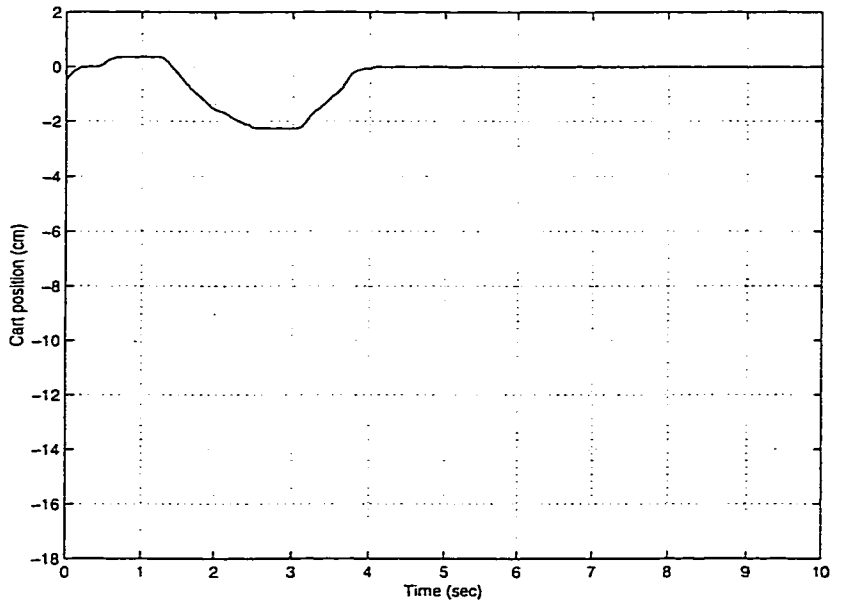

Figure 6.11: Response of cart position when pendulum angle is tapped by 7^{o} using the proposed controller in Chapter 5

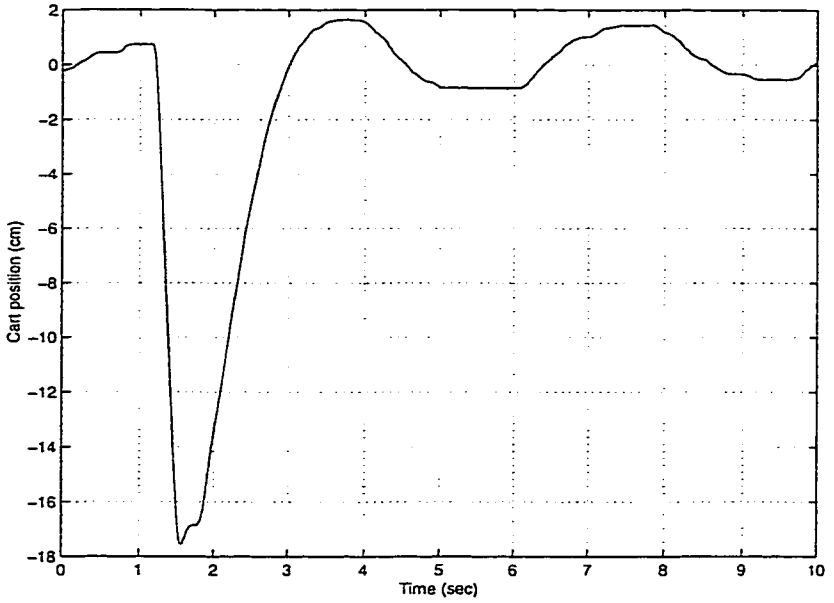

Figure 6.12: Response of cart position when pendulum angle is tapped by 7° using the controller in [23]

Chapter 7

Discussions and conclusions

In this thesis, modelling and control of certain classes of dynamic systems is studied. First, modelling and control approach of dynamic systems that can be well defined and modelled using mathematical equations, is developed. The proposed approach for these systems is a fuzzy-logic gain scheduling-like approach. Also, modelling of some known simple and hard nonlinearities is discussed. Finally, control of systems with unknown dynamics or are difficult to transform into closed form mathematical model (equations of motion), are considered in this study. The proposed approach for these systems is an adaptive fuzzy approach using parameterized Sugeno fuzzy approximators. Sugeno fuzzy model is used for modelling and control of nonlinear systems in all applications presented in this thesis. Details of these contributions are explained below.

The interpolation properties of Sugeno fuzzy system are discussed in Chapter

3. In this chapter the influence of the shape and the distribution of the membership functions and the order of the functional consequent on the interpolation behavior of the Sugeno fuzzy systems is investigated. In general, it can be shown by induction that for a controller with any number of inputs, product conjunction operator, constant consequent and 0.5 complementary triangular membership functions, the output is a linear function with respect to each input. Increasing the order of Sugeno consequent by one will increase the interpolation order by one. For a controller with any number of inputs, product conjunction operator, constant consequent and non-complementary triangular membership functions, the output is a rational function, which is a linear function with respect to each input in the numerator as well as in the denominator. Increasing the order of Sugeno consequent will increase the order of the function in the numerator by one with respect to each input, while the denominator will remain a linear function with respect to each input. Taking the interaction terms into consideration in the rules consequent is a more general case than that of Sugeno. Ignoring the interaction terms is usually the case, which will reduce the number of modifiable parameters in a Sugeno system. In some processes the interaction between variables is more pronounced and the interaction terms should be considered.

Gain scheduling is an effective way of controlling systems whose dynamics change with the operating conditions. It is normally used in the control of nonlinear plants where the relationship between the plant dynamics and operating conditions

is known, and for which a single linear time invariant model is insufficient. Three main issues are involved in gain scheduling namely partitioning of the state space into several linear operating regions, linearizing of the plant and designing of the linear controller in each of the linear regions and interpolation of control action between the linear regions. The restriction to slow variations in the scheduling variables is crucial in the gain scheduling practice. If the time variations in the operating conditions are ignored, instability may result.

In Chapter 4, fuzzy-logic gain scheduling-like controller synthesis technique is developed for MISO nonlinear systems. The resulting controller does not require on line adaptation, estimation, or prediction. Fuzzy logic is used to smoothly schedule independently designed point controllers over the operational and parameter spaces of the system's model. These point controllers are synthesized using techniques chosen by the designer, thus allowing unprecedented amount of design freedom. The proposed approach utilizes the advantages of gain scheduling, fuzzy logic control, neural networks, and optimal control theory.

The result of this research is a systematic methodology resulting in a continuous schedule which provides smooth transitions between point controllers. The use of point controllers based upon established linear control theory allows for the handling of plants whose complex dynamics require dynamic compensation. Nonlinear control action is brought in by fuzzy logic schedule. The proposed approach is based on using fuzzy logic to blend the individual point designs such that for any

trajectory in the parameter space, the system performs adequately.

This approach starts with linearizing the plant dynamics about different equilibrium points. This results in a linear model of the system. Using one of the optimal control algorithms such as LQR, a set of gains are obtained for each equilibrium point. Subtractive clustering and ANFIS are used to construct the rules for the fuzzy gain schedule. The fuzzy gain schedule is used to continuously provide the controller with gains as a function of the states of the system. The drawbacks, mentioned before, for conventional gain scheduling are where to place the point controllers in the operational state space, how to switch between controllers and when to switch. In the proposed approach, the drawbacks of switching are now avoided by the smooth transition a cross a bank of parallel controllers, all now under continuous full operation as determined by the fed back plant's state trajectory. Furthermore, conventional gain schedule cannot work efficiently when controlling a fast varying system, which is a major drawback. The slow variation restriction is not imposed during the development of the fuzzy gain schedule. The fuzzy gain schedule provides a smooth gains without any discontinuities. Therefore, there will be no spikes in the response due to jumping from a set of gains to another.

The effectiveness of the proposed approach is demonstrated by the simulation results of a two dof robot manipulator. A robot manipulator is not stable and displays a strong nonlinearity. The resulting system is observed to be stable for different reference trajectories. The system is also robust for different driving frequencies of

the input. We believe that this approach has great potential in various applications to control different nonlinear systems. The proposed controller is implemented on the unstable inverted pendulum system, as shown in Chapter 6.

For further development on fuzzy gain scheduling approach, the question of system stability should be addressed. In this research stability is addressed by extensive simulation over the operational envelop. One could attempt a general stability analysis of either the general fuzzy controller structure, or a stability proof of a specific controller after the design has been made. An important improvement towards a better fuzzy gain scheduled control is to achieve the gain scheduling in closed loop form, which means providing corrective action to inappropriate schedule on-line.

For nonlinear systems with poorly understood dynamics, various adaptive control schemes have been proposed. Such systems are approximated by parameterized fuzzy approximators which are expressed as a series RBF expansion. Tuning or changing the parameters of the RBF will change the shape of the RBF. This property could be employed to capture fast varying system dynamics, reduce approximation error and improve control performance. In Chapter 5, the controller is designed based on the well known Sugeno first-order fuzzy system. The consequent part of IF-THEN rules is a linear combination of input variables and a constant term, and the final output is the weighted average of each rule's output. This introduces additional parameter vectors to be updated, but improves the tracking performance due

to a better approximation ability of a higher order Sugeno model. Global boundedness of the adaptive system is established. Finally, simulation results, in Chapter 5, are used to demonstrate the effectiveness of the proposed controller. The simulation example shows an improvement in the control performance. This improved performance has been achieved with less number of rules compared with that in [23]. A comparison between the implementation of the proposed approach in Chapter 5 and that of the approach in [23] on the unstable inverted pendulum system is shown in Chapter 6. This comparison verifies that a higher order Sugeno model has a better approximation ability compared with a lower order model.

The proposed adaptive fuzzy approach in Chapter 5 can be further improved by developing a general controller which guarantees stability of the system regardless of the structure of the fuzzy approximator. Also, neural networks can be incorporated to enhance the ability of the proposed controller in updating the parameters of the fuzzy approximator.

Appendix A

Input-output data for LQR-fuzzy control

A.1 Input-output data for LQR-fuzzy control of a robot manipulator

The LQR gains are computed for a two link manipulator by linearizing the system at various equilbrium points. The A and B matrix coefficients were calculated as shown in Chapter 4. This Appendix contains the gains computed using LQR for several equilibrium points for each link. θ_1 is the angular position of the first link in radians, θ_2 is the angular position of the second link in degrees. The LQR gains are claculated for each combination between equilibrium points of each link. For example, for $\theta_1 = .175$ rad. and $\theta_2 = -70$ degrees $k_{11} = 1.5875$. Tables A.1-A.4

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2 (deg.)									
-70	1.5875	2.8665	4.2695	5.6418	6.9096	8.0203	8.9271	9.5875	9.9655
-50	1.6992	3.0819	4.5665	5.9976	7.3010	8.4231	9.3182	9.9475	10.2805
-30	1.8901	3.3892	4.9454	6.4138	7.7236	8.8247	9.6769	10.2481	10.5155
-10	2.1462	3.7298	5.3229	6.7977	8.0882	9.1481	9.9425	10.4452	10.6396
10	2.4074	4.0203	5.6150	7.0749	8.3346	9.3486	10.0840	10.5184	10.6396
30	2.6111	4.2121	5.7872	7.2221	8.4481	9.4174	10.0969	10.4659	10.5155
50	2.7374	4.3068	5.8506	7.2532	8.4417	9.3655	9.9905	10.2974	10.2805
70	2.7944	4.3235	5.8270	7.1886	8.3333	9.2091	9.7821	10.0348	9.9655
90	2.7899	4.2736	5.7295	7.0427	8.1390	8.9679	9.4975	9.7136	9.6187

Table A.1: Gain k_{11} computed using LQR for different combinations of θ_1 and θ_2

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2 (deg.)									
-70	0.5684	0.7666	0.9393	1.0848	1.2071	1.3083	1.3888	1.4478	1.4838
-50	0.6102	0.8251	1.0076	1.1588	1.2835	1.3844	1.4623	1.5168	1.5474
-30	0.6662	0.8946	1.0822	1.2342	1.3565	1.4526	1.5241	1.5714	1.5946
-10	0.7255	0.9577	1.1443	1.2931	1.4104	1.5001	1.5642	1.6039	1.6195
10	0.7699	0.9969	1.1791	1.3237	1.4364	1.5206	1.5784	1.6111	1.6195
30	0.7891	1.0073	1.1845	1.3256	1.4346	1.5144	1.5669	1.5933	1.5946
50	0.7868	0.9957	1.1675	1.3047	1.4100	1.4855	1.5328	1.5531	1.5474
70	0.7715	0.9709	1.1361	1.2680	1.3684	1.4389	1.4808	1.4953	1.4838
90	0.7489	0.9387	1.0960	1.2210	1.3153	1.3802	1.4170	1.4273	1.4127

Table A.2: Gain k_{12} computed using LQR for different combinations of θ_1 and θ_2

are the LQR gains ($k_{11} - k_{14}$) for the first control input (the torque on the first arm). Tables A.5-A.8 are the gains ($k_{21} - k_{24}$) corresponding to the second input torque. Table 4.2 is the fuzzy inference system for the first gain corresponding to the first input, columns 2 and 3 represent the rule antecedents while columns 4,5 and 6 represent the coefficients of first order Sugeno consequent.

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2 (deg.)									
-70	-0.0946	-0.0542	0.0092	0.0865	0.1732	0.2663	0.3623	0.4567	0.5444
-50	-0.0490	0.0351	0.1412	0.2560	0.3736	0.4891	0.5977	0.6945	0.7740
-30	0.0155	0.1463	0.2910	0.4348	0.5708	0.6941	0.8002	0.8848	0.9440
-10	0.1088	0.2699	0.4395	0.5963	0.7347	0.8505	0.9407	1.0024	1.0338
10	0.1953	0.3839	0.5598	0.7130	0.8391	0.9355	1.0006	1.0333	1.0338
30	0.2797	0.4675	0.6329	0.7683	0.8707	0.9391	0.9733	0.9743	0.9440
50	0.3402	0.5105	0.6512	0.7570	0.8267	0.8609	0.8614	0.8312	0.7740
70	0.3697	0.5087	0.6127	0.6794	0.7100	0.7074	0.6756	0.6193	0.5444
90	0.3622	0.4588	0.5174	0.5399	0.5303	0.4938	0.4365	0.3648	0.2852

Table A.3: Gain k_{13} computed using LQR for different combinations of θ_1 and θ_2

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2 (deg.)									
-70	0.0375	0.0538	0.0684	0.0813	0.0928	0.1032	0.1124	0.1202	0.1263
-50	0.0535	0.0762	0.0956	0.1122	0.1264	0.1385	0.1485	0.1564	0.1619
-30	0.0701	0.0980	0.1210	0.1397	0.1551	0.1675	0.1771	0.1840	0.1881
-10	0.0848	0.1154	0.1397	0.1590	0.1741	0.1858	0.1942	0.1996	0.2020
10	0.0939	0.1242	0.1480	0.1665	0.1806	0.1909	0.1977	0.2014	0.2020
30	0.0949	0.1229	0.1448	0.1615	0.1739	0.1825	0.1875	0.1893	0.1881
50	0.0885	0.1127	0.1316	0.1457	0.1556	0.1619	0.1649	0.1648	0.1619
70	0.0765	0.0959	0.1106	0.1212	0.1281	0.1317	0.1324	0.1305	0.1263
90	0.0607	0.0744	0.0843	0.0907	0.0942	0.0950	0.0937	0.0905	0.0859

Table A.4: Gain k_{14} computed using LQR for different combinations of θ_1 and θ_2

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2(deg.)									
-70	-0.6981	-1.0483	-1.3804	-1.6376	-1.7863	-1.8073	-1.6943	-1.4548	-1.1090
-50	-0.6068	-0.8908	-1.1519	-1.3441	-1.4405	-1.4277	-1.3048	-1.0817	-0.7775
-30	-0.4256	-0.6009	-0.7649	-0.8871	-0.9476	-0.9365	-0.8526	-0.7018	-0.4963
-10	-0.1319	-0.1763	-0.2464	-0.3224	-0.3860	-0.4243	-0.4297	-0.3995	-0.3348
10	0.2530	0.3195	0.3094	0.2395	0.1315	0.0049	-0.1236	-0.2403	-0.3348
30	0.6489	0.7770	0.7829	0.6834	0.5039	0.2712	0.0112	-0.2520	-0.4963
50	0.9632	1.1012	1.0867	0.9358	0.6769	0.3424	-0.0339	-0.4182	-0.7775
70	1.1359	1.2446	1.1862	0.9768	0.6466	0.2322	-0.2263	-0.6870	-1.1090
90	1.1456	1.2007	1.0882	0.8255	0.4446	-0.0148	-0.5084	-0.9900	-1.4154

Table A.5: Gain k_{21} computed using LQR for different combinations of θ_1 and θ_2

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2(deg.)									
-70	-0.1346	-0.1820	-0.2155	-0.2335	-0.2363	-0.2242	-0.1985	-0.1606	-0.1129
-50	-0.1046	-0.1420	-0.1672	-0.1790	-0.1776	-0.1638	-0.1389	-0.1046	-0.0630
-30	-0.0523	-0.0746	-0.0903	-0.0980	-0.0971	-0.0880	-0.0715	-0.0486	-0.0206
-10	0.0229	0.0158	0.0059	-0.0032	-0.0094	-0.0119	-0.0105	-0.0052	0.0037
10	0.1083	0.1104	0.1006	0.0847	0.0665	0.0481	0.0310	0.0160	0.0037
30	0.1819	0.1863	0.1723	0.1472	0.1157	0.0811	0.0457	0.0113	-0.0206
50	0.2273	0.2292	0.2094	0.1753	0.1322	0.0840	0.0337	-0.0161	-0.0630
70	0.2401	0.2370	0.2111	0.1698	0.1182	0.0607	0.0008	-0.0581	-0.1129
90	0.2241	0.2145	0.1833	0.1370	0.0810	0.0194	-0.0436	-0.1043	-0.1592

Table A.6: Gain k_{22} computed using LQR for different combinations of θ_1 and θ_2

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2(deg.)									
-70	0.7198	0.7356	0.7554	0.7828	0.8204	0.8699	0.9321	1.0067	1.0920
-50	0.7699	0.7971	0.8305	0.8736	0.9276	0.9928	1.0679	1.1504	1.2370
-30	0.8593	0.9050	0.9567	1.0154	1.0805	1.1506	1.2230	1.2943	1.3612
-10	0.9949	1.0627	1.1294	1.1944	1.2562	1.3132	1.3631	1.4036	1.4330
10	1.1704	1.2518	1.3199	1.3741	1.4145	1.4406	1.4522	1.4495	1.4330
30	1.3567	1.4329	1.4834	1.5099	1.5145	1.4994	1.4669	1.4198	1.3612
50	1.5117	1.5623	1.5795	1.5677	1.5314	1.4753	1.4043	1.3232	1.2370
70	1.5990	1.6099	1.5863	1.5350	1.4627	1.3761	1.2816	1.1851	1.0920
90	1.5986	1.5657	1.5038	1.4217	1.3273	1.2281	1.1303	1.0392	0.9585

Table A.7: Gain k_{23} computed using LQR for different combinations of θ_1 and θ_2

θ_1 (rad.)	0.175	0.349	0.524	0.698	0.870	1.050	1.220	1.400	1.570
θ_2(deg.)									
-70	0.0895	0.0876	0.0872	0.0884	0.0912	0.0956	0.1013	0.1083	0.1162
-50	0.0900	0.0889	0.0894	0.0916	0.0954	0.1005	0.1068	0.1139	0.1216
-30	0.0965	0.0972	0.0990	0.1018	0.1056	0.1102	0.1153	0.1209	0.1265
-10	0.1099	0.1127	0.1151	0.1175	0.1199	0.1224	0.1249	0.1273	0.1294
10	0.1282	0.1321	0.1340	0.1347	0.1347	0.1340	0.1329	0.1313	0.1294
30	0.1465	0.1499	0.1504	0.1489	0.1460	0.1420	0.1373	0.1321	0.1265
50	0.1602	0.1621	0.1606	0.1568	0.1514	0.1448	0.1374	0.1296	0.1216
70	0.1669	0.1666	0.1632	0.1576	0.1505	0.1424	0.1336	0.1248	0.1162
90	0.1663	0.1638	0.1587	0.1521	0.1443	0.1360	0.1275	0.1193	0.1118

Table A.8: Gain k_{24} computed using LQR for different combinations of θ_1 and θ_2

α (rad)	k_1 (V/m)	k_2 (V/rad)	k_3 (V/m)	k_4 (V/rad)
-0.5700	28.8675	-82.1423	10.3574	0.1308
-0.5600	28.8675	-82.1447	10.6880	0.2357
-0.5500	28.8675	-82.1361	11.0383	0.3468
-0.5400	28.8675	-82.1166	11.4102	0.4646
-0.5300	28.8675	-82.0859	11.8063	0.5899
-0.5200	28.8675	-82.0438	12.2291	0.7235
-0.5100	28.8675	-81.9903	12.6819	0.8664
-0.5000	28.8675	-81.9249	13.1685	1.0197
-0.4900	28.8675	-81.8475	13.6933	1.1847
-0.4800	28.8675	-81.7575	14.2617	1.3631
-0.4700	28.8675	-81.6546	14.8800	1.5568
-0.4600	28.8675	-81.5381	15.5561	1.7680
-0.4500	28.8675	-81.4073	16.2997	1.9999
-0.4400	28.8675	-81.2614	17.1229	2.2560
-0.4300	28.8675	-81.0993	18.0411	2.5409
-0.4200	28.8675	-80.9198	19.0744	2.8607
-0.4100	28.8675	-80.7213	20.2494	3.2234
-0.4000	28.8675	-80.5016	21.6019	3.6398
-0.3900	28.8675	-80.2580	23.1825	4.1250
-0.3800	28.8675	-79.9872	25.0642	4.7011

Table A.9: Gains computed using LQR for inverted pendulum control problem

A.2 Input-output data for LQR-fuzzy control of the inverted pendulum

The LQR gains are computed for the inverted pendulum by linearizing the system at various equilbrium points. The A and B matrix coefficients were calculated as shown in Chapter 5. Tables A.9-A.12 are the LQR gains ($k_1 - k_4$).

α (rad)	k_1 (V/m)	k_2 (V/rad)	k_3 (V/m)	k_4 (V/rad)
-0.3700	28.8675	-79.6841	27.3584	5.4015
-0.3600	28.8675	-79.3418	30.2443	6.2800
-0.3500	28.8675	-78.9499	34.0348	7.4304
-0.3400	28.8675	-78.4914	39.3387	9.0354
-0.3300	28.8675	-77.9354	47.5609	11.5160
-0.3200	28.8675	-77.2126	63.0695	16.1803
-0.3100	28.8675	-76.0760	115.1212	31.7835
-0.3000	-28.8675	-93.8883	-129.5948	-41.3937
-0.2900	-28.8675	-95.5291	-79.6144	-26.4992
-0.2800	-28.8675	-96.7188	-63.8333	-21.8150
-0.2700	-28.8675	-97.7077	-55.4572	-19.3385
-0.2600	-28.8675	-98.5713	-50.1027	-17.7614
-0.2500	-28.8675	-99.3444	-46.3262	-16.6534
-0.2400	-28.8675	-100.0469	-43.4949	-15.8259
-0.2300	-28.8675	-100.6909	-41.2826	-15.1817
-0.2200	-28.8675	-101.2851	-39.5015	-14.6650
-0.2100	-28.8675	-101.8353	-38.0352	-14.2413
-0.2000	-28.8675	-102.3460	-36.8074	-13.8876
-0.1900	-28.8675	-102.8207	-35.7654	-13.5885
-0.1800	-28.8675	-103.2620	-34.8717	-13.3329
-0.1700	-28.8675	-103.6721	-34.0991	-13.1126
-0.1600	-28.8675	-104.0528	-33.4267	-12.9215
-0.1500	-28.8675	-104.4056	-32.8389	-12.7549
-0.1400	-28.8675	-104.7315	-32.3233	-12.6092
-0.1300	-28.8675	-105.0317	-31.8703	-12.4815
-0.1200	-28.8675	-105.3070	-31.4719	-12.3695
-0.1100	-28.8675	-105.5581	-31.1220	-12.2713
-0.1000	-28.8675	-105.7856	-30.8154	-12.1855
-0.0900	-28.8675	-105.9901	-30.5479	-12.1108
-0.0800	-28.8675	-106.1720	-30.3161	-12.0462
-0.0700	-28.8675	-106.3317	-30.1173	-11.9909
-0.0600	-28.8675	-106.4695	-29.9491	-11.9441

Table A.10: Gains computed using LQR for inverted pendulum control problem

α (rad)	k_1 (V/m)	k_2 (V/rad)	k_3 (V/m)	k_4 (V/rad)
-0.0500	-28.8675	-106.5857	-29.8096	-11.9054
-0.0400	-28.8675	-106.6805	-29.6974	-11.8742
-0.0300	-28.8675	-106.7540	-29.6113	-11.8504
-0.0200	-28.8675	-106.8065	-29.5503	-11.8335
-0.0100	-28.8675	-106.8379	-29.5140	-11.8235
0.0000	-28.8675	-106.8484	-29.5020	-11.8201
0.0100	-28.8675	-106.8379	-29.5140	-11.8235
0.0200	-28.8675	-106.8065	-29.5503	-11.8335
0.0300	-28.8675	-106.7540	-29.6113	-11.8504
0.0400	-28.8675	-106.6805	-29.6974	-11.8742
0.0500	-28.8675	-106.5857	-29.8096	-11.9054
0.0600	-28.8675	-106.4695	-29.9491	-11.9441
0.0700	-28.8675	-106.3317	-30.1173	-11.9909
0.0800	-28.8675	-106.1720	-30.3161	-12.0462
0.0900	-28.8675	-105.9901	-30.5479	-12.1108
0.1000	-28.8675	-105.7856	-30.8154	-12.1855
0.1100	-28.8675	-105.5581	-31.1220	-12.2713
0.1200	-28.8675	-105.3070	-31.4719	-12.3695
0.1300	-28.8675	-105.0317	-31.8703	-12.4815
0.1400	-28.8675	-104.7315	-32.3233	-12.6092
0.1500	-28.8675	-104.4056	-32.8389	-12.7549
0.1600	-28.8675	-104.0528	-33.4267	-12.9215
0.1700	-28.8675	-103.6721	-34.0991	-13.1126
0.1800	-28.8675	-103.2620	-34.8717	-13.3329
0.1900	-28.8675	-102.8207	-35.7654	-13.5885
0.2000	-28.8675	-102.3460	-36.8074	-13.8876
0.2100	-28.8675	-101.8353	-38.0352	-14.2413
0.2200	-28.8675	-101.2851	-39.5015	-14.6650
0.2300	-28.8675	-100.6909	-41.2826	-15.1817
0.2400	-28.8675	-100.0469	-43.4949	-15.8259
0.2500	-28.8675	-99.3444	-46.3262	-16.6534
0.2600	-28.8675	-98.5713	-50.1027	-17.7614

Table A.11: Gains computed using LQR for inverted pendulum control problem

α (rad)	k_1 (V/m)	k_2 (V/rad)	k_3 (V/m)	k_4 (V/rad)
0.2700	-28.8675	-97.7077	-55.4572	-19.3385
0.2800	-28.8675	-96.7188	-63.8333	-21.8150
0.2900	-28.8675	-95.5291	-79.6144	-26.4992
0.3000	-28.8675	-93.8883	-129.5948	-41.3937
0.3100	28.8675	-76.0760	115.1212	31.7835
0.3200	28.8675	-77.2126	63.0695	16.1803
0.3300	28.8675	-77.9354	47.5609	11.5160
0.3400	28.8675	-78.4914	39.3387	9.0354
0.3500	28.8675	-78.9499	34.0348	7.4304
0.3600	28.8675	-79.3418	30.2443	6.2800
0.3700	28.8675	-79.6841	27.3584	5.4015
0.3800	28.8675	-79.9872	25.0642	4.7011
0.3900	28.8675	-80.2580	23.1825	4.1250
0.4000	28.8675	-80.5016	21.6019	3.6398
0.4100	28.8675	-80.7213	20.2494	3.2234
0.4200	28.8675	-80.9198	19.0744	2.8607
0.4300	28.8675	-81.0993	18.0411	2.5409
0.4400	28.8675	-81.2614	17.1229	2.2560
0.4500	28.8675	-81.4073	16.2997	1.9999
0.4600	28.8675	-81.5381	15.5561	1.7680
0.4700	28.8675	-81.6546	14.8800	1.5568
0.4800	28.8675	-81.7575	14.2617	1.3631
0.4900	28.8675	-81.8475	13.6933	1.1847
0.5000	28.8675	-81.9249	13.1685	1.0197
0.5100	28.8675	-81.9903	12.6819	0.8664
0.5200	28.8675	-82.0438	12.2291	0.7235
0.5300	28.8675	-82.0859	11.8063	0.5899
0.5400	28.8675	-82.1166	11.4102	0.4646
0.5500	28.8675	-82.1361	11.0383	0.3468
0.5600	28.8675	-82.1447	10.6880	0.2357
0.5700	28.8675	-82.1423	10.3574	0.1308

Table A.12: Gains computed using LQR for inverted pendulum control problem

Appendix B

Derivatives of G and L with respect to $\overset{*}{\xi}$ and $\overset{*}{\sigma}$

G'_{ξ}, and L'_{ξ} are derivatives of $G(x, \overset{*}{\xi}, \overset{*}{\sigma})$ and $L(x, \overset{*}{\xi}, \overset{*}{\sigma})$ with respect to $\overset{*}{\xi}$ at $\hat{\xi}$, expressed as

$$G'_{\xi} = G'_{\xi}(x, \hat{\xi}, \hat{\sigma}) = \frac{\partial G(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\xi}} \Bigg| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$L'_{\xi} = L'_{\xi}(x, \hat{\xi}, \hat{\sigma}) = \frac{\partial L(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\xi}} \Bigg| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

also, G'_{σ}, and L'_{σ} are derivatives of $G(x, \overset{*}{\xi}, \overset{*}{\sigma})$ and $L(x, \overset{*}{\xi}, \overset{*}{\sigma})$ with respect to $\overset{*}{\sigma}$ at $\hat{\sigma}$, expressed as

$$G'_{\sigma} = G'_{\sigma}(x, \hat{\xi}, \hat{\sigma}) = \frac{\partial G(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\sigma}} \Bigg| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$L'_{\sigma} = L'_{\sigma}(x, \hat{\xi}, \hat{\sigma}) = \frac{\partial L(x, \overset{*}{\xi}, \overset{*}{\sigma})}{\partial \overset{*}{\sigma}} \Bigg| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$G'_{\xi} = G'_{\xi}(x, \hat{\xi}, \hat{\sigma}) = \left[g'_{\zeta 1}(\hat{\sigma}_1 \left\| X - \hat{\xi}_1 \right\|), g'_{\zeta 2}(\hat{\sigma}_2 \left\| X - \hat{\xi}_2 \right\|), \ldots, g'_{\zeta N}(\hat{\sigma}_N \left\| X - \hat{\xi}_N \right\|) \right]^T$$

$$g'_{\zeta j}(\hat{\sigma}_j \left\| X - \hat{\xi}_j \right\|) = \frac{\partial}{\partial \overset{*}{\xi}_j} g_j(\overset{*}{\sigma}_j \left\| X - \overset{*}{\xi}_j \right\|) \Big| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$= \frac{\partial}{\partial \overset{*}{\xi}_j} \left(\prod_{i=1}^{n} \exp\left[-(\overset{*}{\sigma}_j^i (x_i - \overset{*}{\xi}_j^i))^2 \right] \right) \Big| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$= 2 \cdot g_j(\hat{\sigma}_j \left\| X - \hat{\xi}_j \right\|) \cdot (\hat{\sigma}_j^i (x_i - \hat{\xi}_j^i) \equiv g_\zeta$$

$$g_\zeta = 2 \cdot \left(\prod_{i=1}^{n} \exp\left[-(\hat{\sigma}_j^i (x_i - \hat{\xi}_j^i))^2 \right] \right) \cdot (\hat{\sigma}_j^i (x_i - \hat{\xi}_j^i)$$

$$= 2 \cdot \prod_{i=1}^{n} \frac{a}{\exp^{a^2}}$$

where $a = \hat{\sigma}_j^i (x_i - \hat{\xi}_j^i)$, It is easy to show that g_ζ is bounded with the parameter, i.e., g_ζ is bounded with respect to $\hat{\xi}_j^i$.

$$G'_\sigma = G'_\sigma(x, \hat{\xi}, \hat{\sigma}) = \left[g'_{\sigma 1}(\hat{\sigma}_1 \left\| X - \hat{\xi}_1 \right\|), g'_{\sigma 2}(\hat{\sigma}_2 \left\| X - \hat{\xi}_2 \right\|), \ldots, g'_{\sigma N}(\hat{\sigma}_N \left\| X - \hat{\xi}_N \right\|) \right]^T$$

$$g'_{\sigma j}(\hat{\sigma}_j \left\| X - \hat{\xi}_j \right\|) = \frac{\partial}{\partial \overset{*}{\sigma}_j} g_j(\overset{*}{\sigma}_j \left\| X - \overset{*}{\xi}_j \right\|) \Big| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$= \frac{\partial}{\partial \overset{*}{\sigma}_j} \left(\prod_{i=1}^{n} \exp\left[-(\overset{*}{\sigma}_j^i (x_i - \overset{*}{\xi}_j^i))^2 \right] \right) \Big| \overset{*}{\xi} = \hat{\xi}, \overset{*}{\sigma} = \hat{\sigma}$$

$$= -2 \cdot g_j(\hat{\sigma}_j \left\| X - \hat{\xi}_j \right\|) \cdot (\hat{\sigma}_j^i (x_i - \hat{\xi}_j^i)^2) \equiv g_\sigma$$

$$g_\sigma = -2 \cdot \left(\prod_{i=1}^{n} \exp\left[-(\hat{\sigma}_j^i (x_i - \hat{\xi}_j^i))^2 \right] \right) \cdot (\hat{\sigma}_j^i (x_i - \hat{\xi}_j^i)^2)$$

$$= -2 \cdot \prod_{i=1}^{n} \frac{(x_i - \hat{\xi}_j^i)^2 b}{\exp^{(x_i - \hat{\xi}_j^i)^2 b^2}}$$

also we can see that g_σ is bounded with the parameter, i.e., g_σ is bounded with respect to $\hat{\xi}_j^i$.

Appendix C

Fuzzy model of Hysteresis nonlinearity

Consider the system of one input α and one output C_n described in Chapter 4. The input-output experimental data pairs are given in Tables C.1-C.5. The input-output experimental data is modified by adding another input which is $\dot{\alpha}$, as discussed in Chapter 4. ANFIS is applied on a system of two inputs (α and $\dot{\alpha}$) and one output (C_n). Table C.6 shows the fuzzy model of 8 rules for C_n. Figures C.1 and C.2 show the membership function of α and $\dot{\alpha}$, respectively.

α	$\dot{\alpha}$	C_n
0.9344	0.3974	0.0033
0.9399	0.3985	0.0053
0.9483	0.4003	0.0079
1.0195	0.4149	0.0128
1.3937	0.4841	0.0256
1.7180	0.5365	0.0332
2.0915	0.5908	0.0451
3.0653	0.7114	0.0642
3.6455	0.7733	0.0900
4.2799	0.8350	0.1178
4.8996	0.8903	0.1435
5.5608	0.9449	0.1661
6.2633	0.9988	0.1882
7.0021	1.0516	0.2127
7.7672	1.1027	0.2407
11.2094	1.2979	0.3751
12.0196	1.3374	0.4169
13.0410	1.3843	0.4600
13.9882	1.4252	0.5015
14.9514	1.4646	0.5407
16.0385	1.5064	0.5794
17.0632	1.5435	0.6219
18.1248	1.5797	0.6706
19.2166	1.6148	0.7259
20.2172	1.6452	0.7849
21.3406	1.6773	0.8446
22.4767	1.7078	0.9037
23.7463	1.7397	0.9631
24.9264	1.7673	1.0242
26.1348	1.7936	1.0875
27.2465	1.8163	1.1521
28.5038	1.8401	1.2176
29.7798	1.8623	1.2838
31.0711	1.8830	1.3511
32.3760	1.9020	1.4188
33.6924	1.9193	1.4864
34.8842	1.9335	1.5542
36.2130	1.9476	1.6225

Table C.1: Input-output data for hysteresis modeling problem

α	$\dot{\alpha}$	C_n
38.8952	1.9708	1.7494
40.1218	1.9792	1.7940
41.5017	1.9868	1.8172
42.7542	1.9923	1.8173
44.1508	1.9966	1.7964
45.5464	1.9992	1.7594
46.8023	2.0000	1.7130
48.2041	1.9992	1.6650
49.6193	1.9966	1.6224
50.9067	1.9926	1.5874
52.3496	1.9863	1.5584
53.6457	1.9790	1.5310
55.0773	1.9691	1.5029
56.5131	1.9572	1.4736
57.9540	1.9433	1.4436
59.3998	1.9272	1.4127
60.8484	1.9090	1.3812
63.7547	1.8657	1.3250
65.2165	1.8404	1.3042
66.6821	1.8125	1.2883
68.1431	1.7821	1.2763
69.5902	1.7493	1.2680
71.0190	1.7141	1.2636
72.4316	1.6763	1.2625
73.8324	1.6359	1.2624
75.2225	1.5925	1.2625
73.8324	1.6359	1.2624
75.2225	1.5925	1.2603
76.5976	1.5462	1.2542
77.9503	1.4970	1.2439
79.2734	1.4450	1.2323
80.5625	1.3903	1.2225
81.8134	1.3328	1.2161
83.0207	1.2729	1.2103
84.1764	1.2106	1.2002
85.2721	1.1465	1.1827

Table C.2: Input-output data for hysteresis modeling problem

α	$\dot{\alpha}$	C_n
86.3031	1.0809	1.1601
87.2692	1.0139	1.1382
88.1721	0.9453	1.1215
89.0117	0.8752	1.1109
89.7840	0.8040	1.1052
90.4839	0.7322	1.1047
91.1095	0.6604	1.1113
91.6621	0.5886	1.1251
92.1432	0.5173	1.1408
92.5507	0.4473	1.1495
92.8793	0.3811	1.1448
93.1257	0.3222	1.1285
93.2947	0.2744	1.1112
93.4010	0.2394	1.1050
93.4636	0.2161	1.1144
93.4948	0.2035	1.1331
93.7380	-0.0000	1.1480
93.6730	-0.1053	1.1494
93.5197	-0.1928	1.1363
93.2614	-0.2845	1.1168
92.8998	-0.3766	1.1011
92.4482	-0.4660	1.0959
91.9172	-0.5520	1.1002
91.3065	-0.6358	1.1077
90.6080	-0.7186	1.1120
89.8166	-0.8008	1.1107
88.9373	-0.8817	1.1080
87.9828	-0.9602	1.1098
86.9673	-1.0355	1.1200
85.9020	-1.1071	1.1356
84.7969	-1.1750	1.1594
83.6623	-1.2389	1.1594
82.5094	-1.2988	1.1628
81.3469	-1.3548	1.1631
80.1778	-1.4070	1.1619
78.9993	-1.4561	1.1600

Table C.3: Input-output data for hysteresis modeling problem

α	$\dot{\alpha}$	C_n
77.8044	-1.5025	1.1579
76.5857	-1.5466	1.1579
75.3365	-1.5888	1.1616
74.0529	-1.6292	1.1695
72.7359	-1.6678	1.1802
71.3936	-1.7043	1.1914
70.0396	-1.7385	1.2014
68.6876	-1.7701	1.2078
67.3443	-1.7991	1.2095
66.0072	-1.8257	1.2062
64.6668	-1.8502	1.2003
63.3118	-1.8729	1.1948
61.9354	-1.8938	1.1919
60.5373	-1.9131	1.1920
59.1240	-1.9304	1.1947
57.7052	-1.9458	1.2001
56.2892	-1.9592	1.2100
54.8797	-1.9706	1.2262
53.4755	-1.9800	1.2481
52.0718	-1.9876	1.2719
50.6755	-1.9934	1.2916
49.2978	-1.9973	1.3032
48.2978	-1.9991	1.3077
46.6624	-2.0000	1.3098
45.4150	-1.9990	1.3143
44.0341	-1.9963	1.3215
42.7950	-1.9924	1.3281
41.4171	-1.9864	1.3304
40.0344	-1.9786	1.3278
38.6468	-1.9690	1.3215
37.3967	-1.9587	1.3123
36.0129	-1.9456	1.2993
34.6365	-1.9307	1.2808
33.4033	-1.9157	1.2566
32.0360	-1.8972	1.2277
30.5324	-1.8746	1.1949

Table C.4: Input-output data for hysteresis modeling problem

α	$\dot{\alpha}$	C_n
29.1646	-1.8518	1.1583
27.7956	-1.8269	1.1165
26.4261	-1.7997	1.0687
25.0578	-1.7702	1.0143
23.6949	-1.7384	0.9546
22.3427	-1.7043	0.8917
21.0049	-1.6679	0.8275
19.6814	-1.6291	0.7633
18.3706	-1.5878	0.6998
17.0738	-1.5439	0.6377
15.7972	-1.4973	0.5783
14.5497	-1.4484	0.5228
13.3361	-1.3973	0.4714
12.1546	-1.3437	0.4226
10.8863	-1.2815	0.3744
9.7620	-1.2218	0.3257
8.6776	-1.1593	0.2772
7.6521	-1.0952	0.2318
6.7001	-1.0305	0.1929
5.7387	-0.9589	0.1618
4.9364	-0.8934	0.1368
4.2286	-0.8302	0.1135
3.4319	-0.7512	0.0888
2.7432	-0.6742	0.0622
2.1854	-0.6036	0.0374
1.6510	-0.5262	0.0181
1.1947	-0.4487	0.0057
0.8032	-0.3687	-0.0021
0.4343	-0.2716	-0.0089
0.6948	-0.3431	-0.0166

Table C.5: Input-output data for hysteresis modeling problem

R_i	σ_1^i, c_1^i	σ_2^i, c_2^i	a_{i1}	a_{i2}	a_{i0}
R_1	13.2, 0.4387	1.645, -2.021	-0.1203	-3.471	-4.785
R_2	13.2, 0.4387	1.643, 2.024	0.16	-3.4	4.565
R_3	13.18, 31.5	1.645, -2.021	0.02543	0.6001	1.177
R_4	13.18, 31.5	1.643, 2.024	0.1382	-2.529	1.917
R_5	13.17, 62.59	1.645, -2.021	0.02096	0.7288	3.208
R_6	13.17, 62.59	1.643, 2.024	0.1137	9.075	-23.74
R_7	13.18, 93.68	1.645, -2.021	0.01959	-2.416	-5.132
R_8	13.18, 93.68	1.643, 2.024	-0.294	-4.422	33.75

Table C.6: 8 rules for hysteresis fuzzy model

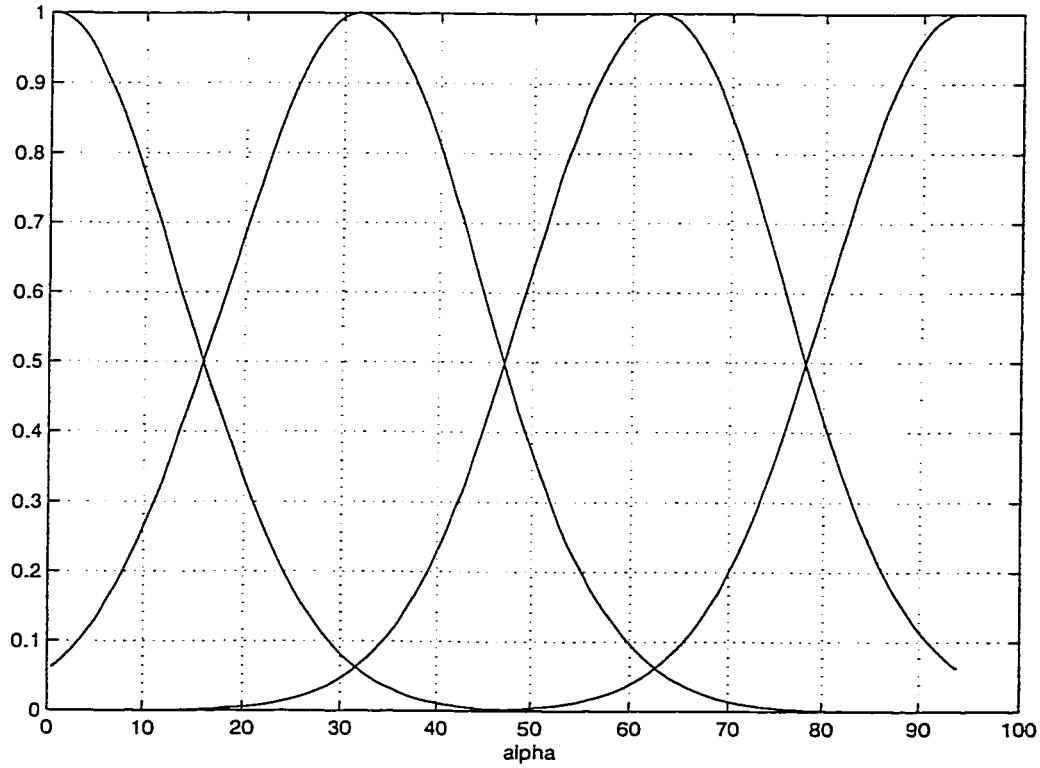

Figure C.1: Membership functions of α

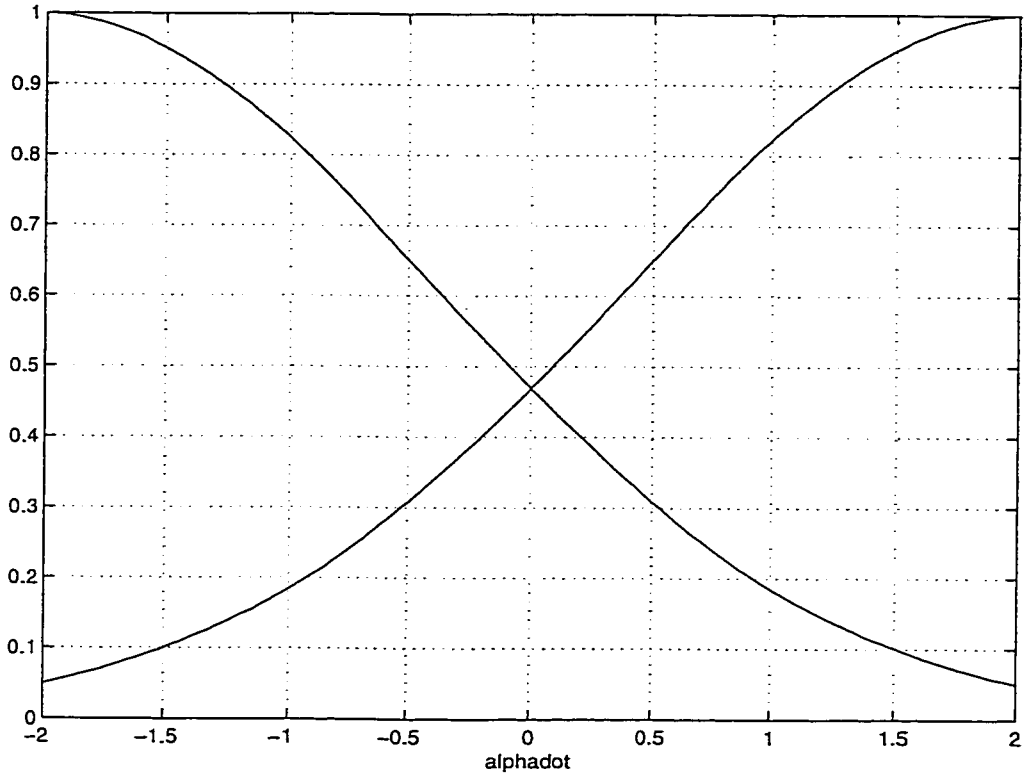

Figure C.2: Membership functions of $\dot{\alpha}$

Bibliography

[1] J. Shamma and M. Athans, "Analysis of nonlinear gain scheduled control systems," *IEEE Transactions on Automatic Control*, vol. AC-35, no. 8, pp. 898–907, 1990.

[2] J. Shamma and M. Athans, "Gain scheduling: potential hazards and possible remedies," *IEEE Control Systems*, pp. 101–107, 1992.

[3] J. Jiang, "Optimal gain scheduling controller for a diesel engine," *IEEE Control Systems*, pp. 42–48, 1994.

[4] J. Shamma and M. Athans, "Guaranteed properties of gain scheduled control of linear parameter-varying plants," *Automatica*, vol. 27, pp. 559–564, 1991.

[5] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Transactions on Systems Man and Cybernetics*, vol. SMC-3, pp. 28–44, 1973.

[6] E. H. Mamdani and S. Assillian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man Machine Studies*, vol. 7, no. 1, pp. 1–13, 1975.

[7] B. Kosko, *Neural Networks and Fuzzy Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[8] M. Brown and C. Harris, *Neurofuzzy Adaptive Modeling and Control*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[9] S. Tan and Y. Yu, "Adaptive fuzzy modeling of nonlinear dynamical systems," *Automatica*, vol. 32, pp. 637–643, 1996.

[10] M. Sugeno and T. Yasukawa, "A fuzzy -logic-based approach to qualitative modeling," *IEEE Transactions on Fuzzy Systems*, vol. FS-1, pp. 7–31, 1993.

[11] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller-part I," *IEEE Transactions on Systems Man and Cybernetics*, vol. 20, no. 2, pp. 404–418, 1990.

[12] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller-part II," *IEEE Transactions on Systems Man and Cybernetics*, vol. 20, no. 2, pp. 419–435, 1990.

[13] L.-X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least-squares learning," *IEEE Transactions on Neural Networks*, vol. 3, pp. 807–814, 1992.

[14] B. Kosko, "Fuzzy systems as universal approximators," in *Proceedings of IEEE International Conference on Fuzzy Systems*, March 1992.

[15] J. L. Castro and M. Delgado, "Fuzzy systems with defuzzification are universal approximators," *IEEE Transactions on Systems Man and Cybernetics*, vol. 26, pp. 149–152, 1996.

[16] H. T. Nguyen, V. Kreinovich, and O. Sirisaengtaksin, "Fuzzy control as a universal control tool," *Fuzzy Sets and Systems*, vol. 80, pp. 71–86, 1996.

[17] J. L. Castro, "Fuzzy logic controllers are universal approximators," *IEEE Transactions on Systems Man and Cybernetics*, vol. 25, pp. 629–635, 1995.

[18] L.-X. Wang, "Stable adaptive fuzzy control of nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 1, no. 2, pp. 146–155, 1993.

[19] C.-Y. Su and Y. Stepanenko, "Adaptive control of a class of nonlinear systems with fuzzy logic," *IEEE Transactions on Fuzzy Systems*, vol. 2, pp. 285–294, 1994.

[20] B. S. Chen, C. H. Lee, and Y. C. Chang, "H_∞ tracking design of uncertain nonlinear SISO systems: adaptive fuzzy approach," *IEEE Transactions on Fuzzy Systems*, vol. 4, pp. 32–43, 1996.

[21] H. Lee and M. Tomizuka, "Robust adaptive control using a universal approximator for SISO nonlinear systems," *IEEE Transactions on Fuzzy Systems*, vol. 8, pp. 95–106, 2000.

[22] D. L. Tsay, H.-Y. Chung, and C.-J. Lee, "The adaptive control of nonlinear systems using Sugeno-type of fuzzy logic," *IEEE Transactions on Systems Man and Cybernetics*, vol. 7, pp. 225–229, 1999.

[23] H. Han, C.-Y. Su, and Y. Stepanenko, "Adaptive control of a class of nonlinear systems with nonlinearly parameterized fuzzy approximators." Accepted for publication in IEEE Transactions on Fuzzy Systems, December 2000.

[24] K. Demirli and P. Muthukumaran, "Fuzzy system identification with higher order subtractive clustering." accepted for publication in Journal of Intelligent and Fuzzy Systems.

[25] M. Alata, K. Demirli, and A. Bulgak, "Interpolation behavior of TS fuzzy controllers," in *Proceedings of the 1999 IEEE international Symposium on Intelligent Control/Intelligent Systems and Semiotics, Cambridge, MA*, Sept. 1999.

[26] M. Alata, M. Jarrah, K. Demirli, and A. Bulgak, "Fuzzy gain scheduling for position control of a robot manipulator," *Journal of Intelligent and Fuzzy Systems*, no. 8, pp. 111–120, 2000.

[27] M. Alata, C.-Y. Su, and K. Demirli, "Adaptive control of a class of nonlinear systems with a first-order parameterized Sugeno fuzzy approximator." Submitted for publication in IEEE Transactions on Systems Man and Cybernetics, Part B: Cybernetics, May 2000.

[28] M. Alata, K. Demirli, and C.-Y. Su, "Adaptive/fuzzy control of a class of nonlinear systems with a first-order parameterized Sugeno fuzzy approximator." Submitted to American Control Conference (ACC2001), Sept. 2000.

[29] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338–353, 1965.

[30] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*. Academic Press, New York, 1980.

[31] D. Dubois and H. Prade, *New results about properties and semantics of fuzzy set-theoretic operators*. P. P. Wang and S. K. Chang, editors. Plenum Press, New York, 1980.

[32] M. Mizumoto, "Fuzzy sets and their operations," *Information and Control*, vol. 48, pp. 30–48, 1981.

[33] M. M. Gupta and J. Qi, "Theory of t-norms and fuzzy inference methods," *Fuzzy Sets and Systems*, vol. 40, pp. 431–450, 1991.

[34] S. Weber, "Fuzzy connectives, negations and implications," *Fuzzy Sets and Systems*, vol. 11, pp. 115–134, 1983.

[35] K. Demirli, *A unified and extended framework for operator selection in generalized modus ponens type fuzzy reasoning*. PhD thesis, University of Toronto, 1995.

[36] R. R. Yager, "Connectives and quantifiers in fuzzy sets," *Fuzzy Sets and Systems*, vol. 40, pp. 39–75, 1991.

[37] H.-J. Zimmermann, *Fuzzy sets. Decision Making, and Expert Systems.* Kluwer Academic Publishers, 1987.

[38] H.-J. Zimmermann and P. Zysno, "Latnet connectives in human decision making," *Fuzzy Sets and Systems*, vol. 4, pp. 37–51, 1980.

[39] D. Dubois and H. Prade, "Fuzzy sets in approximate reasoning, part I: Inference with possibility distributions," *Fuzzy Sets and Systems*, vol. 40, pp. 143–202, 1991.

[40] D. Dubois and H. Prade, "Fuzzy sets in approximate reasoning, part II: Logical approaches," *Fuzzy Sets and Systems*, vol. 40, pp. 203–244, 1991.

[41] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine studies*, vol. 7, pp. 1–13, 1975.

[42] L. A. Zadeh, "A rationali for fuzzy control," *Transactions for ASME, Journal of Dynamic system Measurements and Control*, vol. 94, pp. 3–4, 1972.

[43] D. A. Rutherford and G. C. Bloore, "The implementation of fuzzy algorithms for control," *in Proceedings of IEEE*, vol. 64, no. 4, pp. 572–573, 1976.

[44] J. J. Ostergaard, *Fuzzy logic control of a heat exchange process.* Fuzzy Automata and Decision Processes, North-Holland, 1977.

[45] D. Willaeys, "Optimal control of fuzzy systems," in *Proceedings of International Congress on Applied Systems Research and Cybernetics, Acapulco*, Dec. 1980.

[46] S. Fukami, M. Mizumoto, and K. Tanaka, "Some considerations of fuzzy conditional inference," *Fuzzy Sets and Systems*, vol. 4, pp. 243–273, 1980.

[47] T. Takagi and M. Sugeno, "Derivation of fuzzy control rules from human operator's control actions," in *Proceedings of IFAC Symposium on Fuzzy Information, Knowledge Representation and Decision Analysis, France*, July 1983.

[48] S. Yasunobo and S. Miyamoto, *Automatic train operation by predictive control.* M. Sugeno, Ed. Amsterdam: North-Holland, 1985.

[49] M. Sugeno and K. Murakami, "Fuzzy parking control of model car," in *23rd IEEE Conference on Decision and Control, Las Vegas*, July 1984.

[50] J. B. Kiszka, M. M. Gupta, and P. Nikiforuk, "Energestic stability of fuzzy dynamic systems," *IEEE Transactions of Systems Man and Cybernetics*, vol. SMC-15, no. 5, pp. 783–792, 1985.

[51] T. Yamakawa, "High speed fuzzy controller hardware system," in *Proceedings of 2nd Fuzzy System Symposium, Japan*, 1986.

[52] W. J. Kickert and H. V. N . Lemke, "Application of a fuzzy controller in a warm water plant," *Automatica*, vol. 12, no. 4, pp. 301–308, 1976.

[53] R. M. Tong, M. B. Beck, and A. Latten, "Fuzzy control of the activated sludge wastewater treatment process," *Automatica*, vol. 16, no. 6, pp. 695–701, 1980.

[54] C. Pappis and E. H. Mamdani, "A fuzzy logic controller for a traffic junction," *IEEE Transactions of Systems Man and Cybernetics*, vol. SMC-7, no. 10, pp. 707–717, 1977.

[55] P. M. Larsen, "Industrial applications of fuzzy control," *International Journal of Man Machine Studies*, vol. 12, pp. 3–10, 1980.

[56] L. I. Larkin, *A fuzzy logic controller for aircraft flight control.* M. Sugeno, Ed. Amsterdam: North-Holland,. 1985.

[57] M. Uragami, M. Mizumoto, and K. Tanaka, "'fuzzy robot control," *Cybernetics*, vol. 6, pp. 39–64, 1976.

[58] M. Scharf and N. J. Mandic, *The application of a fuzzy controller to the control of a multi-degree-of-freedom robot arm.* M. Sugeno, Ed. Amsterdam: North-Holland, 1985.

[59] A. M. Wakileh and K. F. Gill, "Use of fuzzy logic in robotics," *Computers in Industry*, vol. 10, no. 1, pp. 35–46, 1988.

[60] C. Isik, "Identification and fuzzy rule-based control of a mobile robot motion," in *Proceedings of IEEE International Symposium on Intelligent Control, Philadelphia, PA*, 1987.

[61] T. I. Tsay and J. H. Huang, "Robust nonlinear control of robot manipulators," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 1994.

[62] F. Y. Hsu and L. C. Fu, "Adaptive robust fuzzy control of robot manipulators," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, 1994.

[63] K. Demirli and I. B. Turksen, "Mobile robot navigation with generalized modus ponens type fuzzy reasoning," in *Proceeding 1995 IEEE International Conference on Systems Man and Cybernetics*, 1995.

[64] M. Sugeno and K. Murakami, *An experimental study on fuzzy parking control using a model car.* M. Sugeno, Ed. Amsterdam: North-Holland, 1985.

[65] M. Sugeno and M. Nishida, "Fuzzy control of model car." *Fuzzy Sets and Systems*, vol. 16, pp. 103–113, 1985.

[66] S. Murakami, "Application of fuzzy controller to automobile speed control system," in *Proceedings of the IFAC Symposium on Fuzzy Information, Knowledge Representation and Decision Analysis, France*, 1983.

[67] J. C. Bezdek, "Cluster validity with fuzzy sets," *Cybernetics*, vol. 3, pp. 58–71, 1974.

[68] R. M. Tong, *The construction and evaluation of fuzzy models.* M. M. Gupta, R. K. Ragade and R. R. Yager, Ed. Amsterdam: North-Holland, 1979.

[69] W. Pedrycz, "Identification in fuzzy systems," *IEEE Transactions on Systems Man and Cybernetics*, vol. 14, pp. 361–366, 1984.

[70] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems Man and Cybernetics*, vol. 15, pp. 116–132, 1985.

[71] S. Chiu, "Fuzzy model identification based on cluster estimation," *Journal of Intelligent and Fuzzy Systems*, vol. 2, pp. 267–278, 1994.

[72] M. R. Emami, I. B. Turksen, and A. A. Goldenberg, "Development of a systematic methodology of fuzzy logic modeling," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 3, pp. 346–361, 1998.

[73] K. Demirli, S. Cheng, and P. Muthukumaran, "Fuzzy modeling of job sequencing with subtractive clustering." Submitted to Fuzzy Sets and Systems, 1999.

[74] E. H. Mamdani, "Advances in the linguistic synthesis of fuzzy controllers," *International Journal of Man Machine Studies*, vol. 8, no. 6, pp. 669–678, 1976.

[75] C. V. Nogoita, "On the stability of fuzzy systems," in *Proceedings of IEEE International Conference on Cybernetics and Society*, 1978.

[76] M. Braae and D. A. Rutherford, "Theoretical and linguistic aspects of the fuzzy logic controller," *Automatica*, vol. 15, no. 5, pp. 553–577, 1979.

[77] W. Pedrycz, "An approach to the analysis of fuzzy systems," *International Journal of Control*, vol. 34, no. 3, pp. 403–421, 1981.

[78] K. S. Ray and D. D. Majumder, "Application of circle criteria for stability analysis of linear SISO and MIMO systems associated with fuzzy logic controllers," *IEEE Transactions of Systems Man and Cybernetics*, vol. SMC-14, no. 2, pp. 345–349, 1984.

[79] R. Langari and M. Tomizuka, "Analysis and design of fuzzy linguistic control systems," in *Proceedings of ASME Winter Annual Meeting*. 1990.

[80] R. Langari and M. Tomizuka, "Stability of fuzzy linguistic control systems," in *Proceedings of IEEE Conference on Decision and Control*, 1990.

[81] R. Langari and M. Tomizuka, "Analysis of stability of a class of fuzzy linguistic controllers with internal dynamics," in *Proceedings of ASME*, 1993.

[82] R. Langari, "A nonlinear formulation of a class of fuzzy linguistic control algorithms," in *Proceedings of 1992 American Control Conference, Chicago, IL*, 1992.

[83] K. Tanaka and M. Sugeno, "Stability analysis and design of fuzzy control systems," *Fuzzy Sets and Systems*, vol. 45, pp. 135–156, 1992.

[84] K. Tanaka and M. Sano, "Fuzzy stability criterion of a class of nonlinear systems," *Information Science*, vol. 7, pp. 3–26, 1993.

[85] K. Tanaka and M. Sano, "A robust stabilization problem of fuzzy controller systems and its applications to backing up control of a truck-trailer," *IEEE transactions on Fuzzy Systems*, vol. 2, pp. 119–134, 1994.

[86] G. Chen and H. Ying, "BIBO stability of nonlinear fuzzy PI control systems," *Journal of Intelligent and Fuzzy Systems*, vol. 5, pp. 245–256, 1998.

[87] S. Quail and S. Adnan, *State of the art in household appliances using fuzzy logic*. J. Yen and R. Langari and L. Zadeh, Ed. IEEE Press, College Station, TX, 1992.

[88] A. Nowe, "Sugeno and fuzzy Mamdani controllers put in a uniform interpolation framework," *International Journal of Intelligent Systems*, vol. 13, pp. 243–256, 1998.

[89] B. Kosko, "Fuzzy systems as universal approximators," *IEEE Transactions on Computers*, vol. 43, no. 11, pp. 1329–1333, 1994.

[90] H. Ying, W. Silver, and J. Buckley, "Fuzzy control theory: a nonlinear case," *Automatica*, vol. 26, no. 3, pp. 513–520, 1990.

[91] Y. Kim, D. Kang, W. Bang, J. Kim, and Z. Bien, "Balancing position control of a circular inverted pendulum system using self-learning fuzzy controller," in *Proceedings of Seventh IFSA Word Congress, Prague*, 1997.

[92] J. Tao and J. Luh, "Application of neural network with real-time training to robust position/force control of multiple robots," in *Proceedings of 1993 IEEE International Conference on Robotics and Automation, Atlanta*, 1993.

[93] G. Rovithakis and M. Christodoulou, "Adaptive control of unknown plants using dynamical neural networks," *IEEE Transactions on Systems Man and Cybernetics*, vol. 24, no. 3, pp. 400–412, 1994.

[94] D. Nguyen and B. Widrow, "Neural networks for self-learning control systems," *IEEE Control Systems Magazine*, pp. 18–23, 1990.

[95] J. R. Jang, "Anfis: Adaptive-network-based fuzzy inference system," *IEEE Transactions on Systems Man and Cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[96] J. Jang, C. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing.* Prentice Hall, 1997.

[97] M. Scharf and N. J. Mandic, *The application of a fuzzy controller to the control of a multi-degree-of-freedom robot arm.* M. Sugeno, Ed. Amsterdam: North-Holland, 1985.

[98] C. M. Lim and T. Hiyama, "Application of fuzzy logic control to a manipulator," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 5, pp. 688–691, 1991.

[99] H. Ying, "Sufficient conditions on uniform approximation of multivariate functions by general Takagi-Sugeno fuzzy systems with linear rule consequent," *IEEE Transactions on Systems Man and Cybernetics-part A: Systems and Humans*, vol. 28, pp. 515–520, 1998.

[100] J. Apkarian, *A comprehensive and modular laboratory for control systems design and implementation.* Quanser Consulting, 1995.