Amirkabir University of Technology
(Tehran Polytechnic)

Introduction to Computational Intelligence Course Project

Design a Fuzzy Logic Controller for a Rotary Flexible Joint Robotic Arm

By

Mohammad Arshia Samoudi
Mahban Gholi Jafari
Niloofar Tavahodi
Mohammad Arabzadeh

******

Supervisor:
Dr. I. Sharifi

Tehran Polytechnic
2022

Abstract

The purpose of this research is to design a fuzzy logic feedback controller (FLC) in order to control a desired tip angle position a rotary flexible joint robotic arm. The FLC is also employed to dampen the vibration emanated from a rotary flexible joint robotic arm when reaching a desired tip angle position. The performance of FLC is tested in simulation and experiment. It is found that the FLC is successfully designed, applied and tested. The results show that fuzzy logic controller performed satisfactorily control a desired tip angle position and reduce the oscillations.

Contents

List of Figures

List of Tables

Chapter 1

Introduction to SRV02

1.1 Introduction to Rotary Flexible Joint module

The Quanser Rotary Flexible Joint module, pictured in Figure 1.1, consists of a rigid beam mounted on a flexible joint that rotates via a DC motor. The joint deflection is measured using a sensor.

This module is designed to mount to a Quanser rotary servo plant (SRV02). The sensor shaft is aligned with the motor shaft. One end of a rigid link is mounted to the sensor shaft. The link rotation is counteracted by two extension springs anchored to the solid frame resulting in an instrumented flexible joint.

The spring anchor points are adjustable to three locations to obtain various stiffness constants. Three types of springs are supplied with the system resulting in a total of 9 possible stiffness values. The link is also adjustable in length thus allowing for variations in inertia.

Fig. 1.1 Rotary Flexible Joint system

This system is similar in nature to the control problems encountered in large geared robot joints where flexibility is exhibited in the gearbox. The Rotary Flexible Joint is an ideal experiment intended to model a flexible joint on a robot or spacecraft.

This experiment is also useful in the study of vibration analysis and resonance. The SRV02 Rotary Flexible Joint module is equipped with a1024 line optical encoder to sense arm's angular position.

1.2 Components

The Rotary Flexible Joint components are identified in subsection 1.2.1. Some of those components are then described in subsection 1.2.2.

1.2.1 Component Nomenclature

The components of the Rotary Flexible Joint module are listed in Table 1.1 below and labeled in Figure 1.2 and Figure 1.3.

Table 1.1 Listing of ROTFLEX Components

ID	Components	ID	Components
1	ROTFLEX base	7	Springs
2	Thumbscrews	8	Adjustable load
3	ROTFLEX arm	9	Encoder connector
4	Arm sensor (Encoder)	10	ROTFLEX pivot
5	Base anchor points	11	Adjustable load anchor points
6	Arm anchor points	12	SRV02

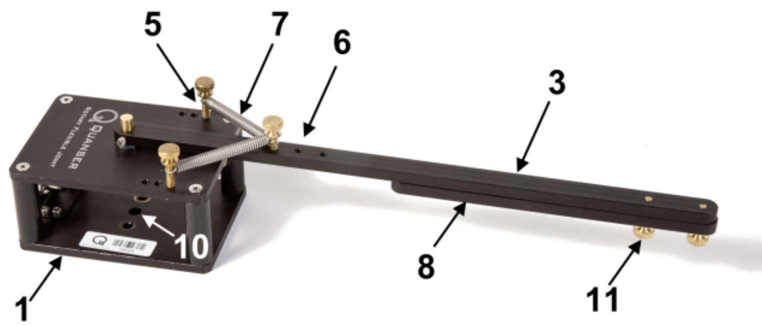

Fig. 1.2 ROTFLEX Components - Top View

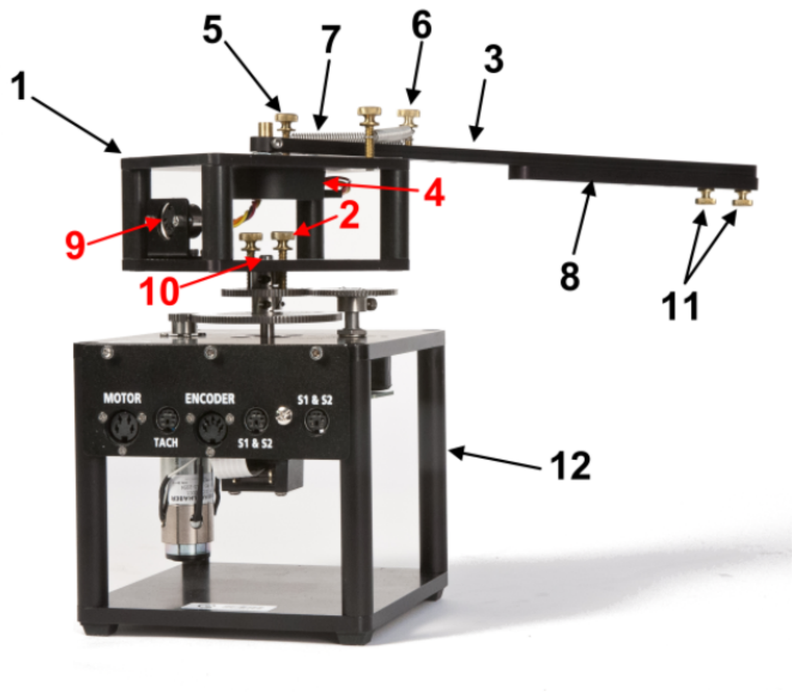

Fig. 1.3 ROTFLEX Components - Back View when installed on SRV02

1.2.2 Component Description

The ROTFLEX option comes with an optical encoder used to measure the arm's angular position. The model used is a US Digital Optical Kit Encoder. It offers high resolution (4096 counts in quadrature), and measures the relative angle of the arm. The internal wiring of the encoder and the 5-pin DIN connector on the ROTFLEX module is illustrated in Figure 1.4.

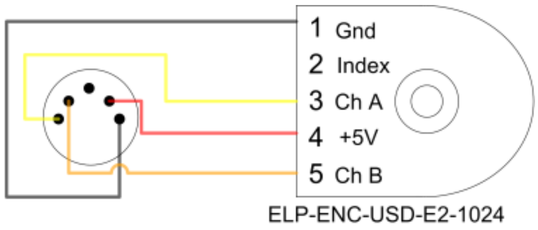

Fig. 1.4 Encoder Wiring

1.3 System Specification

Table 1.2, below, lists and characterizes the main parameters associated with the ROTFLEX module. Some of the parameters listed in Table 1.2 are used in the mathematical model.

Table 1.2 Rotary Flexible Joint specifications

Symbol	Description	Value	Unit
	Module Dimensions	$10 \times 8 \times 5$	cm^3
L_1	Main arm length	29.8	cm
L_2	Load arm length	15.6	cm
d_{11}	Arm Anchor Point 1	21.0	cm
d_{12}	Arm Anchor Point 2	23.5	cm
d_{13}	Arm Anchor Point 3	26.0	cm
	Module body mass	0.3	kg
m_1	Main arm mass	0.064	kg
m_2	Load arm mass	0.03	kg
K_{enc}	Encoder resolution (in quadrature mode)	4096	Counts/Rev
K_1	Spring 1 stiffness	187	N/m
K_2	Spring 2 stiffness	313	N/m
K_3	Spring 3 stiffness	565	N/m

Chapter 2

Mathematical Equation

2.1 Structure

In brief, Figure 1 depicts the flexible joint module coupled to the SRV02 plant. The Module is attached to the SRV02 load gear by two thumbscrews.

The Main Arm is attached to the module body by two identical springs thus resulting in the flexible joint. In this paper, only the state space representation of the complete system is given. This allows us to investigate the fuzzy logic control performance in simulation.

The detailed derivation of the mathematical equation of flexible joint module can be found in [1]

Figure 2.1 is a model depicting the Rotary Flexible Joint system. The ROTFLEX module has been designed to allow many configurations. As shown in Figure 2.1, there are three anchor positions on the arm as well as three anchor positions on the body. The force exerted by the springs can be varied by attaching the springs in different anchor points.

The ROTFLEX system is also supplied with three sets of springs of different stiffness constants (values shown in Table 1.2). The secondary load arm attaches underneath the main arm and can be connected to different anchor points, allowing the total arm length to be changed.

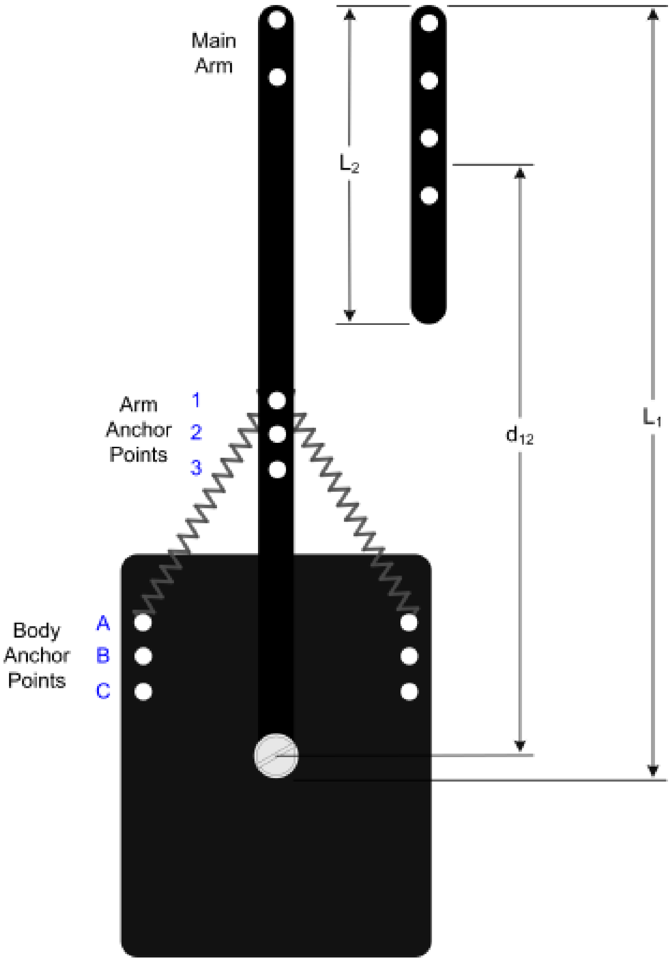

Fig. 2.1 Rotary Flexible Joint Module

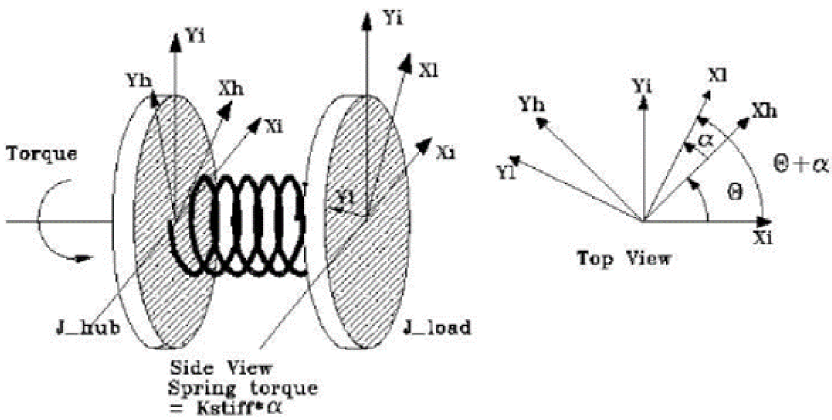

Fig. 2.2 Spring Torque

$$F_1 = k(L_1 - L) + F_r \tag{2.1}$$

$$F_{1x} = \frac{F_1 L_{1x}}{L_1} \qquad F_{1y} = \frac{F_1 L_{1y}}{L_1} \tag{2.2}$$

$$F_2 = k(L_2 - L) + F_r \tag{2.3}$$

$$F_{2x} = \frac{F_2 L_{2x}}{L_1} \qquad F_{2y} = \frac{F_2 L_{2y}}{L_2} \tag{2.4}$$

$$F_x = F_{2x} - F_{1x} \tag{2.5}$$

$$F_y = F_{2y} + F_{1y} \tag{2.6}$$

$$M_x = R \times F_x = RF_x \sin(\frac{\pi}{2} - \alpha) \tag{2.7}$$

$$M_y = R \times F_y = RF_y \sin(2\pi - \alpha) \tag{2.8}$$

$$M = M_x + M_y = R\cos(\alpha)(F_{2x} - F_{1x}) - R\sin(\alpha)(F_{2y} + F_{1y}) \tag{2.9}$$

with M Calculation, with the Balanced point Linearization, **K** will be Obtained:

$$K_{stiff} = (\frac{\partial M}{\partial \alpha})|_{\alpha=0} \tag{2.10}$$

2.2 Equation of Motion

2.2.1 Lagrangian Modeling

Elegant and powerful methods have also been devised for solving dynamic problems with constraints. One of the best known is called Lagrange's equations. The Lagrangian L is defined as L = T - V, where T is the kinetic energy and V the potential energy of the system in question.

Generally speaking, the potential energy of a system depends on the coordinates of all its particles; this may be written as V = V(x1, y1, z1, x2, y2, z2, . . .). The kinetic energy generally depends on the velocities, which, using the notation vx = dx/dt = $\dot{x}$, may be written $T = T(\dot{x}_1, \dot{y}_1, \dot{z}_1, \dot{x}_2, \dot{y}_2, \dot{z}_2, ...)$.

Thus, a dynamic problem has six dynamic variables for each particle that is, x, y, z and $\dot{x}, \dot{y}, \dot{z}$ and the Lagrangian depends on all 6N variables if there are N particles.

$$Lagrangian : L = T - U \tag{2.11}$$

$$Total\ \ kinetic\ \ energy : T = T_{hub} + T_{arm} \tag{2.12}$$

$$T = \frac{1}{2} J_{hub} \dot{\theta}^2 + \frac{1}{2} J_{arm} (\dot{\theta} + \dot{\alpha})^2 \quad U = \frac{1}{2} K_{stiff} \alpha^2 \tag{2.13}$$

$$L = \frac{1}{2} J_{hub} \dot{\theta}^2 + \frac{1}{2} J_{arm} (\dot{\theta} + \dot{\alpha})^2 - \frac{1}{2} K_{stiff} \alpha^2 \tag{2.14}$$

$$Euler - Lagrange\ \ Equation : \frac{\mathrm{d}}{\mathrm{d}t} (\frac{\delta L}{\delta \dot{q}}) - \frac{\delta L}{\delta q} = F \tag{2.15}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} (\frac{\delta L}{\delta \dot{\theta}}) - \frac{\delta L}{\delta \theta} = T_{output} - B_{eq} \dot{\theta} \tag{2.16}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} (\frac{\delta L}{\delta \dot{\alpha}}) - \frac{\delta L}{\delta \alpha} = 0 \tag{2.17}$$

2.2.2 System Equations

$$T_{output} = \frac{\eta_m \eta_g k_t k_g (v_m - k_g k_m \dot{\theta})}{R_m} \tag{2.18}$$

$$J_{eq}\ddot{\theta} + J_{arm}(\ddot{\theta} + \ddot{\alpha}) = T_{output} - B_{eq}\dot{\theta} \tag{2.19}$$

$$J_{Arm}(\ddot{\theta} + \ddot{\alpha}) + K_{stiff}\alpha = 0 \tag{2.20}$$

2.2.3 State Space Equations

State Space of Linear System:

$$\dot{x} = Ax + Bu \quad y = Cx + Du \tag{2.21}$$

$$x = \begin{bmatrix} x1 = \theta \\ x2 = \alpha \\ x3 = \dot{\theta} \\ x4 = \dot{\alpha} \end{bmatrix} \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{K_{stiff}}{J_{eq}} & \frac{-\eta_m \eta_g k_t k_m k_g^2 + B_{eq} R_m}{J_{eq} R_m} & 0 \\ 0 & \frac{-K_{stiff}(J_{eq} + J_{arm})}{J_{eq} J_{arm}} & \frac{\eta_m \eta_g k_t k_m k_g^2 + B_{eq} R_m}{J_{eq} R_m} & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ \frac{\eta_m \eta_g k_t k_g}{J_{eq} R_m} \\ \frac{-\eta_m \eta_g k_t k_g}{J_{eq} R_m} \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Table 2.1 State Space Form parameters

Symbol	Value
J_{eq}	0.0026
J_{arm}	0.0035
K_g	70
$K_t = K_m$	0.00767
K_{stiff}	1.2485
η_m	0.69
η_g	0.9
B_{eq}	0.0040
R_m	2.6

$$B = \begin{bmatrix} 0 \\ 0 \\ 49.32 \\ -49.32 \end{bmatrix} \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 480.19 & -24.94 & 0 \\ 0 & -836.91 & 28.02 & 0 \end{bmatrix}$$

2.2.4 Transfer Function

From the below equation, the Transfer function will be obtained:

$$G(s) = C(sI - A)^{-1}B + D \tag{2.22}$$

$$G(s) = \begin{cases} \frac{\theta(s)}{v_m(s)} = \frac{123300s^2+43983576}{2500s^4+62350s^3+2092275s^2+18544029s} \\ \\ \frac{\alpha(s)}{v_m(s)} = \frac{379764-123300s}{2500s^3+62350s^2+2092275s^1+18544029} \end{cases}$$

Poles of Two Transfer Function are Listed Below:

$$G(s)|_{Poles} = \begin{cases} \frac{\theta(s)}{v_m(s)} = [\ 0,\ \ -10.8436,\ \ -7.0482 \pm 25.1868j\ ] \\ \\ \frac{\alpha(s)}{v_m(s)} = [\ -10,\ \ -7.0482 \pm 25.1868j\ ] \end{cases}$$

Chapter 3

System Stability

In this section, we've studied the stability of G_θ and G_α transfer functions using the Routh Hurwitz and root locus methods. If all the roots of the characteristic equation of a system lie on the left half of the s-plane, then the system is said to be a stable system.

Routh Hurwitz criterion states that any system can be stable if and only if all the roots of the first column of the Routh table have the same sign; if it does not has the same sign or there is a sign change, then the number of sign changes in the first column is equal to the number of roots of the characteristic equation in the right half of the s-plane i.e. equals to the number of roots with positive real parts.

In control theory and stability theory, root locus analysis is a graphical method for examining how the roots of a system change with variation of a certain system parameter, commonly a gain within a feedback system. This is a technique used as a stability criterion in the field of classical control theory developed by Walter R.

Evans which can determine stability of the system. The root locus plots the poles of the closed loop transfer function in the complex s-plane as a function of a gain parameter. The root locus gives information about the stability and transient response of feedback control systems.

To study the Routh Hurwitz stability conditions, we've provided a MATLAB script to evaluate the stability based on the system's characteristic equation. For the root locus, we've used MATLAB's rlocus function, which calculates and plots the root locus of the closed-loop pole trajectories as a function of the feedback gain k assuming negative feedback.

3.1 Stability of G_θ

3.1.1 Routh Hurwitz Stability

As seen in Figure 3.1, based on the Routh table of the G_θ's open loop charactristic equation, we can see that the system is stable because it satisfies the Routh Hurwitz stability criterion.

```
Routh Table :
/    2500,     2092275, 0 \
|                         |
|    62350,   18544029, 0 |
|                         |
| 1681865475              |
| ----------,      0,   0 |
|    1247                 |
|                         |
|  18544029,       0,   0 |
|                         |
\  18544029,       0,   0 /


 No sign changes and hence system is stable
```

Fig. 3.1 Routh table of G_θ

3.1.2 Root Locus Stability

According to the root lucas diagram of G_θ calculated by MATLAB if Figure 3.2, most of the diagram is on the left side of the imaginary axis and there is a very small part in the right hand side that shows the instability of the system in those gains (Figure 3.3). The stability of this system occurs on approximately k<50.

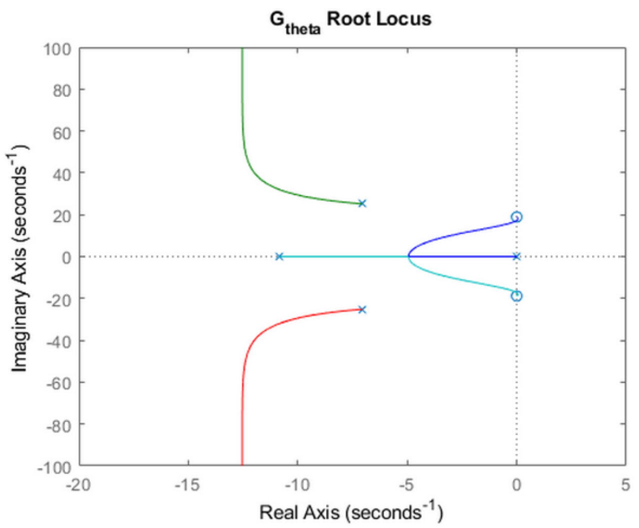

Fig. 3.2 Root locus of G_θ

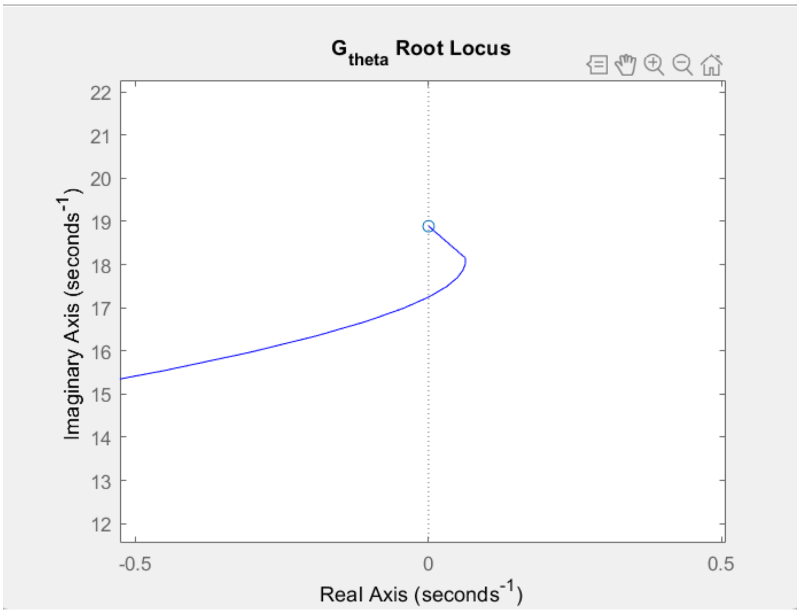

Fig. 3.3 A closer look on the root locus of G_θ

3.2 Stability of G_α

3.2.1 Routh Hurwitz Stability

As seen in Figure 3.4, when we run the Routh Hurwitz MATLAB script for the G_α's open-loop characteristic equation, it declares the system stable because it satisfies the Roth-Herwitz stability criterion.

```
Routh Table :
/    2500,      2092275 \
|                       |
|    62350,   18544029 |
|                       |
| 1681865475            |
| ----------,     0     |
|    1247               |
|                       |
\  18544029,      0     /


 No sign changes and hence system is stable
```

Fig. 3.4 Routh table of G_α

3.2.2 Routh Hurwitz Stability

According to the root lucas diagram of G_α's transfer function calculated by MATLAB in Figure 3.5, there are some parts on the left hand side of the imaginary axis and some on the right of the imaginary axis. Studying the diagram, shows that the system is stable for approximately k<10.

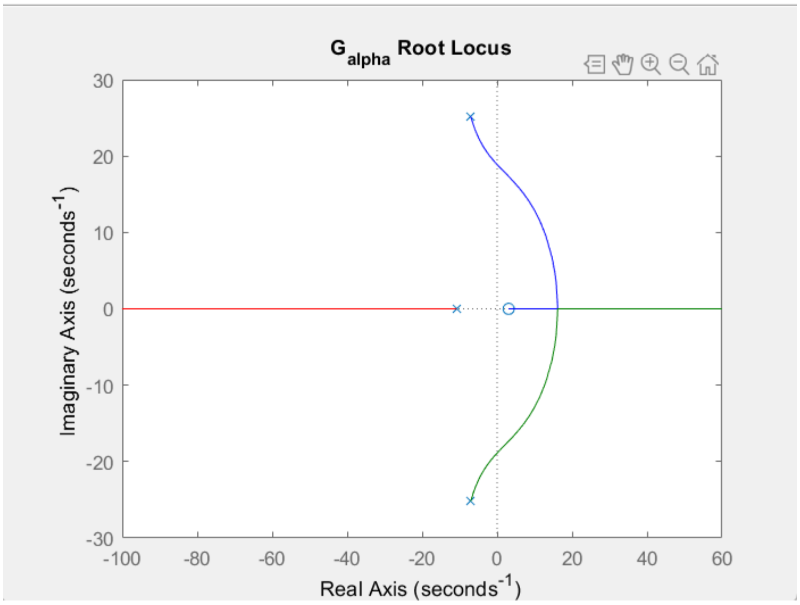

Fig. 3.5 Root locus of G_α

Chapter 4

Fuzzy Controller Design

The goal of this project is to track the desired angle in a rotary flexible joint robotic arm with fuzzy systems. The fuzzy logic controller is used because it is an innovative technology to design solutions for multi-parameter and non-linear control problems. In addition, it uses human experience and experimental results rather than a mathematical model for the definition of a control strategy [2], [3] . Mathematical models and difference equations generate crisp descriptions of systems. This is fine if the mathematical properties and physical laws of the system are known or can be calculated. However, for non-linear processes, the underlying dynamics of the system can be too difficult, or indeed, impossible to model. In these situations, it is more useful to describe the system as a series of if-then rules. This is essentially what a fuzzy model is. It is a mapping of input space to output space by means of a rule base. Hence, the model requires no strict mathematical equations and the range of uses for the fuzzy model is vast. Figure 4.1 shows how fuzzy logic is used to control the position of the rotary flexible joint robotic arm using θ angle and its derivative $\dot{\theta}$.

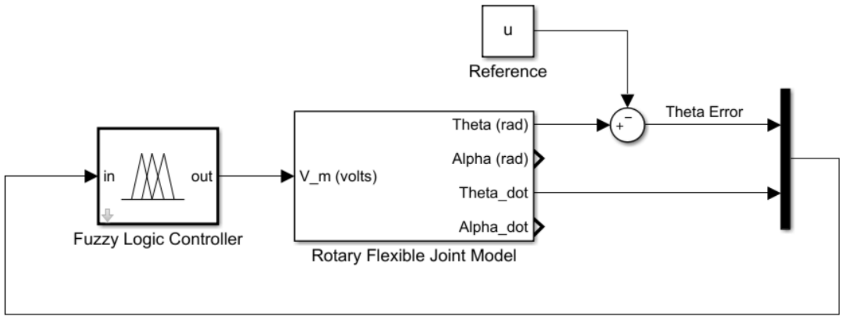

Fig. 4.1 System block diagram

In general, the fuzzy controller has four components namely fuzzification module, rule base, interface engine, and defuzzification module as shown in Figure 4.2. A brief description of each component is as follows:

- ***The fuzzification module***
  The fuzzification interface can be regarded as the input interface. It has the function of modifying/scaling the inputs so that they may compare to the rules in the rule base.
- ***The rule base***
  Here, the knowledge for the control of the system is held as a set of IF–THEN statements.
- ***The interface engine***
  The inference mechanism evaluates which control rules are relevant at the current time and then decides what the input to the plant should be.
- ***The defuzzification module***
  The defuzzification interface acts as an output interface. The output from the inference mechanism is converted so that it may be fed into the input for the process.

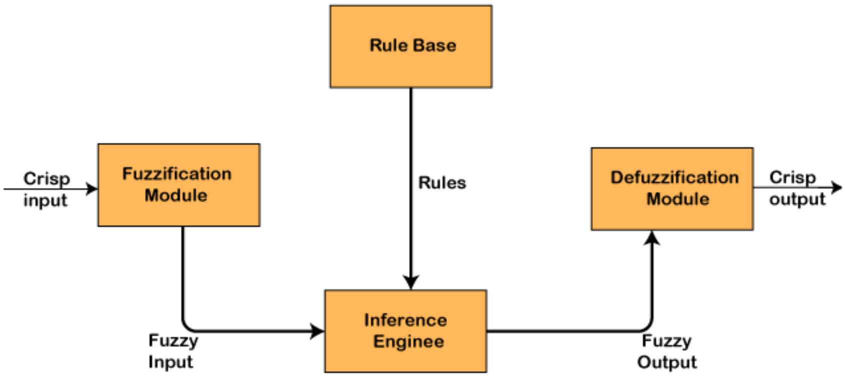

Fig. 4.2 Structure of a fuzzy system

4.1 Fuzzy Mamdani Controller

4.1.1 Input and Output Variables and Set of Terms

Each fuzzy system has some linguistic variables and each variable has a set of terms. To control the θ angle of the given system, we give the θ_{error} and $\dot{\theta}$ as inputs to the fuzzy controller. The controller then calculates the voltage V_m for us to use it as input to our system. Therefore the input linguistic variables are θ_{error} and $\dot{\theta}$, and the output linguistic variable is V_m. The set of terms used for each of these linguistic variables are defined in Table 4.1.

Table 4.1 Set of terms for the linguistic variables

Linguistic variable	Set of terms
θ_{error}	neg-high, neg-low, zero, pos-low, pos-high
$\dot{\theta}$	neg, zero, pos
V_m	neg-high, neg-low, zero, pos-low, pos-high

4.1.2 Membership Functions

Each of the defined terms in Table 4.1, has a membership assigned to it. All membership functions used for the fuzzy controller are shown in Figures 4.3, 4.4, and 4.5.

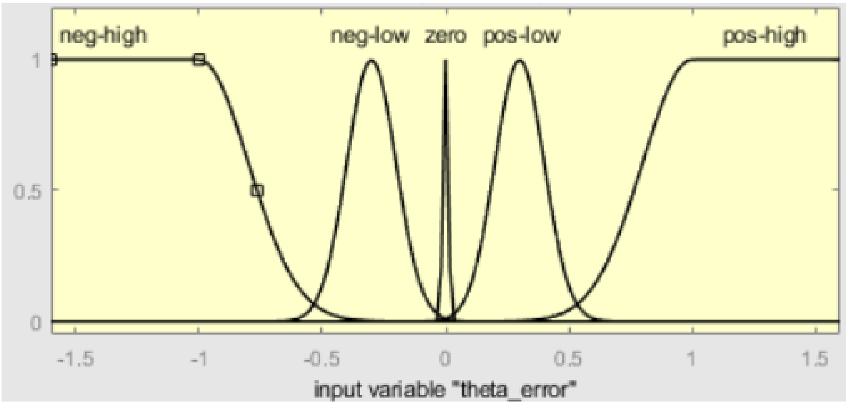

Fig. 4.3 θ_{error} membership functions

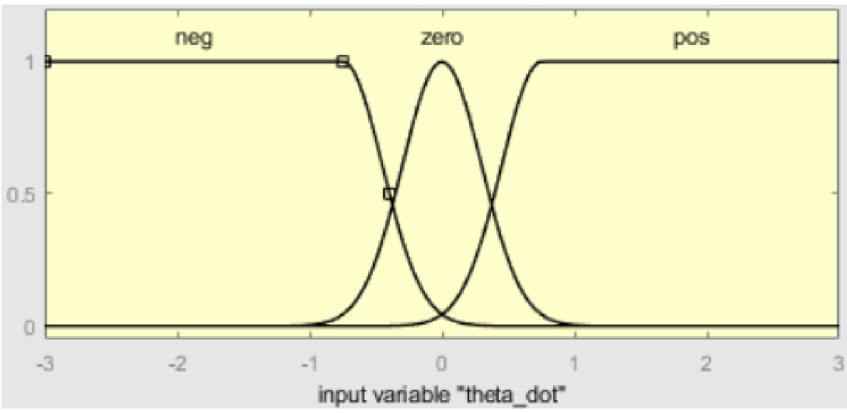

Fig. 4.4 $\dot{\theta}$ membership functions

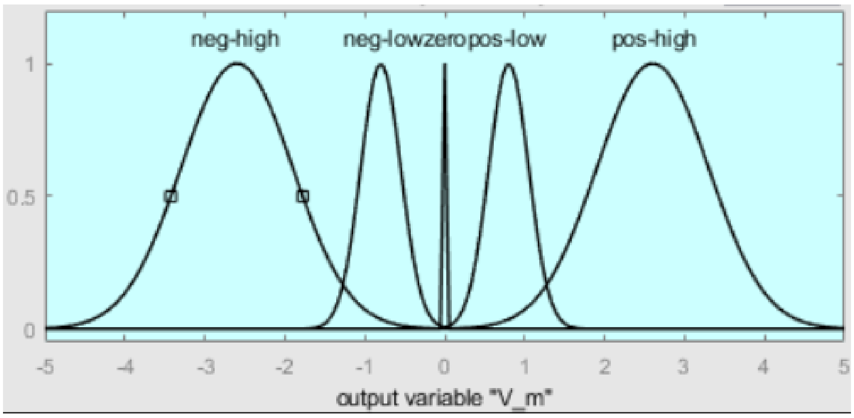

Fig. 4.5 V_m membership functions

4.1.3 Inference Engine

The inference engine uses a set of methods to do the calculations on the fuzzy signals. These methods are defined in Table 4.2.

Table 4.2 Logical methods for the inference engine

Operation	Method
And	Min
Or	Max
Implication	Min
Aggregation	Max
Defuzzification	Centroid

4.1.4 Rule Base

As mentioned before, the fuzzy controller uses a set of rules to control the system. Fuzzy logic rule base applied for the rotary flexible joint robotic arm are written in Table 4.3.

Table 4.3 The fuzzy logic controller rule base

Rule	Description
1	IF (θ_{error} is neg-high) THEN (V_m is pos-high)
2	IF (θ_{error} is pos-high) THEN (V_m is neg-high)
3	IF (θ_{error} is neg-low) AND ($\dot{\theta}$ is pos) THEN (V_m is zero)
4	IF (θ_{error} is pos-low) AND ($\dot{\theta}$ is neg) THEN (V_m is zero)
5	IF (θ_{error} is neg-low) AND ($\dot{\theta}$ is neg) THEN (V_m is pos-high)
6	IF (θ_{error} is pos-low) AND ($\dot{\theta}$ is pos) THEN (V_m is neg-high)
7	IF (θ_{error} is neg-low) AND ($\dot{\theta}$ is zero) THEN (V_m is pos-low)
8	IF (θ_{error} is pos-low) AND ($\dot{\theta}$ is zero) THEN (V_m is neg-low)
9	IF (θ_{error} is zero) AND ($\dot{\theta}$ is neg) THEN (V_m is pos-low)
10	IF (θ_{error} is zero) AND ($\dot{\theta}$ is pos) THEN (V_m is neg-low)
11	IF (θ_{error} is zero) AND ($\dot{\theta}$ is zero) THEN (V_m is zero)

Chapter 5

Fuzzy Controller Stability

5.1 Introduction

Stability is one of the most important concepts of analysis and design of control systems. Stability analysis of fuzzy control systems has been difficult since fuzzy systems are essentially nonlinear systems. Recently, stability analysis techniques based on nonlinear stability theory have been reported.

Langari and Tomizuka analyzed stability of fuzzy linguistic control systems. Kitamura realized stability analysis of fuzzy control systems by using extended circle criterion. Farinwata and Kato utilized phase plane analysis type of techniques. Tanaka and co-authors applied Lyapunov approach to fuzzy control systems and recently developed LMI-based designs, where LMI stands for nonlinear matrix inequality.

The purpose is to show a methodology for stability analysis and design of fuzzy control systems. The methodology discussed here is simple and natural. The stability analysis is based on Lyapunov stability theory. The design utilizes the concept of the so-called "parallel distributed compensation" (PDC). The basic principle for stability analysis and design will be discussed.

Fuzzy controls have been applied to a wide variety of industrial applications. On the other hand, we have experienced difficulties in design of fuzzy controllers, i.e., derivation of fuzzy control rules and determination of membership functions, since we have lacked methodologies for analysis and design of fuzzy control systems.

A keyword for the answer is a model-based fuzzy control. Section 5.2 shows the Takagi-Sugeno's fuzzy model (T-S fuzzy model) and stability conditions based on Lyapunov approach.

5.2 Stability Conditions Based On Lyapunov Approach

5.2.1 Takagi and Sugeno's fuzzy model

The fuzzy model proposed by Takagi and Sugeno is described by fuzzy IF-THEN rules which represent local linear input-output relations of a nonlinear system. The i-th rules of the T-S fuzzy model are of the following forms:

<Continuous fuzzy system: CFS>
Plant Rule i: IF Z1 (t) is Mi1 and ... and zp(t) is Mip, THEN

$$\dot{x}_i(t) = A_i x(t) + B_i u(t), i = 1, 2, ..., r$$

<Discrete fuzzy system: DFS>
Plant Rule i: IF Z1 (t) is Mi1 and ... and zp(t) is Mip, THEN

$$\dot{x}_i(t+1) = A_i x(t) + B_i u(t), i = 1, 2, ..., r$$

M_{ij} is the fuzzy set. CFS and DFS denote the continuous fuzzy system and the discrete fuzzy system, respectively. $x(t) \in R^n$ is the state vector, $u(t) \in R^m$ is the input vector, $A_i \in R^{nxn}$, $B_i \in R^{nxm}$, and r is the number of IF-THEN rules.

zl (t),. . . ,zp(t) are the premise variables. It is assumed in this chapter that the premise variables do not depend on the input variables u(t). It should be emphasized that the stability analysis shown in this chapter can be applied even to the case that the premise variables zl (t),...,zp(t) depend on the input variables u(t).

This restriction is required only for avoidance of complicated defuzzification process. Each linear consequent equation represented by AiX(t) + Biu(t) is called "subsystem". Given a pair of (x(t), u(t)), the final outputs of the fuzzy systems are inferred as follows, where .

$$< CFS >$$

$$\dot{x}(t) = \frac{\sum_{i=1}^{r} w_i(z(t))[A_i x(t) + B_i u(t)]}{\sum_{i=1}^{r} w_i(z(t))} = \sum_{i=1}^{r} h_i(z(t))[A_i x(t) + B_i u(t)]$$

$$< DFS >$$

$$\dot{x}(t+1) = \frac{\sum_{i=1}^{r} w_i(z(t))[A_i x(t) + B_i u(t)]}{\sum_{i=1}^{r} w_i(z(t))} = \sum_{i=1}^{r} h_i(z(t))[A_i x(t) + B_i u(t)]$$

After calculation we will have:

$$w_i(z(t)) = \prod_{j=1}^{p} M_{ij}(z_j(t)), \left\{ \begin{array}{c} \sum_{i=1}^{r} w_i(z(t))) > 0 \\ w_i(z(t)) \geq 0, i = 1, 2, ..., r \end{array} \right\}$$

For all t. $M_{ij}(z_j(t))$ is the grade of membership of Zj(t) in M_{ij}. After more calculation we will have:

$$\begin{array}{c} \sum_{i=1}^{r} h_i(z(t)) = 1 \\ h_i(z(t)) \geq 0, i = 1, 2, ..., r \end{array}$$

For all t. Plant Rule i: IF x(t) is Mil and ... and x(t - n + 1) is Min, THEN

$$x_i(t+1) = A_i x(t) + B_i u(t), i = 1, 2, ..., r$$

5.2.2 Stability Conditions

The open-loop systems are defined as follows:

$$< CFS >$$
$$\dot{x}(t) = \frac{\sum_{i=1}^{r} w_i(z(t))[A_i x(t)]}{\sum_{i=1}^{r} w_i(z(t))} = \sum_{i=1}^{r} h_i(z(t))[A_i x(t)]$$

$$< DFS >$$
$$\dot{x}(t+1) = \frac{\sum_{i=1}^{r} w_i(z(t))[A_i x(t)]}{\sum_{i=1}^{r} w_i(z(t))} = \sum_{i=1}^{r} h_i(z(t))[A_i x(t)]$$

Stability conditions for ensuring stability of the system can be derived via Lyapunov approach. The conditions are given in Theorems 1 and 2. Theorem 1 <CFS>: The equilibrium of a continuous fuzzy system is asymptotically stable in the large if there exists a common positive definite matrix P such that

$$A_i^T P + PA_i < 0$$

For i = 1, 2,. . . , r, i.e., for all the subsystems.

Theorem 2 (DFS): The equilibrium of a discrete fuzzy system is asymptotically stable in the large if there exists a common positive definite matrix P such that

$$A_i^T PA_i - P < 0$$

For i = 1, 2, ... , r, i.e., for all the subsystems.

These theorems are reduced to the Lyapunov stability theorems for linear continuous systems and linear discrete systems respectively when r=1. Theorems 1 and 2 give sufficient conditions for ensuring stability of the open-loop systems. For the open-loop systems, a question naturally arises is whether the systems are stable if all its subsystems are stable, i.e., all $A_i's$ are stable. The answer is no in general.

Theorem 3: Assume that A_i is a stable matrix for i = 1,2, ... ,r. A_iA_j is a stable matrix for i, j = 1,2, ... , r, if there exists a common positive definite matrix P satisfying $A_i^T PA_i - P < 0$. The contraposition of this theorem means that there does not exist a common P satisfying $A_i^T PA_i - P < 0$ if one of the AiAj's is at least an unstable matrix. Hence, the only problem is how to find a common Lyapunov function satisfying $A_i^T PA_i - P < 0$ or $A_i^T PA_i + P < 0$.

In our system:

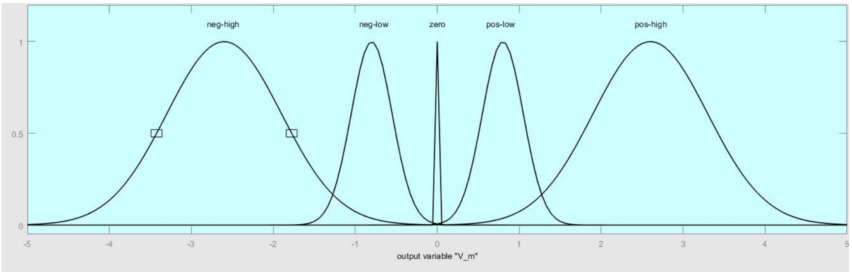

Fig. 5.1 Structure of Fuzzy Controller

For P to be positive definite we wrote a MATLAB code:

```
m = 11;
ii = rand(m);
P1 = ii*ii.';
```

Fig. 5.2

This will generate an 11*11 positive definite P1.

P1 is:

	1	2	3	4	5	6	7	8	9	10	11
1	2.6074	1.9713	1.9741	1.3991	1.6790	2.3664	1.3107	2.9256	1.5171	2.9285	1.6066
2	1.9713	3.8268	2.0515	2.1267	3.0272	3.1889	2.0984	3.8555	3.2357	3.6191	2.2483
3	1.9741	2.0515	2.8884	2.1964	2.7272	3.0609	2.5883	3.0031	2.2796	3.1050	2.1548
4	1.3991	2.1267	2.1964	2.6799	2.4575	2.9579	2.3381	2.5921	2.3115	2.4502	2.3016
5	1.6790	3.0272	2.7272	2.4575	3.5455	3.4611	2.9386	3.2752	2.8420	3.2144	2.4522
6	2.3664	3.1889	3.0609	2.9579	3.4611	4.9876	3.4749	4.5278	3.1958	3.8042	2.7354
7	1.3107	2.0984	2.5883	2.3381	2.9386	3.4749	3.7471	3.0439	2.6869	2.9344	1.7676
8	2.9256	3.8555	3.0031	2.5921	3.2752	4.5278	3.0439	6.1336	3.9852	4.8377	2.7523
9	1.5171	3.2357	2.2796	2.3115	2.8420	3.1958	2.6869	3.9852	3.6007	3.6539	2.1355
10	2.9285	3.6191	3.1050	2.4502	3.2144	3.8042	2.9344	4.8377	3.6539	5.1629	2.4598
11	1.6066	2.2483	2.1548	2.3016	2.4522	2.7354	1.7676	2.7523	2.1355	2.4598	2.5045

Fig. 5.3

For pos-high:

A1 =

	1	2	3	4	5	6	7	8	9	10	11
1	2.5306e-26	4.9665e-20	1.2664e-14	4.1955e-10	1.8058e-06	0.0010	0.0734	0.6926	0.8494	0.1353	0.0028
2	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

Fig. 5.4

Then $A_i^T P A_i + P < 0$

	1	2	3	4	5	6	7	8	9	10	11
1	3.9427	3.8268	2.0515	2.1267	3.0272	3.1915	2.2897	5.6613	5.4503	3.9720	2.2556
2	3.8268	1.9581e-19	2.4965e-14	8.2707e-10	3.5599e-06	0.0020	0.1446	1.3653	1.6744	0.2668	0.0055
3	2.0515	2.4965e-14	4.9999e-14	8.2823e-10	3.5648e-06	0.0020	0.1448	1.3672	1.6767	0.2672	0.0055
4	2.1267	8.2707e-10	8.2823e-10	1.1740e-09	2.5273e-06	0.0014	0.1027	0.9690	1.1884	0.1894	0.0039
5	3.0272	3.5599e-06	3.5648e-06	2.5273e-06	6.0640e-06	0.0017	0.1232	1.1628	1.4261	0.2272	0.0047
6	3.1915	0.0020	0.0020	0.0014	0.0017	0.0048	0.1749	1.6418	2.0115	0.3232	0.0083
7	2.2897	0.1446	0.1448	0.1027	0.1232	0.1749	0.1923	1.1224	1.2246	0.3922	0.1215
8	5.6613	1.3653	1.3672	0.9690	1.1628	1.6418	1.1224	4.0523	3.5356	2.4241	1.1208
9	5.4503	1.6744	1.6767	1.1884	1.4261	2.0115	1.2246	3.5356	2.5771	2.6927	1.3688
10	3.9720	0.2668	0.2672	0.1894	0.2272	0.3232	0.3922	2.4241	2.6927	0.7927	0.2256
11	2.2556	0.0055	0.0055	0.0039	0.0047	0.0083	0.1215	1.1208	1.3688	0.2256	0.0090

Fig. 5.5

The determinant of this matrix is negative (-8.2966e-113).

For pos-low we have:

A2 =

	1	2	3	4	5	6	7	8	9	10	11
1	1.3264e-117	8.9298e-81	6.7655e-51	5.7683e-28	5.5346e-12	0.0060	0.7261	9.9295e-06	1.5280e-17	2.6460e-36	5.1566e-62
2	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

Fig. 5.6

$A_i^T P A_i + P < 0$

	1	2	3	4	5	6	7	8	9	10	11
1	3.9427	3.8268	2.0515	2.1267	3.0272	3.2044	3.9917	3.8555	3.2357	3.6191	2.2483
2	3.8268	3.5207e-80	1.3337e-50	1.1371e-27	1.0911e-11	0.0118	1.4315	1.9574e-05	3.0121e-17	5.2162e-36	1.0165e-61
3	2.0515	1.3337e-50	2.6711e-50	1.1387e-27	1.0926e-11	0.0118	1.4335	1.9601e-05	3.0163e-17	5.2234e-36	1.0869e-50
4	2.1267	1.1371e-27	1.1387e-27	1.6141e-27	7.7436e-12	0.0084	1.0160	1.3893e-05	2.1378e-17	1.6892e-27	9.2671e-28
5	3.0272	1.0911e-11	1.0926e-11	7.7436e-12	1.8585e-11	0.0100	1.2192	1.6672e-05	8.3963e-12	1.6208e-11	8.8916e-12
6	3.2044	0.0118	0.0118	0.0084	0.0100	0.0283	1.7262	0.0175	0.0091	0.0175	0.0096
7	3.9917	1.4315	1.4335	1.0160	1.2192	1.7262	1.9035	2.1244	1.1016	2.1265	1.1666
8	3.8555	1.9574e-05	1.9601e-05	1.3893e-05	1.6672e-05	0.0175	2.1244	5.8099e-05	1.5064e-05	2.9078e-05	1.5952e-05
9	3.2357	3.0121e-17	3.0163e-17	2.1378e-17	8.3963e-12	0.0091	1.1016	1.5064e-05	4.6361e-17	4.4747e-17	2.4548e-17
10	3.6191	5.2162e-36	5.2234e-36	1.6892e-27	1.6208e-11	0.0175	2.1265	2.9078e-05	4.4747e-17	1.5498e-35	4.2510e-36
11	2.2483	1.0165e-61	1.0869e-50	9.2671e-28	8.8916e-12	0.0096	1.1666	1.5952e-05	2.4548e-17	4.2510e-36	1.6569e-61

Fig. 5.7

The determinant of this matrix is negative (-1.0746e-111).

For zero: A3 =

	1	2	3	4	5	6	7	8	9	10	11
1	1.3264e-117	8.9298e-81	6.7655e-51	5.7683e-28	5.5346e-12	0.0060	0.7261	9.9295e-06	1.5280e-17	2.6460e-36	5.1566e-62
2	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

Fig. 5.8

Then $A_i^T P A_i + P < 0$

	1	2	3	4	5	6	7	8	9	10	11
1	3.9427	3.8268	2.0515	2.1267	3.0272	3.2044	3.9917	3.8555	3.2357	3.6191	2.2483
2	3.8268	3.5207e-80	1.3337e-50	1.1371e-27	1.0911e-11	0.0118	1.4315	1.9574e-05	3.0121e-17	5.2162e-36	1.0165e-61
3	2.0515	1.3337e-50	2.6711e-50	1.1387e-27	1.0926e-11	0.0118	1.4335	1.9601e-05	3.0163e-17	5.2234e-36	1.0869e-50
4	2.1267	1.1371e-27	1.1387e-27	1.6141e-27	7.7436e-12	0.0084	1.0160	1.3893e-05	2.1378e-17	1.6892e-27	9.2671e-28
5	3.0272	1.0911e-11	1.0926e-11	7.7436e-12	1.8585e-11	0.0100	1.2192	1.6672e-05	8.3963e-12	1.6208e-11	8.8916e-12
6	3.2044	0.0118	0.0118	0.0084	0.0100	0.0283	1.7262	0.0175	0.0091	0.0175	0.0096
7	3.9917	1.4315	1.4335	1.0160	1.2192	1.7262	1.9035	2.1244	1.1016	2.1265	1.1666
8	3.8555	1.9574e-05	1.9601e-05	1.3893e-05	1.6672e-05	0.0175	2.1244	5.8099e-05	1.5064e-05	2.9078e-05	1.5952e-05
9	3.2357	3.0121e-17	3.0163e-17	2.1378e-17	8.3963e-12	0.0091	1.1016	1.5064e-05	4.6361e-17	4.4747e-17	2.4548e-17
10	3.6191	5.2162e-36	5.2234e-36	1.6892e-27	1.6208e-11	0.0175	2.1265	2.9078e-05	4.4747e-17	1.5498e-35	4.2510e-36
11	2.2483	1.0165e-61	1.0869e-50	9.2671e-28	8.8916e-12	0.0096	1.1666	1.5952e-05	2.4548e-17	4.2510e-36	1.6569e-61

Fig. 5.9

The determinant of this matrix is negative (-1.0746e-111).

For neg-low:

A4 =

	1	2	3	4	5	6	7	8	9	10	11
1	1.3264e-117	8.9298e-81	6.7655e-51	5.7683e-28	5.5346e-12	0.0060	0.7261	9.9295e-06	1.5280e-17	2.6460e-36	5.1566e-62
2	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

Fig. 5.10

Then $A_i^T P A_i + P < 0$

	1	2	3	4	5	6	7	8	9	10	11
1	3.9427	3.8268	2.0515	2.1267	3.0272	3.2044	3.9917	3.8555	3.2357	3.6191	2.2483
2	3.8268	3.5207e-80	1.3337e-50	1.1371e-27	1.0911e-11	0.0118	1.4315	1.9574e-05	3.0121e-17	5.2162e-36	1.0165e-61
3	2.0515	1.3337e-50	2.6711e-50	1.1387e-27	1.0926e-11	0.0118	1.4335	1.9601e-05	3.0163e-17	5.2234e-36	1.0869e-50
4	2.1267	1.1371e-27	1.1387e-27	1.6141e-27	7.7436e-12	0.0084	1.0160	1.3893e-05	2.1378e-17	1.6892e-27	9.2671e-28
5	3.0272	1.0911e-11	1.0926e-11	7.7436e-12	1.8585e-11	0.0100	1.2192	1.6672e-05	8.3963e-12	1.6208e-11	8.8916e-12
6	3.2044	0.0118	0.0118	0.0084	0.0100	0.0283	1.7262	0.0175	0.0091	0.0175	0.0096
7	3.9917	1.4315	1.4335	1.0160	1.2192	1.7262	1.9035	2.1244	1.1016	2.1265	1.1666
8	3.8555	1.9574e-05	1.9601e-05	1.3893e-05	1.6672e-05	0.0175	2.1244	5.8099e-05	1.5064e-05	2.9078e-05	1.5952e-05
9	3.2357	3.0121e-17	3.0163e-17	2.1378e-17	8.3963e-12	0.0091	1.1016	1.5064e-05	4.6361e-17	4.4747e-17	2.4548e-17
10	3.6191	5.2162e-36	5.2234e-36	1.6892e-27	1.6208e-11	0.0175	2.1265	2.9078e-05	4.4747e-17	1.5498e-35	4.2510e-36
11	2.2483	1.0165e-61	1.0869e-50	9.2671e-28	8.8916e-12	0.0096	1.1666	1.5952e-05	2.4548e-17	4.2510e-36	1.6569e-61

Fig. 5.11

The determinant of this matrix is negative (-1.0746e-111).

For neg-high:

A5 =

	1	2	3	4	5	6	7	8	9	10	11
1	1.3264e-117	8.9298e-81	6.7655e-51	5.7683e-28	5.5346e-12	0.0060	0.7261	9.9295e-06	1.5280e-17	2.6460e-36	5.1566e-62
2	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

Fig. 5.12

Then $A_i^T P A_i + P < 0$

	1	2	3	4	5	6	7	8	9	10	11
1	3.9427	3.8268	2.0515	2.1267	3.0272	3.2044	3.9917	3.8555	3.2357	3.6191	2.2483
2	3.8268	3.5207e-80	1.3337e-50	1.1371e-27	1.0911e-11	0.0118	1.4315	1.9574e-05	3.0121e-17	5.2162e-36	1.0165e-61
3	2.0515	1.3337e-50	2.6711e-50	1.1387e-27	1.0926e-11	0.0118	1.4335	1.9601e-05	3.0163e-17	5.2234e-36	1.0869e-50
4	2.1267	1.1371e-27	1.1387e-27	1.6141e-27	7.7436e-12	0.0084	1.0160	1.3893e-05	2.1378e-17	1.6892e-27	9.2671e-28
5	3.0272	1.0911e-11	1.0926e-11	7.7436e-12	1.8585e-11	0.0100	1.2192	1.6672e-05	8.3963e-12	1.6208e-11	8.8916e-12
6	3.2044	0.0118	0.0118	0.0084	0.0100	0.0283	1.7262	0.0175	0.0091	0.0175	0.0096
7	3.9917	1.4315	1.4335	1.0160	1.2192	1.7262	1.9035	2.1244	1.1016	2.1265	1.1666
8	3.8555	1.9574e-05	1.9601e-05	1.3893e-05	1.6672e-05	0.0175	2.1244	5.8099e-05	1.5064e-05	2.9078e-05	1.5952e-05
9	3.2357	3.0121e-17	3.0163e-17	2.1378e-17	8.3963e-12	0.0091	1.1016	1.5064e-05	4.6361e-17	4.4747e-17	2.4548e-17
10	3.6191	5.2162e-36	5.2234e-36	1.6892e-27	1.6208e-11	0.0175	2.1265	2.9078e-05	4.4747e-17	1.5498e-35	4.2510e-36
11	2.2483	1.0165e-61	1.0869e-50	9.2671e-28	8.8916e-12	0.0096	1.1666	1.5952e-05	2.4548e-17	4.2510e-36	1.6569e-61

Fig. 5.13

The determinant of this matrix is negative (-1.0746e-111).

Therefore, the fuzzy system is asymptotically stable.

Chapter 6

Results

6.1 Step Response

The step response of the system without and with the fuzzy logic controller is shown in Figures 6.1 and 6.2. Based on these results, we can see that the fuzzy logic controller has improved the system's behaviour. Based on the step response of the controlled system, we can find out the transient state performance of the system which is shown in Table 6.1.

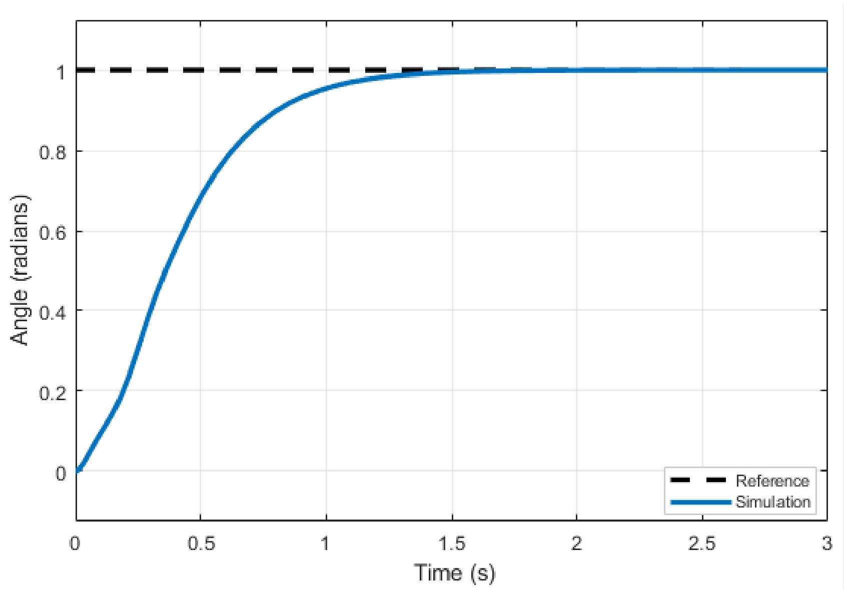

Fig. 6.1 Step response of the system without controller

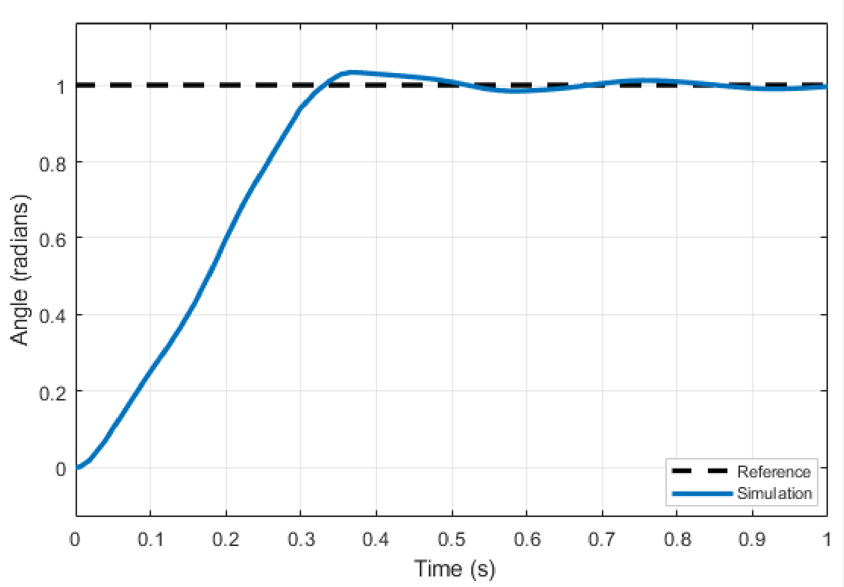

Fig. 6.2 Step response of the controlled system

Table 6.1 The controlled system transient state behaviour

Metric	Value
Rise time (T_r)	0.337s
Peak time (T_p)	0.366s
Settling time (T_s)	0.457s
%Overshoot	3.3%

6.2 Angle Tracking Performance

Here, we've simulated the system with different input references to evaluate the controller performance under various situations. Figures 6.3 and 6.4 show the outputs with pulse and sine inputs respectively and Figures 6.5 and 6.6 show the outputs of the same inputs, but with white Gaussian noise added to the system's feedback loops. As we can see in the figures, the system's performance is very good without noise, but when we added the noise to the feedback loops, the fuzzy logic controller wasn't able to provide acceptable results. Damping the noise effects, needs more complicated controlling methods which are outside the scope of this project [4].

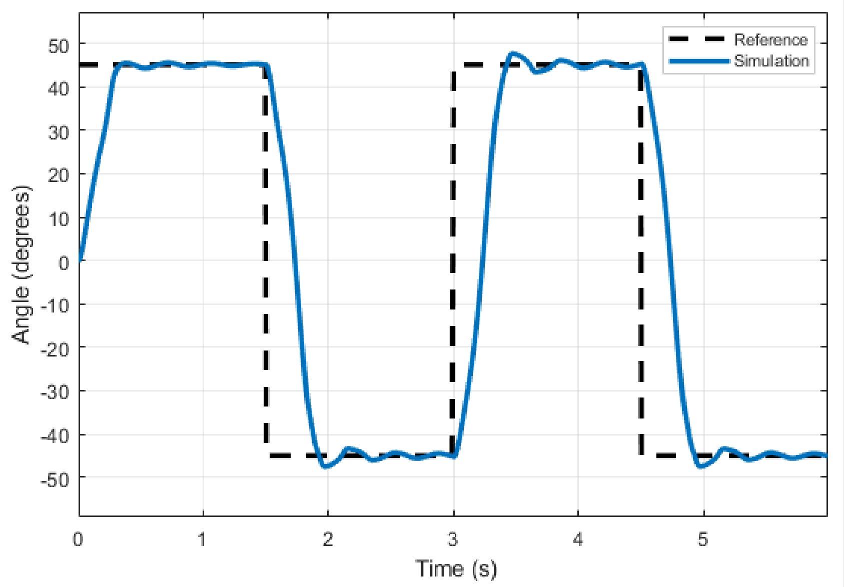

Fig. 6.3 Controlled system's output with pulse input

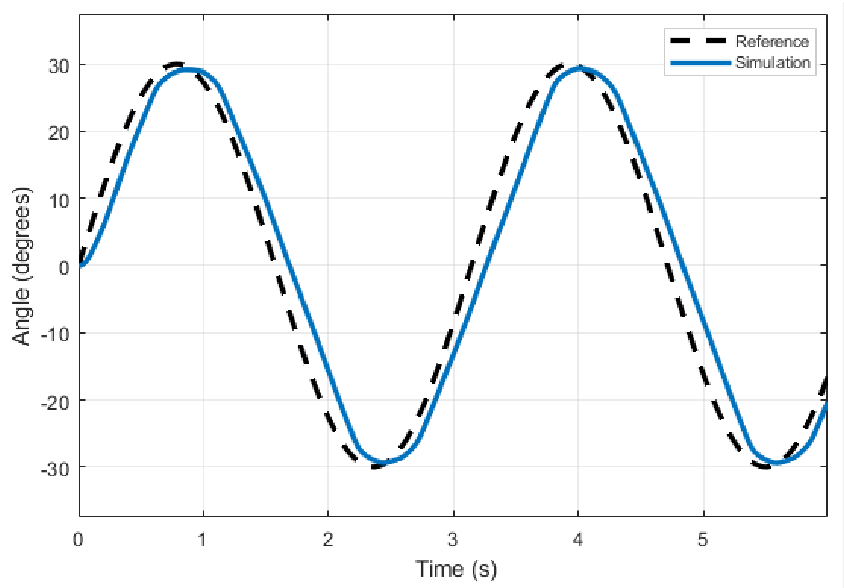

Fig. 6.4 Controlled system's output with sine input

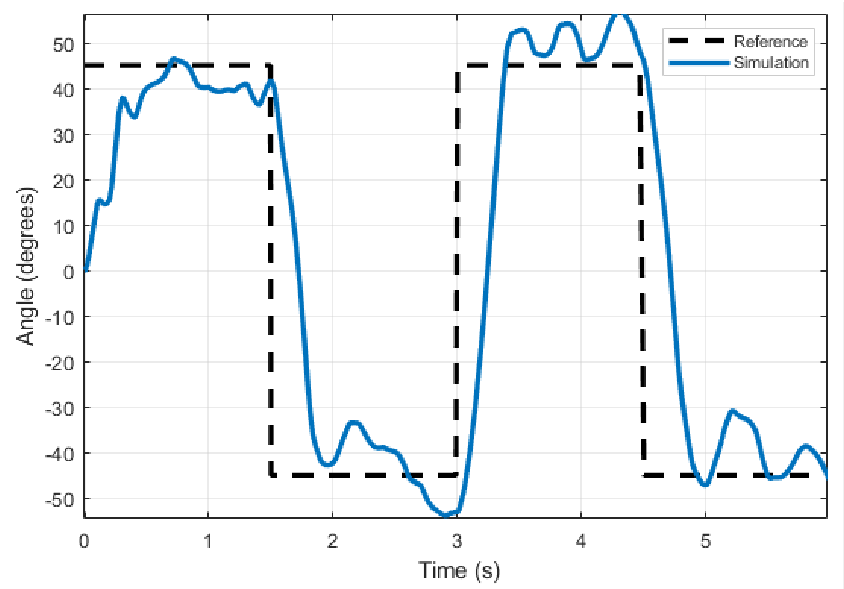

Fig. 6.5 Controlled system's output with pulse input and white Gaussian noise (power=0.005) added to the system's feedback loops

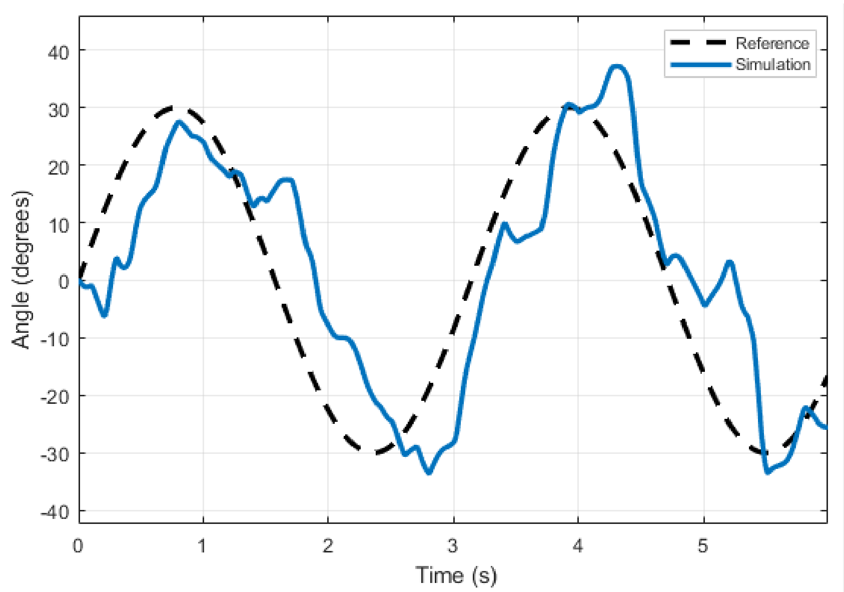

Fig. 6.6 Controlled system's output with sine input and white Gaussian noise (power=0.005) added to the system's feedback loops

Chapter 7

Conclusion

The fuzzy logic controller was applied, designed and tested in the Rotary Flexible Joint Robotic Arm system.

The Gaussian membership function was considered to develop the fuzzy rule. The results showed that fuzzy controller performed satisfactorily to suppress the oscillations. In future, the FLC should be further tuned so that the performances for servo load angle and arm deflection angle become better.

The design may consider different membership function such as triangle and trapezoidal. The Fuzzy Sugeno may also be considered for future control design.

References

[1] T. doersnd o. hammerschmidt, “an adaptive fuzzy control for a three fingered robot gripper,” proceeding of third australian and new zealand conference on intelligent information systems. anziis, pp. 158–163, 1995.

[2] Zadeh and l. a, “fuzzy sets,” information and control, vol. 8, no. 3, pp. 338–353, 1965.

[3] S. thongchai and k. kawamura, “application of fuzzy control to a sonar-based obstacle avoidance mobile robot,” proceeding of the ieee international conference on control applications, pp. 425–430, 2000.

[4] M. rostami kandroodi, m. mansouri, m. aliyari shoorehdeli, and m. teshnehlab, “control of flexible joint manipulator via reduced rulebased fuzzy control with experimental validation,” isrn artificial intelligence, vol. 2012, pp. 1–8, 2012.

[5] Tanaka, k. and yoshioka, k. (1995) design of fuzzy controller for backerupper of a five-trailers and truck, 4th ieee international conference on fuzzy systems, 3, 1543-1548.

[6] Tanaka, k. and sano, m. (1994) a robust stabilization problem of fuzzy controller systems and its applications to backing up control of a trucktrailer, ieee trans. on fuzzy systems, 2(2), 119-134.

[7] Tanaka, k. and sugeno, m. (1990) stability analysis offuzzy systems using lyapunov’s direct method, proceedings of nafips’90, 133-136.

[8] Kitamura, s. and kurozumi, t. (1991) extended circle criterion and stability analysis of fuzzy control systems, proceedings of the international fuzzy eng. symp. ’91, 2, 634-643.

[9] Langari, r. and tomizuka, m. (1990) stability of fuzzy linguistic control systems, proceedings of ieee conference on decision and control, 2185- 2190.

[10] L. a. zadeh and c. a. desoer, linear system theory: The state space approach. mcgraw-hill, 1963.

[11] Pedro albertos and antonio sala, “fuzzy logic controllers advantages and drawbacks,” in ieee transactions on control system, pp. 833–844.