TELEROBOTIC CONTROL OF A MOBILE COORDINATED ROBOTIC SERVER

NAG-1-1283-2

INTERIM TECHNICAL REPORT

Executive Summary

This interim report is comprised primarily of results from the Master's Degree Thesis of Mr. Robert Stanley, a graduate student supervised by the principal investigator on this project. The goal of this effort is to develop advanced control methods for flexible space manipulator systems. As such, a fuzzy logic controller has been developed in which model structure as well as parameter constraints are not required for compensation. A general rule base is formulated using quantized linguistic terms; it is then augmented to a traditional integral control. The resulting hybrid fuzzy controller stabilizes the structure over a broad range of uncertainties, including unknown initial conditions. An off-line tuning approach using phase portraits gives further insight into the algorithm. The approach was applied to a three-degree-of-freedom manipulator system - the prototype of the coordinated flexible manipulator system currently being designed and built at North Carolina State University.
January 7, 1993

NASA Center for Aerospace Information
P. O. Box 8757
Baltimore/Washington International Airport
Maryland 21240

Gentlemen:

Enclosed are two copies of the Interim Technical Report for research performed under NASA Cooperative Research Agreement NAG-1-1283 entitled "Telerobotic Control of a Mobile Coordinated Robotic Servicer."
# TABLE OF CONTENTS

LIST OF TABLES ........................................................................................................... v

LIST OF FIGURES ......................................................................................................... vi 

LIST OF SYMBOLS ....................................................................................................... vii

1. INTRODUCTION & LITERATURE REVIEW .................................................... 1

2. BRIEF OVERVIEW OF FUZZY LOGIC CONTROL .......................................... 4

3. DEVELOPMENT OF THE FUZZY LOGIC ALGORITHM ................................ 7

   3.1 Membership function ..................................................................................... 7

   3.2 Linguistic rules............................................................................................. 10

      3.2.1 Error, change in error, and control input .............................................. 11

      3.2.2 Qualitative linguistic terms ................................................................... 12

   3.3 Rule base ...................................................................................................... 15

      3.3.1 Logic product ..................................................................................... 16

      3.3.2 Logic sum ........................................................................................... 18

      3.3.3 Center of gravity ................................................................................ 21

   3.4 The Fuzzy Control Algorithm ...................................................................... 23

4. ILLUSTRATIVE EXAMPLES ............................................................................ 25

   4.1 The horizontal pendulum ............................................................................. 25

      4.1.1 Number of linguistic terms, universe of discourse, and sign convention .. 26

      4.1.2 Quantization functions ......................................................................... 27

         4.1.2.1 Quantized error .......................................................................... 27

         4.1.2.2 Quantized change in error ............................................................. 29

         4.1.2.3 Dequantized input ........................................................................ 30

      4.1.3 Number of rules ................................................................................... 31

      4.1.4 Population of the rule base ................................................................... 33

      4.1.5 Tuning ................................................................................................. 34

      4.1.6 Varying inertia load ............................................................................. 41

      4.1.7 Time delays .......................................................................................... 43

   4.2 The vertical pendulum .................................................................................. 46

      4.2.1 Application of Fuzzy-PD on the vertical pendulum ......................... 46

      4.2.2 Capture method ................................................................................... 47

      4.2.3 An alternative rule base for FPID ....................................................... 48

      4.2.4 Hybrid Fuzzy-PD and traditional integral control ............................ 50

5. APPLICATION OF FUZZY-PID TO A 3-DOF MANIPULATOR ..................... 52

   5.1 Dynamics ..................................................................................................... 54

   5.2 Traditional PID vs. Fuzzy-PID .................................................................... 57
LIST OF TABLES

4.1 Control parameters for a unit step response .......................................................... 37
LIST OF FIGURES

2.1 Simplified block diagram of fuzzy logic controller ................................................ 4

3.1 Membership function for people of normal height .................................................. 8
3.2 Commonly chosen membership functions .............................................................. 9
3.3 Similarities between some different membership functions ................................... 9
3.4 Linguistic quantized qualitative terms and their respective functions ................. 13
3.5 Qualitative linguistic terms defined on a quantized universe from -6 to +6 .......... 14
3.6 Seven rules used for an inverted pendulum ....................................................... 17
3.7 Logic sum .............................................................................................................. 20
3.8 Final inference produced by the FLC ............................................................... 21
3.9 Center of gravity method ...................................................................................... 22

4.1 Sign convention for error and change in error ...................................................... 27
4.2 Quantized error as a function of error in degrees .................................................. 28
4.3 Quantized change in error angle as a function of change in error angle ............... 30
4.4 Torque as a function of quantized control input .................................................. 31
4.5 Step response of a traditional PD controller vs. Fuzzy-PD ................................... 32
4.6 49 rules used in Fuzzy-Proportional-Derivative controller .................................. 34
4.7 Pictorial representation of the Fuzzy-PD step response ......................................... 36
4.8 Phase portrait for the Fuzzy-PD step response ..................................................... 38
4.9 Effects of varying maximum torque delivered by the FPD controller on the step response and the associated phase portrait ...................................... 40
4.10 Step response as end-point mass varies .............................................................. 42
4.11 Step response as the time delay increases ........................................................... 44
4.12 Step response as the time delay increases ........................................................... 45
4.13 Application of the Fuzzy-PD controller to a traditional vertical pendulum ......... 47
4.14 A candidate rule base for Fuzzy-PID controller ................................................. 48
4.15 Step response of Fuzzy-PID controller with alternative rule base .................... 49
4.16 Step response of traditional PID and Fuzzy-PID ............................................... 50

5.1 (a) DR-106 manipulator under construction at MMRC/NCSU ........................... 53
      (b) Coordinate axes .............................................................................................. 53
5.2 Free vibration of links two and three ................................................................***** 56
5.3 Graphical representation of free vibration of links two and three ....................... 57
5.4 Step response of link one .................................................................................... 58
5.5 Step response of link two .................................................................................... 58
5.6 Negative step response of link three ..................................................................... 59
5.7 Phase portrait of link one ................................................................................... 59
5.8 Phase portrait of link two ................................................................................... 60
5.9 Phase portrait of link three ................................................................................ 60
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( )</td>
<td>vector</td>
</tr>
<tr>
<td>$\theta(t)$</td>
<td>angular position</td>
</tr>
<tr>
<td>$\theta(k)$</td>
<td>angular position at time sample k</td>
</tr>
<tr>
<td>$C_i$</td>
<td>$\cos(\theta_i)$</td>
</tr>
<tr>
<td>$C_{ij}$</td>
<td>$\cos(\theta_i + \theta_j)$</td>
</tr>
<tr>
<td>$\theta_d(t)$</td>
<td>desired angular position</td>
</tr>
<tr>
<td>$\theta_d(k)$</td>
<td>desired angular position at time sample k</td>
</tr>
<tr>
<td>( )''</td>
<td>first time derivative</td>
</tr>
<tr>
<td>( )'''</td>
<td>second time derivative</td>
</tr>
<tr>
<td>$\cap$</td>
<td>intersection</td>
</tr>
<tr>
<td>$\mathbb{R}$</td>
<td>manipulator vector with nonlinear components</td>
</tr>
<tr>
<td>$\Delta t$</td>
<td>sample period</td>
</tr>
<tr>
<td>$S_i$</td>
<td>$\sin(\theta_i)$</td>
</tr>
<tr>
<td>$S_{ij}$</td>
<td>$\sin(\theta_i + \theta_j)$</td>
</tr>
<tr>
<td>( )$ij$</td>
<td>the $i,j$th entry of a matrix</td>
</tr>
<tr>
<td>$\tau$</td>
<td>torque</td>
</tr>
<tr>
<td>$T_{\text{max}}$</td>
<td>torque maximum</td>
</tr>
<tr>
<td>$\tau_{\text{max}}$</td>
<td>torque maximum</td>
</tr>
<tr>
<td>$\tau_{\text{FPD}}$</td>
<td>torque supplied by FPD controller</td>
</tr>
<tr>
<td>$T$</td>
<td>torque vector</td>
</tr>
<tr>
<td>$\cup$</td>
<td>union</td>
</tr>
<tr>
<td>$A$</td>
<td>event</td>
</tr>
</tbody>
</table>
A/D ................................................................. analog to digital
AND ................................................................. intersection
B ................................................................. universe of event
CE ................................................................. change in error
CE(k) ............................................................... change in error at k
CEA ................................................................ change in error angle
DOF ............................................................... degree of freedom
DR-106 .......................................................... robot under construction at MMRC/NCSU
E ................................................................. error
e(k) ................................................................. error at time sample k
e(k-1) .............................................................. error at previous time sample
FLC ............................................................... fuzzy logic control
FPD ............................................................... fuzzy proportional derivative controller
FPID ............................................................. fuzzy proportional integral derivative controller
G ................................................................. proportional gain matrix
g ............................................................... proportional gain
H ................................................................. derivative gain matrix
h ................................................................. derivative gain
I ................................................................. control input
I ................................................................. integral gain matrix
i ................................................................. integral gain
k ................................................................. discrete time sample
L ................................................................. length
LN ............................................................... large negative
LP ............................................................... large positive
M ............................................................... mass matrix
m ........................................................ mass
max ........................................................ maximum
min ........................................................ minimum
MN ........................................................ medium negative
MP ........................................................ medium positive
OR ........................................................ union
PD ........................................................ proportional derivative control
QCE ........................................................ quantized change in error
QCEA ...................................................... quantized change in error angle
QE ........................................................ quantized error
QI ........................................................ quantized input
rad ......................................................... radians
sec ........................................................ seconds
SN ........................................................ small negative
SP ........................................................ small positive
U ........................................................ universe of discourse
u ........................................................ membership value
X(n) ...................................................... \( n \)th state
ZE ........................................................ zero
1. INTRODUCTION

Many future NASA missions require robotics to assist in the assembly, maintenance and servicing of spacecraft. Such scenarios may include one or more multi-linked manipulator arms which, because of their lightly damped characteristics, require vibration suppression as well as end point tracking in a somewhat uncertain environment. Due to the flexibility in the joints/links and the inherent vibration due to the mobility of the robotics system, adaptability to the environment and varying inertia is a requirement.

Several methodologies have been suggested for robot control based upon known tasks and environments. Classical proportional-integral-derivative (PID) control has been employed in industry for many years. The approach assumes complete knowledge of all pertinent system and environmental characteristics. It also requires tuning the PID gains to meet some performance specifications. When the system or environmental parameters change, the gains must be re-tuned accordingly. Thus, unknown disturbances or changing environmental conditions may result in performance degradation.

To address the issue of uncertainty or time-varying conditions, several adaptive control algorithms have been suggested. These include joint-space control [18] and global linearization [4] methods in which some nonlinear or discrete matrix polynomial equation set must be solved in order to construct the controller. While these methods may guarantee stability under certain restrictions, the computation time may limit their implementation for multi-linked robotics systems.

Fuzzy logic control offers an alternative approach in which the structure of the system model is not required for control design [25,2]. Fuzzy control algorithms have been applied to several process control and automotive systems [8,14,13] in which the time constants were somewhat large. The use of fuzzy logic for robotics systems has yielded
some success [16,5] although issues such as time delays and initial conditions sometimes limit the applicability of these algorithms.

This thesis develops a fuzzy logic control algorithm which can be applied to systems with uncertainties. These uncertainties may include unknown initial conditions, and undetermined system dynamics. Unknown initial conditions may exist in space manipulator systems due to sensor inaccuracies.

The concepts of fuzzy logic control are presented in a progressive manner. First, an extensive development concentrating on the theories of fuzzy logic control is presented. Secondly, fuzzy logic control is applied to two simple systems; the first being a horizontal pendulum while the second example is a vertical pendulum. Then the algorithm is applied to a three degree-of-freedom robotic manipulator.

The horizontal pendulum provides an environment with which to develop a FLC that produces performance characteristics similar to traditional proportional derivative control. In the course of this development seven linguistic terms are presented and defined over a quantized Universe of Discourse. The membership function used to define the linguistic terms differs from classical methods. A novel means to quantize the change in error is presented. Utilization of the fully populated rule base results in a general fuzzy logic controller that accomplishes vibration suppression (Fuzzy-PD). By examination of the phase portraits an off-line tuning approach is considered.

The application of Fuzzy-PD to the vertical pendulum provides the groundwork for the development of a fuzzy logic controller that not only accomplishes vibration suppression but also compensates for steady-state errors. In the course of this development, a capture method and an alternative rule base are provided to compensate for system biases. The ultimate result is the hybridization of the Fuzzy-PD controller with traditional integral control resulting in what is referred to as Fuzzy-PID.
At this point the unrestricted fuzzy logic controller (Fuzzy-PID) is applied to the highly coupled, second order, nonlinear dynamics associated with a three degree-of-freedom robotics manipulator. The results of such an application as compared to traditional PID control illustrate the performance of Fuzzy-PID.

Concurrent to this investigation, a coordinated teleoperated mobile manipulator system is being designed and fabricated. The system will contain two DR-106 six-degree of freedom manipulators with flexible joints/links supported by a mobile platform. Several algorithms such as the fuzzy logic controller developed in this thesis are being considered whereby the human operator inputs the desired trajectory and the controller tracks the desired trajectory while suppressing vibration and compensating for platform motion. Such performance measures are typical for in-space robotics operations.
2. BRIEF OVERVIEW OF FUZZY LOGIC CONTROL

Fuzzy logic control is quite confusing when initially introduced. However, like many concepts in life, once a global understanding is obtained the confusion associated with the specifics diminishes. Therefore, before more complicated deliberation on the uses of FLC may be developed, it may be in the readers best interest to consider a brief overview of the fuzzy logic control algorithm (FLC). The block diagram of the fuzzy logic control algorithm is illustrated in Figure 2.1

![Figure 2.1: Simplified block diagram of fuzzy logic controller](image)

Fuzzy logic control is a rule based controller. As the term "rule base" indicates, the fuzzy logic control algorithm is based on a number of rules which are accessed and processed in a specific fashion so as to provide the desired control input to a system. In order to construct such a controller, consideration must be given to the development of the rules and how they interact to form the control input.

Individual rules are constructed using qualitative terms in conjunction with IF...THEN statements. Some examples of common qualitative terms are big, small, large, hot, normal, fast, slow, etc...

A linguistic rule used in the process of balancing a stick may read: IF the stick is inclined moderately to the left AND is almost still THEN move the hand to the left quickly. In order for this rule to be useful a process must be implemented by which the
linguistic terms "moderately", "almost still", and "quickly" are converted into some numeric value. Fuzzy set theory does just that.

Notice how the qualitative linguistic terms are vague in their meaning. This is a desired result because it closely resembles how humans think. The process provided by fuzzy set theory which enables a linguistic term to take on range of values is called the membership function. The membership function is defined over a domain referred to as the Universe of Discourse and assumes a value which ranges from 0 to 1. This value is referred to as the membership value. In general the membership value is a way to weigh how much of a particular linguistic term is present.

How does one determine what is "quick"? For example, if the term "quick" is defined to be 100 mi/hr. its associated membership value would be 1. Any variations, either positive or negative from the speed 100 mi/hr. would result in a membership value of less than 1. For example, 90 m/hr. may correspond to a membership value of .9. Therefore one may conclude that only 90% of the linguistic term "quick" is present.

At this point various rules using linguistic terms in conjunction with IF...THEN statements may be developed. However, because a large number of qualitative linguistic terms exist in the human language it is desirable to choose an appropriate number of linguistic terms and to define what region they are valid.

For control applications the linguistic terms tend to read as Large, Medium, Small, and Zero. By considering both positive and negative values of the linguistic terms listed one has seven distinct qualitative linguistic terms with which to construct rules. (Larger Positive, Medium Positive, Small Positive, Zero, Small Negative, Medium Negative, Large Negative). These linguistic terms must be defined over a region called the Universe of Discourse. It is common practice to define the qualitative terms over a quantized Universe of Discourse. By doing so the qualitative linguistic terms may be used to describe more than one state in a system. Take for example the rule: IF error is
Large Positive AND the rate of error change is Large Positive THEN the control input should be Large Negative. Even though the qualitative linguistic term defining error and error change is the same (Large Positive) the membership value associated with both of the states may differ.

With the concepts used to develop a rule presented and the foregone conclusion that a fuzzy logic controller requires more than one rule to accomplish any reasonable task, a brief discussion of the techniques used to process the rule base follows. The rule base is simply all of the rules created to perform a particular task. Given the states of one such particular task, it is likely that some of the rules will be appropriate to the conditions presented and others will not. The rules that are inappropriate are discarded by the use of the Logic Product and the others are combined using the Logic Sum. The result of this combination of rules is a weighted area. By finding the Center of Gravity of this area a single numeric value results. This value is the final fuzzy inference and is a quantized value. Dequantizing this fuzzy inference results in the final control input.

With this brief overview complete, the reader may proceed further where a more detailed description of fuzzy logic control and its applications to various dynamic systems is presented.
3. DEVELOPMENT OF
THE FUZZY LOGIC CONTROL ALGORITHM

3.1 MEMBERSHIP FUNCTION

The basis for Fuzzy Logic Control (FLC) is the membership function, commonly referred to as the membership shape. A membership function is defined over a domain called the Universe of Discourse (U) and assumes a range from zero to one, referred to as the membership value (u). The universe includes all events that can take place in the context of a particular situation. Restated, the universe exists over the boundaries of a given situation.

The membership value (u) describes the probability of an event occurring, given a particular universe[15]. Probability is defined as the ratio of two numbers. The numerator represents the events in the universe on which interest is focused, and the denominator represents the universe of all possible events. Therefore, the numerator is a subset of the denominator. With this in mind, the probability of an event will range from zero to one indicating from non-membership to total-membership in the universe. Symbolically $u(A|B)$, reads as follows: "the membership value (u) is the probability of event A given the universe of events B," where

$$u(A|B) = \frac{A}{B}$$  \hspace{1cm} \text{Eq. (3.1)}

and

$$0 \leq u(A|B) \leq 1$$  \hspace{1cm} \text{Eq.(3.2)}

FLC is based on a set of heuristic rules. These rules use qualitative words which are defined mathematically in the form of membership functions. To illustrate this
relationship between qualitative words and membership functions consider the following example. Suppose one was to look at the normal height of males. Say for instance that a reasonable normal height is 5 ft. 9 ins. This is not to say that people who are of height of 5 ft. 4 ins. or 6 ft. 2 ins. are not of normal height. Those are normal heights, but somehow the feeling of "normal height" is not as strong. However, at heights of less than 5 ft. 0 ins. or more than 6 ft. 6 ins. one could categorically say this is "short" or this is "tall". Therefore, as one moves from short to tall, the feeling of normal gradually rises and than gradually falls. If values from zero to one are assigned to this feeling of "normal height", the result could be the bell shaped curve shown in Figure 3.1.

![Figure 3.1: Membership function for people of normal height](image)

This bell shaped curve is called the membership function and is defined over the Universe of Discourse (heights ranging between 5 ft. 4 ins. and 6 ft. 2 ins.) to take on values from zero to one, where one is the strongest feeling of "normal". With this concept of membership function defined, one now has the ability to quantitatively deal with inexact or ambiguous issues such as the qualitative words short, normal, tall.
The idea of a membership function differs from a binary approach. The dotted box in Figure 3.1 illustrates binary theory where there are only two membership values: zero and one. According to this, any person in the range of 5 ft. 4 in. to 6 ft. 2 in. takes on the membership value one and is of perfect normal height. Heights less than 5 ft. 4 ins. and greater than 6 ft. 2 ins. correspond to a zero membership value and are thus completely non-normal; clearly this is an unnatural situation.

It has been shown that the exact shape of the membership function is relatively arbitrary and may be chosen based on user preference[9]. Figure 3.2 illustrates some commonly chosen membership functions. Figure 3.3 illustrates how similar they are by superimposing them.

![Bell Shaped](a) ![Trapezoidal](b) ![Triangular](c) ![Sinusoidal](d)

*Figure 3.2: Commonly chosen membership functions*

Due to the complexity of the bell shape (Figure 3.2a) and the piece-wise continuous behavior of the trapezoidal shape and triangular shape (Figure 3.2b & 3.2c), the smooth continuous and easily calculated sinusoidal shape has been selected as the membership function for this study (Figure 3.2d).
3.2 **LINGUISTIC RULES**

To develop a fuzzy controller, it is necessary to interpret linguistic rules that are based on experience so as to form a control surface that provides output values of the controller, corresponding to situations of interest[11]. The basis for the linguistic rule is the "IF...THEN" statement. One linguistic rule or "production rule" describes a portion of a particular problem or task in words. The antecedent blocks ("if" phrases) describe the states, and the consequent block ("then" phrase) describes how the controller should respond to the states.

For example, asking a first shift operator on an assembly line to describe a single portion of his/her task, a typical response may be:

IF the parts are running "far behind" and they "have been" for a period of time THEN I increase the line speed "alot".

This particular response is based on the operator's experience and is to be interpreted to produce a production rule. However, a complication arises when the same question is asked of the second shift line operator. The response may differ in the actual vocabulary used, but the premise would remain the same. Therefore, it is necessary to define a common or universal set of linguistic terms (common vocabulary) which may be used to specifically define the production rule. Refining the operator's response using specific linguistic terms results in a typical fuzzy linguistic rule:

*IF (the error is "large negative") AND (the change in error is "zero") THEN (the control input should be "large positive").*

Notice the change in the antecedents blocks and the consequence block. The term "parts . . . far behind" corresponds to error being *large negative*, while the term "have been for a
period of time" corresponds to change in error being zero. Further, the term "increase...\nor alot" corresponds to a control input large positive.

In general, in order to develop and interpret this rule, or any other fuzzy linguistic rule, the following concepts need to be addressed.

1. How are error, change in error and control input defined?
2. How are the qualitative linguistic terms "large positive", "zero", and "large negative" defined?

### 3.2.1 ERROR, CHANGE IN ERROR, AND CONTROL INPUT

\textit{Error} is defined as the difference between the process output and the desired output:

\[ e(k) = \theta(k) - \theta_d(k) \quad \text{Eq.}(3.3) \]

where

\[ e(k) = \text{error at time sample } k \]
\[ \theta(k) = \text{position at time sample } k \]
\[ \theta_d(k) = \text{desired position at time sample } k \]

The change in error is the difference between the error from the current process output and the error from the last process output.

\[ ce(k) = e(k) - e(k - 1) \quad \text{Eq.}(3.4) \]

where

\[ ce(k) = \text{change in error at current sample} \]
\[ e(k) = \text{error at current sample} \]
\[ e(k - 1) = \text{error at previous sample} \]

All of the examples to follow are maneuvers of mechanical dynamic systems whose dependent variable is an angle \( \theta(t) \) given in either radians or degrees. The control input is the input torque applied to the process.
3.2.2 QUALITATIVE LINGUISTIC TERMS

As illustrated in the line operator example, it is important to develop a set of qualitative linguistic terms to be used in the controller. In the same way the qualitative linguistic term "normal" was defined over the universe of heights ranging from 5 ft. 4 ins. to 6 ft. 2 ins., the linguistic terms for the FLC will span a quantized universe or domain defined from -6 to +6. These limits from -6 to +6 are not hardfast, rather they are chosen such that the individual membership functions begin and end on a whole number. As previously mentioned sinusoidal membership functions will be used to define the linguistic terms. Figure 3.4 illustrates seven such qualitative linguistic terms defined over a quantized Universe of Discourse ranging from -6 to +6 and their respective defining functions: large positive(LP), medium positive(MP), small positive(SP), zero(ZE), small negative(SN), medium negative(MN), and large negative(LN).

The purpose for defining the qualitative terms large-positive through large-negative on a quantized Universe of Discourse is to allow their universal use in defining error, change in error and the control input to the system. This may be accomplished by simply quantizing the values of error, change in error and the control input to the system to the values -6 to +6 on the Universe of Discourse (see Figure 3.5).
Figure 3.4: Linguistic quantized qualitative terms and their respective functions
Figure 3.5: Qualitative linguistic terms defined on a quantized universe from -6 to +6

For example, suppose the measured error and calculated change in error in a particular system after A/D is 22 degrees and 33 degrees/sec, respectively. If the quantization function for error, equals one tenth of the measured error, then the quantized error is 2.2. That is,

$$\text{quantized error} = \left( \frac{6}{60} \right) \times \text{(error)} \quad \text{Eq.(3.5)}$$

Notice in Figure 3.5 that if a vertical line is drawn through the point 2.2, it intersects the membership functions SP and MP. Therefore, the error is a combination of a weight of the membership function Small Positive and a weight of the membership function Medium Positive.

If the quantization function for change in error is defined as

$$\text{quantized change in error} = \left( \frac{6}{90} \right) \times \text{(change in error)} \quad \text{Eq.(3.6)}$$
then the quantized change in error is 2.2. Notice again in Figure 3.5 that this corresponds to some membership value of the linguistic term Small Positive and some membership value of Medium Positive. Collectively, error and change in error have been shown to possess the same quantized value and linguistic values SP and MP while retaining different actual values, thereby demonstrating the transcendental usefulness of the quantized Universe of Discourse.

It has been shown that the number of linguistic terms is arbitrary. As the number of linguistic terms increases, the resolution of the controller increases as a direct result of the induced ability to define each linguistic rule with more accuracy. In most FLC it is common practice to use only three to five linguistic terms. However, due to the complexity associated with the controller for robotics systems, seven linguistic terms are employed.

3.3 RULE BASE

In order to develop a fuzzy logic controller, a series of rules must be assembled. It is the assembly of production rules in which a repertoire of learned problem-solving actions (consequences) is associate with conditions (antecedents), to form condition-action pairs. Once a situation is recognized, the conditions constitute cues or indices for corresponding actions. This is how FLC attempts to model the heuristic problem solving approach of humans[15]. In the assembly line operator example, one production rule governing the case when the error is large negative and change in error is zero was developed. However, in order to handle other cases such as the error being medium positive and the change in error being large positive, one must develop other rules. Therefore, for each particular situation of interest, there exists a corresponding production rule. Combination of the production rules results in what is referred to as a "rule base". In order to assimilate the rule base, the concepts of logic product, logic sum, center of gravity, and
the quantization functions must be developed. An illustrative example of an inverted pendulum is presented in order to mature these concepts.

3.3.1 LOGIC PRODUCT

Figure 3.6 illustrates seven rules which are commonly used for vibration suppression of an inverted pendulum.
Logic product is the first of the concepts to be developed. The physical significance of taking the logic product is to discard any and all rules that are not relevant to a given pair of error and change in error values. In set theory, the logic product is the AND function. It can simply be defined as taking the minimum of the corresponding numeric...
entries of two sets. Mathematically, "the intersection of two sets, A \cap B, corresponds to
the AND function and is define by
\[ u(A \text{ AND } B) = \min(u_a(x), u_b(x)) \] 
Eq.(3.7)

Figure 3.6, demonstrates how the antecedent block or the quantized error and
quantized change in error are joined by the AND function, resulting in a logic product.
The dashed vertical lines in the quantized error (QE) and quantized change in error
(QCE) columns represent a given QE of 3.2 and QCE of .5. Examining the first row of
Figure 3.6, which corresponds to Rule 1, one notices the quantized error membership
value equals 0.8 \((u=0.8)\) and the change in error membership value equals 0.9 \((u=0.9)\).
The minimum of these two values is 0.8, therefore \(u(0.8 \text{ AND } 0.9)=0.8\). The same procedure
is performed for all of the rules as illustrated in Figure 3.6.
\[ \{\min(.8,.9), \min(.6,.4), \min(.6,0), \min(0,.9), \min(0,0), \min(0,.4), \min(0,.9)\} \]
results in the logic product (a set).
\[ \{.8,.4,0,0,0,0,0\} \]
The logic product is used because it provides the condition in which both error and
change in error are satisfied. As shown in rule one (Fig.3.6) the membership value of
\(u=0.8\) satisfies both conditions; therefore it is transferred to the consequence block.

3.3.2 LOGIC SUM
The logic sum is an operator such that the contribution of each individual rule is
combined to produce the final fuzzy inference. Mathematically stated, it is the OR
function. The union of two sets, \(A \cup B\), corresponds to the OR function and is defined
by
\[ u(A \text{ OR } B) = \max(u_a(x), u_b(x)) \] 
Eq.(3.8)
The OR function is applied to the consequence block because even if an individual rule's
influence or contribution is small, it should still be reflected in the resulting control input.
Using the previous values of $QE$ and $QCE$ one can view how the contributions of each rule are transferred to the consequence block.
The logic sum is the OR function applied to the shaded area in the "Control Input" column shown in Figure 3.7. This shaded area is referred to as the conclusion of the fuzzy inference. Figure 3.8 illustrates the result of imposing all of the contributions on the Universe of Discourse.

The conclusion of the fuzzy inference is an area and cannot be used directly to produce a control command. Therefore, a conversion technique is needed to convert the fuzzy inference into a control quantity. The most common means by which to accomplish this goal is the use of the center of gravity method.

![Figure 3.8: Final inference produced by the FLC](image)

### 3.3.3 CENTER OF GRAVITY

The center of gravity method is the most commonly applied way of combining the individual consequences of each rule to get a specific control quantity that may be sent to the process under control. The shaded area in Figure 3.8 is the final inference produced by fuzzy controller. This area, however, cannot be used directly to control the output of
the system. Therefore, the center of gravity of this area is taken. In general the equation for the center of gravity is

\[ QI = \frac{\sum_{i}^{n}(u_{a} \times U_{a})}{\sum_{i}^{n}u_{a}} \]  

Eq.(3.9)

where

- \( QI \) = Quantized Input
- \( u_{a} \) = membership value
- \( U_{a} \) = Universe of discourse

Referring to the final inference illustrated in Figure 3.8, and using Eq.(3.9)

\[ QI = (.8 \times -4) + (.4 \times -2)/(.8 + .4) \]  

Eq.(3.10)

or

\[ QI = -3.3 \]  

Eq.(3.11)

Figure 3.9 illustrates the position of the center of gravity.
The quantized value of the input Q\(_i\)=-3.3 can now be directly related to a control input applied to the system by simply dequantizing it into an applied torque. For example, if the dequantizing function is

\[
\text{Input} = \left( \frac{150}{6} \right) \times (\text{quantized input}) \quad \text{Eq.}(3.12)
\]

then the torque applied to the system

\[
I = \left( \frac{150}{6} \right) \times (-3.3) \quad \text{Eq.}(3.13)
\]

is I=83.33 in-ounces.

This is the defuzzification operation. The method of defuzzification that employs center of gravity is known as the Mamdani method.

3.4 THE FUZZY CONTROL ALGORITHM

The overall general fuzzy control algorithm may now be summarized as follows(Fig. 2.1). First a pair of error and change in error values are measured and calculated respectively(Eqs. 3.3 & 3.4). These two states are then converted into quantized error and quantized change in error(Eqs. 3.5 & 3.6). These quantized values correspond to particular qualitative linguistic terms(Figs. 3.4 & 3.5). The linguistic terms are then applied to the antecedent block of the control rules. If both conditions in the antecedent block are met then a resulting consequence is registered(Eq. 3.7 & Fig. 3.6). All of the individual consequences are then combined by the use of the logic sum(Eq. 3.8 & Fig. 3.7). This results in a final fuzzy inference(Fig 3.8). This final inference may then be converted into a quantized input by application of the center of gravity method(Fig. 3.9 &
Eq. 3.9). The final step is to dequantize the quantized input to the control input to be applied to the system (Eq. 3.12).

The algorithm for FLC just presented is not specific, and it may be applied to a multitude of problems. The intent here, however, is to develop a FLC for robotics systems. Towards this end, two particular examples, a horizontal pendulum and a vertical pendulum, are supplied. These are given in the Chapters 4.
4. ILLUSTRATIVE EXAMPLES

Before applying the fuzzy logic control algorithm developed in Chapter 3 to a robotics example (the coupled nonlinear dynamics of a revolute three degree-of-freedom robot in this study), some physical insight into the behavior of the fuzzy logic controller is desirable. This may be obtained by first applying the fuzzy control algorithm to a simple second-order, linear system and comparing the response to a step input to that of a traditional PD controller. Once this is accomplished, the development of a FLC to handle the slightly more complicated dynamics of a tradition vertical pendulum will be considered.

4.1 THE HORIZONTAL PENDULUM

Consider a massless rod of length (L) in the horizontal plane with a concentrated mass (m) at the endpoint, and an input torque (τ) supplied by a motor. This system is referred to as a horizontal pendulum. The equation of motion for the system is:

\[ \ddot{\theta} - \frac{\tau}{mL^2} = 0 \]  

Eq.(4.1)

where θ(t) is the dependent variable defining the angular position of the pendulum.
When looking at the dynamics of this simple system notice that a traditional proportional derivative (PD) controller is adequate for vibration suppression. This is accomplished by defining the torque as:

\[ \tau = -g(\theta - \theta_e) - h(\dot{\theta} - \dot{\theta}_e) \]

Eq.(4.2)
where g and h are gains chosen to meet desired performance specifications. For this study the selection of the gains is based on the Theory of Natural Control[17].

Substituting Eq.(4.2) into Eq.(4.1) gives the overall closed loop system dynamics.

\[ \ddot{\theta} + h(\dot{\theta} - \dot{\theta}_d) + g(\theta - \theta_d) = 0 \]  

Eq.(4.3)

Consider a specific fuzzy logic controller that provides the same performance characteristics as the traditional PD controller when applied to the horizontal pendulum. For clarification purposes, this fuzzy logic controller is referred to as "Fuzzy-PD" or "FPD". It is noticed in Chapter 3 that in order to develop a FLC, the following need to be defined:

1. The number of qualitative linguistic terms used.
2. The Universe of Discourse.
3. Sign convention on error and change in error.
4. The quantization functions.
5. The number of rules in the rule base.

### 4.1.1 NUMBER OF LINGUISTIC TERMS, UNIVERSE OF DISCOURSE, AND SIGN CONVENTION

Equations 3.3-3.6, and 3.12 were presented in such a way that they are consistent with the Fuzzy-PD controller now under consideration. Therefore, the seven qualitative terms, and the quantized Universe of Discourse used in the previous Chapter will now be applied(Fig.3.4 & Fig. 3.5). It is interesting to note that the error, change in error and input to the system do not have to be quantized to the same seven qualitative linguistic values. For example, it may be decided that only three linguistic terms are needed for an accurate description of error (ex: Positive, Zero, Negative) but five linguistic terms may
be needed for change in error (MN, SN, ZE, SP, MP). This is acceptable when developing a FLC for a particular application. For simplicity, this controller does quantize error, change in error and input to the same seven linguistic values ranging from large positive to large negative.

The sign convention for the error and the change in error are defined for this physical system is shown in Figure 4.1.

![Figure 4.1: Sign convention for error and change in error](image)

4.1.2 QUANTIZATION FUNCTIONS

In Chapter 3 the idea of developing rules based on error, change in error and control input was presented. This approach is now applied to the FPD controller. It was noted in Chapter 3 that quantization functions were required for all three parameters (E, CE, I). (see Eqs. 3.5, 3.6, & 3.12). Therefore, before the Fuzzy-PD rule base can be developed it is important to specifically define how the antecedent and consequence blocks are quantized.

4.1.2.1 QUANTIZED ERROR

In general it is desired to quantize the error and change in error such that the maximum and minimum quantized values correspond to the maximum and minimum actual values. For example, consider the quantization function for the error antecedent.
QE = \left( \frac{6}{\text{maximum expected error}} \right) \times E \quad \text{Eq.(4.4)}

Keeping in mind that the ultimate intent for this thesis study is to develop a FLC to control a 3-DOF revolute manipulator, one notices that the DR-106 (Fig. 5.1) possesses physical limitations pertaining to the working space. The maximum working space for any of the three links is restricted to plus or minus 60 degrees. Therefore, the maximum expected error is set to 60 degrees.

\[ QE = \left( \frac{6}{60} \right) \times E \quad \text{Eq.(4.5)} \]

Figure 4.2 shows the error quantization function in Eq.(4.5). Notice that quantized error is clipped to either +6 or -6 if the error exceeds the expected limits.
4.1.2.2 QUANTIZED CHANGE IN ERROR

The quantization function for change in error is slightly more involved. The change in error is the first derivative of error or the slope of the error curve at a particular time. If the error changes rapidly with respect to time, the CE(slope) approaches infinity. As previously discussed, in order to use CE in the antecedent block it must be quantized. How does one quantize change in error values ranging from negative to positive infinity?

A novel solution to this problem is provided by taking the inverse tangent of the slope of the error with respect to the sampling period. This mapping operation provides a bounded domain for the change in error between negative 90 and positive 90 degrees. With this domain defined on a closed set, it is quite easy to parameterize the change in the error as a change in error angle (CEA).

\[ CEA = \tan^{-1}\left( \frac{e(k) - e(k-1)}{\Delta t} \right) \]  

Eq.(4.6)

where the \(-90 \leq CEA \leq +90\). Using the bounds of CEA and the same premise developed for quantizing error results in quantized change in error angle (QCEA) being defined as:

\[ QCEA = \left( \frac{6}{90} \right) \times CEA \]  

Eq.(4.7)

The graphical representation of this quantization procedure is illustrated in Figure 4.3.
4.1.2.3 DEQUANTIZED INPUT

With the antecedent quantization functions accounted for, the last quantization function to be selected is associated with the input to the system. This process is actually a dequantization process. Given the control input in quantized terms, it is dequantized to a torque value which is then applied to the system (Eq. 4.8). For the example under consideration, the torque varies from +150 to -150 inch-ounces. This number was initially selected and adjusted according to the behavior of the step response. In general:

\[ I = \left( \frac{\tau_{\text{max}}}{6} \right) \times QI \quad \text{Eq.}(4.8) \]

and specifically:

\[ I = \left( \frac{150}{6} \right) \times QI \quad \text{Eq.}(4.9) \]

where
$QI = \text{quantized input}$

$I = \text{control input to the system (in-ounces)}$

This quantization procedure is illustrated in Figure 4.4.

![Figure 4.4: Torque as a function of quantized control input](image)

### 4.1.3 Number of Rules

As a preliminary investigation leading up to the final Fuzzy-PD controller, seven rules were applied to the horizontal pendulum. This resulted in a poor performance associated with the controller's lack of ability to handle various wide ranges in initial conditions. To compensate for this poor performance, an increase in the rule base to 36 rules followed. This control surface also failed due to a lack of robustness.

Given the fact that there are only seven different values of $QE$ and $QCEA$, one concludes that there are a total of 49 different possible combinations, corresponding to 49 distinct rules. Therefore, considering the poor performance of the 36 rules, it was decided to use a control surface fully populated with all 49 different rules. This final approach produced acceptable behavior and did not hinder the computational time.
associated with the fuzzy logic controller. This is due to the fact that, given any pair of error and change in error values, only four of the 49 rules are applied at that given instance.

The step responses of the horizontal pendulum controlled by both a tradition PD controller and the Fuzzy-PD controller are illustrated in Figure 4.5. Both systems possess a similar rise times but the Fuzzy-PD controller has considerably less overshoot. This is due to the fact that the controller is based on humanistic rules or the rule base as opposed to a mathematical function governed by the damping envelope.

Since the system is second order, the PD gains could be specifically chosen to produce an identical step response to that demonstrated by the Fuzzy-PD controller. However, in order to provide an objective test environment, the maximum torque that either controller could apply to the system was set to 15 in-ounces. With this in mind Fuzzy-PD out performed the traditional PD controller.

![Figure 4.5: Step response of a traditional PD controller vs. Fuzzy-PD](image)
With a working Fuzzy-PD controller now fully matured, consideration is given to how the individual rules can be formulated and how the Fuzzy-PD controller can be tuned.

### 4.1.4 POPULATION OF THE RULE BASE

The rule base was populated by simply allowing the quantized change in error angle to be set to zero and looking at how the system behaves as a function of only quantized error. For example, if the quantized error is "large positive" and the quantized change in error angle is "zero", then the quantized input to the system should be "large negative". This rule may be viewed graphically in Figure 4.6 where it is labeled as rule 4. The complement of rule 4 may be written as: If quantized error is "large negative" and the quantized change in error angle is "zero", then the quantized input to the system should be "large positive" (Fig. 4.6, rule 46). The intermediate rules may be found by interpolating between these two points. Extruding this slope, both in the positive and negative directions of the CEA axis results in an unacceptable controller surface. Therefore, the same technique may be applied to obtain the rules along the CEA axis. With both the error and change in error axes defined, one may then interpolate across all 49 different rules.

Looking at the quantized control input to the system as a function of quantized error and quantized change-in-error-angle, one may plot the surface as in Figure 4.6.

\[ QI = 3(QE, QCEA) \]  
Eq.(4.10)
Interpolating between the discrete values produces a smooth fuzzy logic control surface.

4.1.5 TUNING

The second topic to be elaborated on before development of the more complicated Fuzzy-PID controller is a tuning scheme. There are numerous ways to tune a fuzzy controller. First, and most obvious, the rule base itself may be altered. Secondly, the
shape of the membership function may be altered (sinusoidal, bell, trapezoid, etc...).

Also, the overlap of the membership functions may be altered. And lastly, the quantizing schemes or functions may be also changed. Assuming that the rule base, the membership function, and the overlap of the membership function have been chosen appropriately, the only method of tuning seems to be to change the quantizing functions.

However, when trying to tune the quantizing functions, a problem arises. This problem is associated with the fact that the control input is a function of three quantized terms: the error, change in error angle, and the input. With no guidance governing the relationship between these three quantized values, it may be just as effective to randomly choose quantizing functions.

If two of the three quantization functions were constrained to certain limits, then there would remain only one independent variable to alter. Since, the error is bounded (between +60 and -60 degrees) and the change in error angle is bounded (the inverse tangent of the slope of the error curve lies between +90 and -90 degrees), the quantized control input-to-torque relationship remains the only quantization function that may be altered. Therefore, it stands to reason that the quantized input to the system is the only choice with which to tune the fuzzy controller.

As an off-line approach to tuning, the effect of varying the dequantization function for QI can be examined through a combination of the resulting step response in conjunction with its respective phase portrait. Figure 4.7 shows the system at eleven steps in time. From this, the quantized error and quantized CEA can be tabulated as in Table 4.1. A plot of the quantized error versus the quantized change in error angle may then be obtained; this is the phase portrait. Figure 4.8 illustrates the phase portrait for the step response illustrated in Figure 4.5 and 4.7.
Figure 4.7: Pictorial representation of the Fuzzy-PD step response
**Table 4.1: Control parameters for a unit step response**

<table>
<thead>
<tr>
<th>Time Step</th>
<th>E</th>
<th>CEA</th>
<th>QE</th>
<th>QCEA</th>
<th>QI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-57</td>
<td>0</td>
<td>-5.7</td>
<td>0</td>
<td>5.6</td>
</tr>
<tr>
<td>1</td>
<td>-54</td>
<td>24</td>
<td>-5.4</td>
<td>1.6</td>
<td>3.9</td>
</tr>
<tr>
<td>2</td>
<td>-47</td>
<td>50</td>
<td>-4.7</td>
<td>3.3</td>
<td>1.7</td>
</tr>
<tr>
<td>3</td>
<td>-38</td>
<td>58</td>
<td>-3.8</td>
<td>3.9</td>
<td>.0</td>
</tr>
<tr>
<td>4</td>
<td>-28</td>
<td>60</td>
<td>-2.8</td>
<td>4.1</td>
<td>-1.2</td>
</tr>
<tr>
<td>5</td>
<td>-18</td>
<td>59</td>
<td>-1.8</td>
<td>4.0</td>
<td>-2.0</td>
</tr>
<tr>
<td>6</td>
<td>-10</td>
<td>54</td>
<td>-1.0</td>
<td>3.6</td>
<td>-2.1</td>
</tr>
<tr>
<td>7</td>
<td>-4</td>
<td>47</td>
<td>-.4</td>
<td>3.2</td>
<td>-2.1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>36</td>
<td>.3</td>
<td>2.5</td>
<td>-1.1</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>28</td>
<td>.3</td>
<td>1.3</td>
<td>-.0</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>17</td>
<td>.5</td>
<td>.3</td>
<td>.1</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>0</td>
<td>.5</td>
<td>-.4</td>
<td>0</td>
</tr>
</tbody>
</table>
Figure 4.8: Phase portrait for the Fuzzy-PD step response

By varying the maximum torque ($\tau_{\text{max}}$) in the dequantization function (see Eq. 4.8) between the values 2.5, 15, and 80 in-oz, one obtains three different time responses and three different phase portraits associated with these time responses. Figure 4.9(b) shows a phase portrait of an underdamped system when a maximum torque of 2.5 in-oz is applied. The corresponding time response of this system is given in Figure 4.9(a).

Figure 4.9(d) is a plot of a phase portrait when a maximum torque of 15 in. ounces is applied to the horizontal pendulum. The associated time response is illustrated in Figure 4.9(c). It may be noted that the phase portrait illustrated in Figure 4.9(d) corresponds to a quick rise time with a minimal amount of peak overshoot. Therefore, this phase portrait is the desired curve.
Figure 4.9(f) is a phase portrait for a maximum torque value of 80 in. ounces and Figure 4.9(e) is its corresponding time response. This particular system is over responsive. These three phase portraits and their corresponding time responses are critical to accomplishing tuning of the complicated dynamics associated with the three-link revolute manipulator.
Figure 4.9: Effect of varying maximum torque delivered by the FPD controller on the step response and the associated phase portrait.
4.1.6 VARYING INERTIA LOAD

Due to the varying task requirements associated with on orbit assemble issues it is of interest to investigate how the Fuzzy-PD controller compares to traditional PD control when the end-point mass varies. Figure 4.10 illustrates the step response of the horizontal pendulum when controlled by both Fuzzy-PD and traditional PD. Figure 4.10(a) illustrates the response when both controllers are tuned properly. Figures 4.10(b)-4.10(d) demonstrates that both controllers are approximately producing the same control response for a given situation. This is expected because the fuzzy rules discretely approximate the PD control surface.
Figure 4.10: Step response as end-point mass varies
4.1.7 TIME DELAYS

Another issue of particular interest is that of time delays. By tuning Fuzzy-PD and PD to the same performance characteristics and then introducing time delays a comparison of the robustness of each controller may be made. Figures 4.11 and 4.12 illustrate the performance degradation associated with increasing the delay time. Figure 4.11(a) is the step response without any time delays. Figure 4.11(b) demonstrates that both controllers behave approximately the same with a time delay of one sample period (the sample period used here was .05 sec.). At a time delay of two sampling periods the Fuzzy-PD controller becomes marginally stable while the PD controller performs with slight indifference(Fig. 4.11(c)). Figure 4.12 illustrates that it is not until a five sampling period time delay that the PD controller becomes marginally stable. Figure 4.12(c) illustrates that as the time delay increase past 4 the system remains marginally stable with an increase in amplitude and a slower frequency.

The results of introducing time delays indicate that Fuzzy-PD control does not perform as well as traditional PD control. This is also an expected result due to the fact that the traditional PD controller is continuous while the Fuzzy-PD controller possesses a discrete number of rules.
Figure 4.11: Step response as the time delay increases.
Figure 4.12: Step response as the time delay increases.
4.2 THE VERTICAL PENDULUM

As a second example, consider applying the Fuzzy-PD controller just developed to a second-order nonlinear system with a constant system bias. In particular, consider a traditional vertical pendulum. This pendulum possesses two equilibrium points. However, it does not operate about either one, thereby maintaining a constant system bias due to gravity. The dynamics associated with the pendulum are:

\[ \ddot{\theta} + \frac{g}{L} \sin \theta = \tau \quad \text{Eq.}(4.11) \]

As in the first illustration, to implement traditional proportional integral derivative control, consider applying the Theory of Natural Control\[17\], where,

\[ \tau = -g(\theta - \theta_d) - h(\dot{\theta} - \dot{\theta}_d) - i \int_0^t (\theta - \theta_d) ds \quad \text{Eq.}(4.12) \]

Linearizing Eq.(4.11), substituting equation (4.12) into the linearized version of Eq.(4.11) and solving for the desired gains \( g, h, \) and \( i \) results in a PID controller which may then be applied to the nonlinear dynamics of Eq.(4.11).

NOTE: The term Fuzzy-PID represents the fuzzy controller that when applied to a system behaves like a tuned traditional PID controller.

4.2.1 APPLICATION OF FUZZY-PD ON THE VERTICAL PENDULUM

The first attempt at controlling this system using fuzzy logic is to apply the fuzzy controller developed for the horizontal pendulum to the dynamics of the vertical pendulum. The results of such an approach is illustrated in Figure 4.13 (the curve labeled
"course"). The desired position in this case is one and the position obtained is approximately 0.425; therefore, there is a steady-state error.

![Figure 4.13](image)

**Figure 4.13: Application of the Fuzzy-PD controller to a traditional vertical pendulum**

### 4.2.2 CAPTURE METHOD

In order to compensate for this steady-state error, a capture method was implemented. This capture method consists of redefining the quantized error function every time the error of the system falls within certain limits.

For example, if the error was originally quantized to a maximum value of plus or minus 60 degrees (coarse), when the error lies between $\pm 30^\circ$ the function could be redefined to have bounds of $\pm 30^\circ$; this would be the medium quantization function. The same could be done for a fine quantization function by changing the bounds to $\pm 15^\circ$. Figure 4.13 illustrates how this method succeeds in reducing the steady state error but never accomplishes the ultimate goal of no steady state error.
4.2.3 **AN ALTERNATIVE RULE BASE FOR FPID**

Due to the inability of the Fuzzy-PD controller to deal with the steady-state error, a second method using an altered rule base was developed. Figure 4.14 illustrates the new rule base.

![Figure 4.14: A candidate rule base for a Fuzzy-PID controller](image-url)

*Figure 4.14: A candidate rule base for a Fuzzy-PID controller*
The motivation for defining this new rule base came from observing the phase portraits of the time responses illustrated in Figure 4.13. It was noticed in these phase portraits that the portion of the original 49 rule base that compensated for overshoot was producing a steady-state error in the time response.

Therefore, the rules that compensated for overshoot were eliminated; thus the rule base illustrated in Figure 4.14 was assembled. The time response of the vertical pendulum to the new rules is illustrated in Figure 4.15.

![Figure 4.15](image)

*Figure 4.15: Step response of Fuzzy-PID controller with an alternative rule base*

It is noticed that this system has a quick rise time; however once the system reaches its desired position, the control input produces no torque. This results in a rapid drop in the vertical pendulum back towards its equilibrium position. This cycle repeats itself ultimately resulting in a marginally stable system. Therefore, the method of altering the rule base to compensate for system biases fails and another method is needed.
4.2.4 HYBRID FUZZY-PD AND TRADITIONAL INTEGRAL CONTROL

In order to compensate for steady-state errors, a fuzzy integral control parameter may be augmented to the existing Fuzzy-PD controller. Due to the complexity associated with defining the necessary 343 rules, a hybrid controller was designed. This hybrid controller uses fuzzy proportional-derivative control (FPD) and classical integral control to produce a fuzzy-proportional-integral-derivative controller (FPID).

\[ \tau = \tau_{FPD} + i \int_{0}^{t} (\theta - \theta_d) \, ds \quad \text{Eq.(4.13)} \]

The Fuzzy-PD controller handles the vibration suppression as demonstrated earlier and the integral term compensates for any system bias.

The time responses of the vertical pendulum the Fuzzy-PID controller and traditional PID control are shown in Figure 4.16.

![Figure 4.16: Step response of traditional PID and Fuzzy-PID](image-url)
This hybrid Fuzzy-PID controller not only suppresses vibration but also compensates for steady-state error. The time response of the Fuzzy-PID controller compared to that of a tuned traditional PID controller is hardly distinguishable. With the hybrid Fuzzy-PID controller now developed, the approach can be applied to robotics systems. The dynamics of the 3-link manipulator are investigated in the next chapter.
5. APPLICATION OF FUZZY-PID TO A 3-DOF MANIPULATOR

With the hybrid fuzzy controller developed, its applicability to a robotics system can be investigated. A telerobotic flexible manipulator system is currently being developed at the Mars Mission Research Center in order to investigate several control algorithms for real-time implementation. A model of one of the robotic manipulator arms is selected for preliminary studies of the fuzzy control algorithm.

Figure 5.1(a) is a drawing illustrating the (DR-106) three-degree-of-freedom revolute manipulator being constructed at North Carolina State University. Figure 5.1(b) illustrates the coordinate axis definition.
Figure 5.1: (a) DR-106 manipulator under construction at MMRC / NCSU

(b) Coordinate axes
5.1 DYNAMICS

The nonlinear dynamics for the three-degree-of-freedom revolute manipulator can be developed using the Lagrangian approach[22]. This results in

\[
\begin{bmatrix}
26 + 8C_{23}^2 + 29C_{23}^2 + 24C_{23}C_{32} & 0 & 0 \\
0 & 43 + 24C_3 & 8 + 12C_3 \\
0 & 8 + 12C_3 & 8
\end{bmatrix}
\begin{bmatrix}
\ddot{\theta}_1 \\
\ddot{\theta}_2 \\
\ddot{\theta}_3
\end{bmatrix}
= \begin{bmatrix}
(16C_{23}S_{23} + 24C_{23}S_{23})\dot{\theta}_1(\dot{\theta}_2 + \dot{\theta}_3) + (24S_{23}C_{23} + 58S_2C_3)\dot{\theta}_1\dot{\theta}_2 \\
-(8C_{23}S_{23} + 12S_2C_{23} + 12C_{23}S_2 + 29C_2S_2)\dot{\theta}_1^2 + 24S_2\dot{\theta}_2\dot{\theta}_3 + 12S_3\dot{\theta}_3^2 - 20gC_2 - 6gC_{23} \\
-(8C_{23}S_{23} + 12C_2S_{23})\dot{\theta}_1^2 - 12S_3\dot{\theta}_2^2 - 6gC_{23}
\end{bmatrix}
\]

Eq.(5.1)

Eq.(5.1) may be written more compactly as

\[
M\ddot{\theta} = R + T
\]

Eq.(5.2)

where

- \(M\) = mass matrix
- \(\ddot{\theta}\) = acceleration vector
- \(R\) = vector containing nonlinear terms
- \(T\) = torque vector

Premultiplying Eq.(5.2) by \(M^{-1}\)

\[
\ddot{\theta} = M^{-1}R + M^{-1}T
\]

Eq.(5.3)

Equation (5.3) may now be written in state space form and numerically integrated to find \(\theta(t)\).
To apply proportional-integral-derivative control to the system, the appropriate relationship between the torque vector and the position vector must be found. Once again the methods developed in Natural Control Theory [17] will be utilized. First, make a linear approximation by simply dropping the nonlinear terms in Eq. (5.2). This results in

$$M\ddot{\theta} = T \quad \text{Eq.(5.4)}$$

Secondly, assume

$$T = -G\dot{\theta} - H\dot{\theta} - I\int_0^t \dot{\theta} dt \quad \text{Eq.(5.5)}$$

where \(G, H,\) and \(I\) are control gain matrices. Letting

\[
G = gM \quad H = hM \quad I = iM
\]

Eqs.(5.6)

Substitution of Equations (5.6) into Eq.(5.5) results in the following torque vector:

\[
\begin{bmatrix}
\tau_1 \\
\tau_2 \\
\tau_3 
\end{bmatrix} = -
\begin{bmatrix}
g_{m_{11}}x(2) + h_{m_{11}}x(3) \\
g[m_{22}x(5) + m_{23}x(8)] + h[m_{22}x(6) + m_{23}x(9)] + i[m_{22}x(4) + m_{23}x(7)] \\
g[m_{23}x(5) + m_{33}x(8)] + h[m_{23}x(6) + m_{33}x(9)] + i[m_{23}x(4) + m_{33}x(7)]
\end{bmatrix} \quad \text{Eq.(5.7)}
\]

where \(x(i)\) is the \(i\)th state defined as:

\[
x(1) = \int_0^t (\dot{\theta}_i - \dot{\theta}_{id}) dt \\
x(2) = \dot{\theta}_i - \dot{\theta}_{id} \\
x(3) = \dot{\theta}_i
\]

Eq.(5.8)
\[ x(4) = \int_0^t \theta_2 - \theta_{2d} \, ds \]
\[ x(5) = \theta_2 - \theta_{2d} \]
\[ x(6) = \dot{\theta}_2 \]
\[ x(7) = \int_0^t \theta_3 - \theta_{3d} \, ds \]
\[ x(8) = \theta_3 - \theta_{3d} \]
\[ x(9) = \dot{\theta}_3 \]

Eq.(5.9)

Eq.(5.10)

Figure 5.2 illustrates how the dynamics of the system behaves. With no torque applied the links vibrate freely. Figure 5.3 provides a graphical representation of how links 2 and 3 behave when the links are released from a horizontal position and allowed to move freely. Gravity is acting in this system and the two links oscillate about their respective equilibrium points.

Figure 5.2: Free vibration of links 2 and 3
5.2 TRADITIONAL PID VS. FUZZY-PID

The fuzzy-PID hybrid controller applied to the vertical pendulum will now be applied to this manipulator. Each link of the manipulator is independently controlled by a separate fuzzy logic control algorithm. Each of the three links are subjected to a unit step forcing function; links 1 and 2 have positive unit steps and link 3 has a negative unit step. Figure 5.4 shows the response of the first link using a FPIID hybrid and a traditional PID controller. The step response shown in this figure is the robot’s maneuver in the horizontal plane of rotation.

Figure 5.3: Graphical representation of the free vibration of links 2 and 3
Figure 5.4: Step response of link one

Figure 5.5 illustrates the response of the second link of the manipulator. This graph illustrates a slower rise time for the fuzzy-PID controller as compared to the traditional PID. However, the fuzzy-PID controller has considerably less overshoot than PID.

Figure 5.5: Step response of link two
Figure 5.6 shows the response of the robot's link 3. This graph is the most dramatic of the three links in terms of the difference between PID and FPID. The fuzzy logic controller not only produces a quicker rise time but also exhibits hardly any overshoot as compared to the traditional PID controller.

![Graph showing response of link 3](image)

*Figure 5.6: Negative step response of link three*

The phase portraits associated with these step maneuvers are Figures 5.7, 5.8 and 5.9, respectively. Collectively these figures illustrate that the fuzzy-PID control provides a better or equivalent time response than classical PID control.

![Phase portrait of link one](image)

*Figure 5.7: Phase portrait of link one*
Figure 5.8: Phase portrait of link two

Figure 5.9: Phase portrait of link three
6. CONCLUSION AND SUGGESTIONS FOR FUTURE WORK

This thesis presented a fuzzy-PD controller made up of 49 rules with a sinusoidal membership function. This general fuzzy-PD controller was then augmented with traditional integral control to produce a fuzzy-PID controller. The fuzzy-PID controller was then used to control nonlinear robotics models. Fuzzy-PID showed promise when compared to traditional PID due to PID's requirement of a model and the complexity associated with developing the gains. Because robotics models, including space robotics systems, contain uncertainties, exact model-based controllers are difficult to implement; hence the fuzzy approach may be more appealing.

Precise response characteristics using fuzzy controllers may be difficult, however. In [5] Cela and Hamam present some stability issues associated with fuzzy control systems. Although exact tuning of such systems may be difficult, this study has shown that this process is less difficult than PID gain tuning. Phase portraits provide a feasible off-line tuning approach.

The use of a fully populated quantized error and quantized change in error angle rule base provides a more effective controller while not hindering the processing time of the controller. Further, utilization of the inverse target function provides a novel means with which to bound the quantized change in error angle term.

Several issues may be considered as future activities, in order to extend the technique to space robotics systems and in particular, the ground testbed being constructed at North Carolina State University. First, fuzzy logic control may be constructed in such a fashion so as to tune a traditional PID controller. This approach would result in an adaptive controller that could compensate for varying inertia loads, and time delays; ultimately resulting in an ideal candidate for space applications.
Secondly, other calibration techniques including on-line methods are being considered. These methods include requantizing other variables besides the control input. Such extensions will be investigated and compared to other control methods being developed and implemented for space robotics systems.
7. REFERENCES


8. APPENDICES

8.1 PROPORTIONAL-INTEGRAL-DERIVATIVE CONTROL ON 3-DOF MANIPULATOR

PROGRAM PID ON ROBOT

PURPOSE: To apply PID control to the nonlinear dynamics of a three link microbot.

AUTHOR: Robert J. Stanley II

DATE: 8/5/92

VARIABLES:

T#: Torque applied to respective links
Td#: The desired angle of each link
NEQ: Number of equations
NSTEP: Number of times runga-kutta subroutine is called
DT: Time interval \( \delta T \) in rad/sec
TIME: Independent variable
X(I): Dependent variable
F(I): State equations

INTEGER COUNT, NEQ, NSTEP
REAL*8 X(10), DT, TIME, T1, T2, T3, Td1, Td2, Td3
COMMON/MYCOMM/Td1, Td2, Td3

OPEN (8, FILE='pidr.dat', STATUS='unknown')
OPEN (9, FILE='pidrT.dat', STATUS='unknown')
OPEN (8, FILE='PIDR1G.dat', STATUS='unknown')
OPEN (9, FILE='PIDR2G.dat', STATUS='unknown')

15 FORMAT(1X, F6.2, 1X, F8.4, 1X, F10.4, 1X, F8.4, 1X, F10.4, 1X, F8.4, 1X, F10.4)
25 FORMAT(1X, F6.2, 1X, F10.4, 1X, F10.4, 1X, F10.4)

NEQ=9
DT=.01D0
NSTEP=1000
TIME=0.0D0
X(1)=0.0
X(2)=-1.0
X(3)=0.0
X(4)=0.0
X(5)=-1.0
X(6)=0.0
X(7)=0.0
X(8)=1.0
X(9)=0.0
Td1 = 1.0
Td2 = 1.0
Td3 = -1.0

CALL TORQUE(X,T1,T2,T3)
WRITE(8,15)TIME,X(2)+Td1,T1,X(5)+Td2,T2,X(8)+Td3,T3
WRITE(9,25)TIME,X(2)+Td1,X(5)+Td2,X(8)+Td3

DO 100 COUNT=1,NSTEP
   CALL RUNGA(X,DT,NEQ,TIME)
   CALL TORQUE(X,T1,T2,T3)
   WRITE(8,15)TIME,X(2)+Td1,T1,X(5)+Td2,T2,X(8)+Td3,T3
   WRITE(9,25)TIME,X(2)+Td1,X(5)+Td2,X(8)+Td3
100 CONTINUE
CLOSE(8)
END

SUBROUTINE TORQUE(X,T1,T2,T3)

REAL*8 X(10),Td1,Td2,Td3
REAL*8 T1,T2,T3,M11,M22,M33,M23,M23,C23,S23,C2,C3,S2
REAL*8 S3,S23S23,S2C23,S2C23,S2C2,S2C2
REAL*8 H1,H2,H3,G1,G2,G3,I1,I2,I3,ALPHA,BETA
COMMON /MYCOMM/Td1,Td2,Td3

ALPHA=2.46051702
BETA=3.14
\begin{verbatim}
G1=3*ALPHA**2+BETA**2
H1=3*ALPHA
I1=0.0
G2=3*ALPHA**2+BETA**2
H2=3*ALPHA
I2=ALPHA*(ALPHA**2+BETA**2)
G3=3*ALPHA**2+BETA**2
H3=3*ALPHA
I3=ALPHA*(ALPHA**2+BETA**2)

C23=COS(X(5)+Td2+X(8)+Td3)
S23=SIN(X(5)+Td2+X(8)+Td3)
C2=COS(X(5)+Td2)
C3=COS(X(8)+Td3)
S2=SIN(X(5)+Td2)
S3=SIN(X(8)+Td3)
C23S23=C23*S23
C2S23=C2*S23
S2C23=S2*C23
C2S2=C2*S2

M11=26.+8.**2*2+29.*C2**2+24.*C2*C23
M22=43.+24.*C3
M23=8.+12.*C3
M33=8.

T1=-1*(G1*M11*X(2)+H1*M11*X(3)+I1*M11*X(1)) T2=-
1*(G2*(M22*X(5)+M23*X(8))+H2*(M22*X(6)+M23*X(9))
++I2*(M22*X(4)+M23*X(7))) T3=-
1*(G3*(M23*X(5)+M33*X(8))+H3*(M23*X(6)+M33*X(9))
++I3*(M23*X(4)+M33*X(7)))
RETURN
END

SUBROUTINE RIGHT(R1,R2,R3,X,M11,M22,M33,M23,DET)
PURPOSE: To calculate the respective mass matrix entries and
the respective nonlinear contributions R#.

AUTHOR: Robert J. Stanley II
DATE: 8/5/92

VARIABLES:
R#: The nonlinear terms of link # respectively
F: State space
G: Gravity
DET: The determinate of the mass matrix divided by M11

RETURN
END
\end{verbatim}
REAL*8 X(10),F(10),TIME,Td1,Td2,Td3
REAL*8 T1,T2,T3,R1,R2,R3,M11,M22,M33,M23,G,C23,S23,C2,C3,S2
REAL*8 S3,C23S23,C2S23,S2C3,S2C2,C2S23,DET
COMMON /MYCOMM/ Td1,Td2,Td3

G=9.8
C23=COS(X(5)+Td2+X(8)+Td3)
S23=SIN(X(5)+Td2+X(8)+Td3)
C2=COS(X(5)+Td2)
C3=COS(X(8)+Td3)
S2=SIN(X(5)+Td2)
S3=SIN(X(8)+Td3)
C23S23=C23*S23
C2S23=C2*S23
S2C23=S2*C23
S2C2=S2*C2
C2S2=C2*S2

M11=26.+8.*C23**2+29.*C2**2+24.*C2*C23
M22=43.+24.*C3
M23=8.+12.*C3
M33=8.
DET=M22*M33-M23*M23

CALL TORQUE(X,T1,T2,T3)
R1=(16.*C23S23+24.*C2S23)*X(3)*(X(6)+X(9))
++(24.*S2C23+58.*S2C2)*X(3)*X(6)+T1
R2=-(8.*C23S23+12.*S2C23+12.*C2S23+29.*C2S2)*X(3)**2
++24.*S3*X(6)*X(9)+12.*S3*X(9)**2-20.*G*C2-6.*G*C23+T2
R3=-(8.*C23S23+12.*C2S23)*X(3)**2-12.*S3*X(6)**2-6.*G*C23+T3
RETURN
END

SUBROUTINE STATE(F,X,TIME)

PURPOSE: To compute the present state of the dynamic system.

AUTHOR: Robert J. Stanley II

DATE: 8/5/92

VARIABLES:

All variables already defined.

REAL*8 X(10),F(10),TIME,Td1,Td2,Td3
REAL*8 R1,R2,R3,M11,M22,M33,M23,DET
COMMON /MYCOMM/ Td1,Td2,Td3
TIME = TIME * 1.0

CALL RIGHT(R1, R2, R3, X, M11, M22, M33, M23, DET)
F(1) = X(2)
F(2) = X(3)
F(3) = R1 / M11
F(4) = X(5)
F(5) = X(6)
F(6) = (R2 * M33 / DET) - (R3 * M23 / DET)
F(7) = X(8)
F(8) = X(9)
F(9) = -(R2 * M23 / DET) + (R3 * M22 / DET)

RETURN
END

SUBROUTINE RUNGA(X, DT, NEQ, TIME)

PURPOSE: Use a Runge Kutta routine to compute the next state vector

AUTHOR: Robert J. Stanley II

DATE: 8/5/92

VARIABLES:
G#: Variable gains

REAL*8 X(10), Y(10), F(10), DT, TIME, G1(10), G2(10), G3(10), G4(10)
INTEGER I, NEQ

DO 1 I = 1, NEQ
  Y(I) = X(I)
1  CALL STATE(F, Y, TIME)

DO 2 I = 1, NEQ
  G1(I) = DT * F(I)
2  TIME = TIME + DT / 2.0D0

DO 3 I = 1, NEQ
  Y(I) = X(I) + G1(I) / 2.0D0
3  CALL STATE(F, Y, TIME)

DO 4 I = 1, NEQ
  G2(I) = DT * F(I)
4  Y(I) = X(I) + G2(I) / 2.0D0
  CALL STATE(F, Y, TIME)

DO 5 I = 1, NEQ
  G3(I) = DT * F(I)
5
Y(I) = X(I) + G3(I)
TIME = TIME + DT/2.0D0
CALL STATE(F, Y, TIME)
DO 6 I = 1, NEQ
   G4(I) = DT * F(I)
DO 7 I = 1, NEQ
X(I) = X(I) + (G1(I) + 2.0D0 * (G2(I) + G3(I)) + G4(I))/6.0D0
RETURN
END
8.2 FUZZY LOGIC CONTROL ON 3-DOF MANIPULATOR

PROGRAM FUZZY ON ROBOT

PURPOSE: To apply a hybrid of Fuzzy Logic and traditional integral feedback to a three link microbot. (Highly Non-linear Coupled Second Order Differential Equations)

AUTHOR: Robert J. Stanley II

DATE: 8/4/92

VARIABLES:
- COUNT: Holds the value of the present Runge-Kutta iteration
- TORQUE: The input to the system
- INERTIA: The inertia of the system
- Wn: The natural frequency of the system
- Td: The desired position (Theta Desired)
- TRIGGER: Zero on the first pass and One afterwards
- NEQ: Number of equations
- NSTEP: Number of times runga-kutta subroutine is called
- DT: Time interval delta T in rad/sec
- TIME: Independent variable
- X(I): Dependent variable
- F(I): State equations

INTEGER COUNT, NEQ, NSTEP
REAL*8 X(10), DT, TIME, C2, C3, C23, M11, M22, M23, M33
REAL*8 T1, T2, T3, Td1, Td2, Td3, I2, I3, ALPHA, BETA
INTEGER TRIG1, TRIG2, TRIG3, L1, L2, L3
COMMON/MYCOMM/T1, T2, T3, Td1, Td2, Td3
OPEN (8, FILE='fpidrl1.dat', STATUS='unknown')
OPEN (12, FILE='fpidrl1T.dat', STATUS='unknown')
OPEN (9, FILE='fpidrl11.dat', STATUS='unknown')
OPEN (10, FILE='fpidrl12.dat', STATUS='unknown')
OPEN (11, FILE='fpidrl13.dat', STATUS='unknown')
15 FORMAT(1X, F6.2, 1X, F8.4, 1X, F10.4, 1X, F8.4, 1X, F10.4, 1X, F10.4)
25 FORMAT(1X, F6.2, 1X, F8.4, 1X, F10.4, 1X, F10.4, 1X, F10.4)

ALPHA = .46051702
BETA = 3.14

I2 = ALPHA * (ALPHA**2 + BETA**2)
I3 = ALPHA * (ALPHA**2 + BETA**2)
L1 = 1
L2 = 2
L3=3
NEO=9
DT=.01D0
NSTEP=1000
TIME=0.0D0
X(1)=0.0
X(2)=1.0
X(3)=0.0
X(4)=0.0
X(5)=1.0
X(6)=0.0
X(7)=0.0
X(8)=1.0
X(9)=0.0
Td1=1.0
Td2=1.0
Td3=1.0
TRIG1=0
TRIG2=0
TRIG3=0
CALL FUZZY_LOGIC(X,T1,TRIG1,L1)
CALL FUZZY_LOGIC(X,T2,TRIG2,L2)
CALL FUZZY_LOGIC(X,T3,TRIG3,L3)
TRIG1=1
TRIG2=1
TRIG3=1
C23=COS(X(5)+Td2+X(8)+Td3)
C2=COS(X(5)+Td2)
C3=COS(X(8)+Td3)
M11=26.+8.*C23**2+29.*C2**2+24.*C2*C23
M22=43.+24.*C3
M23=8.+12.*C3
M33=8.

c Agument the torque produced by the Fuzzy controller with that of c the traditional
Integral feedback
T1=T1
T2=T2-12*(M22*X(4)+M23*X(7))
T3=T3-13*(M23*X(4)+M33*X(7))
T1=0.0
T2=0.0
T3=0.0
WRITE(8,15)TIME,X(2)+Td1,T1,X(5)+Td2,T2,X(8)+Td3,T3
WRITE(12,25)TIME,X(2)+Td1,X(5)+Td2,X(8)+Td3
DO 100 COUNT=1,NSTEP
  CALL RUNGAX,DT,NEQ,TIME
  CALL FUZZY_LOGIC(X,T1,TRIG1,L1)
  CALL FUZZY_LOGIC(X,T2,TRIG2,L2)
  CALL FUZZY_LOGIC(X,T3,TRIG3,L3)
  C23=COS(X(5)+Td2+X(8)+Td3)
  C2=COS(X(5)+Td2)
  C3=COS(X(8)+Td3)
M11=26.8\times C23^2+29.2^2+24.2\times C2\times C3
M22=43.2\times C3
M23=8.12\times C3
M33=8.

Agument the torque produced by the Fuzzy controller with that of the traditional Integral feedback

T1=T1
T2=T2-I2(M22\times X(4)+M23\times X(7))
T3=T3-I3(M23\times X(4)+M33\times X(7))

T1=0.0
T2=0.0
T3=0.0

WRITE(8,15)TIME,X(2)+Td1,T1,X(5)+Td2,T2,X(8)+Td3,T3
WRITE(12,25)TIME,X(2)+Td1,X(5)+Td2,X(8)+Td3

100 CONTINUE
CLOSE(8)
CLOSE(9)
CLOSE(10)
CLOSE(11)
END

SUBROUTINE RIGHT(R1,R2,R3,X,M11,M22,M33,M23,DET)

PURPOSE: To calculate the respective mass matrix entries and the respective nonlinear contributions R#.

AUTHOR: Robert J. Stanley II

DATE: 8/5/92

VARIABLES:

R#: The nonlinear terms of link # respectively
F: State space
G: Gravity
DET: The determinate of the mass matrix divided by M11

REAL*8 X(10),Td1,Td2,Td3
REAL*8 T1,T2,T3,R1,R2,R3,M11,M22,M33,M23,G,C23,S23,C2,C3,S2
REAL*8 S3,C23S23,C2S23,S2C23,S2C2,C2S2,DET
COMMON /MYCOMM/ T1,T2,T3,Td1,Td2,Td3

G=9.8
C23=\cos(X(5)+Td2+X(8)+Td3)
S23=\sin(X(5)+Td2+X(8)+Td3)
C2=\cos(X(5)+Td2)
C3=\cos(X(8)+Td3)
S2=\sin(X(5)+Td2)
S3=\sin(X(8)+Td3)
C23S23=C23\times S23
C2S23=C2\times S23
S2C23=S2*C23
S2C2=S2*C2
C2S2=C2*S2

M11=26.+8.*C23**2+29.*C2**2+24.*C2*C23
M22=43.+24.*C3
M23=8.+12.*C3
M33=8.
DET=M22*M33-M23*M23

R1=(16.*C23S23+24.*C2S23)*X(3)*(X(6)+X(9))
++(24.*S2C23+58.*S2C2)*X(3)*X(6)+T1

R2=-(8.*C23S23+12.*S2C23+12.*C2S23+29.*C2S2)*X(3)**2
++24.*S3*X(6)*X(9)+12.*S3*X(9)**2-20.*G*C2-6.*G*C23+T2

R3=-(8.*C23S23+12.*C2S23)*X(3)**2-12.*S3*X(6)**2-6.*G*C23+T3
RETURN
END

SUPROUTINE STATE(F,X,TIME)
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
cc PURPOSE: Define the dynamics in a state form for use in c
cc Runge Kutta Subroutine. c
cc AUTHOR: Robert J. Stanley II c
cc DATE: 8/4/92 c
cccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccccc
REAL*8 X(10),F(10),TIME,Td1,Td2,Td3,T1,T2,T3
REAL*8 R1,R2,R3,M11,M22,M33,M23,DET
COMMON /MYCOMM/ T1,T2,T3,Td1,Td2,Td3

TIME=TIME*1.0

CALL RIGHT(R1,R2,R3,X,M11,M22,M33,M23,DET)
F(1)=X(2)
F(2)=X(3)
F(3)=R1/M11
F(4)=X(5)
F(5)=X(6)
F(6)=(R2*M33/DET)-(R3*M23/DET)
F(7)=X(8)
F(8)=X(9)
F(9)=-(R2*M23/DET)+(R3*M22/DET)
RETURN
END
SUBROUTINE RUNGA(X,DT,NEQ,TIME)

PURPOSE: Use a Runge Kutta algorithm to numerically solve the state equations given in subroutine STATE.

AUTHOR: Robert J. Stanley II

DATE: 8/4/92

REAL*8 X(10),Y(10),F(10),DT,TIME,G1(10),G2(10),G3(10),G4(10)
INTEGER I,NEQ

DO 1 I=1,NEQ
    Y(I)=X(I)
  1

CALL STATE(F,Y,TIME)
DO 2 I=1,NEQ
    G1(I)=DT*F(I)
  2

TIME=TIME+DT/2.0D0
DO 3 I=1,NEQ
    Y(I)=X(I)+G1(I)/2.0D0
  3

CALL STATE(F,Y,TIME)
DO 4 I=1,NEQ
    G2(I)=DT*F(I)
    Y(I)=X(I)+G2(I)/2.0D0
  4

CALL STATE(F,Y,TIME)
DO 5 I=1,NEQ
    G3(I)=DT*F(I)
    Y(I)=X(I)+G3(I)
  5

TIME=TIME+DT/2.0D0
CALL STATE(F,Y,TIME)
DO 6 I=1,NEQ
    G4(I)=DT*F(I)
  6

DO 7 I=1,NEQ
    X(I)=X(I)+(G1(I)+2.0D0*(G2(I)+G3(I))+G4(I))/6.0D0
  7
RETURN
END

SUBROUTINE FUZZY_LOGIC(X,TORQUE,TRIGGER,LINK)

PURPOSE: Given a position calculate a torque required to...
drive the error to zero using Fuzzy Logic Control.

AUTHOR: Robert J. Stanley II

DATE: 8/4/92

VARIABLES:
E: Error
CE: Change in Error
CEA: Change in Error Angle
LASTE: The last error
SET_PT: Set point desired
PI: 3.14
QE: Quantized value of the error
QEC: Quantized value of the Error Change
u: Membership function value
UU: Universe of discourse value
NUM: NUMerator of the input value
DEN: DENominator of the input value
Ye: Temp variable for the Error membership function
Yec: Temp variable for Change in Error membership function
INPUT: The quantized input to the plant
TORQUE: The actual input to the plant
N: Number of rules
I: Count variable
GRID: Tells output which grid is being utilized
FINE: Boolean for the quantized table
MEDIUM: Boolean for the quantized table
COARSE: Boolean for the quantized table
GRID: Indicates which quantized table is being accessed
ELP: Linguistic value Error Large Positive
EMP: Linguistic value Error Medium Positive
ESP: Linguistic value Error Small Positive
EZE: Linguistic value Error Zero
ESN: Linguistic value Error Small Negative
EMN: Linguistic value Error Medium Negative
ELN: Linguistic value Error Large Negative
CELP: Linguistic value Change Error Large Positive
CEMP: Linguistic value Change Error Medium Positive
CESP: Linguistic value Change Error Small Positive
CEZE: Linguistic value Change Error Zero
CESN: Linguistic value Change Error Small Negative
CEMN: Linguistic value Change Error Medium Negative
CELN: Linguistic value Change Error Large Negative

REAL*8 E,CE,LASTE3,PI,QE,QEC,u(50),UU(50),NUM,DEN
REAL*8 Ye,Yec,INPUT,TORQUE,CEA,QECA,LASTE1,LASTE2
REAL*8 EMAX,TOR_MAX,X(10)
INTEGER N,I,TRIGGER,GRID,LINK
LOGICAL FINE,MEDIUM,COARSE,ELP,EMP,ESP,EZE,ESN,EMN,ELN
LOGICAL CELP,CEMP,CESP,CEZE,CESN,CEMN,CELN

FORMAT(1X,F10.4,1X,F10.4,1X,F10.4,1X,F10.4,1X,F10.4,1X,F10.4)

ELP=.FALSE.
EMP=.FALSE.
ESP=.FALSE.
EZE=.FALSE.
ESN=.FALSE.
EMN=.FALSE.
ELN=.FALSE.
CELP=.FALSE.
CEMP=.FALSE.
CESP=.FALSE.
CEZE=.FALSE.
CESN=.FALSE.
CEMN=.FALSE.
CELN=.FALSE.
FINE=.FALSE.
MEDIUM=.FALSE.
COARSE=.FALSE.

N=49
PI=3.14
IF (LINK.EQ.1) THEN
   E=X(2)
   IF (TRIGGER.EQ.0) THEN
      CEA=0.0
   ELSE
      CEA=ATAN2(E-LASTE 1,.01)
   END IF
   LASTE 1=E
   EMAX=60.0
   TOR_MAX=500.0
END IF
IF (LINK.EQ.2) THEN
   E=X(5)
   IF (TRIGGER.EQ.0) THEN
      CEA=0.0
   ELSE
      CEA=ATAN2(E-LASTE2,.01)
   END IF
   LASTE2=E
   EMAX=60.0
   TOR_MAX=500.0
END IF
IF (LINK.EQ.3) THEN
   E=X(8)
   IF (TRIGGER.EQ.0) THEN
      CEA=0.0
ELSE
    CEA=ATAN2(E-LASTE3,.01)
END IF
LASTE3=E
EMAX=60.0
TOR_MAX=150.0
END IF

C Change error and change in error from radians to degrees. E=(180/3.14)*E
CEA=(180/3.14)*CEA
C Determine which quantized table is to be used and find the corresponding quantized values of error and error change.
C IF ((E.LT.25.0).AND.(E.GT.-25.0)) THEN
  FINE=.TRUE.
  QE=E*(6/25.0)
  QECA=CEA*(6/90.)
ELSE
  IF ((E.LT.33.0).AND.(E.GT.-33.0)) THEN
    MEDIUM=.TRUE.
    QE=E*(6/33.)
    QECA=CEA*(6/90.)
  ELSE
    COARSE=.TRUE.
    QE=E*(6/60.)
    QECA=CEA*(6/90.)
  END IF
END IF
COARSE=.TRUE.
QE=E*(6/EMAX)
QECA=CEA*(6/90.)
C Determine which grid is being used
IF (COARSE) THEN
  GRID=1
END IF
IF (MEDIUM) THEN
  GRID=2
END IF
IF (FINE) THEN
  GRID=3
ENDIF
C With the Quantized Error determine which linguistic values are applicable.
IF (QE.GE.6.0) THEN
  QE=6.0
  ELP=.TRUE.
END IF
IF ((QE.GE.4.0).AND.(QE.LT.6.0)) THEN ELP=.TRUE.
  EMP=.TRUE.
END IF
IF ((QE.GE.2.0).AND.(QE.LE.4.0)) THEN EMP=.TRUE.
    ESP=.TRUE.
END IF
IF ((QE.GE.0.0).AND.(QE.LE.2.0)) THEN ESP=.TRUE.
    EZE=.TRUE.
END IF
IF ((QE.GE.-2.0).AND.(QE.LE.0.0)) THEN
    EZE=.TRUE.
    ESN=.TRUE.
END IF
IF ((QE.GE.-4.0).AND.(QE.LE.-2.0)) THEN
    ESN=.TRUE.
    EMN=.TRUE.
END IF
IF ((QE.GE.-6.0).AND.(QE.LE.-4.0)) THEN
    EMN=.TRUE.
    ELN=.TRUE.
END IF
IF (QE.LE.-6.0) THEN
    QE=-6.0
    ELN=.TRUE.
END IF
IF (QECA.GE.6.0) THEN
    QECA=6.0
    CELP=.TRUE.
END IF
IF ((QECA.GE.4.0).AND.(QECA.LT.6.0)) THEN
    CELP=.TRUE.
    CEMP=.TRUE.
END IF
IF ((QECA.GE.2.0).AND.(QECA.LE.4.0)) THEN
    CEMP=.TRUE.
    CESP=.TRUE.
END IF
IF ((QECA.GE.0.0).AND.(QECA.LE.2.0)) THEN
    CESP=.TRUE.
    CEZE=.TRUE.
END IF
IF ((QECA.GE.-2.0).AND.(QECA.LE.0.0)) THEN
    CEZE=.TRUE.
    CESN=.TRUE.
END IF
IF ((QECA.GE.-4.0).AND.(QECA.LE.-2.0)) THEN
    CESN=.TRUE.
    CEMN=.TRUE.
END IF
IF ((QECA.GE.-6.0).AND.(QECA.LE.-4.0)) THEN
    CEMN=.TRUE.
END IF

With the Quantized Error Change determine which linguistic values are applicable.
IF (QECA.GE.6.0) THEN
    QECA=6.0
    CELP=.TRUE.
END IF
IF ((QECA.GE.4.0).AND.(QECA.LT.6.0)) THEN
    CELP=.TRUE.
    CEMP=.TRUE.
END IF
IF ((QECA.GE.2.0).AND.(QECA.LE.4.0)) THEN
    CEMP=.TRUE.
    CESP=.TRUE.
END IF
IF ((QECA.GE.0.0).AND.(QECA.LE.2.0)) THEN
    CESP=.TRUE.
    CEZE=.TRUE.
END IF
IF ((QECA.GE.-2.0).AND.(QECA.LE.0.0)) THEN
    CEZE=.TRUE.
    CESN=.TRUE.
END IF
IF ((QECA.GE.-4.0).AND.(QECA.LE.-2.0)) THEN
    CESN=.TRUE.
    CEMN=.TRUE.
END IF
IF ((QECA.GE.-6.0).AND.(QECA.LE.-4.0)) THEN
    CEMN=.TRUE.
CELN=.TRUE.
END IF
IF (QECA .LE. -6.0) THEN
  QECA = -6.0
  CELN = .TRUE.
END IF

DO 250 I = 1, N
  u(I) = 0.0
  UU(I) = 0.0
250 CONTINUE

c Initialize the membership function value (u) and the universe of discourse value (U) to zero before the rules are applied.

         c Rule one if Error is Large Positive and the Change in Error is Large Negative then contribution is Zero.
         IF (ELP.AND.CELN) THEN
           Ye = SIN(PI/4*(QE-4.0))
           Yec = SIN(PI/4*(QECA+8.0))
           u(1) = MIN(Ye, Yec)
           UU(1) = 0.0
         END IF

c Rule two if Error is Large Positive and the Change in Error is Medium Negative then contribution is Small Negative.
         IF (ELP.AND.CEMN) THEN
           Ye = SIN(PI/4*(QE-4.0))
           Yec = SIN(PI/4*(QECA+6.0))
           u(2) = MIN(Ye, Yec)
           UU(2) = -2.0
         END IF

c Rule three if Error is Large Positive and the Change in Error is Small Negative then contribution is Medium Negative.
         IF (ELP.AND.CESN) THEN
           Ye = SIN(PI/4*(QE-4.0))
           Yec = SIN(PI/4*(QECA+4.0))
           u(3) = MIN(Ye, Yec)
           UU(3) = -4.0
         END IF

c Rule four if Error is Large Positive and the Change in Error is Zero then contribution is Large Negative.
         IF (ELP.AND.CEZE) THEN
           Ye = SIN(PI/4*(QE-4.0))
           Yec = SIN(PI/4*(QECA+2.0))
           u(4) = MIN(Ye, Yec)
           UU(4) = -6.0
         END IF

c Rule five if Error is Large Positive and the Change in Error is Small Positive then contribution is Large Negative.
         IF (ELP.AND.CESP) THEN
           Ye = SIN(PI/4*(QE-4.0))
         END IF
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a})) \]
\[ u(5) = \min(Y_{e}, Y_{ec}) \]
\[ UU(5) = 6.0 \]

**END IF**

C Rule six if Error is Large Positive and the Change in Error is Medium Positive then contribution is Large Negative.

IF (ELP.AND.CEMP) THEN
\[ Y_{e} = \sin(\frac{\pi}{4} * (Q_{e} - 4.0)) \]
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a} - 2.0)) \]
\[ u(6) = \min(Y_{e}, Y_{ec}) \]
\[ UU(6) = 6.0 \]

END IF

C Rule seven if Error is Large Positive and the Change in Error is Large Positive then contribution is Large Negative.

IF (ELP.AND.CELP) THEN
\[ Y_{e} = \sin(\frac{\pi}{4} * (Q_{e} - 4.0)) \]
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a} - 4.0)) \]
\[ u(7) = \min(Y_{e}, Y_{ec}) \]
\[ UU(7) = -6.0 \]

END IF

C Rule eight if Error is Medium Positive and the Change in Error is Large Negative then contribution is Small Positive.

IF (EMP.AND.CELN) THEN
\[ Y_{e} = \sin(\frac{\pi}{4} * (Q_{e} - 2.0)) \]
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a} + 8.0)) \]
\[ u(8) = \min(Y_{e}, Y_{ec}) \]
\[ UU(8) = 2.0 \]

END IF

C Rule nine if Error is Medium Positive and the Change in Error is Medium Negative then contribution is Zero.

IF (EMP.AND.CEMN) THEN
\[ Y_{e} = \sin(\frac{\pi}{4} * (Q_{e} - 2.0)) \]
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a} + 6.0)) \]
\[ u(9) = \min(Y_{e}, Y_{ec}) \]
\[ UU(9) = 0.0 \]

END IF

C Rule ten if Error is Medium Positive and the Change in Error is Small Negative then contribution is Small Negative.

IF (EMP.AND.CESN) THEN
\[ Y_{e} = \sin(\frac{\pi}{4} * (Q_{e} - 2.0)) \]
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a} + 4.0)) \]
\[ u(10) = \min(Y_{e}, Y_{ec}) \]
\[ UU(10) = -2.0 \]

END IF

C Rule eleven if Error is Medium Positive and the Change in Error is Zero then contribution is Medium Negative.

IF (EMP.AND.CEZE) THEN
\[ Y_{e} = \sin(\frac{\pi}{4} * (Q_{e} - 2.0)) \]
\[ Y_{ec} = \sin(\frac{\pi}{4} * (QEC_{a} + 2.0)) \]
\[ u(11) = \min(Y_{e}, Y_{ec}) \]
\[ UU(11) = -4.0 \]
c Rule twelve if Error is Medium Positive and the Change in Error is Small Positive
then contribution is Large Negative.
IF (EMP.AND.CESP) THEN
Ye=SIN(PI/4*(QE-2.0))
Yec=SIN(PI/4*(QECA))
u(12)=MIN(Ye,Yec)
UU(12)=-6.0
END IF
c Rule thirteen if Error is Medium Positive and the Change in Error is Medium Positive
then contribution is Large Negative.
IF (EMP.AND.CEMP) THEN
Ye=SIN(PI/4*(QE-2.0))
Yec=SIN(PI/4*(QECA-2.0))
u(13)=MIN(Ye,Yec)
UU(13)=-6.0
END IF
c Rule fourteen if Error is Medium Positive and the Change in Error is Large Positive
then contribution is Large Negative.
IF (EMP.AND.CELP) THEN
Ye=SIN(PI/4*(QE-2.0))
Yec=SIN(PI/4*(QECA-4.0))
u(14)=MIN(Ye,Yec)
UU(14)=-6.0
END IF
c Rule fifteen if Error is Small Positive and the Change in Error is Large Negative
then contribution is Medium Positive.
IF (ESP.AND.CELN) THEN
Ye=SIN(PI/4*(QE))
Yec=SIN(PI/4*(QECA+8.0))
u(15)=MIN(Ye,Yec)
UU(15)=4.0
END IF
c Rule sixteen if Error is Small Positive and the Change in Error is Medium Negative
then contribution is Small Positive.
IF (ESP.AND.CEMN) THEN
Ye=SIN(PI/4*(QE))
Yec=SIN(PI/4*(QECA+6.0))
u(16)=MIN(Ye,Yec)
UU(16)=2.0
END IF
c Rule seventeen if Error is Small Positive and the Change in Error is Small Negative
then contribution is Zero.
IF (ESP.AND.CESN) THEN
Ye=SIN(PI/4*(QE))
Yec=SIN(PI/4*(QECA+4.0))
u(17)=MIN(Ye,Yec)
UU(17)=-2.0
END IF
c Rule eighteen if Error is Small Positive and the Change in Error is Zero then
contribution is Small Negative.
IF (ESP.AND.CEZE) THEN
    Ye=SIN(PI/4*(QE))
    Yec=SIN(PI/4*(QECA+2.0))
    u(18)=MIN(Ye,Yec)
    UU(18)=-2.0
END IF

Note: Rule nineteen if Error is Small Positive and the Change in Error is c Small Positive then contribution is Medium Negative.

IF (ESP.AND.CESP) THEN
    Ye=SIN(PI/4*(QE))
    Yec=SIN(PI/4*(QECA))
    u(19)=MIN(Ye,Yec)
    UU(19)=-4.0
END IF

Note: Rule twenty if Error is Small Positive and the Change in Error is c Medium Positive then contribution is Large Negative.

IF (ESP.AND.CEMP) THEN
    Ye=SIN(PI/4*(QE))
    Yec=SIN(PI/4*(QECA-2.0))
    u(20)=MIN(Ye,Yec)
    UU(20)=-6.0
END IF

Note: Rule twenty one if Error is Small Positive and the Change in Error is c Large Positive then contribution is Large Negative.

IF (ESP.AND.CELP) THEN
    Ye=SIN(PI/4*(QE))
    Yec=SIN(PI/4*(QECA-4.0))
    u(21)=MIN(Ye,Yec)
    UU(21)=-6.0
END IF

Note: Rule twenty two if Error is Zero and the Change in Error is c Large Negative then contribution is Large Positive.

IF (EZE.AND.CELN) THEN
    Ye=SIN(PI/4*(QE+2.0))
    Yec=SIN(PI/4*(QECA+8.0))
    u(22)=MIN(Ye,Yec)
    UU(22)=6.0
END IF

Note: Rule twenty three if Error is Zero and the Change in Error is c Medium Negative then contribution is Medium Positive.

IF (EZE.AND.CEMN) THEN
    Ye=SIN(PI/4*(QE+2.0))
    Yec=SIN(PI/4*(QECA+6.0))
    u(23)=MIN(Ye,Yec)
    UU(23)=4.0
END IF

Note: Rule twenty four if Error is Zero and the Change in Error is c Small Negative then contribution is Small Positive.

IF (EZE.AND.CESN) THEN
    Ye=SIN(PI/4*(QE+2.0))
    Yec=SIN(PI/4*(QECA+4.0))
c Rule twenty five if Error is Zero and the Change in Error is c Zero then contribution is Zero.

\[
\begin{align*}
\text{IF (EZE.AND.CEZE) THEN} \\
&\quad \text{Ye} = \sin(\pi/4*(QE+2.0)) \\
&\quad \text{Yec} = \sin(\pi/4*(QECA+2.0)) \\
&\quad u(25) = \min(\text{Ye}, \text{Yec}) \\
&\quad UU(25) = 0.0
\end{align*}
\]

END IF

c Rule Twenty six if Error is Zero and the Change in Error is c Small Positive then contribution is Small Negative.

\[
\begin{align*}
\text{IF (EZE.AND.CESP) THEN} \\
&\quad \text{Ye} = \sin(\pi/4*(QE+2.0)) \\
&\quad \text{Yec} = \sin(\pi/4*(QECA)) \\
&\quad u(26) = \min(\text{Ye}, \text{Yec}) \\
&\quad UU(26) = -2.0
\end{align*}
\]

END IF

c Rule Twenty seven if Error is Zero and the Change in Error is c Medium Positive then contribution is Medium Negative.

\[
\begin{align*}
\text{IF (EZE.AND.CEMP) THEN} \\
&\quad \text{Ye} = \sin(\pi/4*(QE+2.0)) \\
&\quad \text{Yec} = \sin(\pi/4*(QECA-2.0)) \\
&\quad u(27) = \min(\text{Ye}, \text{Yec}) \\
&\quad UU(27) = -4.0
\end{align*}
\]

END IF

c Rule Twenty eight if Error is Zero and the Change in Error is c Large Positive then contribution is Large Negative.

\[
\begin{align*}
\text{IF (EZE.AND.CELP) THEN} \\
&\quad \text{Ye} = \sin(\pi/4*(QE+2.0)) \\
&\quad \text{Yec} = \sin(\pi/4*(QECA-4.0)) \\
&\quad u(28) = \min(\text{Ye}, \text{Yec}) \\
&\quad UU(28) = -6.0
\end{align*}
\]

END IF

c Rule Twenty nine if Error is Small Negative and the Change in c Error is Large Negative then contribution is Large Positive.

\[
\begin{align*}
\text{IF (ESN.AND.CELN) THEN} \\
&\quad \text{Ye} = \sin(\pi/4*(QE+4.0)) \\
&\quad \text{Yec} = \sin(\pi/4*(QECA+8.0)) \\
&\quad u(29) = \min(\text{Ye}, \text{Yec}) \\
&\quad UU(29) = 6.0
\end{align*}
\]

END IF

c Rule Thirty if Error is Small Negative and the Change in c Error is Medium Negative then contribution is Large Positive.

\[
\begin{align*}
\text{IF (ESN.AND.CEMN) THEN} \\
&\quad \text{Ye} = \sin(\pi/4*(QE+4.0)) \\
&\quad \text{Yec} = \sin(\pi/4*(QECA+6.0)) \\
&\quad u(30) = \min(\text{Ye}, \text{Yec}) \\
&\quad UU(30) = 6.0
\end{align*}
\]

END IF
c Rule Thirty one if Error is Small Negative and the Change in c Error is Small Negative then contribution is Medium Positive.

IF (ESN.AND.CESN) THEN
Ye=SIN(PI/4*(QE+4.0))
Yec=SIN(PI/4*(QECA+4.0))
\(u(31)=\min(Ye, Yec)\)
UU(31)=4.0
END IF

c Rule Thirty two if Error is Small Negative and the Change in c Error is Zero then contribution is Small Positive.

IF (ESN.AND.CEZE) THEN
Ye=SIN(PI/4*(QE+4.0))
Yec=SIN(PI/4*(QECA+2.0))
\(u(32)=\min(Ye, Yec)\)
UU(32)=2.0
END IF

c Rule Thirty three if Error is Small Negative and the Change in c Error is Small Positive then contribution is Small Positive.

IF (ESN.AND.CESP) THEN
Ye=SIN(PI/4*(QE+4.0))
Yec=SIN(PI/4*(QECA))
\(u(33)=\min(Ye, Yec)\)
UU(33)=2.0
END IF

c Rule Thirty four if Error is Small Negative and the Change in c Error is Medium Positive then contribution is Small Negative.

IF (ESN.AND.CEMP) THEN
Ye=SIN(PI/4*(QE+4.0))
Yec=SIN(PI/4*(QECA-2.0))
\(u(34)=\min(Ye, Yec)\)
UU(34)=-2.0
END IF

c Rule Thirty five if Error is Small Negative and the Change in c Error is Large Positive then contribution is Medium Negative.

IF (ESN.AND.CELP) THEN
Ye=SIN(PI/4*(QE+4.0))
Yec=SIN(PI/4*(QECA-4.0))
\(u(35)=\min(Ye, Yec)\)
UU(35)=-4.0
END IF

c Rule Thirty six if Error is Medium Negative and the Change in c Error is Large Negative then contribution is Large Positive.

IF (EMN.AND.CELN) THEN
Ye=SIN(PI/4*(QE+6.0))
Yec=SIN(PI/4*(QECA+8.0))
\(u(36)=\min(Ye, Yec)\)
UU(36)=6.0
END IF

c Rule Thirty seven if Error is Medium Negative and the Change in c Error is Medium Negative then contribution is Large Positive.

IF (EMN.AND.CEMN) THEN
Ye = \sin(\pi/4(QE+6.0))
Yec = \sin(\pi/4(QECA+6.0))
u(37) = \min(Ye, Yec)
UU(37) = 6.0

END IF

Rule Thirty eight if Error is Medium Negative and the Change in Error is Small Negative then contribution is Large Positive.
IF (EMN.AND.CESN) THEN
Ye = \sin(\pi/4(QE+6.0))
Yec = \sin(\pi/4(QECA+4.0))
u(38) = \min(Ye, Yec)
UU(38) = 6.0
END IF

Rule Thirty nine if Error is Medium Negative and the Change in Error is Zero then contribution is Medium Positive.
IF (EMN.AND.CEZE) THEN
Ye = \sin(\pi/4(QE+6.0))
Yec = \sin(\pi/4(QECA+2.0))
u(39) = \min(Ye, Yec)
UU(39) = 4.0
END IF

Rule Forty if Error is Medium Negative and the Change in Error is Small Positive then contribution is Small Positive.
IF (EMN.AND.CESP) THEN
Ye = \sin(\pi/4(QE+6.0))
Yec = \sin(\pi/4(QECA))
u(40) = \min(Ye, Yec)
UU(40) = 2.0
END IF

Rule Forty one if Error is Medium Negative and the Change in Error is Medium Positive then contribution is Zero.
IF (EMN.AND.CEMP) THEN
Ye = \sin(\pi/4(QE+6.0))
Yec = \sin(\pi/4(QECA-2.0))
u(41) = \min(Ye, Yec)
UU(41) = 0.0
END IF

Rule Forty two if Error is Medium Negative and the Change in Error is Large Positive then contribution is Small Negative.
IF (EMN.AND.CELP) THEN
Ye = \sin(\pi/4(QE+6.0))
Yec = \sin(\pi/4(QECA-4.0))
u(42) = \min(Ye, Yec)
UU(42) = -2.0
END IF

Rule Forty three if Error is Large Negative and the Change in Error is Large Negative then contribution is Large Positive.
IF (ELN.AND.CELN) THEN
Ye = \sin(\pi/4(QE+8.0))
Yec = \sin(\pi/4(QECA+8.0))
u(43) = \min(Ye, Yec)
UU(43)=6.0

END IF

Rule forty four if Error is Large Negative and the Change in Error is
Medium Negative then contribution is Large Positive.

IF (ELN.AND.CEMN) THEN
    Ye=SIN(PI/4*(QE+8.0))
    Yec=SIN(PI/4*(QECA+6.0))
    u(44)=MIN(Ye,Yec)
    UU(44)=6.0
END IF

Rule forty five if Error is Large Negative and the Change in Error is Small
Negative then contribution is Large Positive.

IF (ELN.AND.CESN) THEN
    Ye=SIN(PI/4*(QE+8.0))
    Yec=SIN(PI/4*(QECA+4))
    u(45)=MIN(Ye,Yec)
    UU(45)=6.0
END IF

Rule forty six if Error is Large Negative and the Change in Error is Zero
then contribution is Large Positive.

IF (ELN.AND.CEZE) THEN
    Yz=SIN(PI/4*(QE+8.0))
    Yec=SIN(PI/4*(QECA+2))
    u(46)=MIN(Ye,Yec)
    UU(46)=6.0
END IF

Rule forty seven if Error is Large Negative and the Change in Error is Small
Positive then contribution is Medium Positive.

IF (ELN.AND.CESP) THEN
    Ye=SIN(PI/4*(QE+8.0))
    Yec=SIN(PI/4*(QECA))
    u(47)=MIN(Ye,Yec)
    UU(47)=4.0
END IF

Rule forty eight if Error is Large Negative and the Change in Error is Medium Positive then contribution is Small Positive.

IF (ELN.AND.CEMP) THEN
    Ye=SIN(PI/4*(QE+8.0))
    Yec=SIN(PI/4*(QECA-2.0))
    u(48)=MIN(Ye,Yec)
    UU(48)=2.0
END IF

Rule forty nine if Error is Large Negative and the Change in Error is Large
Positive then contribution is Zero.

IF (ELN.AND.CELP) THEN
    Ye=SIN(PI/4*(QE+8.0))
    Yec=SIN(PI/4*(QECA-4.0))
    u(49)=MIN(Ye,Yec)
    UU(49)=0.0
END IF
c Initialize the NUMerator and DENomenator to zero so that only c contributions occurring on this pass will be considered.
NUM=0.0
DEN=0.0

c Calculate the NUMerator and the DENomenator of the control input c by means of the center of gravity method.
DO 300 I=1,N
   NUM=NUM+u(I)*UU(I)
   DEN=DEN+u(I)
300 CONTINUE

c Setting the DENomentator to 1.0 prevents division by zero and c does not effect the value of the control input.
IF (DEN.LT.0.0001)THEN
   DEN=1.0
END IF
INPUT=NUM/DEN
IF (LINK.EQ.1) THEN
   WRITE(9,16)E,CEA,QE,QECA,
INPUT END IF
IF (LINK.EQ.2) THEN
   WRITE(10,16)E,CEA,QE,QEC
   A,INPUT
END IF
IF (LINK.EQ.3) THEN
   WRITE(11,16)E,CEA,QE,QEC
   A,INPUT
END IF

c Using the correct quantized table convert the input into c a torque to be sent to the plant.
IF (COARSE) THEN
   TORQUE=INPUT*(TOR_MAX/6.0)
END IF
IF (MEDIUM) THEN
   TORQUE=INPUT*(15./6.0)
END IF
IF (FINE) THEN
   TORQUE=INPUT*(17./6.0)
END IF
RETURN
END