Fuzzy Logic Controller for the Inverted Pendulum Problem

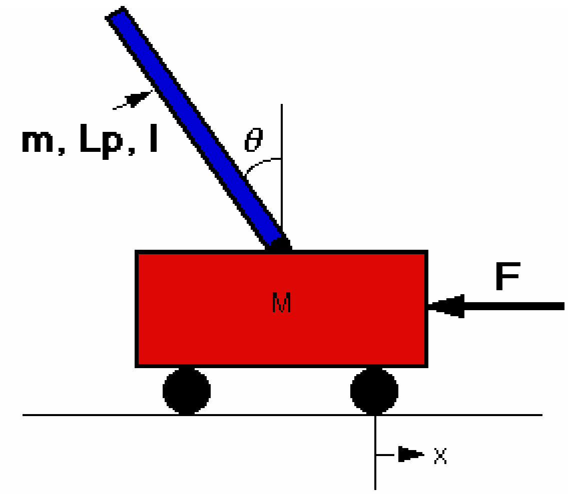
Lawrence Bush
Computer Science Department
Rensselaer Polytechnic Institute
Troy, New York
November 27, 2001

Abstract

The application of a Fuzzy Logic Controller (FLC) to the inverted pendulum problem is presented in this paper. FLCs have been used to control many dynamic systems. The inverted pendulum represents a challenging control problem, which continually moves toward an uncontrolled state. Three versions of an FLC for the inverted pendulum problem are discussed in this paper including their strengths, weaknesses and performance. Techniques for manually tuning a complex FLC are also implemented in this project and addressed in this paper.

1. Problem Description

This project addresses the inverted pendulum-balancing problem. In this project, a simulated[1] inverted pendulum system is used. The system amounts to an inverted pendulum mounted on a cart. The diagram below depicts this system.



System Constants

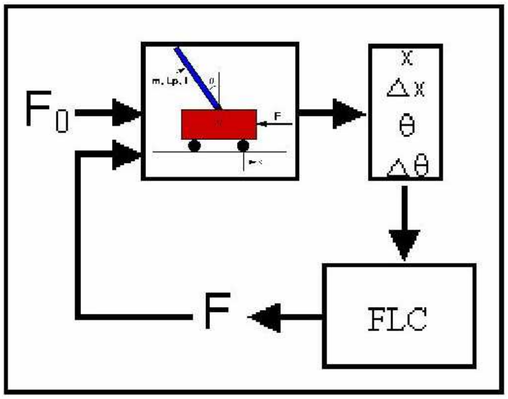
	Name	Value
M	cart mass	0.5 kg
b	cart friction	0.1N/m/sec
m	pendulum mass	0.2kg
Lp	pendulum length (to center of mass)	0.3m
I	pendulum inertia	$0.006 kg*m^2$

System Variables

	Name	Units
F	force applied to cart in	N
x	cart position	m
Δx	cart velocity	m/sec
θ	pendulum angle (from vertical)	radians
$\Delta\theta$	pendulum angular velocity	radians/sec

The system works as follows. The system begins at rest (all of the system variables are zero) with the pendulum standing straight up. An initial disturbance force is applied to the cart, which puts the system out of balance. After this point, it is up to the controller to keep the system in control.

The system simulator provides the system status variables (x, Δx, θ, $\Delta\theta$) to the controller. The controller uses these variables to determine the force to apply to the cart. The diagram below depicts this feedback system.



Solution Evaluation Criteria:

The primary objective of this problem is to keep the pendulum and cart in a controlled state over the simulation period (6 seconds). This means that the controlled system variables are not monotonically increasing or decreasing and that the controlled system variables are within a certain band. Our benchmark for the system variable control band is for θ to be within .5 radians and x to be within .5 meters.

The secondary objective is to tighten the system control. The best possible result would be for the system to return to a balanced rest state (all of the system variables (x, Δx, θ,$\Delta\theta$) equal to zero). The closer the controller comes to achieving this, the better.

A third objective of this problem is to create a robust controller. To test for robustness, the controller is run using a range of initial disturbance forces (-.5 N to + .5 N). The controller should achieve criteria 1 and 2 over the range of disturbance forces.

2. Related Work

Many methods have been used to control the inverted pendulum problem. These include a Proportional Integral-Derivative controller[2], a Neural Network controller[3] and a Genetic Algorithm tuned Fuzzy Logic Controller[4].

A Proportional Integral-Derivative controller uses a three-term transfer function to calculate a signal (u) from an error signal (e). The signal (u) is equal to the proportional gain (Kp) times the magnitude of the error plus the integral gain (Ki) times the integral of the error plus the derivative gain (Kd) times the derivative of the error. This signal (u) will be sent to the plant, and the new output (Y) will be obtained.

A neural network approach to the inverted pendulum problem was presented by Osamu Fujita and Kuniharu Uchimura. Their approach used trial-and-error correlation learning to train a four input neural controller with two hidden neurons.

An example of a controller similar to the system presented in this paper is that of Shyh-Jier Huang and Chuan-Chang Hung. They used a genetic algorithm to tune a fuzzy logic controller for the inverted pendulum problem. Their system used 2 inputs (θ and $\Delta\theta$) and a 5^2 rule-base. A 5^2 rule-base is a 5 by 5 matrix of 25 rules with corresponding consequent functions. A 5^2 rule-base takes 2 inputs and has 5 membership functions for each input.

3. Solution Description

For this project, I created and compared three FLCs for the inverted pendulum problem. The three FLCs are referred to as the 5^2-FLC, 2^4-FLC, and 5^4-FLC.

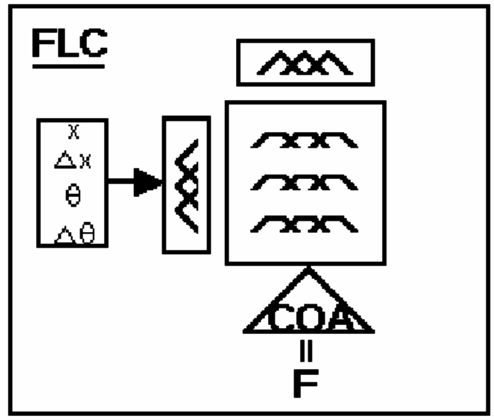
The 5^2-FLC accepts 2 input variables from the simulator and fuzzifies them with 2 sets of 5 triangular membership functions. It then uses a 5 by 5 T-norm (minimum) rule-base with 5 consequent functions to determine the overall output.

The 2^4-FLC accepts 4 input variables from the simulator and fuzzifies them with 4 sets of 2 triangular membership functions. It then uses a 2^4 T-norm (minimum) rule-base with 5 consequent functions to determine the overall output.

The 5^4-FLC accepts 4 input variables from the simulator and fuzzifies them with 4 sets of 5 triangular membership functions. It then uses a 5^4 T-norm (minimum) rule-base with 17 consequent functions to determine the overall output.

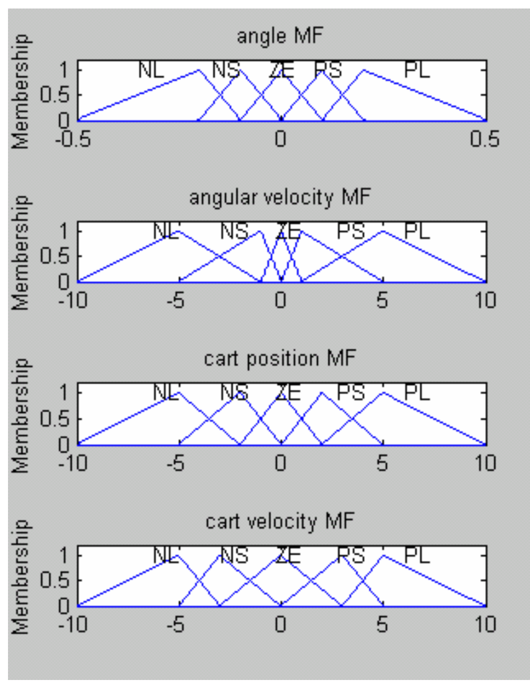
All of the above FLCs use the center of area method to defuzzify the consequent MF matrices.

The following diagram depicts the organization of the controllers.



The 5^4-FLC is described in further detail.

The following is a graph of the membership functions which 5^4-FLC uses to fuzzify the 4 input variables.



The membership functions determine the firing strength for each individual input variable. 5^4-FLC combines these firing strengths using a 2^4 T-norm (minimum) rule-base and applies them to 17 consequent functions. This produces a matrix of qualified consequent MFs.

The st45.m function was created to automate the construction of the consequent functions. Then the controller uses st45 to create a set of n (17) evenly spaced horizontally symmetrical orthogonal trapezoidal consequent functions across a given range [-5.4 : +5.4]. The support of each consequent function is 3 times the width of the core.

Each of these consequent functions is indexed from 1 to 17.

Each rule in the rule-base is associated with a consequent function using the formula:

cmf(i + (j-1) + (-k+5) + (-l+5))

for i, j, k, l = [1 : 5]

The variables **i, j, k** and **l** pertain to the membership of x, Δx, θ and $\Delta\theta$. This formula requires that the positive-negative orientation of the MF-set be appropriately matched to the consequent MF-set. However, the polarity of **k** and **l** were reversed by adjusting the formula. The resulting composed rule-base is a 4 dimensional matrix of diagonally layered isogramatic hyper planes. For example, the following diagonalized matrix is produced by applying the above formula in 2 dimensions resulting in 9 diagonally layered iso-lines.

1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8
5	6	7	8	9

If this notion is extrapolated to 4 dimensions, the result is a 4 dimensional matrix of 17 diagonally layered isogramatic hyper-planes.

An example of a rule derived from the rule-base formula is as follows:

Inputs

Angle is large positive.
Angular Velocity is large positive.
Cart Position is large negative.
Cart Velocity is large negative.

Rule(5, 5, 1, 1)

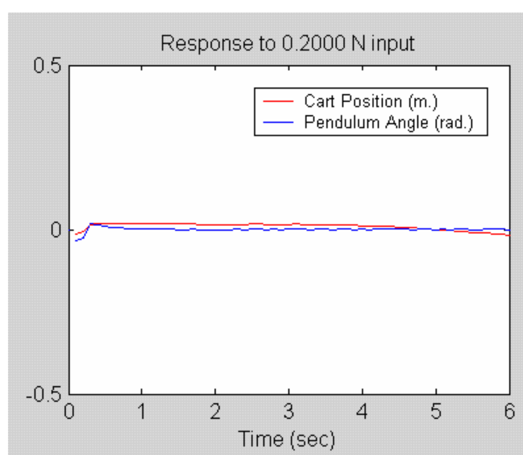
= cmf(i + (j-1) + (-k+5) + (-l+5))
= cmf(17)
= large positive force

The 5^4-FLC then defuzzies the qualified consequent matrix using the center of area method to determine the overall output.

4. Solution Analysis

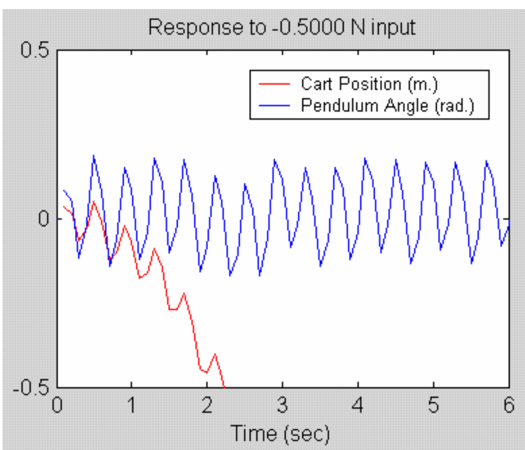
The following graphs show the results of the three controllers for an initial disturbance force of .2N and -.5N.

5^2-FLC



This FLC keeps the pendulum in tight control over the simulation period. However, after the first .5 seconds, the cart position is moving in the negative direction. This is due to the fact that this controller only uses the pendulum angle and angular velocity as inputs and therefore, does not directly control the cart position. It works well for this disturbance force because it was tuned for this force. We can see from the following graph that this controller does not do very well on the robustness test (-.5N). The controller is able to keep the pendulum from falling down, however, the cart moves far away from the starting point. The oscillations are larger but are within the .5m benchmark.

5^2-FLC

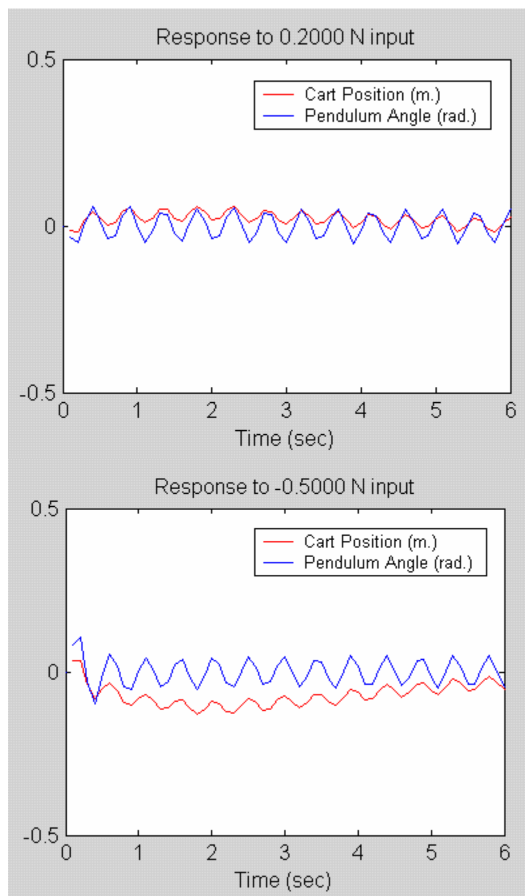


The graph above shows that the 5^2-FLC is nt robust. However, the 5^2-FLC has the advantage of being simpler and easier to tune because it only deals with 2 variables and only has a 25 element rule-base.

2^4-FLC

The following controller performs reasonably well under the standard test and the robustness test. However, the controller is unable to reduce the size of the oscillations in both tests. This is due to the fact that the controller has only 2 MFs for each variable. Therefore, it behaves linearly. In order for a controller to gradually reduce the size of the oscillations, it needs a polynomially shaped relationship between the input and the output. This controller has the advantage of being simpler and easier to tune because it only has a 16 element rule-base.

2^4-FLC

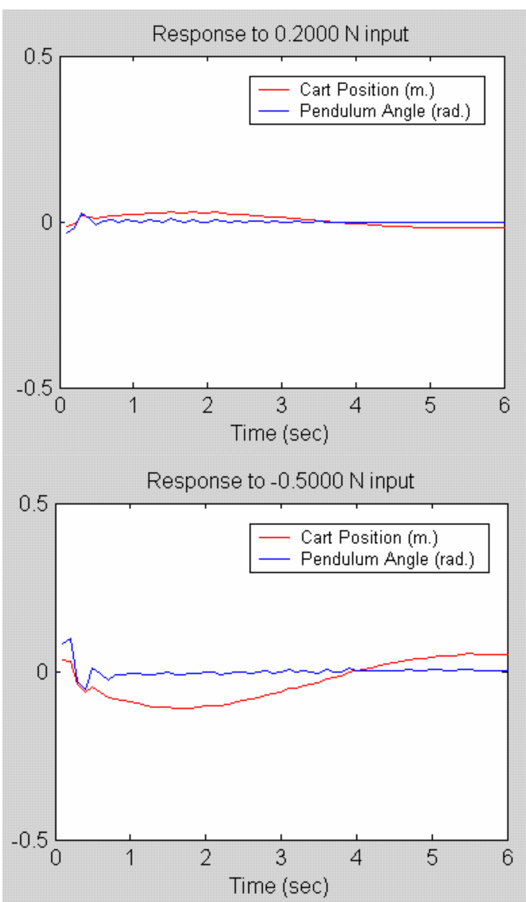


5^4-FLC

The following controller performs very well under the standard test and the robustness test. The controller was able to reduce the size of the oscillations in both tests. This is due to the fact that the controller has 5 MFs for each variable. Therefore, it behaves polynomially.

This controller has the disadvantage of being more complicated and more difficult to tune because it uses 5 MFs for each of the 4 input variables, 17 consequent functions and has a 625 element rule-base. It also takes much more computing time.

5^4-FLC



5. Conclusion

Manually tuned Fuzzy Logic Controllers can be effectively applied to control the inverted pendulum problem. The controllers were able to recover quickly from various deviations and stabilize the pendulum within a tight band of control.

The 5^4-FLC is by far the most accurate and robust model presented here. The 5^4-FLC produced a quick and smooth recovery on both tests. One drawback of the 5^4-FLC is its long processing time. A possible next step is to improve the running time by converting the iterative calculations to matrix calculations.

The techniques explained above address some of the complexities of tuning an FLC. These techniques should be useful in tuning FLCs for other complex systems. An appropriate future area to explore is testing these techniques on other control problems.

References:

1. Copyright (C) 1997 by the Regents of the University of Michigan.

2. Control Tutorials for Matlab developed by Professor Bill Messner of the Department of Mechanical Engineering at Carnegie Mellon University and Professor Dawn Tilbury of the Department of Mechanical Engineering and Applied Mechanics at the University of Michigan, 1997.

3. O. Fujita and K. Uchimura, Trial-and-Error Correlation Learning to Control an Inverted Pendulum, Proceedings of the 3rd International Conference on Fuzzy Logic, Neural Nets and Soft Computing, Iizuka, Japan, August 1994.

4. S. Huang and C. Hung, Genetic-Evolved Fuzzy Systems for Inverted Pendulum Controls, Proceedings of the International Joint Conference of CFSA/IFIS/SOFT '95 on Fuzzy Theory and Applications, December 1995.

Appendix:

The following source code for the 5^4-FLC is provided in the appendix:

controller_4.m	coa.m
mf4b.m	trap_mf.m
st45_mf.m	triangle_mf.m