A Comparative Study of Fuzzy Logic and Classical Control with EPICS*

Johnny Tang and Hamid Shoaee

Continuous Electron Beam Accelerator Facility, Newport News, VA, USA

Abstract

The classical control theory which relies on the mathematical model of the underlying system has been successfully applied to the control of a large variety of simple, non-linear processes. However, it has not been as widely used with complicated, non-linear, time varying systems or with processes suffering from noisy measurements. The main idea of fuzzy control is to build a model of an expert operator who is capable of controlling the plant without thinking in terms of a mathematical model. This paper describes the application of fuzzy control to a feedback system within an EPICS environment. Comparison of the application of a modern controller and a fuzzy controller to an inverted pendulum problem is presented.

I. INTRODUCTION

The American philosopher Charles Pierce pointed out about a hundreds years ago, that "Logicians have too much neglected the study of vagueness, not suspecting the important part it plays in mathematical thought." In 1965, Lotfi Zadeh, Professor of electrical engineering at the University of California at Berkeley, published his paper "Fuzzy Sets.[1]" Today, fuzzy logic has been widely used in automotive control applications and has transformed many commercial product markets in Japan and Germany. Many experts predict that fuzzy technology is on its way to becoming a multibillion-dollar industry.

The main idea behind fuzzy control is to model an expert operator who is able to control the process, instead of using a mathematical formalization of the process as in modern control theory. Although there are many practical successes, fuzzy control has not been viewed as a rigorous science due to a lack of formal synthesis techniques. Fuzzy control should be useful when a workable mathematical model for a plant is very difficult or impossible to derive or the mathematical model is complex and computing it in real time would require expensive processors.

In this paper, we first present the simulation of an inverted pendulum in the EPICS environment. Implementations of modern and fuzzy controllers to the simulation follow and then we compare the results of these two controllers. Finally a summary and future plans are discussed.

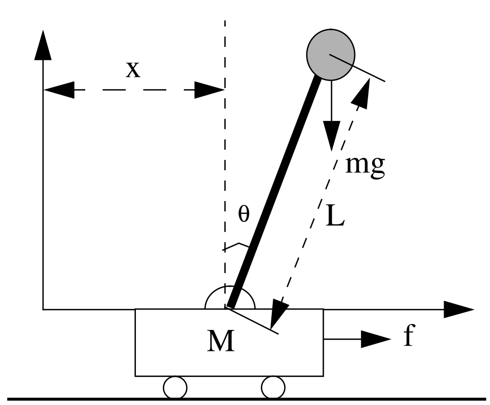
II. SIMULATION

In this section, we present the implementation of a simulation of an inverted pendulum balancing problem in the EPICS environment.

A. Problem

The inverted pendulum problem consists of a variable mass on top of a variable-length rod mounted on a motor-driven cart as shown in Figure 1.

Figure 1. The Inverted Pendulum



* Supported by U.S. DOE contract DE-AC05-84ER40150.

The motion of the inverted pendulum can be described by the following two linearized differential equations with the assumptions of a small θ and a small $\dot{\theta}$ [2]:

$$Ml\ddot{\theta} = \langle M + m\rangle g\theta - f$$

$$M\ddot{x} = f - mg\theta$$

B. Computer Solution

By defining state variables x_1, x_2, x_3, and x_4 as $x_1 = \theta; x_2 = \dot{\theta}; x_3 = x; x_4 = \dot{x}$, the above two linearized differential equations can be transformed to a state-space representation:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{M+m}{Ml}g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{m}{M}g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{1}{Ml} \\ 0 \\ \frac{1}{M} \end{bmatrix} f$$

Applying the Runge-Kutta method[2] to find the solution of a scalar differential equation, $\frac{dx}{dt} = f(t,x), (x(0) = x_0)$, gives

$$x_{i+1} = x_i + \frac{1}{6}\langle K_1 + 2K_2 + 2K_3 + K_4\rangle$$

where

$$K_1 = hf(t_i, x_i)$$

$$K_2 = hf(t_i + \frac{1}{2}h, x_i + \frac{1}{2}K_1)$$

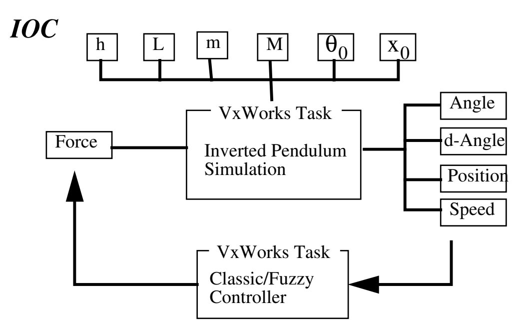
$$K_3 = hf(t_i + \frac{1}{2}h, x_i + \frac{1}{2}K_2)$$

$$K_4 = hf(t_i + h, x_i + K_3)$$

$$h = t_{i+1} - t_i$$

It then becomes straightforward to implement the state-space equations for the motion of the inverted pendulum as a VxWorks task running on an IOC with a sampling rate h. Figure 2 illustrates how the controllers access the plant via the VxWorks variables.

Figure 2. Simulation of an Inverted Pendulum



III. MODERN CONTROL

In this section, an implementation of a modern control to the inverted pendulum balancing problem in an EPICS environment is presented.

A. Modern Control Theory

Two techniques are commonly used in modern control theory to design control systems: pole placement and linear quadratic optimal control. The pole placement approach is to assume that the state feedback control vector is $u = -Kx$ and determine the feedback gain matrix **K** such that the control system will have the desired closed-loop poles. The quadratic optimal control approach is also based on the assumption that the state feedback control vector is $u = -Kx$. However, the feedback gain matrix **K** is determined such that a quadratic performance index is minimized.

B. Modern Control Feedback System

The pole placement technique is used here to design and implement a controller for the inverted pendulum balancing problem in the EPICS environment. The state-space equations for the dynamics of the inverted pendulum can be rewritten as

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu$$
$$y = C\mathbf{x}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{M+m}{Ml}g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{m}{M}g & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 \\ -\frac{1}{Ml} \\ 0 \\ \frac{1}{M} \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \mathbf{x} = \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \end{bmatrix}$$

The state feedback vector is $\mathbf{u} = -\mathbf{K}\mathbf{x}$ and by using the following Ackermann formula[4], the feedback gain matrix **K** can be determined.

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & \ldots & 0 & 1 \end{bmatrix} \begin{bmatrix} B | & AB | & \ldots | & A^{n-1}B \end{bmatrix}^{-1} \phi(A)$$

where

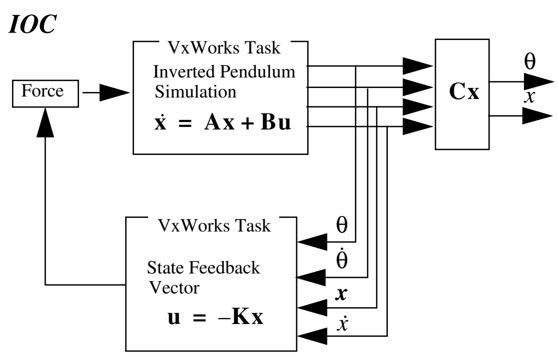
$$\phi(A) = A^n + \alpha_1 A^{n-1} + \ldots + \alpha_{n-1} A + \alpha_n I$$

the values of $\alpha_1, \alpha_2, \ldots, \alpha_n$ can be determined by the following equation:

$$\langle s - \mu_1 \rangle \langle s - \mu_2 \rangle \ldots \langle s - \mu_n \rangle = s^n + \alpha_1 s^{n-1} + \ldots + \alpha_{n-1} s + \alpha_n$$

where $\mu_1, \mu_2, \ldots, \mu_n$ are the desired closed-loop poles.

Figure 3. Modern Control Feedback System



The modern controller is then easily implemented once the state feedback gain matrix **K** is determined as shown in Figure 3.

IV. Fuzzy Control

An alternative approach to the control of any processes is to investigate the control strategies employed by the human operator. An operator usually expresses his control strategy linguistically as a set of heuristic decisions. The fuzzy logic provides the means of expressing linguistic rules in a form suitable for computer implementation. In this section, the design and implementation of a fuzzy controller for the inverted pendulum balancing problem in EPICS environment is presented.

A. Fuzzy Set Theory and Fuzzy Logic

Zadeh[1] expanded the concept of the traditional boolean logic set to that of a fuzzy set. He stated that a fuzzy set is a class of objects with a continuum of grades of membership.

Fuzzy logic is based on fuzzy set theory and provides means for representing partial belief situations. It is a generalization of a conventional boolean logic. It defines the union, intersection, etc. operations of fuzzy sets in a similar way to boolean logic and classic set theory.

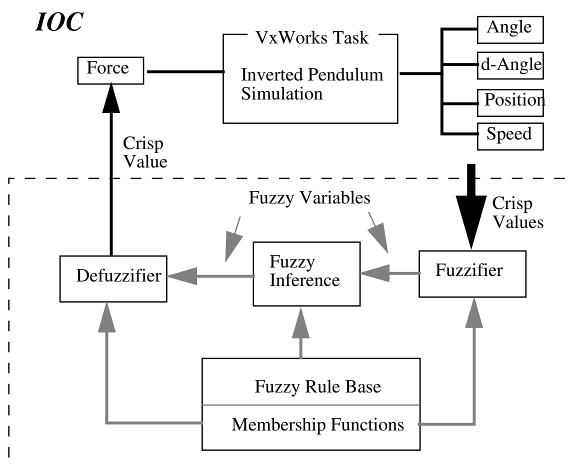
B. Fuzzy Control Feedback System

In problems involving signal processing, the signal is usually transformed to the frequency domain, processed and then transformed back to the time domain. This allows the use of processing techniques which would otherwise be too complicated in the time domain. Similarly, fuzzy control system design consists of a fuzzification step, fuzzy inference and a defuzzification step. Fuzzification involves transforming inputs from crisp values to fuzzy variables. Fuzzy inference is the process of applying fuzzy rules to the fuzzified input values and calculating the fuzzy outputs. In the last step, a defuzzifier transforms the fuzzy outputs back to the crisp values. Figure 4 shows the configuration of a fuzzy controller as applied to the inverted pendulum problem.

i) Fuzzy Rules and Membership Functions

Imagine that you are the operator who is trying to balance the pendulum by pushing or pulling the cart as shown in Figure 1. You will probably come up with some control rules similar to what follows to balance the pendulum.

Figure 4. Fuzzy Control to Feedback System



If the pole is leaning slightly to the left and falling slowly, then push the cart slowly to the left.

If the pole is leaning moderately to the right and rising quickly, then don't move the cart.

Here the "slightly to the left", "moderately to the right" and "rising quickly" are fuzzy terms which can be quantified by the membership functions. There are many ways to define a membership function. The common ones are triangle function, trapezoid function, S-function, Z-function, Pi-function and general fuzzy set defined by a set of pairs[5]

Angular Velocity

Table 1: Fuzzy Rules

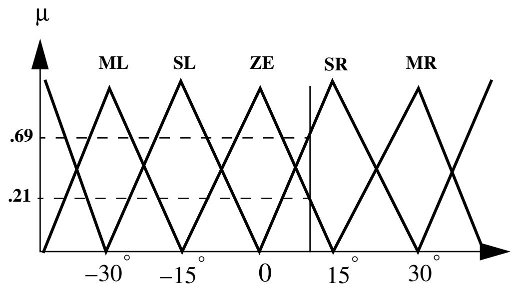
Angle	*Force*	**ML**	**SL**	ZE	SR	MR
	ML	*MR*	*MR*	*MR*	*MR*	*SR*
	SL	*MR*	*MR*	*SR*	*ZE*	*ZE*
	ZE	*SR*	*SR*	*ZE*	*SL*	*SL*
	SR	*ZE*	*ZE*	*SL*	*ML*	*ML*
	MR	*SL*	*ML*	*ML*	*ML*	*ML*

The rules for balancing the inverted pendulum problem are listed in Table 1. Each rule takes two inputs, angle and angular velocity, and produces one output, force. Five membership functions are defined for every input and output variable: ML, SL, ZE, SR and MR where ML stands for Moderately to the Left, SL Slightly to the Left, ZE zero and so forth.

ii) Fuzzification

Consider the pendulum angle θ. Its range extends from -90° to 90° with a current value of 10°. Figure 5 illustrates the process of transferring this crisp value to fuzzy values and identifies its truth values or the degrees of membership.

Figure 5. Fuzzification

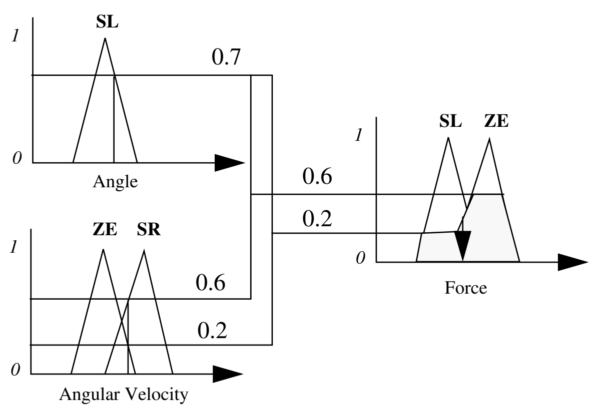


$\mu_{SR}(\theta = 10°) = 0.69 \quad \mu_{ZE}(\theta = 10°) = 0.21$

iii) Inference

In fuzzy inference, all the rules are fired. However, each rule may have a different degree of truth in its premise and will hence contribute differently to the outcome. There are two principal methods of inference in fuzzy systems: the min-max method and the fuzzy additive method. Min-Max inference technique was used in the control of the simulation and is illustrated in Figure 6.

Figure 6. Min-Max Inference Method



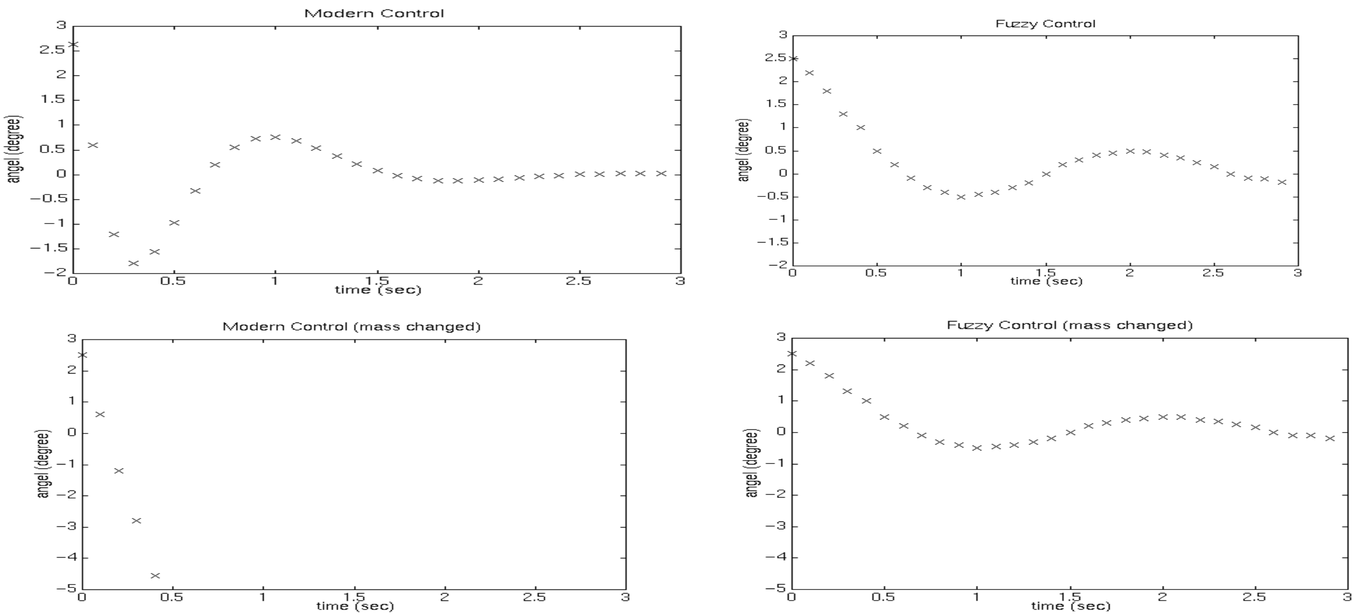
Rule 1: If Angle is SL and d-Angle is ZE then Force is SL (0.2)

Rule 2: If Angle is SL and d-Angle is SR then Force is ZE (0.6)

V. SIMULATION RESULTS

Figure 7 shows two sets of results comparing the application of modern and fuzzy control techniques to the inverted pendulum simulator. In the first set, the fuzzy controller gives less overshoot, shorter settling time, but with more error. The reason for the increase in error in the fuzzy control is due to the coarseness and the discrete nature of the membership function. For example, in the simulation, zero was defined as +0.5 to -0.5 degrees of the vertical. If desired, one may define zero to be a smaller range, e.g., +0.1 to -0.1 degrees. In the second set, we changed the mass of the pendulum without modifying the controllers. The results show that the simulation with fuzzy controller exhibited little performance degradation due to this parameter change. This is due to the fact that fuzzy logic is not based on the mathematical model of the inverted pendulum. The modern controller, on the other hand, totally failed to stabilize the pendulum as it was designed for the initial value of the rod mass.

Figure 7. Simulation Results



VI. SUMMARY and FUTURE PLANS

A fuzzy engine has been implemented in the EPICS and VxWorks environment. It has been used to control an inverted pendulum simulation. The study has identified some of the potential benefits of using fuzzy logic controllers. In comparison with the modern control theory, fuzzy logic is simpler to implement as it eliminates the complicated mathematics of the modelling process and uses a set of control rules instead. Fuzzy control is also more robust to parameter variations.

There are numerous accelerator control problems which involve phenomena that are not amenable to simple mathematical modeling. There is also a class of problems that are simple in nature but are difficult to codify for algorithmic control. Typical problems include the control of electron bunch length in the accelerator or temperature stabilization of beam targets. It is interesting to note that expert operators are able to control many of these problems manually without having a clear understanding of the underlying phenomena. As a follow up to the inverted pendulum study, we intend to simulate and develop a fuzzy controller for the bunch length problem in the CEBAF linac.

Another planned study involves the development of an adaptive fuzzy controller which includes using neural networks to update the membership function. This will then be used in a comparative study of adaptive fuzzy orbit control with optimal control feedback loops that have had their system matrices updated through system identification techniques. A number of these optimal control loops are under development for the CEBAF accelerator facility. The major objective would be to study the robustness of either technique with respect to changes in accelerator parameters, such as quadrupole focusing or cavity energy output.

References

[1] L. Zadeh, "Fuzzy Sets," in *Selected Papers by L. A. Zadeh*
[2] Katsuhiko Ogata, "Modern Control Engineering," Second Edition. 1990, pp. 105
[3] Katsuhiko Ogata, "Modern Control Engineering," Second Edition. 1990, pp. 312
[4] Katsuhiko Ogata, "Modern Control Engineering," Second Edition. 1990, pp. 782
[5] Earl Cox, "The Fuzzy Systems Handbook," 1994
[6] Bart Kosko, "Neural Networks and Fuzzy Systems," 1992