# Development of Fuzzy Algorithms for Servo Systems

Y. F. Li and C. C. Lau

ABSTRACT: This paper investigates the possibility of applying fuzzy algorithms in a microprocessor-based servomotor controller, which requires faster and more accurate response compared with other industrial processes. The performance of proportional-integral-derivative control, model reference adaptive control, and fuzzy controllers is compared in terms of steady-state error, settling time, and response time. Limitations of fuzzy control algorithms are also described.

## Introduction

Servomotors are used in many automatic systems, including drives for printers, tape recorders, and robotic manipulators. Since the development of microprocessors in the 1970s, microprocessor-based servomotor controllers have become more popular because of their superiority over analog controllers. One obvious reason is that the microprocessors can be used to implement intelligent control algorithms to cope with varying environments as a result of load disturbances, process nonlinearities, and changes of plant parameters. A microprocessor-based controller can remember past experiences and predict future consequences caused by present inputs. This is beyond the ability of an analog controller.

Conventional digital control algorithms can be developed by formulating the transfer function of the process [1]. However, in practice it is not always easy to describe an engineering object by means of a discrete transfer function so as to realize ideal compensation. Thus, servomotors are usually controlled by proportional-integral-derivative (PID) algorithms. Such algorithms will be effective enough if the speed and accuracy requirements of the control system are not critical. The usual way to optimize the control action is to tune the PID coefficients, but this cannot cope with a varying control environment or system nonlinearity.

Presented at the 1988 IEEE International Conference on Robotics and Automation, Philadelphia, Pennsylvania, April 24–29, 1988. Y. F. Li and C. C. Lau are with the Department of Electrical and Electronic Engineering, University of Hong Kong, Pokfulam, Hong Kong.

The Model Reference Adaptive Control (MRAC) technique [2] is an approach for coping with environmental variation and system nonlinearities. Its function is to compare the output from the process with that from a reference model. The error is then used for adjusting the parameters of the controller through a suitable adaptation algorithm, either based on physical or chemical laws or a parameter estimation method. These techniques are usually complex and require large amounts of computation time. This restricts the application when a fast response is desirable. The common difficulty with this approach lies in the attempt to formulate the input-output relationship by means of mathematical models, which may be difficult in many cases. Even when such models are developed, they may be too complex to compute in real time.

Facing these problems, investigators realized that incorporating human intelligence into automatic control systems would be a more efficient solution, and this led to the development of fuzzy control algorithms [3]. The fuzzy algorithm is based on intuition and experience, and can be regarded as a set of heuristic decision rules or "rules of thumb." Such nonmathematical control algorithms can be implemented easily in a computer. They are straightforward and should not involve any computational problems.

Mamdani [4] and Mamdani and Assilian [5] reported on the application of fuzzy set theory to control a small laboratory steam engine. The purpose was to regulate engine speed and boiler steam pressure by using heat applied to the boiler and the throttle setting on the engine. At the same time, van Nauta Lemke and Kickert [6]–[8] examined the performance of a fuzzy controller on a warm water plant. The success of these studies led King and Mamdani [9] to attempt to control the temperature of a chemical reactor by using fuzzy algorithms. Furthermore, Rutherford [10] and Ostergaard [11] reported the results of the experiments with fuzzy controllers on a sinter strand and a heat exchanger, respectively. Lastly, R. M. Tong's paper [12] was concerned with the control of a pressurized tank containing liquid. The problem was to use fuzzy algorithms to regulate the total pressure and the level of liquid inside the tank.

The results of those experiments showed that fuzzy controllers performed better than, or at least as good as, a PID controller. They have the common feature of not requiring a detailed mathematical model. However, those experiments were concerned mainly with slow chemical processes. The objective of this paper is to investigate the possibility of applying fuzzy algorithms in faster, and more accurate, controllers, such as servomotor position controllers.

## Fuzzy Control Algorithm

When controlling a process, human operators usually encounter complex patterns of quantitative conditions, which are difficult to interpret accurately. The magnitude of the measurements is usually described as fast, big, slow, high, etc. To represent such inexact information, a nonmathematical approach called "fuzzy set theory" was developed by Zadeh [3].

A fuzzy subset $A$ with an element $x$ has a membership function of $u_A(x)$, which is in the interval between 0 and 1. If $u_A(x)$ is 1, then the element is a member of the set. If $u_A(x)$ is 0, then it is not. Consider a fuzzy subset $A$ with five elements, which have the membership functions 0.7, 0.9, 1, 0.9, and 0.7. From the preceding concept, the element with a membership function of 1 is a full member of $A$, whereas the others are only part member. The membership function determines the degree to which the element belongs to the subset. If a fuzzy set $A$ is defined as "around 10" on a scale from 8 to 12, it might be described by the following, where 0.7, 0.9, and 1 are the membership functions and 8, 9, 10, 11, and 12 are called the universe of discourse.

$$A = (0.7/8, 0.9/9, 1/10, 0.9/11, 0.7/12)$$

## Fuzzy Logic

Fuzzy subset theory involves very complicated theorems, but most of these theorems do not relate to the development of fuzzy control algorithms. The following three definitions form the basis for the decision

0272-1708/89/0400-0065 $01.00 

table that will be used in the control algorithms.

(1) The union of two sets, $A \cup B$, corresponds to the OR function and is defined by

$$u(A \text{ OR } B) = \max(u_A(x), u_B(x))$$

(2) The intersection of two sets, $A \cap B$, corresponds to the AND function and is defined by

$$u(A \text{ AND } B) = \min(u_A(x), u_B(x))$$

(3) The complement of a set $A$ corresponds to the NOT function and is defined by

$$u(\text{NOT } A) = 1 - u_A(x)$$

To illustrate the application of these definitions, consider two qualitative statements, "big" and "medium," with the following membership functions:

$$u(\text{big}) = \{0, 0.3, 0.7, 1.0\}$$

$$u(\text{medium}) = \{0.2, 0.7, 1.0, 0.8\}$$

The three definitions can be applied directly to the two membership functions.

$$\begin{aligned} u(\text{big OR medium}) &= \{\max(0, 0.2), \max(0.3, 0.7), \\ &\quad \max(0.7, 1.0), \max(1.0, 0.8)\} \\ &= \{0.2, 0.7, 1.0, 1.0\} \end{aligned}$$

$$\begin{aligned} u(\text{big AND medium}) &= \{\min(0, 0.2), \min(0.3, 0.7), \\ &\quad \min(0.7, 1.0), \min(1.0, 0.8)\} \\ &= \{0, 0.3, 0.7, 0.8\} \end{aligned}$$

$$\begin{aligned} u(\text{NOT big}) &= \{(1 - 0), (1 - 0.3), (1 - 0.7), \\ &\quad (1 - 1.0)\} \\ &= \{1, 0.7, 0.3, 0\} \end{aligned}$$

To establish the fuzzy controller, it is necessary to interpret rules that are based on experience so as to form a decision table that gives input and output values of the controller corresponding to situations of interest.

A fuzzy algorithm consists of situation and action pairs. Conditional rules expressed in IF and THEN statements are generally used. For example, the control rule might be: If the output is lower than the requirement and the output is dropping moderately, then the input to the system shall be increased greatly. Such a rule has to be converted into a more general statement for application to fuzzy algorithms. To achieve this, the following terms are defined: error equals the set point minus the process output; error change equals the error from the process output minus the error from the last process output; and control input equals the input applied to the process.

In addition, it is necessary to quantize the qualitative statements, and the following linguistic sets are assigned.

- Large positive (LP)
- Medium positive (MP)
- Small positive (SP)
- Zero (ZE)
- Small negative (SN)
- Medium negative (MN)
- Large negative (LN)

Thus, the statement of the example control rule will be: If the error is large positive and the error change is small positive, then the input to the system is large positive.

## Membership Matrix

Having formulated the control rule in fuzzy terms, the next step is to define the membership functions of the linguistic sets, i.e., large positive, medium positive, etc. The shape of the fuzzy set is quite arbitrary and depends on the user's preference. For simplicity, trapezoidal shapes usually are used.

Table 1 is an example of a membership matrix table for a membership function. It includes the error, error change, and control input variables. Each table consists of five sets, including LP, SP, ZE, SN, and LN, and each set consists of *nine* elements, i.e., −4, −3, −2, −1, 0, 1, 2, 3, 4. All error, error change, and control input variables are quantized to these nine levels.

## Control Rules

Suppose four fuzzy rules have been formulated for a system; namely:

(1) If the error is zero and the error change is small positive, then the control input is small negative.

(2) If the error is zero and the error change is zero, then the control input is zero.

(3) If the error is small negative and the error change is small negative, then the control input is small positive.

(4) If the error is small negative and the error change is zero, then the control input is large positive.

These rules are then combined to form a decision table for the fuzzy controller. The table consists of values showing the different situations experienced by the system and the corresponding control input function.

## Decision Table

To show the preparation of the decision table, the four rules are interpreted as functional diagrams, as shown in Fig. 1. Consider a process having an error of −1 and an error change of 1.5. From the diagram, it can be shown that, for such an error/error change pair, rules 1, 2, and 4 are applicable.

The points of intersection between the values of −1 and the graph in the first column (i.e., error) have the membership functions of 0.6, 0.6, 0.6, and 0.6. Likewise, the second column (i.e., error change) shows that an error change of 1.5 has the membership functions of 0.8, 0.2, 0, 0.2. The control input for the four rules is the intersection of the paired values obtained from the graph, i.e., min (0.6, 0.8), min (0.6, 0.2), min (0.6, 0), and min (0.6, 0.2), which reduces to 0.6, 0.2, 0, and 0.2, respectively.

The membership functions representing the control adjustment are weighted according to the corresponding input change and the different control contributions as shown in Fig. 2. Now, for a pair of error and error changes, three sets of control inputs exist. To determine the value of action to be taken from these contributions, we can either choose the maximum value or use the "center-of-gravity" method. In our example, the maximum value is 0.6, which corresponds to a control input of approximately −2 units (Fig. 2). In the center-of-gravity method, the action is given by the center of the summed area,

**Table 1**
**Membership Matrix Table**

| Linguistic Sets | Quantized Levels | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | −4 | −3 | −2 | −1 | 0 | 1 | 2 | 3 | 4 |
| LP | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.6 | 0 |
| SP | 0 | 0 | 0 | 0 | 0 | 0.6 | 1 | 0.6 | 0 |
| ZE | 0 | 0 | 0 | 0.6 | 1 | 0.6 | 0 | 0 | 0 |
| SN | 0 | 0.6 | 1 | 0.6 | 0 | 0 | 0 | 0 | 0 |
| LN | 1 | 0.6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | Membership function | | | | | | | | |

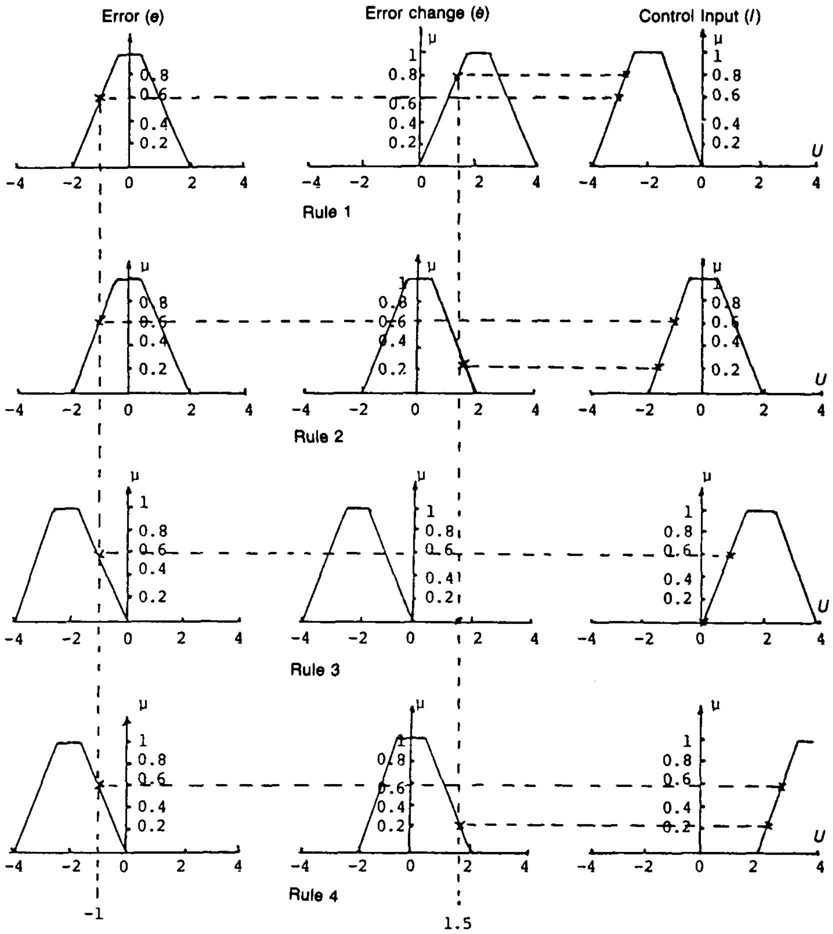


Fig. 1. Graphical representation of control rules.

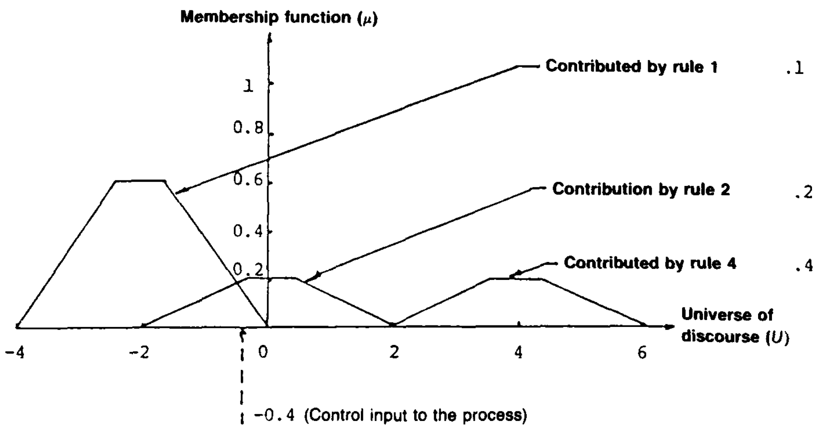


Fig. 2. Determination of the control input to process by means of the center-of-gravity method.

which is contributed by the control inputs. In this study, the latter is chosen because the contributions have the same maximum value. Furthermore, the center-of-gravity method gives a more reliable decision table compared with the "maximum" operation.

Thus, for an error of $-1$ and an error change of 1.5, the control input $I$ will be

$$I = [0.6 \times (-2) + 0.2 \times 0 + 0 \times 0 + 0.2 \times 4]/(0.6 + 0.2 + 0 + 0.2) = -0.4$$

In general terms, the control input can be written as follows, where $u$ is the membership function, $U$ the universe of discourse, and $n$ the number of contributions (i.e., 1, 2, . . . , etc.).

$$I = \sum_{1}^{n} (u_n \times U_n) \Big/ \left( \sum_{1}^{n} u_n \right)$$

Figure 2 shows the control input to the process graphically, and the contributions are developed from rules 1, 2, and 4 of Fig. 1. In most cases, the number of rules that define different input conditions are limited, and there is a good possibility that no rule exists for certain inputs. Such undefined situations lower the efficiency of the fuzzy controller. To overcome this, the effect of all rules is spread around the input situations in all directions to a distance that is determined by the user. Thus, for a given input, the control algorithms will check if there exists a corresponding rule. If no rule exists, then the rules in the immediate neighborhood (within the predetermined distance) will be considered.

In the following example, a distance of 1 unit is chosen. Consider at one instant, the situation where the system has an error of $-3$ and an error change of 2. No rule appears to correspond to this situation. However, rule 4 can be adopted within a distance of 1 unit (Fig. 1).

These calculations can be implemented on a computer. After the calculations, each error/error change pair will have their corresponding control input values. The decision table is stored in memory in the form of a "lookup" table. An example of the lookup table is shown in Table 2.

The following procedure shows how a control input to the process is determined from the lookup table.

- Suppose the set point = 1 unit
- Output of the system at $t_1$ = 4 units
- Output of the system at $t_2$ = 2 units
- Error at $t_1 = 4 - 1 = 3$ units
- Error at $t_2 = 2 - 1 = 1$ unit
- Error change $= 1 - 3 = -2$ units

From the table, the quantized control input for the system at $t_2$ will be 1 unit. If the scaling factor of the quantized control input is 0.5 per unit, then the absolute value for control input is $1 \times 0.5 = 0.5$.

However, the preceding rules may not be adequate to cover the wide range of different situations the system may encounter. In this respect, different lookup tables may be required in order to increase the adaptability of the fuzzy controller.

**Table 2**
**Sample Lookup Table**

| Error | Error Change −4 | −3 | >−2 | −1 | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|---|---|---|---|
| −4 | 5 | 4 | 4 | 3 | 3 | 2 | 1 | 1 | −1 |
| −3 | 5 | 4 | 3 | 2 | 2 | 1 | 0 | 0 | −2 |
| −2 | 4 | 3 | 3 | 2 | 1 | 1 | 0 | −1 | −3 |
| −1 | 4 | 3 | 2 | 1 | 1 | 0 | 1 | −2 | −3 |
| 0 | 3 | 3 | 2 | 1 | 0 | −1 | −2 | −3 | −3 |
| 1 | 3 | 2 | >1 | 0 | −1 | −1 | −2 | −3 | −4 |
| 2 | 3 | 1 | 0 | 0 | −1 | −2 | −3 | −3 | −4 |
| 3 | 2 | 0 | 0 | −1 | −2 | −2 | −3 | −4 | −5 |
| 4 | 1 | −1 | −1 | −1 | −2 | −3 | −4 | −4 | −5 |
| | | | | | Control input | | | | |

The notation > illustrates the example.

## Fuzzy Control Algorithms for Servomotors

The task of the control algorithm is to rotate the shaft of the motor to a set point without overshoot. It is necessary to write a set of fuzzy control statements based on the error signal between the set point and the measured shaft position and the change of error so as to adjust the output of the drive unit.

The inputs of the fuzzy controller are denoted by the following, where $O$ is the output from the shaft encoder, $S$ the set point, $e_1$ the error of the servo system at $t_1$, and $e_2$ the error of the servo system at $t_2$.

$$e = O - S$$

$$\dot{e} = e_1 - e_2$$

The output is denoted by $v$, which is the voltage output from the servo drive unit.

In this investigation, the shaft encoder is an incremental type having a resolution of 1000, and the servo amplifier has an output range of ±30 V. The universes of discourse of these functions are as follows: $e$ equals −1000 to +1000, $\dot{e}$ equals −100 to + 100, and $v$ equals −30 to +30. Table 3 shows the quantized variables. A pure verbal formulation is then carried out to control the servo system.

Counterclockwise and clockwise rotations are defined as positive and negative, respectively. A corresponding output is given for the error and error change detected in each sampling interval. For example, if the error is positive large and the error change is positive small then a large positive drive is used.

All of the preceding strategies are combined to form a series of rules and six numbers as follows:

(1) If $e$ is LP and $\dot{e}$ is any, then $v$ is LP.

(2) If $e$ is SP and $\dot{e}$ is SP or ZE, then $v$ is SP.

(3) If $e$ is ZE and $\dot{e}$ is SP, then $v$ is ZE.

(4) If $e$ is ZE and $\dot{e}$ is SN, then $v$ is SN.

(5) If $e$ is SN and $\dot{e}$ is SN, then $v$ is SN.

(6) If $e$ is LN and $\dot{e}$ is any, then $v$ is LN.

Rule 1 implies a general condition when the present position of the shaft is very far away from the set point. Therefore, it requires a large drive output to turn the motor shaft to the set point quickly. Rule 2 implements the condition when the error starts to decrease and the motor is approaching the required position. Thus, a small drive output is given. Rule 3 implies that the set point is very nearly reached. Because of the inertia of the motor, it is necessary to stop the drive at this instant to keep the overshoot at a minimum. However, rule 4 deals with the condition when overshoot does occur. A small reverse drive signal is given to bring the motor to its set point. Rule 6 implies the reverse condition of rule 1.

These rules are then combined to form the decision table (lookup table) shown in Table 4. However, the lookup table may not provide optimum control when the error is approaching zero. This will lead to overshoot and hunting around the desired position. Thus, Table 4 is used only for coarse control and a second lookup table is formulated for fine control (Tables 5 and 6).

The two lookup tables are stored in the memory of a 6502-based microcomputer, which checks whether the error is within the predetermined limit and then assigns the corresponding lookup table to the control function. In this application, the limit is set between +100 and −100. When the error falls within such limits, the computer will switch to the fine control lookup table. This speeds up the response at regions around the set point. A block diagram of the fuzzy controller is presented in Fig. 3.

## Simulation Results

For comparison purposes, three different control algorithms are implemented in a 6502-based microcomputer with a BASIC interpreter. The first is a conventional digital PI controller, the second is an MRAC controller. The third is the fuzzy algorithm described in the previous section.

A servomotor having the following parameters is chosen for this simulation: moment amplification ($K_t$) equals 1.088 Nm/A; electrical amplification ($K_1$) equals 1.1 V/rad/sec; armature inductance ($L_a$) equals 10 mH; armature resistance ($R_a$) equals 3.5 Ω; and the moment of inertia of the motor ($J$) equals 0.0945 kg-m. Using these constants, the poles of the transfer function of the motor are found to be the following:

$$S(1 + S/3.6)\ (1 + S/350)$$

Because the electrical time constant (0.00285 sec) is much smaller than the mechanical time constant (0.28 sec), the trans-

**Table 3**
**Quantized Variables (Coarse Control)**

| $e$ | $\dot{e}$ | $v$ | Quantized Level |
|---|---|---|---|
| −1000 | −100 | −30 | −5 |
| − 800 | − 80 | −24 | −4 |
| − 600 | − 60 | −18 | −3 |
| − 400 | − 40 | −12 | −2 |
| − 200 | − 20 | − 6 | −1 |
| 0 | 0 | 0 | 0 |
| 200 | 20 | 6 | 1 |
| 400 | 40 | 12 | 2 |
| 600 | 60 | 18 | 3 |
| 800 | 80 | 24 | 4 |
| 1000 | 100 | 30 | 5 |

**Table 4**
**Lookup Table (Coarse Control)**

| | Error Change | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Error** | **−5** | **−4** | **−3** | **−2** | **−1** | **0** | **1** | **2** | **3** | **4** | **5** |
| −5 | −5 | −5 | −4 | −4 | −3 | −1 | 0 | 1 | 1 | 2 | 2 |
| −4 | −5 | −5 | −5 | −4 | −3 | −1 | 0 | 1 | 1 | 2 | 2 |
| −3 | −5 | −5 | −5 | −4 | −3 | −1 | 0 | 1 | 2 | 2 | 2 |
| −2 | −5 | −5 | −4 | −4 | −3 | −1 | 1 | 2 | 3 | 3 | 3 |
| −1 | −5 | −5 | −4 | −3 | −2 | 0 | 1 | 2 | 3 | 3 | 4 |
| 0 | −5 | −4 | −4 | −2 | −2 | 0 | 1 | 3 | 4 | 4 | 4 |
| 1 | −4 | −4 | −3 | −2 | −1 | 0 | 2 | 3 | 4 | 4 | 4 |
| 2 | −4 | −4 | −3 | −2 | −1 | 1 | 2 | 3 | 5 | 4 | 5 |
| 3 | −3 | −4 | −2 | −1 | 0 | 1 | 2 | 3 | 5 | 4 | 5 |
| 4 | −2 | −3 | −2 | −1 | 0 | 1 | 3 | 3 | 5 | 5 | 5 |
| 5 | −2 | −3 | −1 | −1 | 0 | 1 | 3 | 3 | 5 | 5 | 5 |
| | | | | | Control input | | | | | | |

**Table 5**
**Quantized Variables (Fine Control)**

| $e$ | $\dot{e}$ | $v$ | Quantized Level |
|---|---|---|---|
| 100 | 60 | 20 | 4 |
| 75 | 45 | 15 | 3 |
| 50 | 30 | 10 | 2 |
| 25 | 15 | 5 | 1 |
| 0 | 0 | 0 | 0 |
| − 25 | −15 | − 5 | −1 |
| − 50 | −30 | −10 | −2 |
| − 75 | −45 | −15 | −3 |
| −100 | −60 | −20 | −4 |

**Table 6**
**Lookup Table (Fine Control)**

| | Error Change | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Error** | **−4** | **−3** | **−2** | **−1** | **0** | **1** | **2** | **3** | **4** |
| −4 | −4 | −4 | −3 | −3 | −2 | −1 | −1 | 0 | 1 |
| −3 | −4 | −3 | −3 | −2 | −2 | −1 | 0 | 1 | 1 |
| −2 | −3 | −2 | −2 | −1 | −1 | −1 | 0 | 1 | 2 |
| −1 | −2 | −2 | −1 | 0 | 1 | 0 | 1 | 2 | 2 |
| 0 | −2 | −2 | −1 | 0 | 0 | 0 | 1 | 2 | 3 |
| 1 | −2 | −2 | 0 | 1 | 1 | 1 | 2 | 3 | 3 |
| 2 | −1 | −1 | 0 | 1 | 1 | 1 | 2 | 3 | 4 |
| 3 | 0 | 0 | 1 | 2 | 2 | 2 | 3 | 3 | 4 |
| 4 | 0 | 0 | 1 | 2 | 3 | 3 | 3 | 4 | 4 |
| | | | | | Control input | | | | |

fer function of the motor is simplified to

$$G(S) = (K/S)\ (S + 3.6)$$

The partial fraction expansion for $G(S)$ is

$$G(S) = (K/3.6)\ [(1/S) - 1/(S + 3.6)]$$

Taking the sampling period $T$ equal to 0.25 sec, the characteristic equation ($Z$-transform) for this system becomes

$$(Z - 1)\ (Z - 0.4) + 0.165\ KZ = 0$$

By using the Routh criteria, the system is shown to be unstable for $K > 16.97$. In this investigation, $K$ is chosen to be 1. Finally, the transfer function of the servomotor becomes

$$G(Z) = 0.165Z/[(Z - 1) \cdot (Z - 0.4) + 0.165Z]$$

The transfer function of the servo system (Fig. 4) is transformed into the following difference equation:

$$Y_n = 1.235\ Y_{n-1} - 0.4\ Y_{n-1} + 0.165\ X_{n-1}$$

During the simulation, the sampling time and the set points were common to all three algorithms and of the values 0.25 sec and 23 deg (360 × 64/1000), respectively. In order to test the adaptability of the controllers, a disturbance is simulated in the servo system. This disturbance results in a slight change in the mechanical time constant (from 3.6 to 4) after the servo has settled. The difference equation of the modified system transfer function is

$$Y_n = 1.21\ Y_{n-1} - 0.368\ Y_{n-2} + 0.158\ X_{n-1}$$

*PI Controller*

The digital controller used in this investigation is in the usual form, where $U$ is the output of the controller, $E$ the output of the system minus the set point, and $K_1$, $K_2$, and $K_3$ are the coefficients of the controller.

$$U_n = K_1(E_n + K_2 \times U_{n-1}) + (K_3 \times E_{n-1})$$

The coefficients for the PI controller are tuned for best performance, i.e., with minimum overshoot and no steady-state error, and the values are found to be $A$ equals −2.79, $B$ equals 0.90625, and $C$ equals −0.001.

*MRAC*

A block diagram of the MRAC loop is shown in Fig. 4, with the reference model being described by

$$1/S(S + 3.6)$$

Variation on any parameter of the servo system can be adjusted by $K \times e$; $e = Y_m - Y_p$, where $Y_m$ and $Y_p$ are the outputs of the model and process, respectively. The reference model is of the same order as the process and is linear as well.

In order to get the two identical responses, the parameter $K$ must be adjusted. It is obvious that $K$ should be increased, and a reasonable choice for adjustment of $K$ seems to be

$$K(t) = K(0) + B \int e\ dt$$

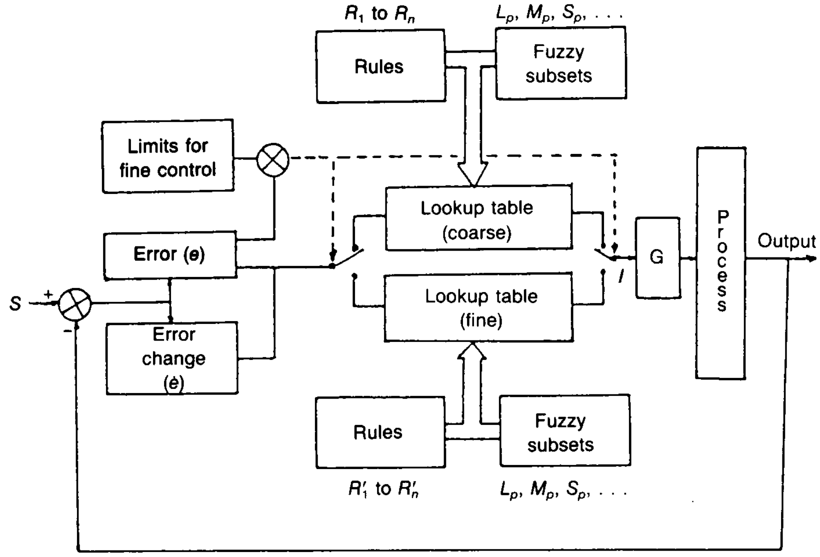



Fig. 3. Block diagram of fuzzy controller.

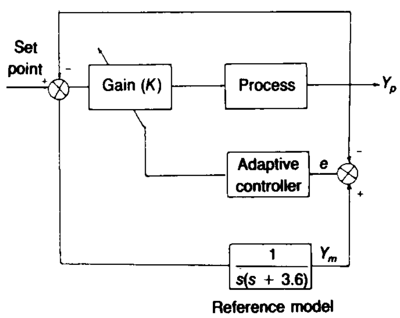



Fig. 4. Block diagram of Model Reference Adaptive Controller (MRAC).

With the gain $B$, the speed of adjustment can be set, and the desired memory function is realized by means of integration.

Note that when the input signal $u$ is inverted, the adjustment of $K$ will be in the wrong direction because of the negative sign of $e$. This will result in an unstable system. Thus, it is necessary to include the sign of the input signal. For instance, by multiplying $e$ and $u$, the result of the parameter adjustment will conform to the adaptation criteria. The adjustment law is modified as follows, where $B$ equals $-0.007$.

$$K(t) = K(0) + B \int (e \times u)\, dt$$

Converting this into a difference equation gives

$$K_n = B \times E_n \times U_n + K_{n-1}$$

### Fuzzy Control

Tables 4 and 6 have been used to implement the fuzzy control algorithm. Entries in these tables are initially estimated based on the characteristics of the servomotor. Then they are fine-tuned by repeated trials. Simulation results are shown in Fig. 5 and Table 7 for comparison purposes. A plot of fuzzy controller output against time is shown in Fig. 6.

To prevent instability problems, the gain for the fuzzy controller is kept as low as possible so that the system remains stable within the range of operation. Unfortunately, this will increase the response time of the system. To overcome this, a third lookup table may be needed for varying the system gain so as to increase the adaptability of the controller.

The simulation programs were written in BASIC. It can be expected that, in practical implementation, where control programs are written in assembly language, faster response times can be expected.

### Summary and Conclusions

The parameters of the controller should be designed so that the servo system can have fast response and minimum steady-state error. To achieve this, the quantization of the parameters is arranged so that the full lookup table can be utilized. Also, two sets of algorithms are used to describe the control

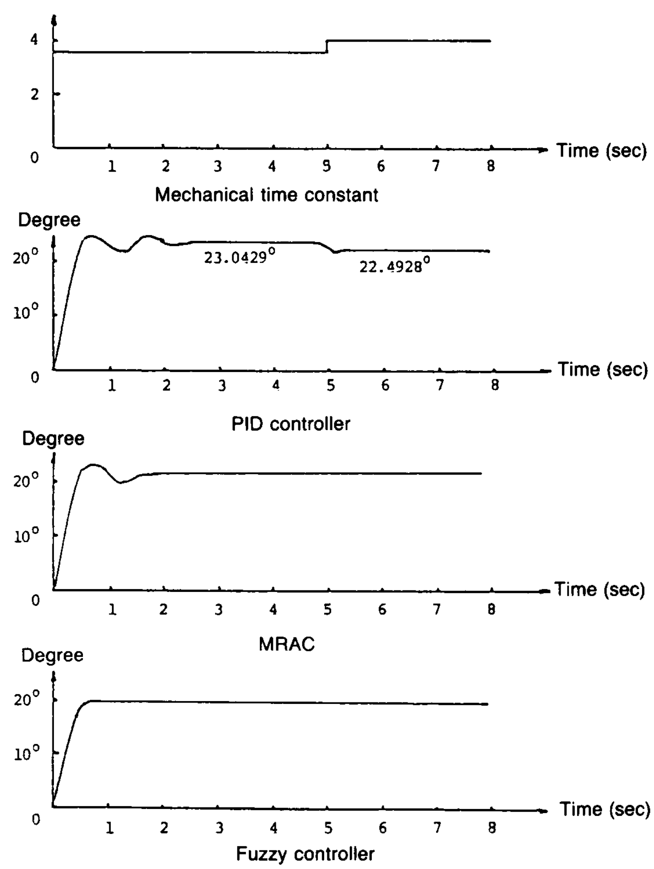



Fig. 5. Simulation results for three different controllers.

**Table 7**
**Comparison of Simulation Results**

| | PI | MRAC | Fuzzy |
|---|---|---|---|
| Settling time, sec | 2.5 | 2 | 1 |
| Overshoot, % | 10 | 6 | 0 |
| Effect of disturbance | Yes (Steady-state error of 3%) | No | No |

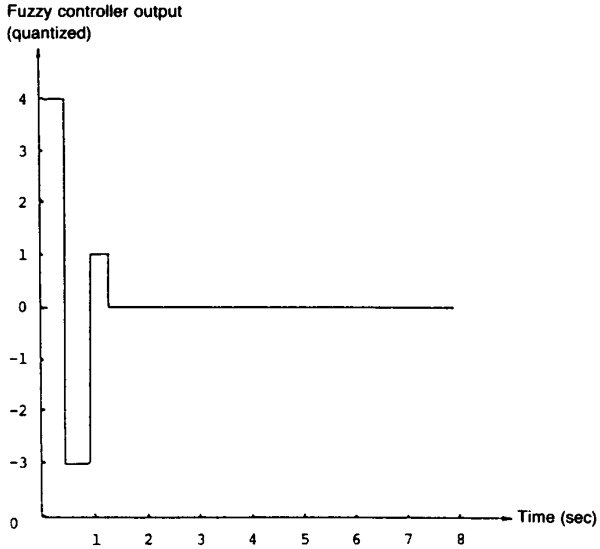


Fig. 6. Fuzzy controller output.

function of the servo system; namely, coarse and fine control. This greatly reduces the system settling time.

By comparing the simulated results obtained from the fuzzy, MRAC, and PI controllers, it can be seen that the performance of the fuzzy controller is better than that of the PI controller and as good as that of the MRAC. As can be seen, the settling time of the fuzzy controller is only one-half that of the MRAC controller and two-fifths that of the PI controller. Both the MRAC and fuzzy controllers are insensitive to the simulated disturbance, which is done by slightly altering the mechanical time constant. It is important to note that the fuzzy algorithms have the distinct advantage of not relying on a mathematical transfer function for formulating control rules. Instead, the fuzzy algorithms rely mainly on the overall knowledge of the designer. However, an optimum response of the fuzzy controller can be expected only for a limited range of inputs, and it is necessary to retune the controller (adjusting the scaling factor and, in some cases, the magnitude of the parameters) for other ranges of input. This is because the controller has been dimensioned and formulated in a very straightforward way on the basis of the basic operational characteristics of the servomotor. The limited sets of rules and lookup tables restrict the adaptability of the controller.

Since there is no mature guidance in fuzzy set theory for the determination of the best shapes for fuzzy sets, it is suggested that different shapes for different set points need to be studied to obtain an optimum solution for various ranges of error/error change pairs. The amount of overlap with the fuzzy sets affects the efficiency of the fuzzy controller. In case of too much overlap, many rules will be applied for a single-input pair, and the situation will not be represented accurately. If there is too little overlap, it will be difficult to derive the lookup table.

In summary, it has been shown that fuzzy controllers offer the following advantages:

(1) They do not require a detailed mathematical model to formulate the algorithms.

(2) Because both error and error change are required to evaluate the control input, the fuzzy controller has more adaptive capability.

(3) By using different sets of control rules, the fuzzy controller can operate for a large range of inputs.

However, certain challenges remain for their use, including the following:

(1) Completeness of the rule base,

(2) Guidance on the shape of the fuzzy linguistic functions,

(3) Guidance on the overlapping of subsets, and

(4) Practical methods for controller calibration.

## References


[1] Raven, *Automatic Control Engineering*, International Student Edition, McGraw-Hill, 1968.

[2] A. J. Dymock, "Analysis of a Type of Model Reference Adaptive Control System," *Proc. IEE*, vol. 112, pp. 743-753, 1965.

[3] L. A. Zadeh, "Fuzzy Sets," *Inform. Contr.*, vol. 8, pp. 338-353, 1965.

[4] E. H. Mamdani, "Application of Fuzzy Algorithms for Control of a Simple Dynamic Plant," *Proc. IEEE*, vol. 121, pp. 1585–1588, 1974.

[5] E. H. Mamdani and S. Assilian, "A Fuzzy Logic Controller for a Dynamic Plant," *Intl. J. Man-Machine Stud.*, vol. 7, pp. 1-13, 1975.

[6] H. R. van Nauta Lemke and W. J. M. Kickert, "The Application of Fuzzy Set Theory to Control a Warm Water Process," *Journal A17*, vol. A17, pp. 8–18, 1976.

[7] W. J. M. Kickert and H. R. van Nauta Lemke, "Application of a Fuzzy Controller in a Warm Water Plant," *Automatica*, vol. 12, pp. 301–308, 1976.

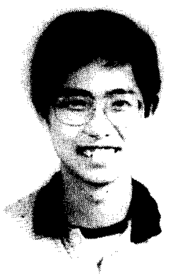
[8] W. J. Kickert, "Further Analysis and Application of Fuzzy Logic Control," Internal Report F/WK 2/75, Queen Mary's College, London, 1975.

[9] P. J. King and E. H. Mamdani, "The Application of Fuzzy Control Systems to Industrial Processes," *Automatica*, vol. 13, pp. 235–242, 1976.

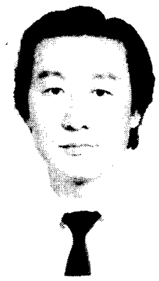
[10] D. Rutherford and G. Z. Carter, "A Heuristic Adaptive Controller for a Sinter Plant," *Proc. 2nd IFAC Symp.*, Johannesburg, 1976.

[11] J. J. Ostergaard, "Fuzzy Logic Control of a Heat Exchanger Process," Internal Report 7601, Electric Power Engineering Dept., Tech. Univ. of Denmark, 1976.

[12] R. M. Tong, "Some Problems with the Design and Implementation of Fuzzy Controllers," Internal Report CUED/F-CAMS/TR127, Engineering Dept., Cambridge Univ., 1976.



**Y. F. Li** received the First Class Honours degree in instrumentation and systems engineering in 1980 from Teerside Polytechnic in England. Until 1983, he was an Assistant Electronic Engineer of the Hong Kong United Dockyard. He is currently an Electronic Engineer of the Marine Department of the Hong Kong Government and a M.Phil. degree candidate at the University of Hong Kong. His research interests include fuzzy theory and its application to microprocessor-based controllers.



**C. C. Lau** received the M.Phil. and Ph.D. degrees from the University of Hong Kong in 1974 and 1983, respectively. His research interests include microprocessor-based controllers and hypercube supercomputers. He is currently a lecturer of the Department of Electrical and Electronic Engineering, University of Hong Kong.

# Future Directions in Control Theory

A major report on the status and future directions of control theory, entitled "Future Directions in Control Theory: A Mathematical Perspective," was released in December 1988 and is being distributed by the Society for Industrial and Applied Mathematics (SIAM). Wendell H. Fleming of Brown University chaired the panel of 17 mathematicians and engineers that produced the report. During various stages of its preparation, the panel solicited, and received, valuable input from more than 50 members of the control community. For a copy of the report, write to SIAM at the address listed at the end of this article.

## Challenges

Control theory has grown dramatically from the linear systems, optimum control, and linear filtering of noisy signals of the 1960s into a vastly diverse family of theories of nonlinear, stochastic, adaptive, distributed parameter, discrete event, and intelligent control. Because control research is driven by the diverse and changing needs of applications, the wide variety of mathematical techniques included in control theory go beyond those associated with traditional applied mathematics.

Control theory faces particular challenges arising from its diverse origins and the wide applicability of its research. The field is both an engineering discipline and applied mathematics discipline and, in addition, is experiencing increasing interaction with computer science and computation. The creative interplay between mathematics and engineering in the solution of control problems has been a major strength of the field, but it also raises questions about the raison d'être and the future direction of the field.

In spite of the rapid growth of the field, the panel found that many fundamental problems—such as control of nonlinear multivariable systems, especially those with many degrees of freedom, and control of nonlinear distributed parameter systems (e.g., those governed by nonlinear partial differential equations)—are not yet understood. These fundamental problems give rise to difficult mathematical questions, many of which cannot be answered within the current theoretical framework.

## Advances

The report describes both striking recent advances in the mathematical theory, such as the robust control theory for linear systems, and successful applications to control technology. Among the latter are the space shuttle control systems, a new hormone therapy that is programmed by a nonlinear feedback linearization and decoupling technique, the fly-by-wire F-16 jet, the hot strip steel mill computer control, and a variety of "small" applications that make modern control systems pervasive in today's technological environment.

The report also identifies a strikingly diverse range of areas in science and technology that could benefit from research in control theory; e.g., robotics, combustion control, fluid flow control, solidification processes, biomedical research, hydrology, and economics.

The report strongly encourages control scientists to make the fullest possible use of advanced scientific computing as a research tool. It predicts that major new advances may become possible because of the dramatic increases in computing power, the proliferation of new computing tools, and, to some extent, the availability of new sensor technologies, which open new possibilities for data collection and experimental research on control.

The panel avoided the all-too-easy approach of calling U.S. federal government agencies to double or triple the dollar amounts spent on research in this area, relying, instead, on the importance of the field and the continuing high quality of research as the guarantors of future funding. Questions were raised, however, about the continuing supply of young talent, training opportunities, and communication barriers.

## Recommendations

The panel recommended that academic institutions promote the development of the field by training Ph.D.s in both mathematics and engineering and by facilitating communication across departmental lines. The success of such programs depends on the critical mass of faculty interested in control research.

The panel further recommended that the mathematical and engineering aspects of fundamental control research become an integral part of new research initiatives sponsored by the federal agencies in many areas of science and technology, such as robotics, space structures, and computation.

The control science community, the academic institutions, and the federal agencies were encouraged to promote greater exchange of ideas among mathematicians, engineers, and computer scientists. One of the goals in this area is integration within the field to overcome the internal communication barriers; another is facilitation of the flow of ideas from other rapidly progressing fields of mathematics into control theory.

The panel members were the following: H.T. Banks, G. Blankenship, R. Brockett, J.A. Burns, W.H. Fleming, R.V. Kohn, A. Krener, A.J. Laub, J.L. Lions, S. Markus, J.E. Marsden, S. Mitter, E. Polak, R.T. Rockefeller, D. Russell, E.D. Sontag, and G. Stein.

In the process of assembling the report, it has become clear that perceptions of the future directions vary widely among the members of the control science community. For instance, some believe that the field will evolve more in the direction of software engineering, artificial intelligence, and intelligent control, with mathematical research taking second place to computer science. These views, however, were not strongly emphasized in the report. In fact, the report definitely takes a mathematical perspective and emphasizes the need to continue the creative interaction of mathematics, computation, and engineering.

Copies of the report can be obtained by writing to: Customer Service, SIAM, 117 South 17th St., 14th Fl., Philadelphia, PA 19103-5052.