Machine intelligence Fuzzy modelling and Control

DR. AHMAD AL-MAHASNEH

Fuzzy modelling

- The general purpose of a model is to describe the functioning of a system in terms of input/output behaviour.
- Traditional techniques of system modelling have significant limitations.
- In many cases it is difficult to describe the system behaviour by a set of mathematical equations when the system is nonlinear and partially known or unknown.
- Moreover, there are many uncertainties and unpredictable dynamics that do not allow the system model to be described mathematically. Such uncertainties and unpredictable behaviour in complicated and ill-defined systems can be modelled using the linguistic approach as a model of human thinking, which introduced fuzziness into systems theory (Zadeh, 1965, 1973).
- Therefore, fuzzy system modelling is an important issue while control of such systems is of concern.

introduction

- Most of the classical design methodologies such as Nyquist, Bode, state-space and optimal control are based on assumptions that the process is linear and stationary and hence is represented by a finite-dimensional constant-coefficient linear model.
- These methods do not suit complex systems well because few of these represent uncertainty and incompleteness in process knowledge or complexity in design. But the fact is that the real world is too complex.
- In particular, many industrial processes are highly nonlinear and complex. As the complexity of a system increases, quantitative analysis and precision become difficult.
- However, many processes that are nonlinear, uncertain, incomplete or non-stationary have subtle and interactive exchanges with the operating environment and are controlled successfully by skilled human operators.
- Rather than mathematically modelling the process, the human operator models the process in a heuristic or experiential manner. It is evident that human knowledge is becoming more and more important in control system design.
- This experiential perspective in controller design requires the acquisition of heuristic and qualitative, rather than quantitative, knowledge or expertise from the human operator.
- During the past several years, fuzzy control has emerged as one of the most active and powerful areas for research in the application of such complex and real-world systems using fuzzy set theory (Zadeh, 1965, 1994).
- The control of complex nonlinear systems has been approached in recent years using fuzzy logic techniques.

introduction

- A fuzzy logic controller (FLC) has the basic configuration

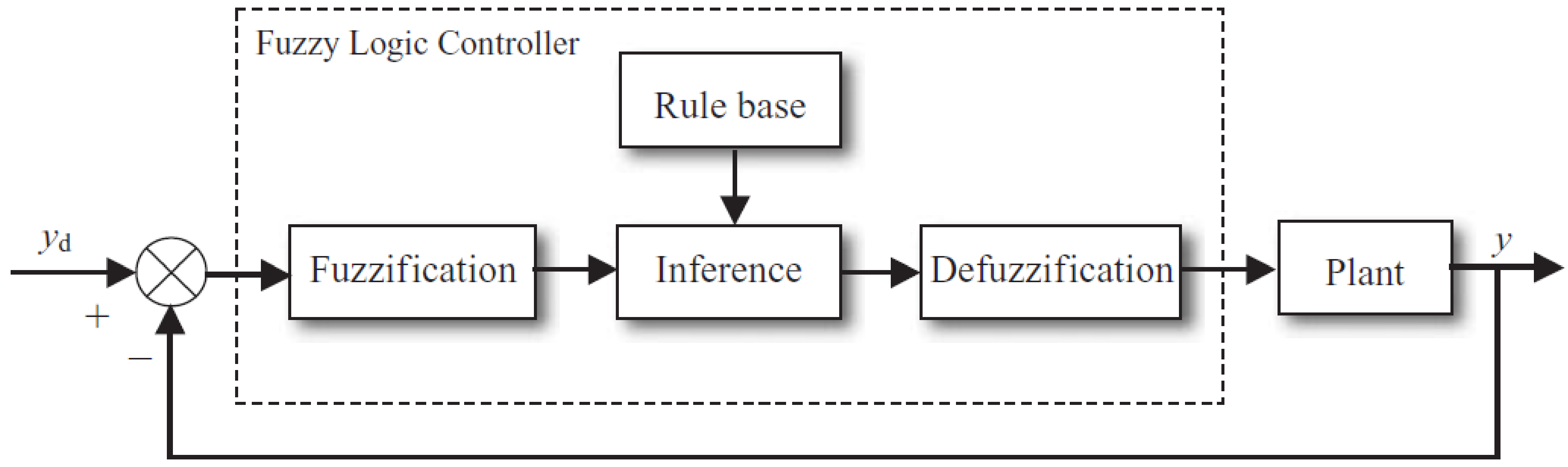


Figure 3.10 Configuration of a fuzzy logic controller

Fuzzification

- Fuzzification is defined as a mapping from an observed input space to fuzzy sets in a certain input universe of discourse.

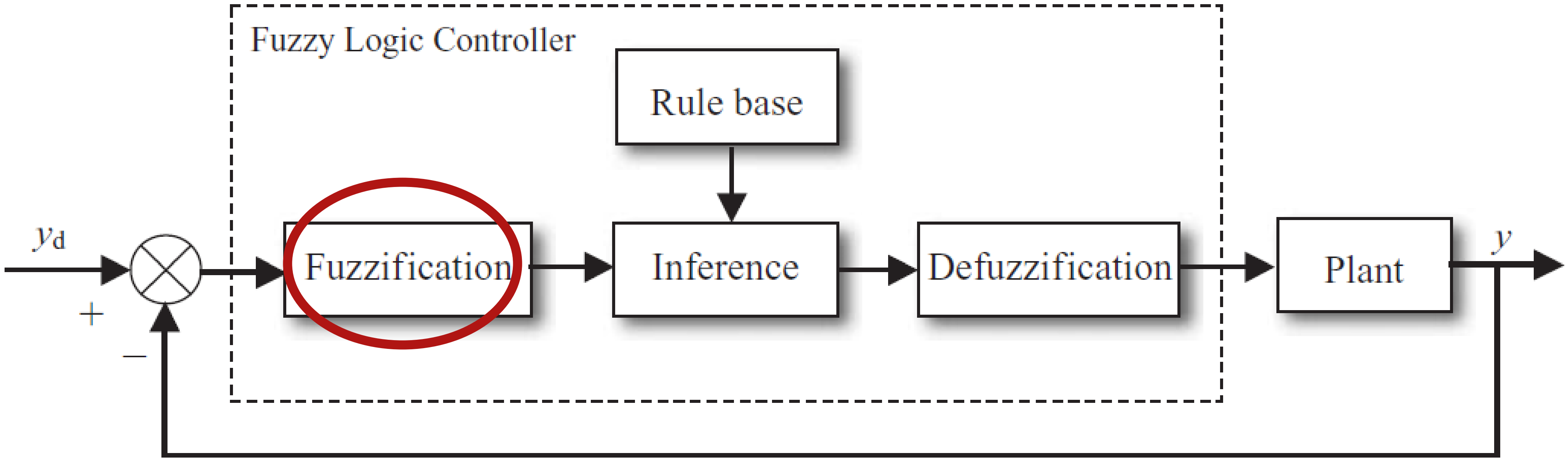


Figure 3.10 Configuration of a fuzzy logic controller

Inference

- Inference is the process of formulating a nonlinear mapping from a given input space to an output space. The mapping then provides a basis from which decisions can be taken. The process of fuzzy inference involves all the membership functions, fuzzy logic operators and if–then rules.

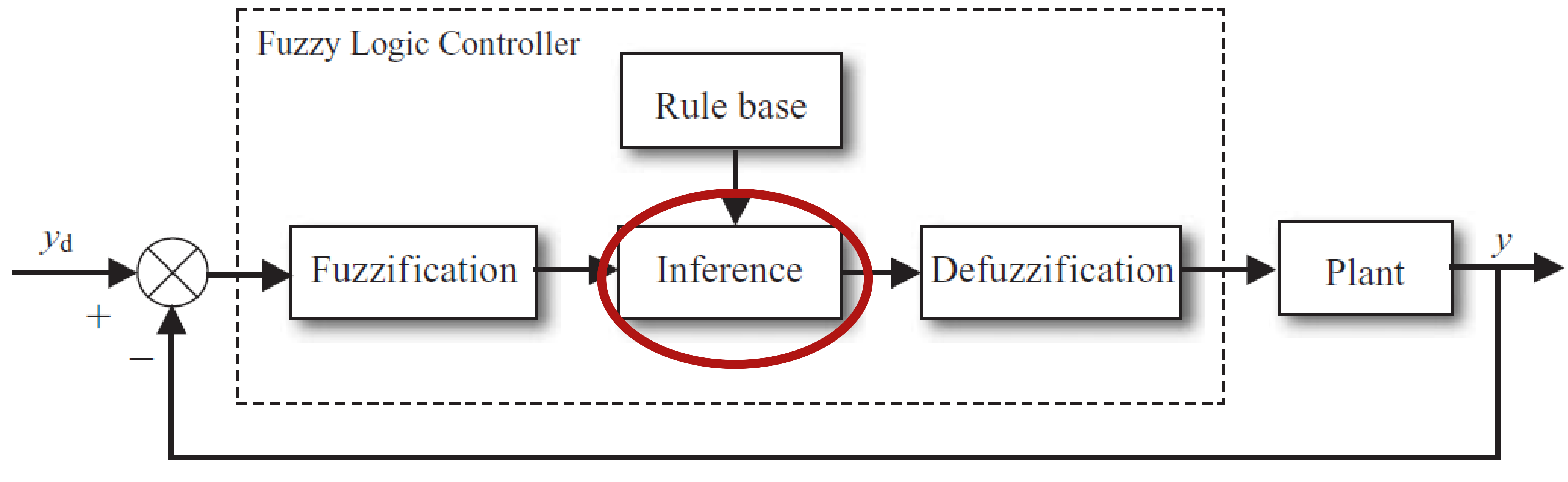


Figure 3.10 Configuration of a fuzzy logic controller

Rule base

- A fuzzy system is characterized by a set of linguistic statements based on expert knowledge.
- The expert knowledge is usually in the form of if–then rules, which are easily implemented
- by fuzzy conditional statements in fuzzy logic (Wong and Lin, 1997).
- The collection of fuzzy rules that are expressed as fuzzy conditional statements forms the rule base or the rule set of an FLC.

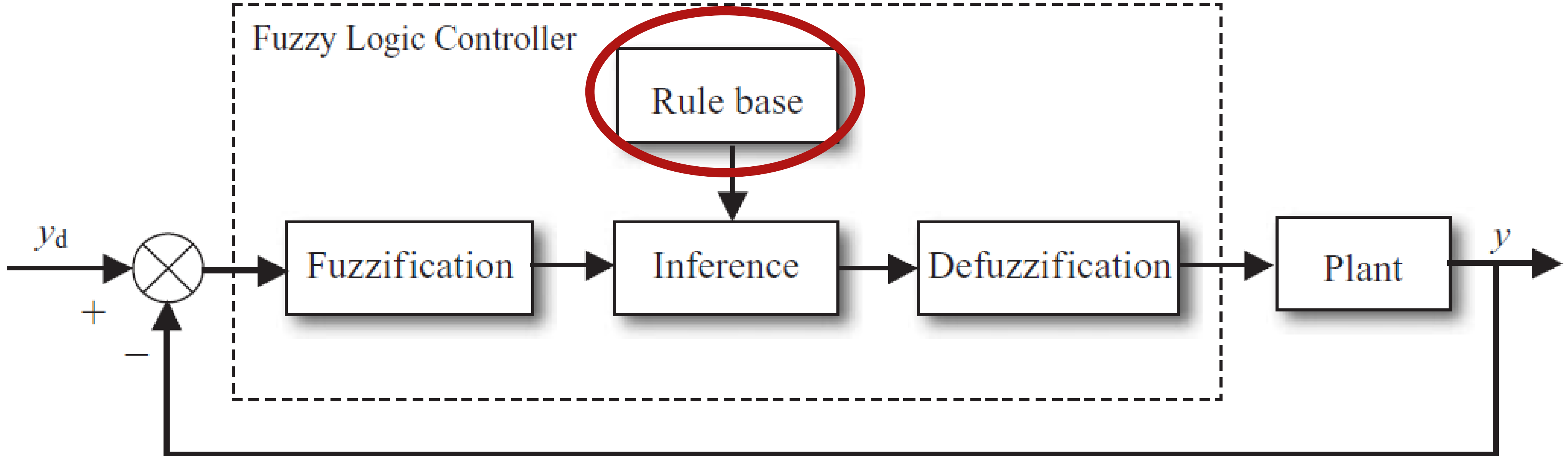


Figure 3.10 Configuration of a fuzzy logic controller

Rule base

- For example, a rule base with two inputs, error and change of error, is shown in Table 3.1.
- Each input/output has five membership functions NB, NS, ZO, PS and PB, where PB=positive big, PS=positive small, ZO=zero, NS=negative small and NB= big.

Table 3.1 FLC rule base with error and change of error

	Change of error				
Error	NB	NS	ZO	PS	PB
NB	PB	PB	PB	PS	ZO
NS	PB	PS	PS	ZO	NS
ZO	PS	ZO	ZO	ZO	NS
PS	PS	ZO	NS	NS	NB
PB	ZO	NS	NB	NB	NB

Rule base

The design parameters of the rule base include:

- Choice of process state and control output variables;
- Choice of the content of the rule antecedent and the rule consequent;
- Choice of term sets for the process state and control output variables;
- Derivation of the set of rules.
- If one has made the choice to design a P-, PD-, PI-, or PID-like fuzzy logic controller this already implies the choice of process state and control output variables, as well as the content of the rule antecedent and rule consequent for each of the rules.

Rule base

- The process state variables are selected as follows:

- Error, denoted by e;
- Change of error, denoted by Δe;
- Sum of error, denoted by Σe.

The control output (process input) variables representing the content of the rule consequent ('then' part of the rule) are selected as follows:

- Control output, denoted by u;
- Change of control output, denoted by Δu.

By analogy with the conventional controller, these are defined as

- $e(k) = y_d - y(k)$
- $\Delta e(k) = e(k) - e(k-1)$
- $\sum_{k=1}^{n} e(k) = \sum_{k=1}^{n-1} e(k) + e(k)$
- $\Delta u(k) = u(k) - u(k-1)$ or $u(k) = u(k-1) + \Delta u(k)$

where y_d is the desired output or set point, y is the process output, k is the sampling time and n is the maximum sample number.

Defuzzification

- Defuzzification is a mapping from a space of fuzzy control actions defined over an output universe of discourse into a space of non-fuzzy (crisp) control actions. In a sense this is the inverse of fuzzification, even though mathematically the maps need not be inverses of one another.

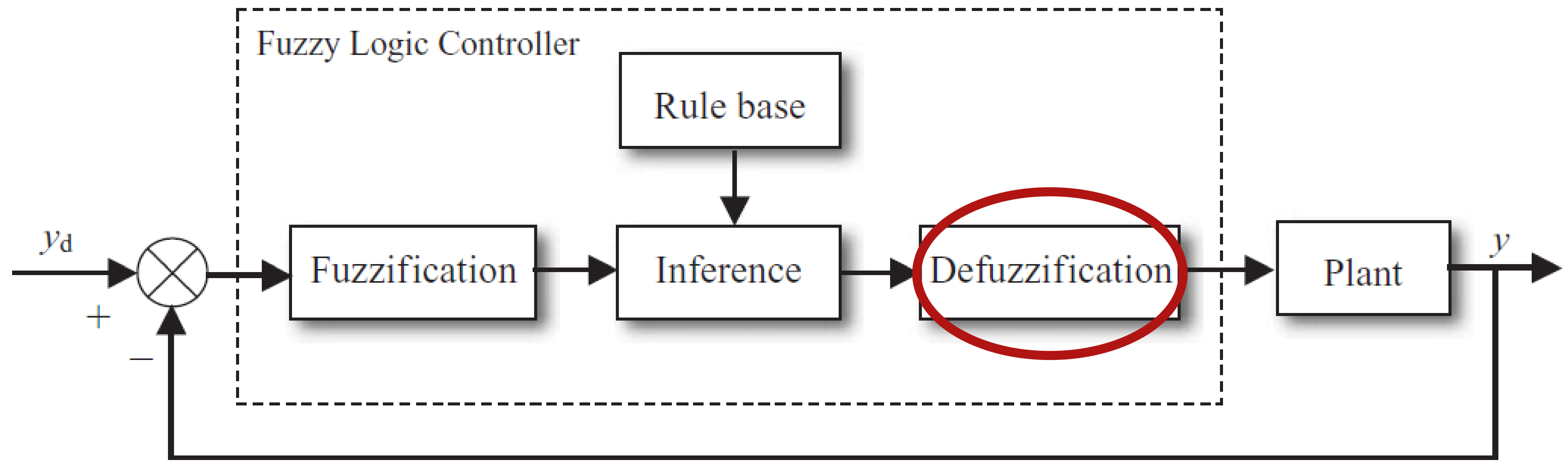


Figure 3.10 Configuration of a fuzzy logic controller

Design of Fuzzy Controller

Let $x = (x_1, \ldots, x_n)$ be a vector of process state variables, y the process output variable and u the process input variable or control variable. The conventional closed-loop model, when linearized around the set point, is given by

$$x(k+1) = A \cdot x(k) + b^T \cdot u(k) \tag{3.14}$$

$$y(k) = c^T \cdot x(k) \tag{3.15}$$

$$u(k) = k \cdot y(k) \tag{3.16}$$

where A is the process matrix, b and c are vectors and k is a scalar. The state equations can be written as

$$x(k+1) = A \cdot x(k) + b^T \cdot u(k) \tag{3.17}$$

$$u(k) = k \cdot c^T \cdot x(k) \tag{3.18}$$

Design of Fuzzy Controller

The fuzzy counterpart of the above model can be described as follows. Let the linguistic variable x_i (e.g., error, change of error, etc.) have the term set X_i (e.g., NB, NS, etc.) and the membership function for X_i be denoted by $\widetilde{X}_i$. Thus, the linguistically defined process state vector is denoted by $\widetilde{X} = (\widetilde{X}_1, \ldots, \widetilde{X}_n)$. Similarly, u takes linguistically defined values U with membership functions $\widetilde{U}$. Thus, the fuzzy model of the closed-loop system can be described as

$$\widetilde{X}(k+1) = \left[\widetilde{X}(k) \times \widetilde{U}(k)\right] \circ \widetilde{A} \tag{3.19}$$

$$\widetilde{U}(k) = \widetilde{X}(k) \circ \widetilde{K} \tag{3.20}$$

where $\widetilde{A}$ is a fuzzy relation on $X \times U \times X$, $\circ$ is the composition operation and $\widetilde{K}$ is the controller which is a fuzzy relation on $X \times U$ representing the meaning of a set of if–then rules of the form

$$\text{If } x_1 \text{ is } X_i \text{ and } \ldots x_n \text{ is } X_j \text{ then } u \text{ is } U_k \tag{3.21}$$

$\widetilde{A}$ can be obtained in explicit form by on/off-line identification or $\widetilde{A}$ is the fuzzy relation giving the overall meaning of a set of production rules.

Fuzzy controller example

Suppose the fuzzy controller has to control the water level of a tank

Figure 3.15 Water level in a tank

Fuzzy controller example *Input/Output Selection*

Rather than going for development of a mathematical model of the system with available states, a fuzzy model using the available states, namely the error e, change of error Δe, sum of error Σe and valve position u at each discrete time step during the control process, can be developed.

The states of water level and state of valve can be measured directly from the system, whereas the error e, change of error Δe and sum of error Σe can be derived from these states as follows:

$$e = y_d - y \tag{3.22}$$

$$\Delta e = e(k) - e(k-1) \tag{3.23}$$

$$\sum e(k) = \sum e(k-1) + e(k) \tag{3.24}$$

where y is the measured water level and y_d is the desired water level.

Fuzzy controller example *Choice of Membership Functions*

Since Lotfi Zadeh introduced fuzzy sets, the main difficulties have been with the meaning and measurement of membership functions as well as their extraction, modification and adaptation to dynamically changing conditions. There is no general rule for choice of membership functions, and this mainly depends on the problem domain. In general, use of narrower membership functions results in a faster response but causes larger oscillations, overshoot and settling time.

Gaussian and bell-shaped membership functions involve calculation of exponential terms and use substantial processing time. Trapezoidal membership functions have four parameters and can burden the optimization procedure. Triangular membership functions are the best choice and used for simplicity.

Fuzzy controller example *Creation of Rule Base*

The fuzzy rules R must be completed and covered by fuzzy partitioning the input space. Figure 3.16 shows an input space partitioning for two-input single-output systems.

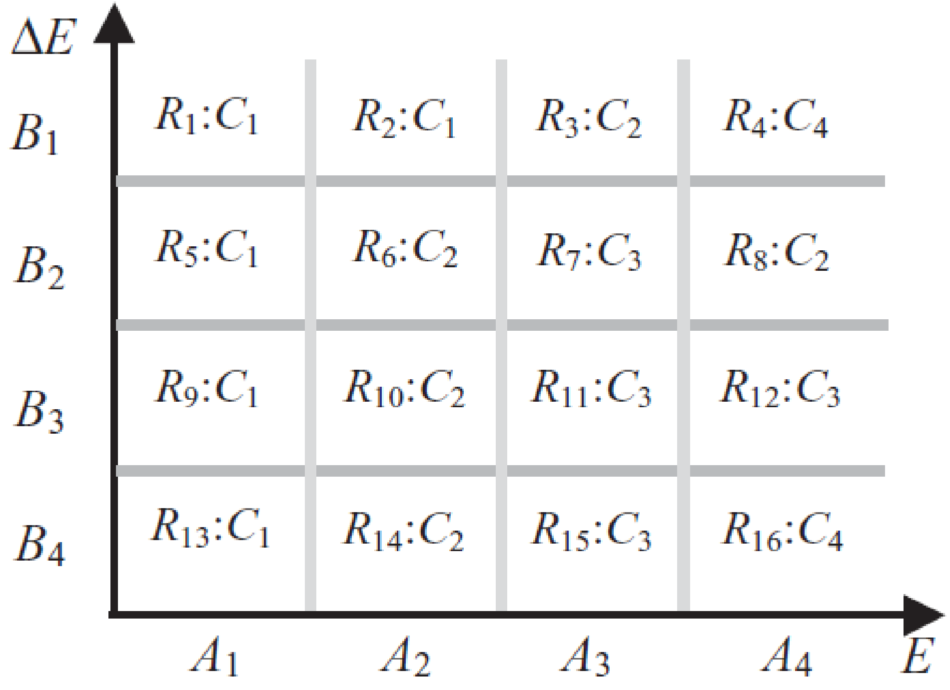


Figure 3.16 Fuzzy input space partitioning

Fuzzy controller example *Creation of Rule Base*

For example, the error and change of error and valve position of a PD-like FLC can be divided into four partitions (i.e., partitioned into four fuzzy sets) as:

$$\text{error } E = \{A1,\ A2,\ A3,\ A4\}$$
$$\text{change of error } \Delta E = \{B_1, B_2, B_3, B_4\}$$
$$\text{valve position } U = \{C_1, C_2, C_3, C_4\}$$

where E, ΔE and U are the universe of discourse for error, change of error and valve position, respectively. The nth rule for the two-input single-output system is

$$R_n : \text{IF } (e \text{ is } A_i) \text{ and } (\Delta e \text{ is } B_j) \text{ THEN } (u \text{ is } C_k)$$

where R_n, $n = 1, 2, \ldots, 16$, is the nth fuzzy rule. A_i, B_j and C_k, $i = 1, 2, \ldots, 4$, $j = 1, 2, \ldots, 4$ and $k = 1, 2, \ldots, 4$, are primary fuzzy sets. There are 16 rules obtained from this uniform partitioning. Initially, fuzzy rules are based on input/output data and these rules are refined through trial and error.

Types of fuzzy controllers

A fuzzy controller can be constructed using e, Δe and Σe as inputs and control input u as output depending on the type of controller, e.g. PD, PI or PID type.

P-like FLC: The equation for a conventional proportional (P)-like controller is given as

$$u = k_p \cdot e(k) \tag{3.25}$$

where k_p is the proportional gain coefficient. The rule for a P-like controller is given in symbolic form as

$$\text{If } e \text{ is } A_i \text{ then } u \text{ is } B_j \tag{3.26}$$

Types of fuzzy controllers

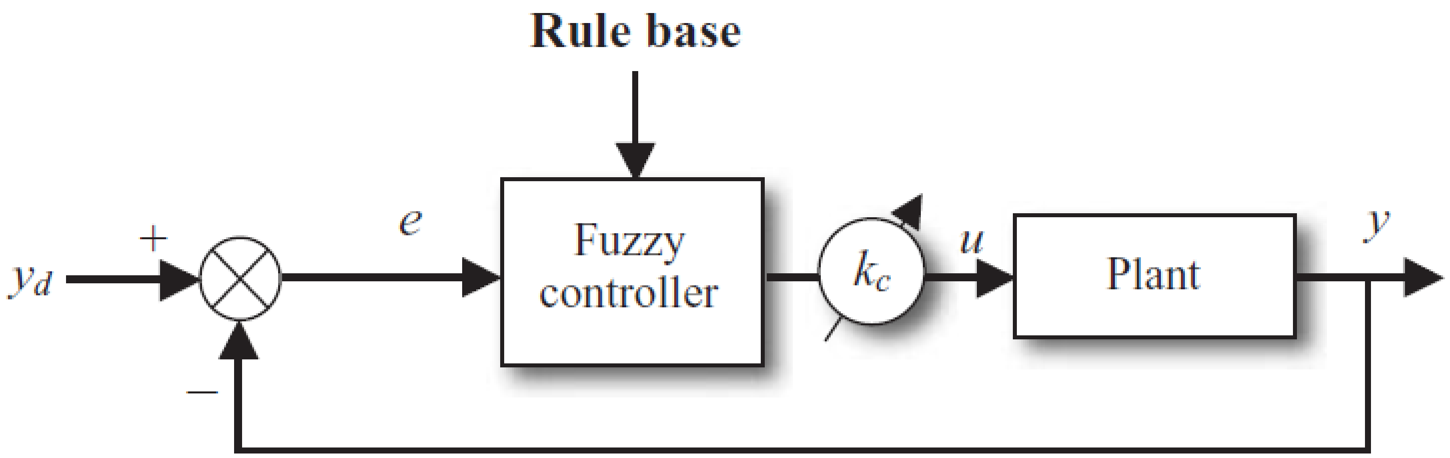


Figure 3.17 Block diagram of a P-type FLC with error

where A_i and B_j, i, $j = 1, 2, \ldots, n$, are the linguistic variables. Figure 3.17 shows the block diagram of a P-type single-input single-output fuzzy controller for a plant. The function of the control output for such a single-input single-output (SISO) system is then a curve, as shown in Figure 3.18 for $n = 4$.

Types of fuzzy controllers

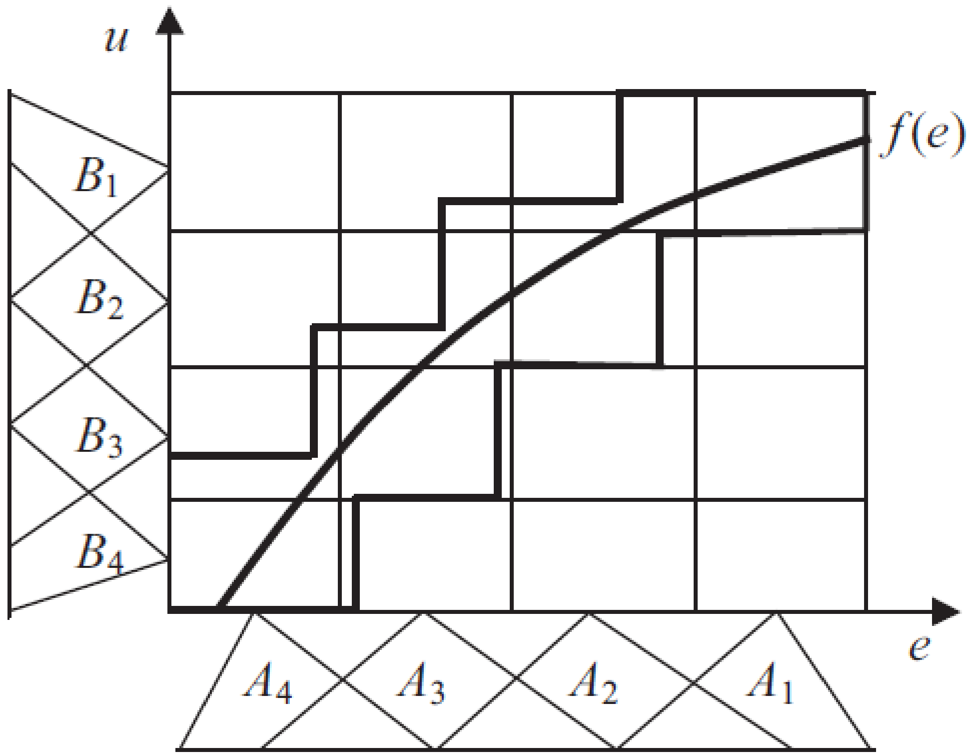


Figure 3.18 Function of control output for SISO systems

Types of fuzzy controllers

PD-like FLC: A conventional proportional differential (PD)-like FLC can be developed by using an error and change of error model as

$$u = k_p.e + k_d.\Delta e \tag{3.27}$$

where k_p and k_d are the proportional and differential gain coefficients and e is the error, Δe is the change of error. In this type of FLC, it is assumed that no mathematical model for the system is available except two states, namely, the error and change of error. Only output y is measured from the system and the error and change of error are derived. The error and change of error are defined as

$$e(k) = y_d - y(k) \tag{3.28}$$

$$\Delta e(k) = e(k) - e(k-1) \tag{3.29}$$

where y_d is the desired output and $y(k)$ is the actual output. Figure 3.19 shows the block diagram of a PD-like FLC with error and change of error as inputs. The PD-like

Types of fuzzy controllers

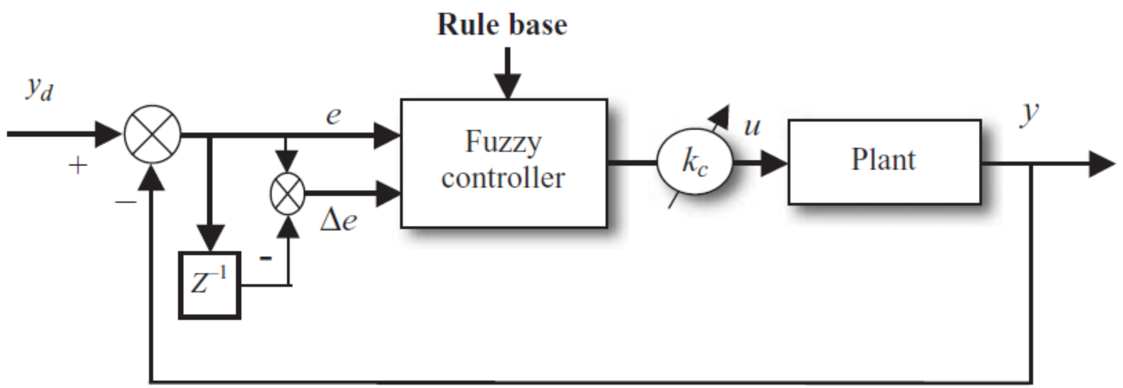


Figure 3.19 Block diagram of PD-like FLC with error and change of error

FLC consists of rules of the form

$$\text{If } e \text{ is } A_i \text{ and } \Delta e \text{ is } B_j \text{ then } u \text{ is } C_k \tag{3.30}$$

where A_i, B_j and C_k are the linguistic variables and $i = 1, \ldots, n_1$, $j = 1, \ldots, n_2$ and $k = 1, \ldots, m$.

The control surface of a two-input single-output (MISO) system is shown in Figure 3.20, where X and Y represent inputs and Z represents the controller output. For a PD-type controller, X represents error and Y represents change of error. For a PI-type controller, X represents error and Y represents sum of error.

Types of fuzzy controllers

Example 3.1: PD-like FLC with error and change of error A simple PD-like FLC is developed for a manipulator. A schematic representation of the flexible-link manipulator system considered here is shown in Figure 3.21, where X_oOY_o and XOY represent the stationary and moving coordinates, respectively and τ represents the applied torque at the hub. E, I, ρ, V, I_h and M_P represent the Young's modulus, area moment of inertia, mass density per unit volume, cross-sectional area, hub inertia and payload of the manipulator, respectively. In this example, the motion of the manipulator is confined to the X_oOY_o plane.

In a PD-type FLC, it is assumed that no mathematical model for the flexible link is available except two states, namely, the hub angle error and change of error. Only the hub angle θ is measured from the system and the error and change of error are derived from θ. The hub angle error and change of error are defined as

$$e(k) = \theta_d - \theta(k) \tag{3.31}$$

$$\Delta e(k) = e(k) - e(k-1) \tag{3.32}$$

where θ_d is the desired hub angle, e is the error and Δe is the change in angle error. Figure 3.22 shows a block diagram of the PD-like FLC with error and change of error as inputs.

Types of fuzzy controllers

Triangular membership functions are chosen for inputs and output. The membership functions for angle error, change of angle error and torque input are shown in Figure 3.23. The universe of discourse for the angle error and change in angle error are chosen as [–36, +36] degrees and [–25, +25] respectively. The universe of discourse of the output (i.e., input torque) is chosen as [–3, +3] volts. To construct a rule base, the angle error, change of angle error and torque input are partitioned into five primary fuzzy sets as

$$\text{hub angle error } E = \{\text{NB, NS, ZO, PS, PB}\}$$
$$\text{change of angle error } C = \{\text{NB, NS, ZO, PS, PB}\}$$
$$\text{torque } U = \{\text{NB, NS, ZO, PS, PB}\}$$

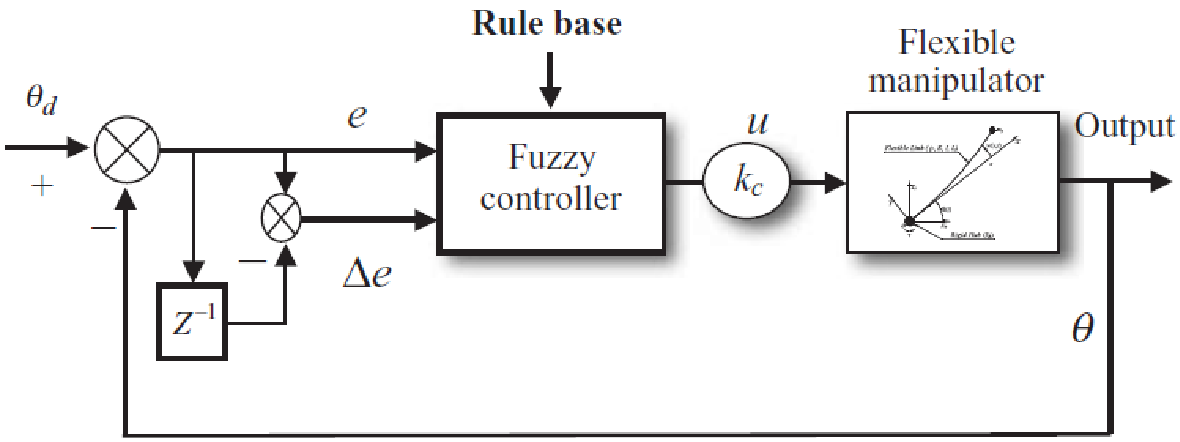
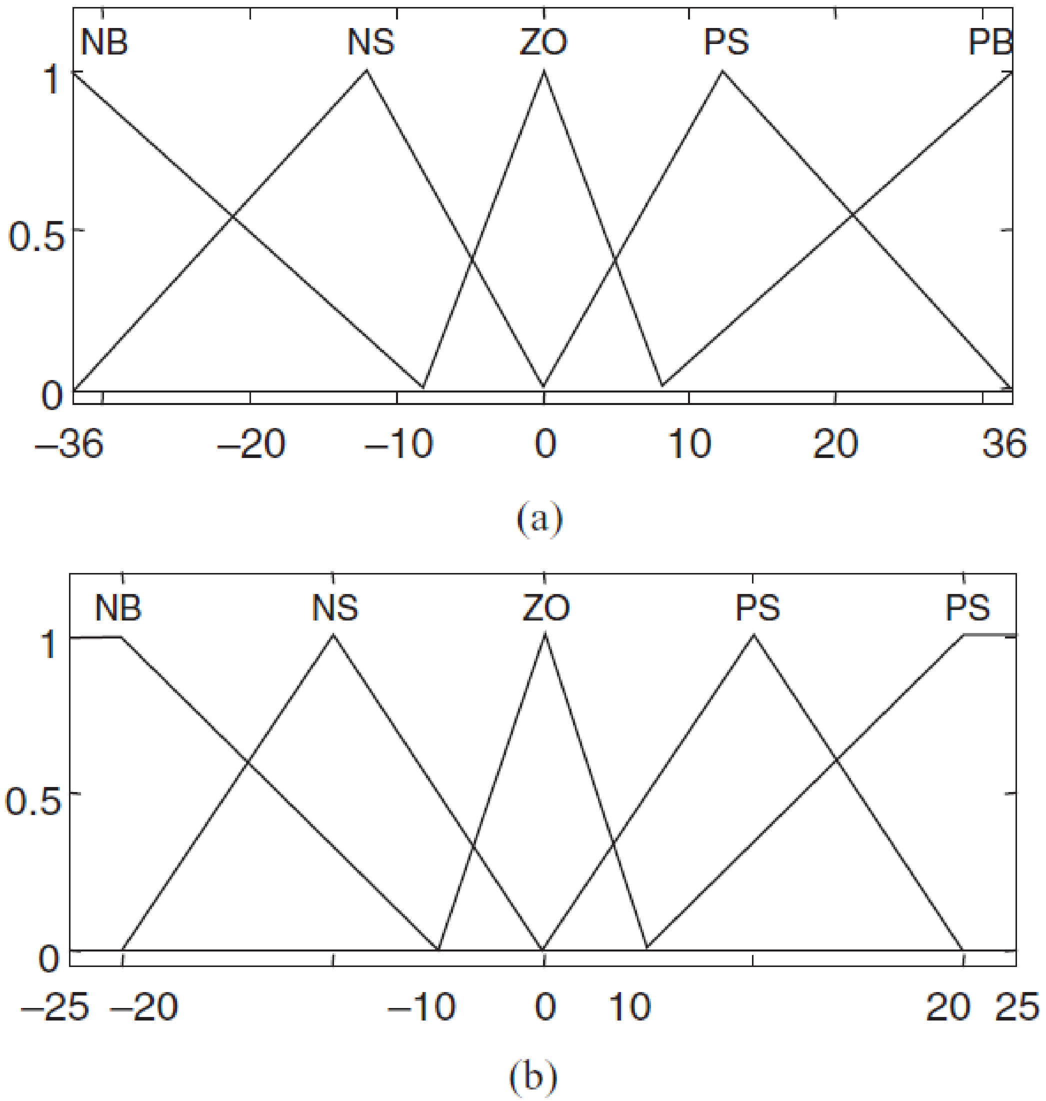
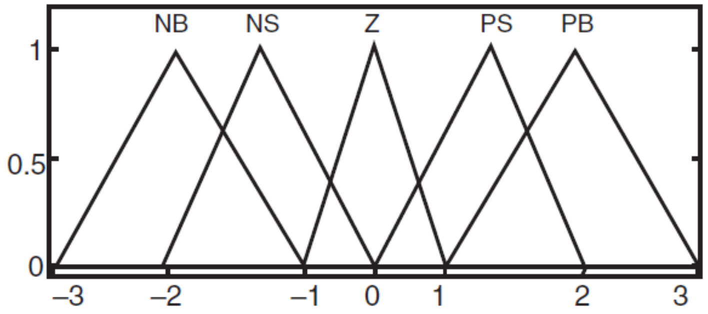


Figure 3.22 PD-like FLC with angle error and change of angle error

Types of fuzzy controllers





(c)

Figure 3.23 Membership functions for inputs and output. (a) Angle error; (b) Change of angle error; (c) Torque input

Types of fuzzy controllers

Table 3.2 FLC rule base with angle error and change of angle error

	Change of error				
Error	NB	NS	ZO	PS	PB
NB	PB	PB	PB	PS	ZO
NS	PB	PS	PS	ZO	NS
ZO	PS	ZO	ZO	ZO	NS
PS	PS	ZO	NS	NS	NB
PB	ZO	NS	NB	NB	NB

Types of fuzzy controllers

PI-like FLC: A conventional proportional-integral (PI)-like controller is described as

$$u = k_P e + k_I \int e dt \tag{3.33}$$

where k_P and k_I are the proportional and integral gain coefficients. Taking the derivative with respect to time of Equation (3.30) yields

$$\dot{u} = k_P \cdot \dot{e} + k_I \cdot e \tag{3.34}$$

which can be rewritten as

$$\Delta u = k_P \cdot \Delta e + k_I \cdot e \tag{3.35}$$

This yields an incremental PI-like controller equation. The PI-like FLC rule base accordingly consists of rules of the form

If e is A_i and Δe is B_j then Δu is C_k

In this case, to obtain the value of the control output $u(k)$, the change of control output $\Delta u(k)$ is added to $u(k-1)$ such that

$$u(k) = \Delta u(k) + u(k-1) \tag{3.36}$$

Types of fuzzy controllers

If e is A_i and Σe is B_j then u is C_k (3.38)

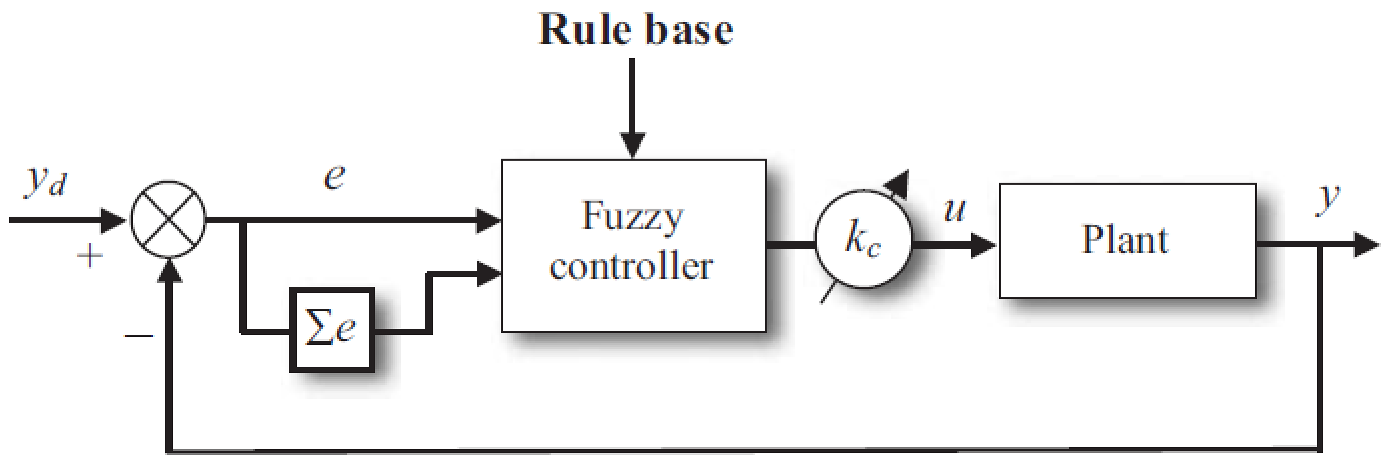


Figure 3.26 Block diagram of a PI-type FLC with error and sum of error