general, caution should be exercised if one wants to call a fuzzy controller fuzzy PID controller when its analytical structure is unknown.

Like linear PID control, fuzzy PID control also has a position form and an incremental form. Their definitions are the same as the respective forms of the PID control. The relationships between PI control and PD control mentioned above also hold for fuzzy PI and fuzzy PD control.

More generally, a fuzzy controller is defined as a fuzzy controller of the PID type, if it can be expressed as

$$U(n) \text{ (or } \Delta U(n)) = c_0 + c_1 x_1(n) + \cdots + c_M x_M(n), \tag{3.7}$$

where c_i, $0 \leq i \leq M$, can be either constant gain or variable gain changing with time. By definition, fuzzy PID control is a special case of fuzzy control of PID type when $c_0 = 0$ and $M = 3$. In this book, only a fuzzy controller of PID type with at most three input variables is called a fuzzy PID controller. When more than three input variables are involved, the name "fuzzy controller of the PID type" is used.

We make this classification to reflect the special value of fuzzy PID control in fuzzy control, just like the important role that its classical counterpart plays in conventional control.

If a fuzzy controller is not of the PID type, it is simply defined as a fuzzy controller of non-PID type.

3.4. FUZZY PI/PD CONTROLLERS AS LINEAR PI/PD CONTROLLERS

Although linear fuzzy controllers have little practical value, they are simpler than nonlinear ones, and their structures are easier to derive and understand. Hence, they provide an excellent stepping stone towards understanding and analysis of more complicated fuzzy controllers.

3.4.1. Fuzzy PI Controller Configuration

The fuzzy controller uses two identical input fuzzy sets, namely Positive and Negative, for scaled input variables, $E(n)$ and $R(n)$. The fuzzy sets are shown in Fig. 3.1a. Using $\tilde{P}$ and $\tilde{N}$ to represent Positive and Negative, respectively, we find that the membership functions of the fuzzy sets for $E(n)$ are

$$\mu_{\tilde{P}}(e) = \begin{cases} 0, & E(n) < -L \\ \dfrac{K_e e(n) + L}{2L}, & -L \leq E(n) \leq L \\ 1, & E(n) > L \end{cases}$$

and

$$\mu_{\tilde{N}}(e) = \begin{cases} 1, & E(n) < -L \\ \dfrac{-K_e e(n) + L}{2L}, & -L \leq E(n) \leq L, \\ 0, & E(n) > L \end{cases}$$

and the membership functions for $R(n)$ are

$$\mu_{\tilde{P}}(r) = \begin{cases} 0, & R(n) < -L \\ \dfrac{K_r r(n) + L}{2L}, & -L \le R(n) \le L \\ 1, & R(n) > L \end{cases}$$

and

$$\mu_{\tilde{N}}(r) = \begin{cases} 1, & R(n) < -L \\ \dfrac{-K_r r(n) + L}{2L}, & -L \le R(n) \le L \\ 0, & R(n) > L. \end{cases}$$

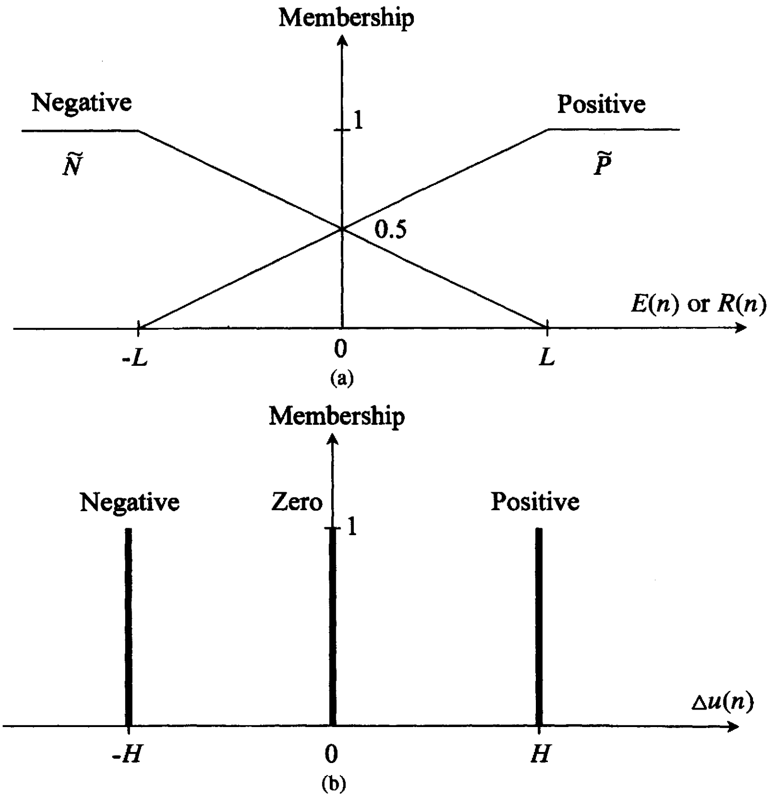

Figure 3.1 Graphical definitions of input and output fuzzy sets used by the linear fuzzy PI controller: (a) two input fuzzy sets Positive and Negative for $E(n)$ and $R(n)$, and (b) three singleton output fuzzy sets, Positive, Zero, and Negative.

In the definitions, L is a constant design parameter. Note that

$$\mu_{\tilde{P}}(e) + \mu_{\tilde{N}}(e) = 1, \quad \text{for } E(n) \in (-\infty, \infty) \tag{3.8}$$

$$\mu_{\tilde{P}}(r) + \mu_{\tilde{N}}(r) = 1, \quad \text{for } R(n) \in (-\infty, \infty). \tag{3.9}$$

The fuzzy PI controller uses the following four fuzzy rules:

IF $E(n)$ is Positive AND $R(n)$ is Positive THEN $\Delta u(n)$ is Positive (r1)

IF $E(n)$ is Positive AND $R(n)$ is Negative THEN $\Delta u(n)$ is Zero (r2)

IF $E(n)$ is Negative AND $R(n)$ is Positive THEN $\Delta u(n)$ is Zero (r3)

IF $E(n)$ is Negative AND $R(n)$ is Negative THEN $\Delta u(n)$ is Negative (r4)

where the output fuzzy sets are of the singleton type and their nonzero values are at H, 0, and $-H$, respectively for Positive, Zero, and Negative, as shown in Fig. 3.1b.

These four rules are sufficient to cover all possible situations, as illustrated in Fig. 3.2. Rule r1 covers the situation in which system output is below the setpoint and is still decreasing. Obviously, controller output should be increased. Rule r4 deals with the opposite circumstance: system output is larger than the setpoint and still rising. Naturally, controller output should be reduced. There are only two remaining scenarios: (1) system output is below the setpoint but is increasing, and (2) system output is above the setpoint but is decreasing. In either case, it is desirable to let controller output stay at the same level, hoping system output will land on the setpoint smoothly on its own. This is what rules r2 and r3 do.

In evaluating the ANDs in the fuzzy rules, the product fuzzy logic AND operator is used. The results of product AND operations in the four fuzzy rules are

$$\mu_{\tilde{P}}(e)\cdot\mu_{\tilde{P}}(r) \quad \text{for } H,$$
$$\mu_{\tilde{P}}(e)\cdot\mu_{\tilde{N}}(r) \quad \text{for } 0,$$
$$\mu_{\tilde{N}}(e)\cdot\mu_{\tilde{P}}(r) \quad \text{for } 0,$$
$$\mu_{\tilde{N}}(e)\cdot\mu_{\tilde{N}}(r) \quad \text{for } -H.$$

The Lukasiewicz fuzzy logic OR operation is applied to combine the membership values from rules r2 and r3, as there exists an implied OR between the two rules for the same output fuzzy set, Zero. Since

$$\mu_{\tilde{P}}(e)\cdot\mu_{\tilde{N}}(r)+\mu_{\tilde{N}}(e)\cdot\mu_{\tilde{P}}(r)=1-\mu_{\tilde{P}}(e)\cdot\mu_{\tilde{P}}(r)-\mu_{\tilde{N}}(e)\cdot\mu_{\tilde{N}}(r)\le 1,$$

the result of the Lukasiewicz fuzzy OR operation is $\mu_{\tilde{P}}(e)\cdot\mu_{\tilde{N}}(r)+\mu_{\tilde{N}}(e)\cdot\mu_{\tilde{P}}(r)$.

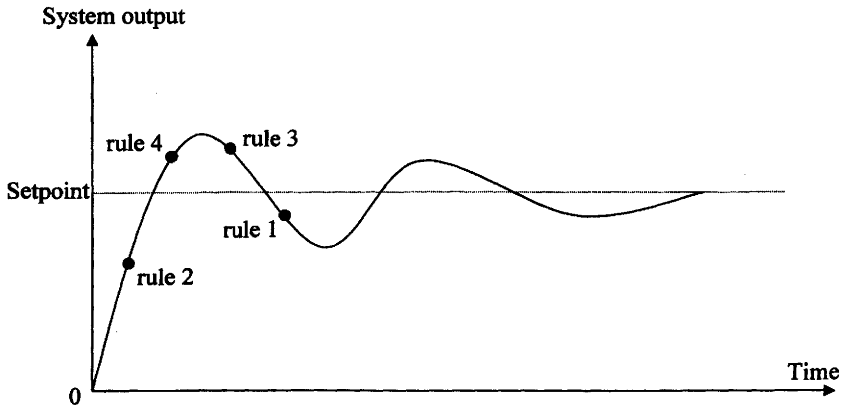

Figure 3.2 Illustration of how merely four fuzzy rules can cover all possible situations.

Due to the use of the singleton output fuzzy sets, the fuzzy inference result is the same no matter which one of the four inference methods in Table 1.1 is employed. Defuzzified by the centroid defuzzifier, the fuzzy controller output is

$$\Delta U(n) = K_{\Delta u} \frac{\mu_{\tilde{P}}(e)\mu_{\tilde{P}}(r) \cdot H + \mu_{\tilde{N}}(e)\mu_{\tilde{N}}(r) \cdot (-H)}{\mu_{\tilde{P}}(e)\mu_{\tilde{P}}(r) + \mu_{\tilde{P}}(e)\mu_{\tilde{N}}(r) + \mu_{\tilde{N}}(e)\mu_{\tilde{P}}(r) + \mu_{\tilde{N}}(e)\mu_{\tilde{N}}(r)}.$$

3.4.2. Derivation and Resulting Structures

Utilizing (3.8) and (3.9), we find that the denominator of the above expression becomes 1. This is because

$$\mu_{\tilde{P}}(e)\mu_{\tilde{P}}(r) + \mu_{\tilde{P}}(e)\mu_{\tilde{N}}(r) + \mu_{\tilde{N}}(e)\mu_{\tilde{P}}(r) + \mu_{\tilde{N}}(e)\mu_{\tilde{N}}(r) = \mu_{\tilde{P}}(e) + \mu_{\tilde{N}}(e) = 1.$$

Replacing the membership notations in the numerator by their mathematical definitions, we obtain

$$\Delta U(n) = \frac{K_{\Delta u} K_e H}{2L} e(n) + \frac{K_{\Delta u} K_r H}{2L} r(n).$$

This fuzzy PI controller is a linear PI controller in incremental form for the entire input space.

For the fuzzy controller, if we replace $\Delta u(n)$ by $u(n)$ in the fuzzy rules r1 to r4, then, based on the relationship between the PI controller in incremental form and the PD controller in position form (see Section 3.2.2), the modified fuzzy controller will be a fuzzy PD controller, which is a linear PD controller.

This study confirms that whether or not a fuzzy controller is linear depends on its configuration (i.e., input fuzzy sets, fuzzy rules, fuzzy logic AND/OR operators, defuzzifier, etc.). No method is available that can directly judge, without explicit knowledge of the controller's input-output relationship, whether a fuzzy controller is a linear controller. The only way is to derive its structure. There are other more complicated fuzzy PID controllers that actually are linear PID controllers (e.g., [20][186]).

Fuzzy control should always be used as nonlinear control, as it does not make any sense to implement fuzzy control as linear control. The linear fuzzy controller shown here serves as a reminder: There are fuzzy controllers that are actually just linear controllers. Thus, certain configurations of fuzzy controllers should not be used to avoid linear fuzzy controllers. Moreover, to be sure that a specific configuration does not lead to a linear controller, one must derive its analytical structure.

3.5. FUZZY PI/PD CONTROLLERS AS PIECEWISE LINEAR PI/PD CONTROLLERS

We now investigate a fuzzy controller that differs from the linear fuzzy PI controller above in the following aspects: (1) the Zadeh fuzzy logic AND operator is used, (2) either the Zadeh or the Lukasiewicz fuzzy logic OR operator is used, and (3) the linear defuzzifier (1.8) is utilized. As will be seen, the new configuration results in a piecewise linear fuzzy controller in that the controller output is a piecewise linear function of its inputs.

Due to the use of the Zadeh fuzzy AND operator, in order to obtain analytical expressions of the AND evaluation results, it is necessary to divide the $E(n) - R(n)$ plane into 12 regions, each of which is called an Input Combination (IC, for short). They are labeled from IC1 to IC12, as shown in Fig. 3.3. The purpose of dividing the input space into these 12