



# Online tuning of fuzzy PID controllers via rule weighing based on normalized acceleration

Onur Karasakal\*, Mujde Guzelkaya, Ibrahim Eksin, Engin Yesil, Tufan Kumbasar

Istanbul Technical University, Faculty of Electrical and Electronics Engineering, Control Engineering Department, Maslak, TR-34469 Istanbul, Turkey

## ARTICLE INFO

### Article history:

Received 2 December 2011

Received in revised form

11 April 2012

Accepted 8 June 2012

Available online 11 July 2012

### Keywords:

Fuzzy PID controller

Self-tuning control

Fuzzy rule weighing

Normalized acceleration

pH neutralization process

## ABSTRACT

In this study, an on-line tuning method is proposed for fuzzy PID controllers via rule weighing. The rule weighing mechanism is a fuzzy rule base with two inputs namely; “error” and “normalized acceleration”. Here, the normalized acceleration provides relative information on the fastness or slowness of the system response. In deriving the fuzzy rules of the weighing mechanism, the transient phase of the unit step response of the closed loop system is to be analyzed. For this purpose, this response is assumed to be divided into certain regions, depending on the number of membership functions defined for the error input of the fuzzy logic controller. Then, the relative importance or influence of the fired fuzzy rules is determined for each region of the transient phase of the unit step response of the closed loop system. The output of the fuzzy rule weighing mechanism is charged as the tuning variable of the rule weights; and, in this manner, an on-line self-tuning rule weight assignment is accomplished. The effectiveness of the proposed on-line weight adjustment method is demonstrated on linear and non-linear systems by simulations. Moreover, a real time application of this new method is accomplished on a pH neutralization process.

© 2012 Elsevier Ltd. All rights reserved.

## 1. Introduction

Fuzzy logic is extensively used in processes where system dynamics is either very complex or exhibit a highly non-linear character. Among various types of fuzzy logic controllers (FLCs), PID type FLCs are most common and practical (Karasakal et al., 2005; Fuente et al., 2006; Chen et al., 2009; Saad and Arrofiq, 2011). From the input–output relationship point of view, the structures of fuzzy PID type controllers are analogous to that of the conventional PID controllers (Galichet and Foulloy, 1995; Moon, 1995; Huang et al., 1999). In literature, there are several researches on the structure of PID type FLCs (Qiao and Mizumoto, 1996; Li and Gatland, 1996; Guzelkaya et al., 2001; Duan et al., 2008).

The design parameters of the fuzzy PID controllers can be categorized as structural and tuning parameters (Hu et al., 1999). Structural parameters include input/output (I/O) variables to fuzzy inference, fuzzy linguistic sets, membership functions, fuzzy rules, inference mechanism and defuzzification mechanism. Tuning parameters include input/output scaling factors and parameters of membership functions.

Various heuristic and non-heuristic scaling factor tuning algorithms have been presented in the case of the systems that own nonlinearities, parameter changes, modeling errors, disturbances (Qiao and Mizumoto, 1996; Chung et al., 1998; Mudi and Pal, 1999; Woo et al., 2000; Guzelkaya et al., 2003; Bhattacharya et al., 2004; Oh et al., 2004, 2009; Bouallegue et al., 2011). Moreover, the literature includes some methods for tuning the membership functions and fuzzy rule weights of the fuzzy controllers. Juang et al. (2008) and Fang et al. (2008) tuned the membership functions by means of genetic algorithm and particle swarm optimization method to improve the system performance. Ahn and Truong (2009) used a robust extended Kalman filter to tune the membership functions of the fuzzy controller during the system operation process to improve the control performance. Teng et al. (2004) proposed a genetic weighted fuzzy rule based system in which the parameters of membership functions including position and shape of the fuzzy rule set and weights of the rules are evolved using a genetic algorithm. In Sharkawy (2010), each classical PID parameter is tuned via a fuzzy logic system whose membership functions are optimally determined in an off-line manner using a modified genetic algorithm. Moreover, Mona et al. (2011) tuned the rules, rule weights and membership functions of fuzzy controller via genetic algorithm for trajectory control of PUMA 560 industrial robot. Alcala et al. (2005) optimized the rule weights of fuzzy controller by genetic algorithm in order to simplify the fuzzy rule base. Karasakal et al. (2011)

\* Corresponding author. Tel.: +90 212 285350; fax: +90 212 2852920.  
E-mail address: [onur\\_karasakal@yahoo.com](mailto:onur_karasakal@yahoo.com) (O. Karasakal).

presented an on-line rule weighting method for fuzzy PID controllers using simple functions of the system error in an adequate arrangement.

In this study, an on-line tuning method is proposed for fuzzy PID controllers via rule weighing. The rule weighing mechanism is a fuzzy rule base with two inputs and one output. The inputs are the “error” and the “normalized acceleration”, where the normalized acceleration provides relative information on the fastness or slowness of the system response. In order to tune the rule weights of the fuzzy controller, the transient phase of the closed loop system response is assumed to be divided into certain regions in accordance with the number of membership functions defined for the error input of the fuzzy PID controller. Then, the relative importance or influence of the fired rules of the rule base is determined for each region and the meta-rules are derived for the adjustment of their weight values to obtain an “efficient” and “appropriate” control signal. The output of the fuzzy mechanism is directly used for the assignment of the rule weights by an adequate arrangement in accordance with the meta-rules derived. The effectiveness of the proposed rule weighting method are demonstrated on linear and non-linear systems by simulations and real time applications are done on the G.U.N.T. RT-552 pH value Control Trainer.

The outline of the paper can be summarized as follows: Section 2 includes the fuzzy PID controller structure without a tuning mechanism, the interpretation of rule weights, the concept of normalized acceleration and the proposed on-line rule weight adjustment method. Section 3 presents simulations for linear and non-linear systems, Section 4 includes the experimental studies which are implemented on the pH neutralization process experimentally, Section 5 provides the discussions and the conclusions, respectively.

## 2. The proposed rule weight adjustment method

### 2.1. Standard fuzzy PID controller

In this study, we will deal with standard fuzzy PID controllers, formed using a fuzzy PD controller with an integrator and a summation unit at the output, as it is shown in Fig. 1 (Qiao and Mizumoto, 1996).

The output of the fuzzy PID controller is given by

$$u = \alpha U + \beta \int U dt \quad (1)$$

The error ( $e$ ) and the change of error ( $\Delta e$ ) are used as the inputs and the change of the control signal ( $U$ ) is used as the output of the FLC. The input scaling factors  $K_e$  for error and  $K_d$  for the change of error normalize the inputs to the range in which the membership functions of the inputs are defined. The output scaling factors  $\beta$  and  $\alpha$  normalize the output of FLC to an applicable value. The universes of discourse are chosen to be  $[-1, 1]$  for the inputs  $e$ ,  $\Delta e$  and output  $U$ .

The symmetrical triangular uniformly distributed membership functions are assigned to each fuzzy linguistic value of the error ( $e$ ) and the change of error ( $\Delta e$ ) as shown in Fig. 2. The linguistic values of  $e$  and  $\Delta e$  are denoted as  $N_i$  where  $i = \{-n, -n+1, \dots, -2, -1, 0, 1, 2, \dots, n-1, n\}$  and  $M_j$  where  $j = \{-m, -m+1, \dots, -2, -1, 0, 1, 2, \dots, m-1, m\}$ , respectively. In addition, the cores of fuzzy sets  $N_i$  and  $M_j$  are denoted as  $e_i$  and  $\Delta e_j$ . Singleton fuzzy membership functions are assigned to the output of the FLC.

### 2.2. The fuzzy rule weights

A comprehensive analysis on the fuzzy rule weights can be found in Nauck and Kruse (1998). The rule weight ( $w$ ) is a real

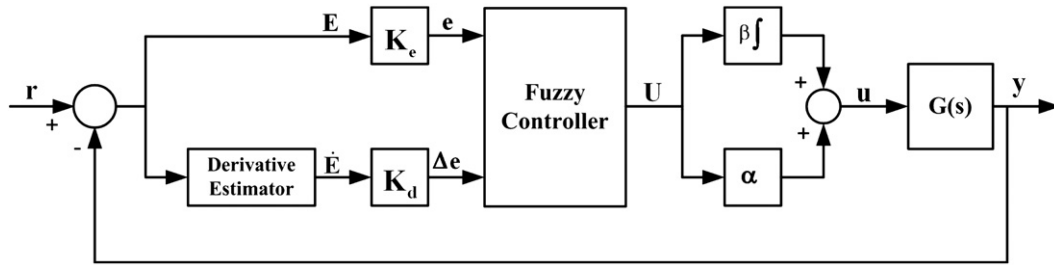


Fig. 1. The closed-loop control structure for fuzzy PID controller.

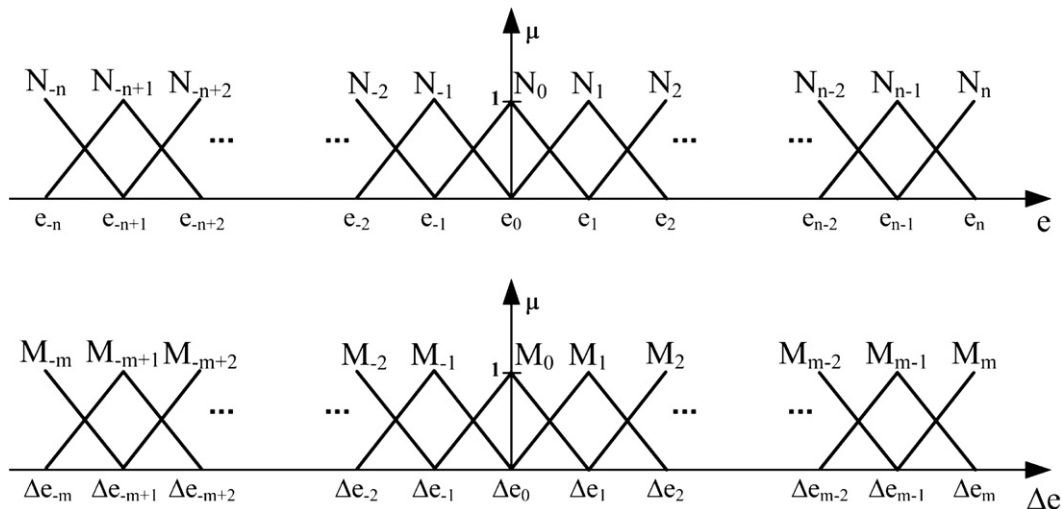


Fig. 2. The triangular membership functions for  $e$  and  $\Delta e$ .

value variable that can change for each rule and it is generally added to a rule with the phrase “with  $w$ ”. The weight values can be considered as a structural parameter of a fuzzy PID controller. The weight of the rule shows the importance or influence of that fuzzy rule for the inputs at a certain time. The fuzzy rules of FLCs with weights are generally written as

$R_k$ : IF the error ( $e$ ) is  $N_i$  AND the change of error ( $\Delta e$ ) is  $M_j$   
THEN the change of control signal ( $U$ ) is  $C_r$  with  $w_k$

Generally the weights affect the rules in two different ways (Nauck and Kruse, 1998):

- (i) The weights are applied so as to affect the complete rule. In this case, the degree of fulfillment of the rule is changed by the weight value.
- (ii) The weights are applied so as to affect the consequent part of the rules. In this case, the output of the rule is changed by the weight value.

When the singleton membership functions are used at the consequent part of the rules, the rule weights appear in the numerator and denominator for the case (i). The output of FLC is calculated as

$$U = \frac{\sum_i^m t_k c_k w_k}{\sum_i^m t_k w_k} \quad (2)$$

For the case (ii), the rule weights occur in the numerator of the (2), so the output of FLC is calculated as

$$U = \frac{\sum_i^m t_k c_k w_k}{\sum_i^m t_k} \quad (3)$$

where  $t_k$ ,  $w_k$  and  $c_k$  are the firing strength, weight and the output singleton membership value of  $k$ th rule, respectively.

Since the membership degree of a variable changes in the interval of  $[0, 1]$ , the weights must be chosen also in the interval of  $[0, 1]$  in order to keep the  $N_i$ ,  $M_j$ , and  $C_r$  as fuzzy sets. Keeping the weights within range of  $[0, 1]$  enables us to compute the output of FLC in the usual way. In this study, the case (i) where

the weights are applied to complete rules has been taken into consideration.

### 2.3. The concept of normalized acceleration

The normalized acceleration gives “relative rate” information about the fastness or slowness of the system response (Eksin et al., 2001; Guzelkaya et al., 2003). The normalized acceleration  $r_v(k)$  is defined as follows:

$$r_v(k) = \frac{de(k) - de(k-1)}{de(\cdot)} = \frac{dde(k)}{de(\cdot)} \quad (4)$$

Here,  $e(k)$  is the error value and  $de(k)$  is the incremental change in error which is given by

$$de(k) = e(k) - e(k-1) \quad (5)$$

and  $dde(k)$  is called the acceleration in error and it is given by

$$dde(k) = de(k) - de(k-1) \quad (6)$$

In (4),  $de(\cdot)$  is chosen as follows:

$$de(\cdot) = \begin{cases} de(k) & \text{if } |de(k)| \geq |de(k-1)| \\ de(k-1) & \text{if } |de(k)| < |de(k-1)| \end{cases} \quad (7)$$

The normalized acceleration  $r_v(k)$  given in (4) yields us “relative rate” information about the system response within a range of  $[-1, 1]$ . If the system response is very fast,  $r_v(k)$  approaches to “ $-1$ ”, and if the system response is very slow, it approaches to “ $-1$ ”. When the system response increases or decreases with a constant rate, it is considered as a “medium” rate and  $r_v(k)$  takes the value of zero.

### 2.4. The normalized acceleration based on-line rule weight adjustment method

For the fuzzy PID controller structure shown in Fig. 1, the step response of the closed loop system can be divided into  $2(n+1)$  main regions as illustrated in Fig. 3, depending on the number of membership functions which are assigned to the input ( $e$ ).

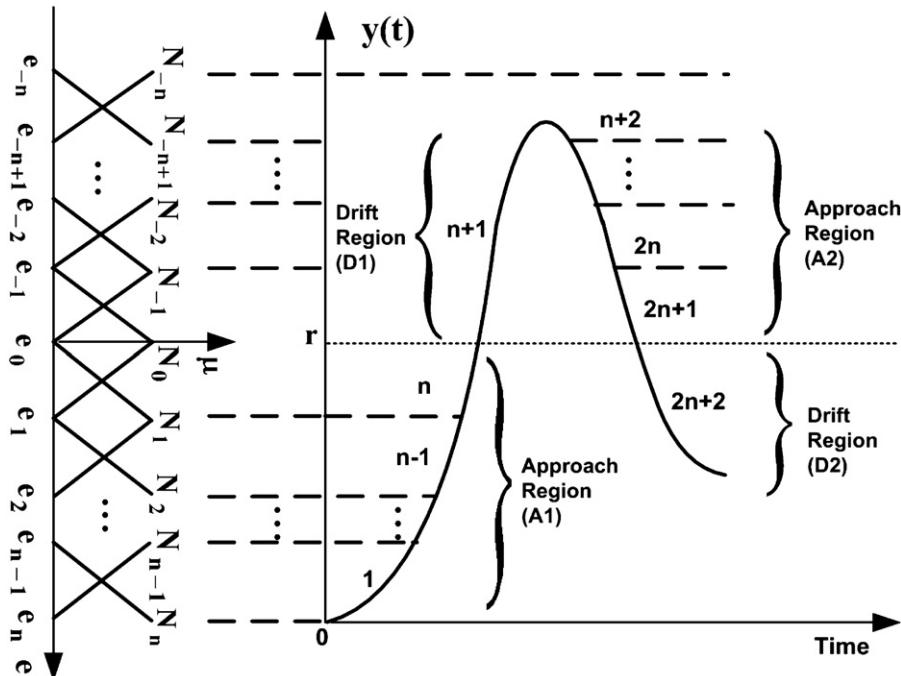


Fig. 3. The general partitioning of the step response into regions depending on the number of membership functions of the error.

It is clear that the output value of FLC should take appropriate values at different regions to obtain a desired system performance. For example, at the regions from 1 to approximately the middle of A1, where the value of the system error is positive big, a positive big value of control signal is needed to fasten the system response appropriately. At the regions from approximately the middle of A1 to D1, where the system response approaches to set-point and the value of the system error decreases but is still positive, it is required to reduce the control signal to slow down the system response. At the region D1 where the value of the system error becomes negative and the system response drifts apart from the set-point, the control signal value must be kept negative maximum in order to prevent the overshoot. For the regions from  $n+2$  to approximately the middle of A2 where the value of the system error is negative big and the system response approaches to the set-point, a negative big value of control signal is needed to fasten the system response sufficiently. At the regions from approximately the middle of A2 to D2 where the system response approaches to set-point and the value of the system error increases but still negative, it is required to increase the control signal value to slow down the system response. Finally, at the region D2 where the value of the system error becomes positive and the system response drifts apart from the set-point, the control signal value must be kept positive maximum.

According to the above discussion, the following meta-rules can be derived to form the required control signal in order to obtain a satisfactory system performance.

- (i) When the system error value is positive and the system response approaches the set-point, the control signal must be positive but its magnitude has to be decreased in time so as to obtain an appropriate fast system response.
- (ii) When the system error value is negative and the system response drifts apart from the set-point, a maximum negative big control signal is needed to prevent the overshoot.
- (iii) When the system error value is negative and the system response approaches the set-point, the control signal value must be negative but its magnitude is to be increased in time so as to obtain an appropriate fast system response.
- (iv) When the system error value is positive and the system response drifts apart from the set-point, a maximum positive big control signal is needed to prevent the undershoot.

These meta-rules can be associated with the importance of the fuzzy rules in order to obtain the required control signal for a satisfactory system performance. For example; when a positive big control signal is needed, the importance of the fuzzy rules which have positive rule consequent will be greater than the importance of the fuzzy rules that have negative rule consequent. Similarly, when there is a need to decrease the control signal in time, the importance of the fuzzy rules that have negative rule consequent will be greater than the importance of the fuzzy rules that have positive rule consequent. Since the importance of a fuzzy rule is related to its weight value, the change of the importance of a fuzzy rule at any region can be expressed as changing the weight value of that rule.

There exists a symmetrical structure between the Approach Regions (A1 and A2) and Drift Regions (D1 and D2) as shown in Fig. 3. For example, when the system response is at any point in the region “ $n$ ”, the control signal value should be the same strength with opposite sign at its symmetrical point in the region “ $2n+1$ ” due to the symmetrical structure. Using this symmetrical structure and the meta-rules derived above, the fuzzy rule weights of the fuzzy rules should be adjusted symmetrically

according to the intersection of the cores  $e_0$  and  $\Delta e_0$  as illustrated in Fig. 4.

Appropriate rule weight values can be generated in an online manner using a function or a mechanism that depends on the input, output and internal system information such as error, rate of error, normalized acceleration, and/or control signal magnitude. In Karasakal et al. (2011), simple functions of the system error have been directly used in generation of the rule weight values by an adequate arrangement in accordance with the meta-rules given above. In this study, the normalized acceleration “ $r_v$ ” is also taken into consideration as well as the system error “ $e$ ” in the generation of the rule weight values. Thus, the relative rate information about the system response is inserted into the structure of FLC using rule weight values. The weight tuning is accomplished via a fuzzy mechanism which makes use of the system error “ $e$ ” and the normalized acceleration “ $r_v$ ” as its inputs. The fuzzy mechanism is not in the need of any input scaling factor adjustment since the variables “ $e$ ” and “ $r_v$ ” are both in the range of  $[-1, 1]$ . The output of the fuzzy mechanism is designated as “ $\gamma$ ” and the block diagram of the proposed method is illustrated in Fig. 5. Since the interval for rule weight values should lie within the range  $[0, 1]$ , the interval of the  $\gamma$  lies within the range  $[0, 1]$ . Then, the functions

$$f_1 = b\gamma \quad (8)$$

$$f_2 = 1 - \gamma \quad (9)$$

are directly assigned as the rule weight values by an adequate arrangement shown in Fig. 4. The parameter  $\gamma$  then becomes the tuning variable of the rule weights by these assignments. Since the error value possesses its extreme value at the first region a special precaution is required in the arrangements of the functions  $f_1$  and  $f_2$  in Fig. 4. In that respect, in order to prevent the excessive acceleration of the system and thus the possible overshoots, the value of “ $b$ ” in (8) is defined as

$$b = \frac{1}{(2n-1)^2} \quad (10)$$

This has to be done only for the region number “1” in the adjustment of the weight of a rule that has positive rule consequent. The value of the parameter “ $b$ ” is taken to be as “1” for the rest of the regions.

In accordance with the arrangement of the functions  $f_1$  and  $f_2$  in the Fig. 4, the meta-rules for determining  $\gamma$  can be deduced as follows:

- (a) If the system response is fast,  $\gamma$  should be small (so  $f_2$  will be big and  $f_1$  will be small). Thus, a big negative (for the regions A1 and D1)/positive (for the regions A2 and D2) control value is obtained and the system response can be retrained. If the system response is slow,  $\gamma$  should be big (so  $f_2$  will be small and  $f_1$  will be big). Thus, a big positive (for the region A1)/negative (for the region A2) control value is obtained and the system response can be fastened.
- (b) If the system response is fast then small  $\gamma$  values should be increased slightly in the regions where system error value is around zero. This would provide a smoother control signal.
- (c) If the system response is slow and the system error value is big then  $\gamma$  should be big so as to increase the control signal in positive sense (for the region A1)/negative sense (for the region A2). When the system response approaches to reference value,  $\gamma$  value should be decreased relatively in order to produce a smoother control signal.

In order to satisfy the above meta-rules, a fuzzy mechanism which uses the system error “ $e$ ” and the normalized acceleration

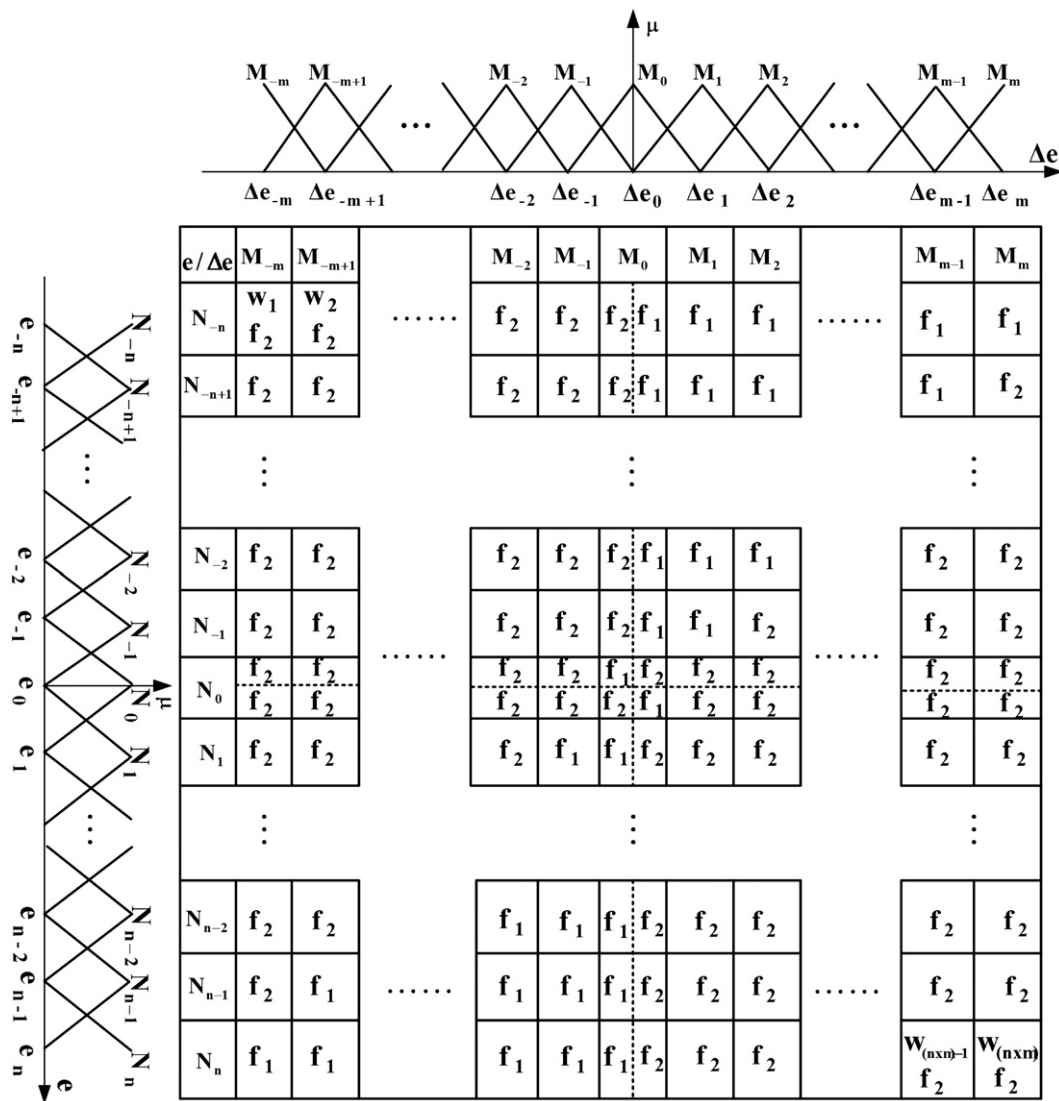
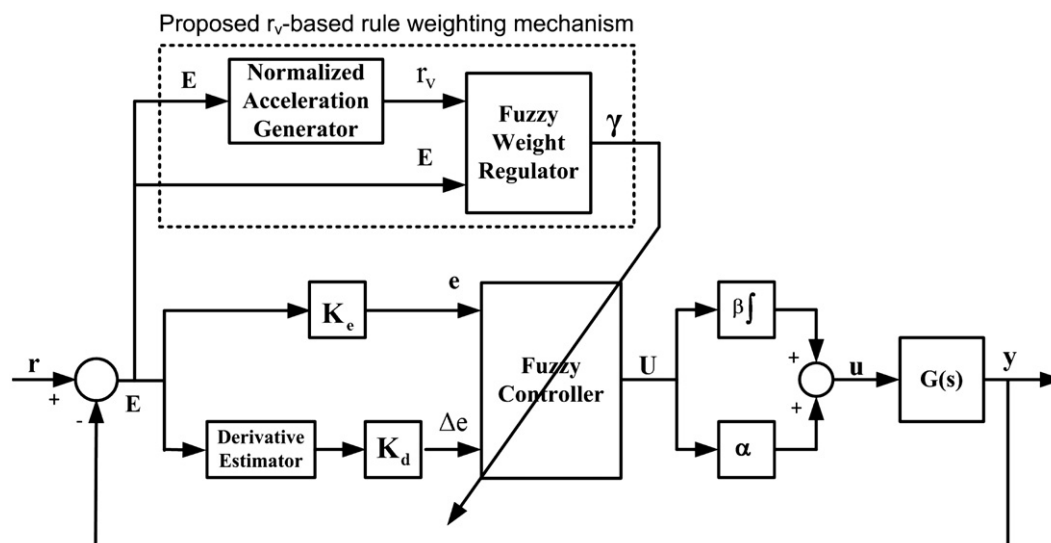
Fig. 4. The rule weight adjustment table for  $n \times m$  input membership functions.

Fig. 5. The closed-loop control structure for fuzzy PID controller using on-line rule weighting method.



" $r_v$ " as its inputs can be proposed. The membership functions and rule base tables are given for the fuzzy mechanisms having 9 and 25 rules as examples.

Symmetrical triangular uniformly distributed membership functions are assigned for the input and output variables for the fuzzy mechanism having 9 rules as shown in Fig. 6. The input variables have been decomposed into three different fuzzy linguistic levels which are Negative (N), Zero (Z), Positive (P) and Slow (S), Medium (M), Fast (F).

The output variable has been decomposed into three different fuzzy linguistic levels as follows: Small (S), Medium (M), Large (L). A fuzzy rule base satisfying the meta-rules given above is presented in Table 1.

Symmetrical triangular uniformly distributed membership functions are assigned for the input and output variables for the fuzzy mechanism having 25 rules as shown in Fig. 7. The input variables have been decomposed into five different fuzzy linguistic levels which are Negative Big (NB), Negative Medium (NM), Zero (Z), Positive Medium (PM), Positive Big (PB) and Slow (S), Slow Medium (SM), Medium (M), Fast Medium (FM), and Fast (F).

The output variable has been decomposed into five different fuzzy linguistic levels as follows: Small (S), Small Medium (SM), Medium (M), Medium Large (ML), Large (L). A rule base satisfying the meta-rules given above is presented in Table 2.

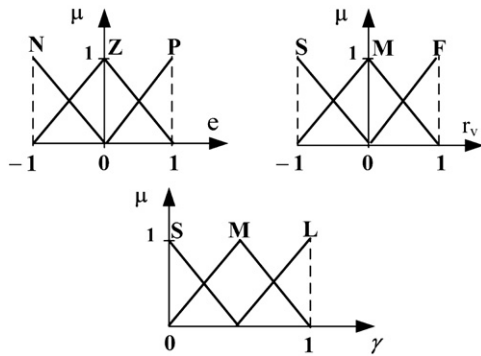


Fig. 6. The membership functions for inputs  $e$ ,  $r_v$  and output  $\gamma$  for 9 rules.

**Table 1**  
The rule base of the fuzzy weight regulator for  $3 \times 3$  membership functions.

$e/r_v$	S	M	F
N	L	M	S
Z	M	S	S
P	L	M	S

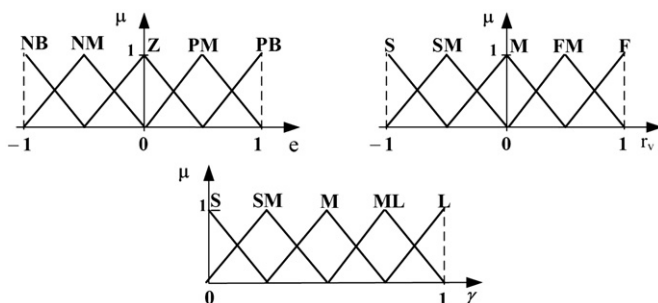


Fig. 7. The membership functions for inputs  $e$ ,  $r_v$  and output  $\gamma$  for 25 rules.

**Table 2**

The rule base of the fuzzy weight regulator for  $5 \times 5$  membership functions.

$e/r_v$	S	SM	M	FM	F
NB	L	M	SM	S	S
NM	M	SM	S	S	S
Z	ML	M	SM	SM	SM
PM	M	SM	S	S	S
PB	L	M	SM	S	S

### 3. Simulation studies

#### 3.1. The fuzzy PID controller structure

In order to explain the method more vividly, a special case is taken in consideration where three membership functions are assigned to both inputs ( $e$ ) and ( $\Delta e$ ) of the FLCs. The membership functions for the inputs are chosen as uniformly distributed triangular functions and for the output singleton membership functions are used. The input variables have been decomposed into three different fuzzy linguistic levels which are Negative (N), Zero (Z) and Positive (P).

The output variable has been decomposed into five different fuzzy linguistic levels as follows: Negative Big (NB), Negative Medium (NM), Zero (Z), Positive Medium (PM) and Positive Big (PB).

The membership functions for the inputs and the output have been illustrated in Fig. 8a and b, respectively. The diagonally symmetrical rule base, which is given in Table 3, has been used for the fuzzy PID controllers. The control surface for this rule base is given in Fig. 8c.

#### 3.2. The rule weight assignment

Since the number of membership functions for the input ( $e$ ) is three, we obtain four main regions for the fuzzy rule weight assignment as illustrated in Fig. 9.

The rule weight adjustment table illustrated in Fig. 10 for  $3 \times 3$  input membership functions case is obtained from the systematic rule weight adjustment structure given in Fig. 4.

#### 3.3. The other methods and performance measures used for comparison

In the simulation studies, the performance of the proposed on-line weight tuning method is compared with four other fuzzy PID controllers: Mudi and Pal (1999) and Karasakal et al. (2011), the conventional fuzzy PID controller with the optimized rule weights and with the conventional fuzzy PID controller without rule weighting.

It is obvious that if the rule weights were set the same for all the rules, i.e.  $f_1 = f_2$ , this would resemble the adjustment of the output scaling factor. For this reason the fuzzy PID controller with the proposed rule weight adjustment method can be compared with the fuzzy PID controller proposed by Mudi and Pal (1999) which uses a self-tuning scheme for the output scaling factor.

Moreover, the proposed weight adjustment method is compared with the weight adjustment method presented in Karasakal et al. (2011). In that study, rule weighting is established by using simple functions of system error in an adequate arrangement. For this purpose, the absolute value of error has been directly used in generation of the rule weight values in accordance with the same meta-rules presented to form the required control signal in this study. By this comparison, the benefits of taking the normalized

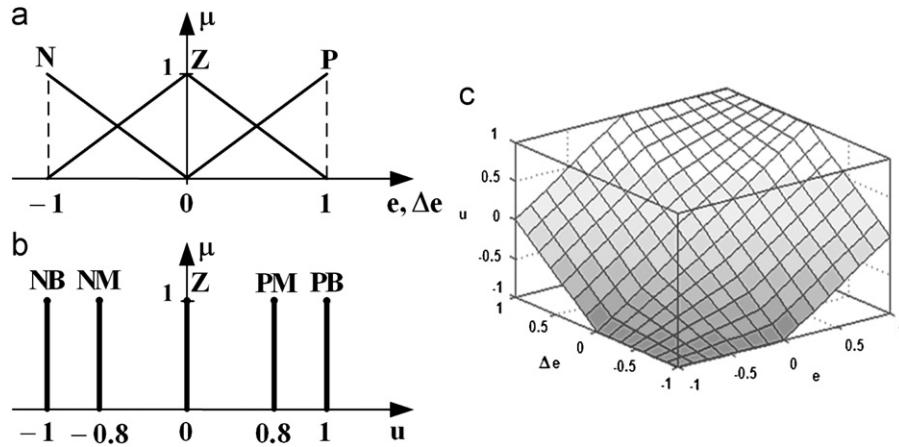


Fig. 8. The membership functions for (a) inputs  $e$  and  $\Delta e$ , (b) output  $U$ , and (c) control surface.

**Table 3**  
The rule base of the fuzzy PID controller.

$e/\Delta e$	N	Z	P
N	NB ( $w_1$ )	NM ( $w_2$ )	Z ( $w_3$ )
Z	NM ( $w_4$ )	Z ( $w_5$ )	PM ( $w_6$ )
P	Z ( $w_7$ )	PM ( $w_8$ )	PB ( $w_9$ )

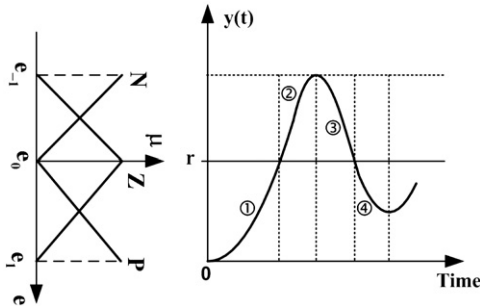


Fig. 9. The partitioning of the step response into regions in accordance with 3 input membership functions for the error.

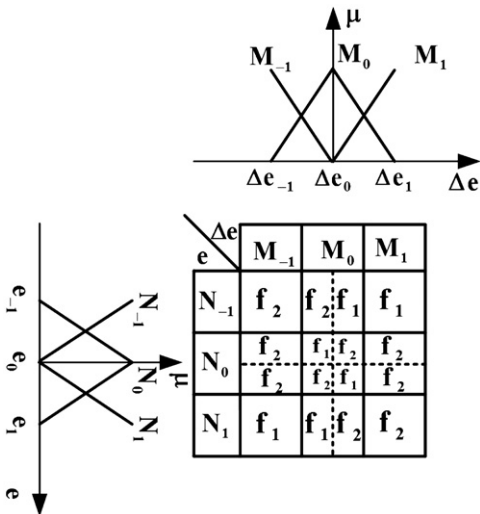


Fig. 10. The rule weight adjustment table for the case of  $3 \times 3$  input membership functions (9 rules).

acceleration “ $r_v$ ” into consideration as well as the system error “ $e$ ” in the generation of the rule weight values is demonstrated.

In comparison of the fuzzy PID controller with proposed method with the fuzzy PID controller whose weights are optimized in off-line manner, genetic algorithm is used to minimize the integral absolute error (IAE) performance measure.

In order to compare the performance of transient responses of the fuzzy PID controller with the proposed normalized acceleration based weight adjustment method with other fuzzy PID controllers, five different performance measures are considered. The two of these performance measures are selected from the classical transient system response criteria; namely the maximum overshoot (%OS) and the settling time ( $T_s$ ). The next three performance measures are considered to be

(i) IAE which is defined as

$$IAE = \int_0^\infty |r(t) - y(t)| dt \quad (11)$$

(ii) integral time absolute error (ITAE) which is defined as

$$ITAE = \int_0^\infty t |r(t) - y(t)| dt \quad (12)$$

(iii) total variation (TV) (Skogestad, 2003) of the control input  $u(t)$ , which is defined as

$$TV = \sum_{i=1}^\infty |u_{i+1} - u_i| \quad (13)$$

### 3.4. Simulation results

In all the simulation studies, discrete simulation method is used with a sampling time of 0.1 s. During the simulations, the conventional fuzzy PID controller without rule weighting is denoted as NWA, the fuzzy PID controller with a self-tuning scheme for the output scaling factor is denoted as OSFT, the conventional fuzzy PID controller with the optimized rule weights is denoted as WO, the fuzzy PID controller with the error based weight adjustment method is denoted as EBWA and the fuzzy PID controller with the proposed normalized acceleration based weight adjustment method is denoted as RVBWA. To be fair, the same input and output membership functions given in Fig. 8 and the same rule base given in Table 3 are used for each of these fuzzy PID controllers. The initial values of all the fuzzy rule weights are chosen as “1”.

### 3.4.1. System I

Since most of the high-order processes can usually be modeled as second order plus dead time linear systems, the following system given by Mudi and Pal (1999) is taken into consideration:

$$G_p(s) = \frac{e^{-0.3s}}{s^2 + s + 0.2} \quad (14)$$

The scaling factor values are also taken from Mudi and Pal (1999) as follows:

$$K_e = 0.9, \quad K_d = 13.5, \quad \beta = 0.02, \quad \alpha = 0.$$

These scaling factors lead to an oscillatory output response with a moderate overshoot but a small rise time. The unit step output responses and the control signals of NWA, OSFT, WO, EBWA and RVBWA are in Fig. 11. The disturbance with the amplitude of 0.2 is applied to the system at time 50 s.

The performance analysis of the fuzzy PID controllers according to the performance measures used is given in Table 4. The proposed RVBWA reduces the overshoot value as it is expected when it is compared to NWA, WO, EBWA and OSFT. As given in Fig. 11, the settling time of the proposed RVBWA is significantly smaller than the others. The ITAE value of RVBWA is drastically less than the other fuzzy PID controllers since the oscillations vanish. The proposed method has a low value of TV that shows that it has the smoothest control signal. Finally, the effect of disturbance is compensated in a shorter time with the use of proposed method.

### 3.4.2. System II

Consider the non-linear process given again by Mudi and Pal (1999) as

$$\frac{d^2y(t)}{dt^2} + \frac{dy(t)}{dt} + 0.25y^2 = u(t-L) \quad (15)$$

The parameter  $L$  is chosen to be 0.5 s and the scaling factors of the fuzzy PI controllers are taken as follows (Mudi and Pal, 1999):

$$K_e = 0.9, \quad K_d = 11, \quad \beta = 0.018, \quad \alpha = 0.$$

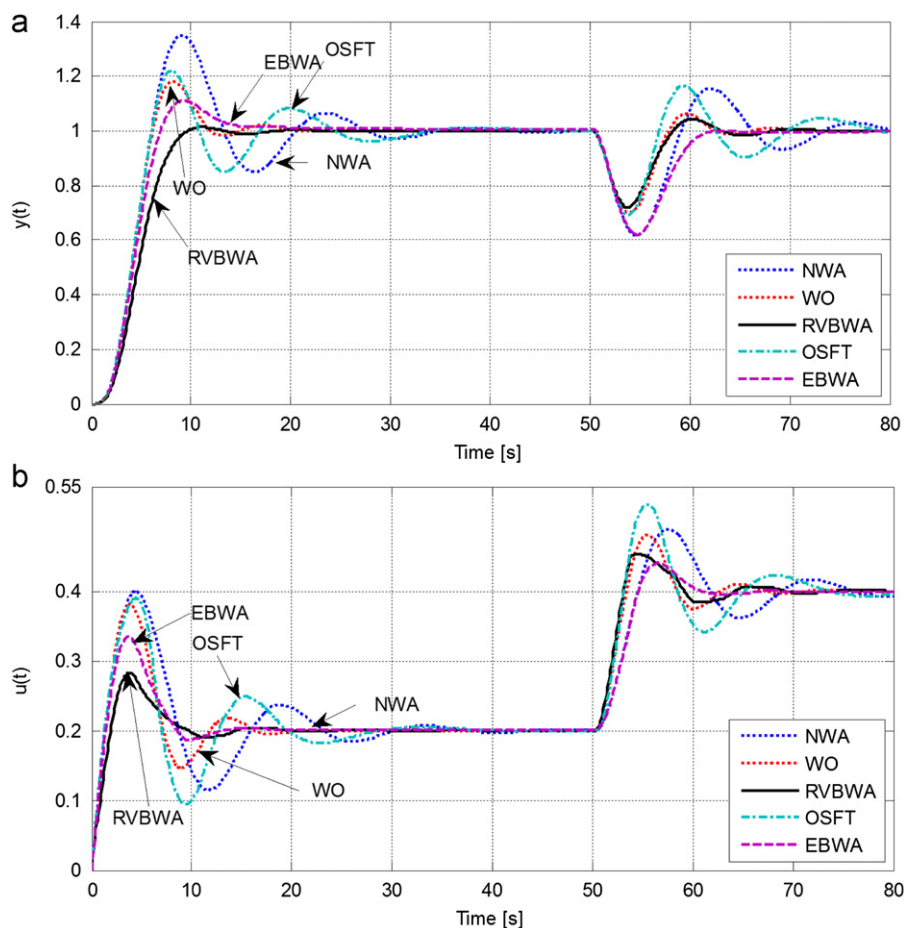
Using these scaling factors an oscillatory output response with small rise time is obtained. The unit step output responses and the control signals of NWA, WO, OSFT, EBWA and RVBWA are shown in Fig. 12. The disturbance with the amplitude of 0.2 is applied to the system at time 35 s.

The comparison of the fuzzy PI controllers according to the performance measures used is given in Table 5. As parallel to the

**Table 4**

The performance analysis of NWA, WO, OSFT, EBWA and RVBWA for System I.

Fuzzy PID controller	%OS	$T_s$ (s)	IAE	ITAE	TV
NWA	35	25.2	9.568	211.0	0.152
WO	18	10.9	6.286	100.0	0.136
OSFT	22	22.4	8.252	177.1	0.167
EBWA	11	12.3	7.280	148.0	0.101
RVBWA	1.3	8.30	6.222	84.49	0.090



**Fig. 11.** Simulation results for System I: (a) system outputs and (b) control signals.



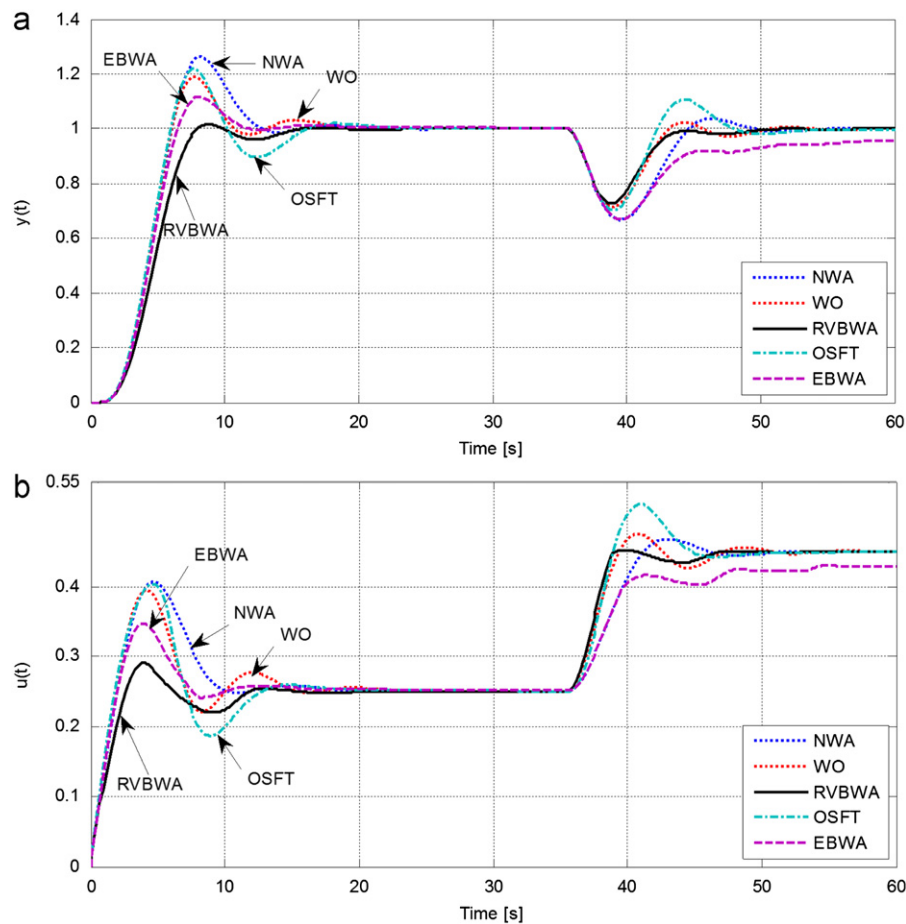


Fig. 12. Simulation results for System II: (a) system outputs and (b) control signals.

**Table 5**  
The performance analysis of NWA, WO, OSFT EBWA and RVBWA for System II.

Fuzzy PID controller	%OS	$T_s$ (s)	IAE	ITAE	TV
NWA	26.3	11.6	7.096	93.2	0.103
WO	19.1	10.0	6.22	69.9	0.118
OSFT	22.1	14.4	6.76	85.1	0.125
EBWA	11.7	10.0	7.50	136	0.089
RVBWA	1.60	7.20	6.10	62.9	0.084

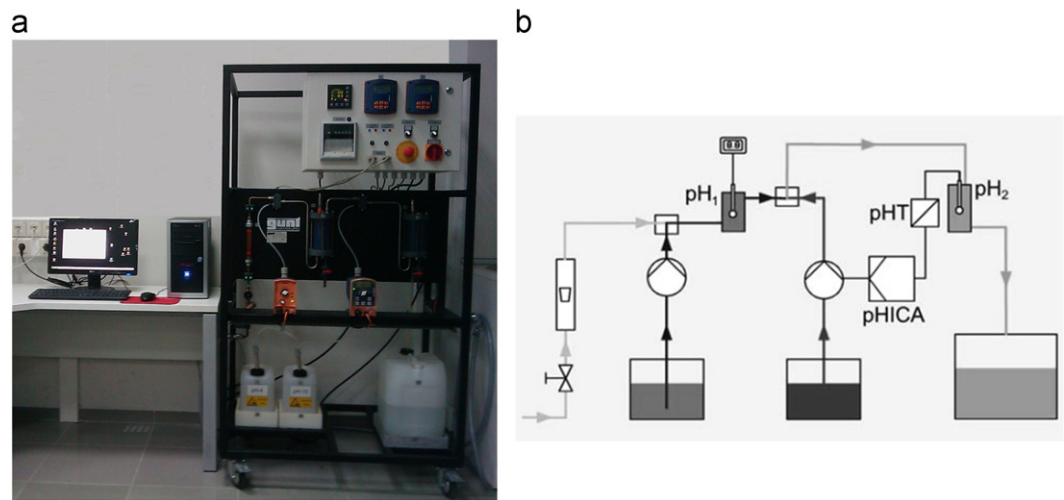


Fig. 13. (a) The general view of the G.U.N.T. RT-552 pH Process set and (b) the schematic working diagram of the RT-552 pH process set.

results obtained for linear system, proposed RVBWA improves the transient performance also for non-linear system. Maximum overshoot and the settling time are decreased. In addition, *IAE*, *ITAE* and *TV* values show the performance improvement capacity of the proposed RVBWA. With the proposed weight assignment method, the disturbance is again compensated in a shorter time as it compared to the other control structures.

#### 4. Experiment

In order to show the effectiveness of the proposed method, real time control system applications have been performed. Both NWA and RVBWA have been applied to a highly non-linear pH control process. The neutralization process is complex industrial process with dominant nonlinearities. Since the pH process inherits severe nonlinearities and time varying behavior, it is a benchmark process to evaluate control performances. The experimental set-up handled in this study is G.U.N.T. pH Value Control

Trainer RT-552, which is illustrated in Fig. 13. This process has two input streams, one containing strong acid (H-Cl) and the other one strong base (Na-OH).

In order to obtain a single input-single output system, the acetic acid stream,  $F_a$ , is considered to be constant at its nominal value while the base flow rate is considered as the manipulated variable ( $F_b$ ). Therefore, the process can be described as the sodium hydroxide stream as input ( $F_b$ ) and the pH ( $pH_2$ ) as output. The parameters of the neutralization process are given in Table 6.

In order to determine the scaling factors of the fuzzy PID controllers, the fuzzy model of this process obtained. For that purpose, an identification experiment is applied to RT-552 with the sampling time of 2 s (Kumbasar et al., 2011a, 2011b, 2011c). The regression vector is constructed by considering  $pH(k-1)$  and  $F_b(k-1)$  values. The collected input and output data has then been trained by ANFIS Toolbox/MATLAB to obtain the fuzzy model. In order to represent the nonlinearities of pH process better, the membership functions of each input in the fuzzy rule base are defined with six Gaussian membership functions (Kumbasar et al., 2011c).

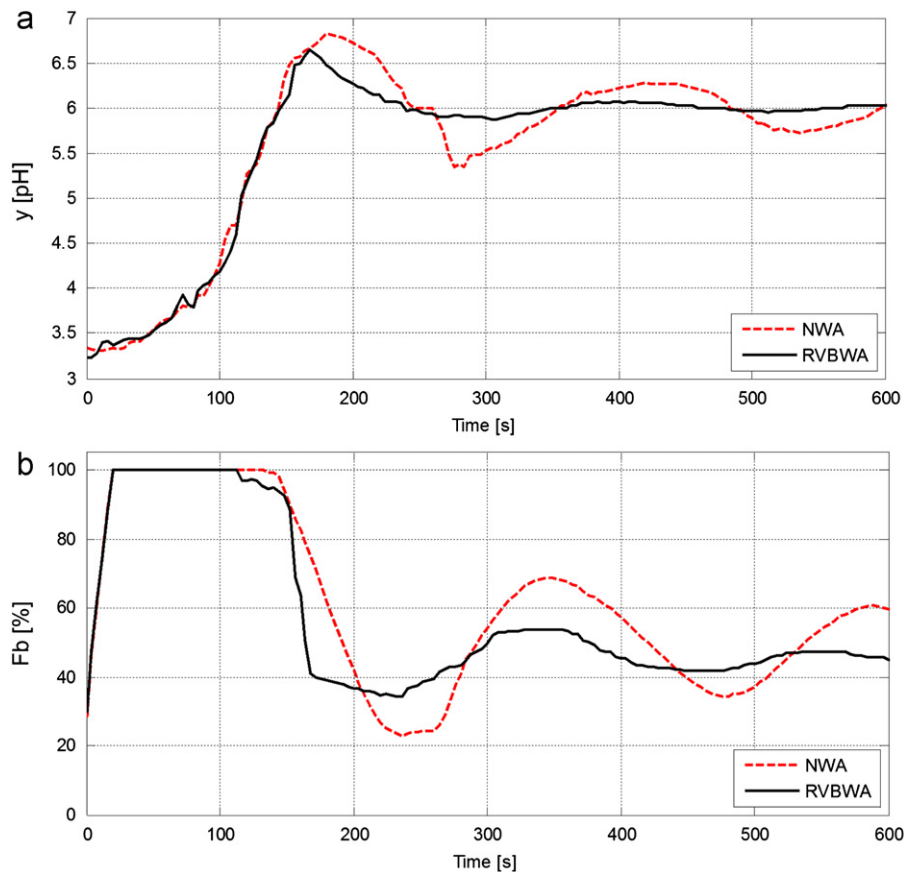
Since the pH process to be controlled has high nonlinearity related to molarities of  $[H]/[OH]$ , the performances of the

**Table 6**  
Parameters of the pH neutralization process.

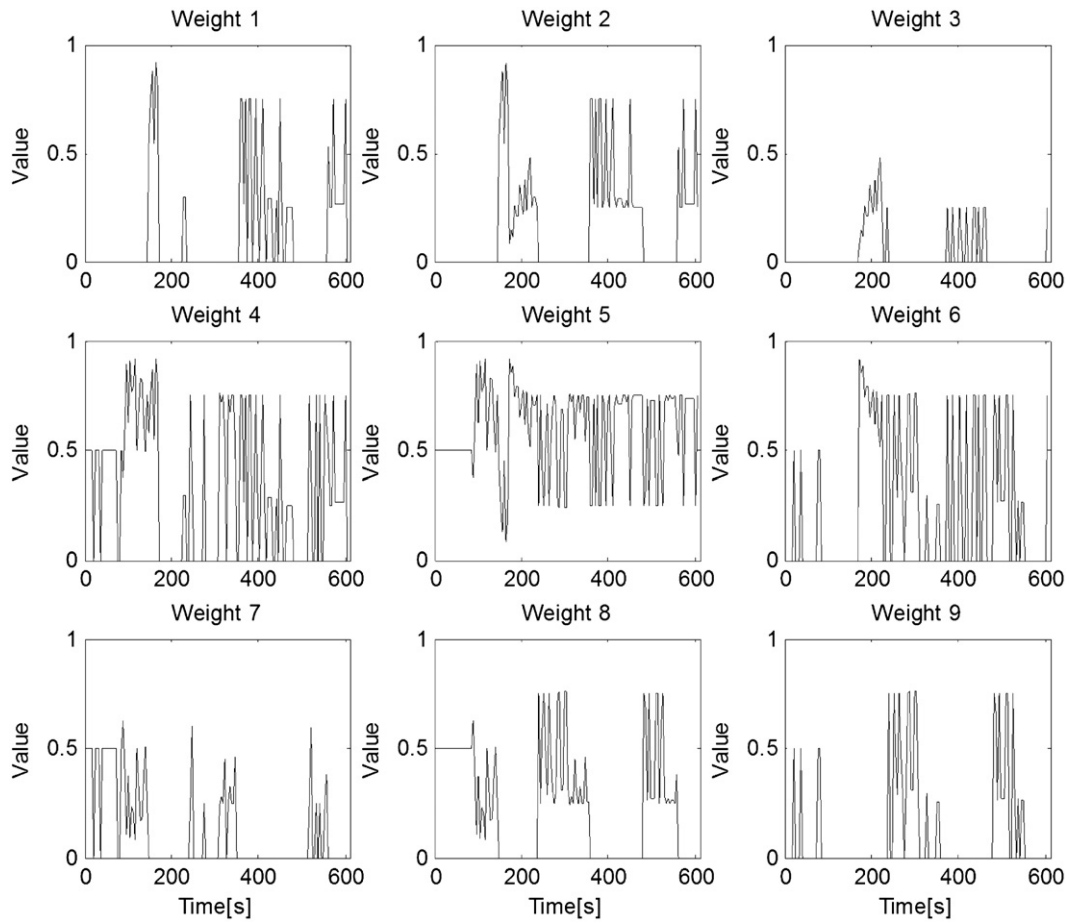
Symbols	Description	Value
$V$	Volume of the CSTR	0.8 l
$F_a$	Flow rate of the influent stream	1 ml/s
$F_b$	Flow rate of the titrating stream	0–2.1 ml/s (0–100%)
$C_a$	Concentration of the influent stream	$6.3096 \times 10^{-4}$ M
$C_b$	Concentration of the titrating stream	$13 \times 10^{-4}$ M

**Table 7**  
The performance analysis of NWA and RVBWA for Case 1.

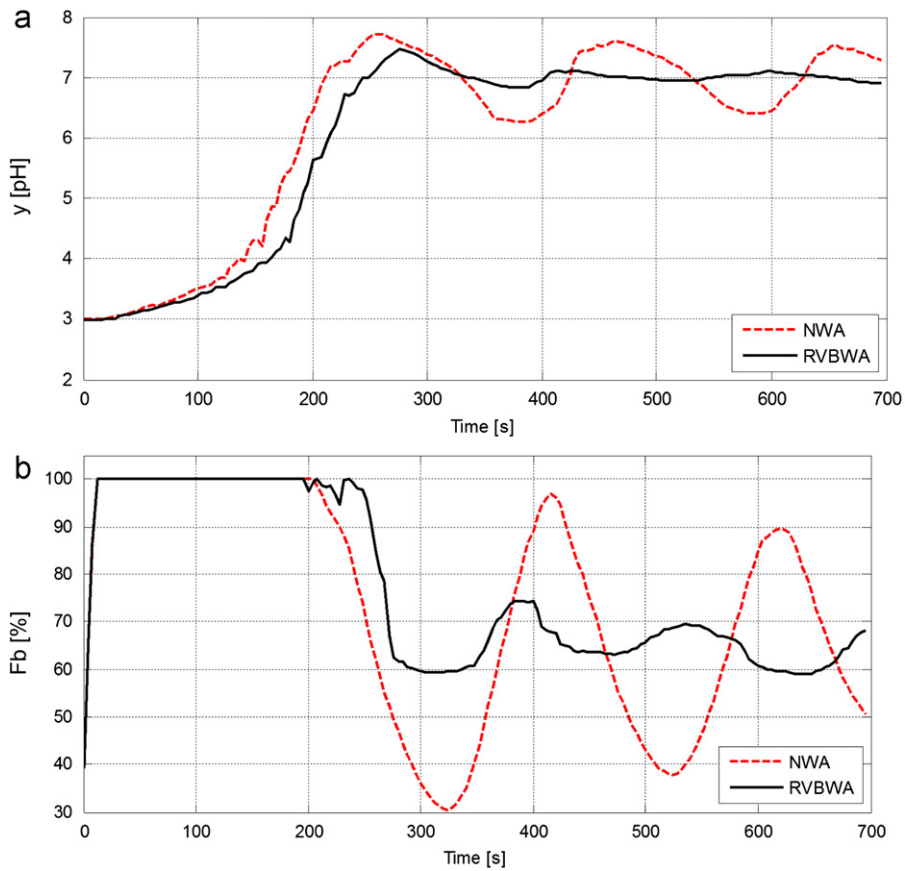
Fuzzy PID controller	%OS	$T_s$ (s)	IAE	ITAE	TV
NWA	13.69	328.0	103.3	14150	292.56
RVBWA	10.81	192.0	81.29	6637	213.53



**Fig. 14.** Experiment results of the pH neutralization process for Case 1: (a) system outputs and (b) control signals.



**Fig. 15.** The value changes of the 9 fuzzy rule weights for the reference value 6 pH.



**Fig. 16.** Experiment results of the pH neutralization process for Case 2: (a) system outputs and (b) control signals.

controller with the proposed method are tested for different reference values at the first three cases. The desired reference values have been chosen as 6, 7 and 8 pH for Case1, Case 2 and Case 3, respectively. An additional experiment has been performed on the Case 4 in order to test the robustness of the proposed method under disturbance. The disturbance is applied as manipulating the flow rate of the acetic acid stream.

The fuzzy PID controller structure given in Section 3.1 is implemented in the real-time control experiments. For the proposed method, the initial values of all the fuzzy rule weights are chosen as “1”.

**Case 1.** The reference value is taken as 6 pH value. The scaling factors of the fuzzy PID controllers are chosen as  $K_e = 0.11$ ,  $K_d = 0.2$ ,  $\beta = 15$  and  $\alpha = 50$  using the system model just to obtain a fast system response for this process. Using these scaling factors an oscillatory output response with a moderate overshoot is obtained. The step input system responses and the control signals of NWA and RVBWA are shown in Fig. 14. The comparison of the fuzzy PID controllers according to the performance measures given in Section 3.3 is shown in Table 7.

The Fig. 15 shows the value changes of the 9 fuzzy rule weights with time for the real time control application with reference value 6 pH. As it is seen from Fig. 15, the fuzzy rule weights take different values at different regions of the system response. For example; approximately between 145 and 170 s, when the system error is negative and the change of error is negative (the region D1 in Fig. 3) the fuzzy rule weights 1, 2, 4 and 5 are active and take appropriate values between 0 and 1 where others are inactive and do not take any value.

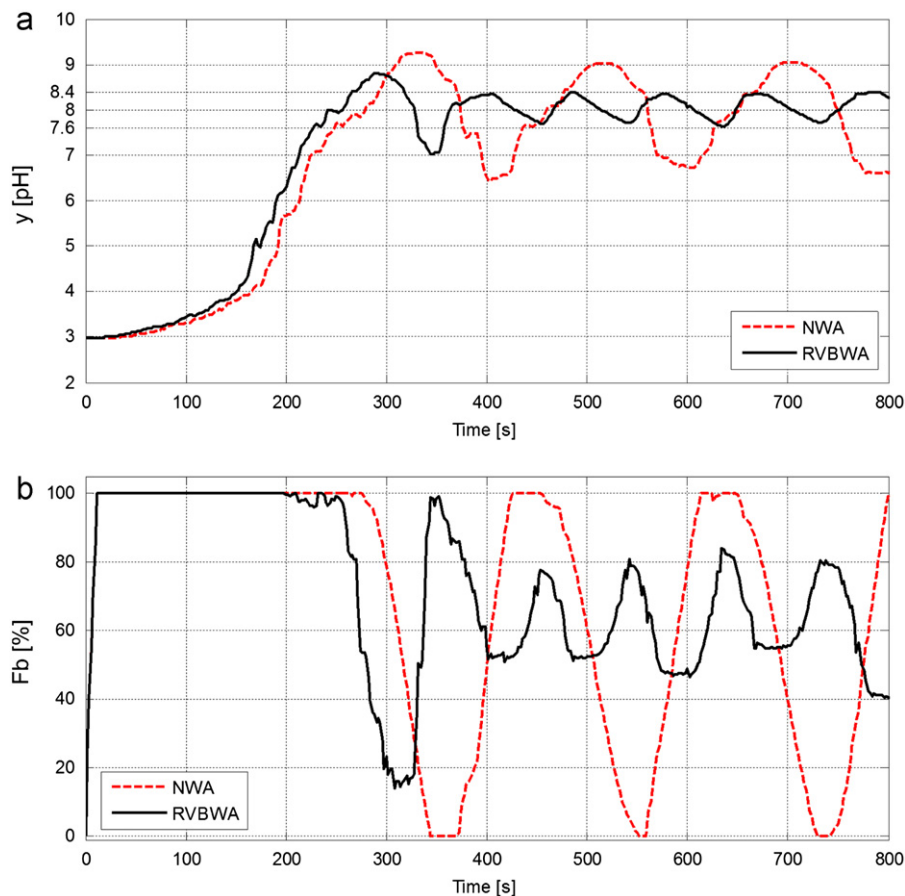
**Case 2.** The reference value is taken as 7 pH value. The scaling factors of the fuzzy PID controllers are chosen as  $K_e = 0.11$ ,  $K_d = 0.2$ ,  $\beta = 15$ ,  $\alpha = 50$  using the system model just to obtain a fast system response for the system. Naturally, an oscillatory output response is obtained with these scaling factors. The step system responses and the control signals of NWA and RVBWA are shown in Fig. 16. The comparison of the fuzzy PID controllers according to the performance measures used is given in Table 8.

**Case 3.** The reference value is taken as 8 pH value. The scaling factors of the fuzzy PID controllers are chosen as  $K_e = 0.11$ ,  $K_d = 0.4$ ,  $\beta = 15$ ,  $\alpha = 50$  using the system model just to

**Table 8**

The performance analysis of NWA and RVBWA for Case 2.

Fuzzy PID controller	%OS	$T_s$ (s)	IAE	ITAE	TV
NWA	10.36	–	210.52	36,078	396.85
RVBWA	6.680	272	191.43	21,299	218.08



**Fig. 17.** Experiment results of the pH neutralization process for Case 3: (a) system outputs and (b) control signals.

obtain a fast system response for the system. These scaling factors also produce an oscillatory output response. The step system responses and the control signals of NWA and RVBWA are shown in Fig. 17. The comparison of the FLCs according to the selected performance measures is given in Table 9. The dashed-lines in Fig. 17 indicate the bounds of the 5% band of the reference value, which actually defines the settling time.

**Case 4.** The fuzzy PID controller with RVBWA has been applied to pH process set in order to test the robustness of the proposed method under disturbances. For this purpose, the flow rate of the influent stream has been manipulated. At first, the neutralization process has been forced to attain the reference value 7 pH. Then, the flow rate of the influent stream has been increased by 50% (1.5 ml/s), and reduced back to the default value to examine the disturbance rejection. Secondly, the flow rate ( $F_a$ ) has been decreased by 40% (0.6 ml/s), and increased back. The above described disturbances are applied to the system at times  $t=116$  s and 342 s, respectively. For this case, the scaling factors of the fuzzy PID controllers are chosen as in Case 2.

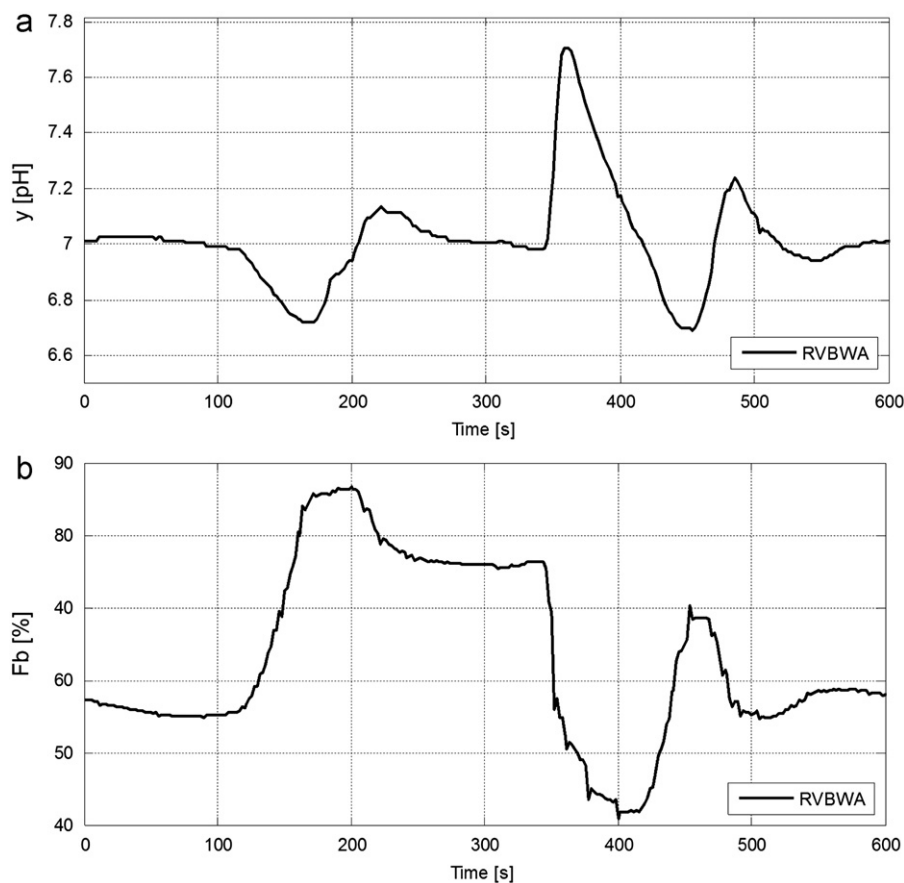
The system response and the control signal of RVBWA are shown in Fig. 18. When the same disturbances are applied to the system without the proposed weight tuning method, the system response becomes unstable and therefore it is not illustrated within Fig. 18.

From the experiments it has been observed that the proposed RVBWA reduces maximum overshoots compared to the NWA. For each case, the settling time of the fuzzy PID controller with the proposed rule weighting method is reduced significantly. The ITAE values of RVBWA are less than the NWA with smaller TV values that show the smoothness of the control signal. The results also show that the disturbances are also compensated fairly well with the proposed method.

Moreover, the average computational time needed for calculating the control signal for experiment runs at each sampling time was measured as 0.029 for the conventional Fuzzy PID controller and 0.058 s for the proposed controller structure. These computations were done on a personal computer with an Intel Pentium Dual Core T2370 1.73 GHz processor, 2.99 GB RAM, and software package MATLAB/Simulink

**Table 9**  
The performance analysis of NWA and RVBWA for Case 3.

Fuzzy PID controller	%OS	$T_s$ (s)	IAE	ITAE	TV
NWA	15.84	–	707.7	175,970	881.77
RVBWA	10.21	360	525.63	84,452	617.91



**Fig. 18.** Experimental results of the pH neutralization process for Case 4: (a) system output and (b) control signal.



7.4.0. This result shows that the proposed fuzzy rule weighting method does not bring any serious additional computational burden. The obtained computational time confirms that the proposed method is feasible even for systems with faster dynamics.

## 5. Conclusion

An on-line adjustment method for the rule weights of the fuzzy PID controller is presented. The adjustment of rule weight values are performed in a simple way using a fuzzy logic mechanism which uses the “normalized acceleration” in addition to “system error” as inputs and the output of this fuzzy mechanism is directly assigned as the weight values of the fuzzy rules.

The proposed on-line fuzzy rule weight assignment method is applied to both fuzzy PI and PID controllers and performances are tested on linear and non-linear processes by simulations. The performance results of the fuzzy PID controller with the proposed method are compared with the fuzzy PID controller proposed by Mudi and Pal (1999) which uses a self-tuning scheme for the output scaling factor, the fuzzy PID controller with error based weight adjustment method presented in Karasakal et al. (2011), the fuzzy PID controller with optimized weights and the conventional fuzzy PID controller with no rule weight adjustment.

In order to make a fair comparison, several performance measures are used such as maximum overshoot, settling time, integral absolute error, integral time absolute error and total variation of the control signal. In addition to simulations, the effectiveness and the robustness of the proposed weight assignment method are demonstrated on a pH neutralization process for various reference values. The results obtained from the simulations and the experiments show the drastic improvements on performance measures and show that the disturbances are also compensated much more effectively with the use of the proposed weight adjustment method.

## References

- Ahn, K.K., Truong, D.Q., 2009. Online tuning fuzzy PID controller using robust extended Kalman filter. *J. Process Control* 19, 1011–1023.
- Alcala, R., Casillas, J., Cordon, O., Gonzalez, A., Herrera, F., 2005. A genetic rule weighting and selection process for fuzzy control of heating, ventilating and air conditioning systems. *Eng. Appl. Artif. Intell.* 18, 279–296.
- Bhattacharya, S., Chatterjee, A., Munshi, S., 2004. A new self-tuned PID-type fuzzy controller as a combination of two-term controllers. *ISA Trans.* 43, 413–426.
- Bouallegue, S., Haggege, J., Ayadi, M., Benrejeb, M., 2011. PID-type fuzzy logic controller tuning based on particle swarm optimization. *Eng. Appl. Artif. Intell.* <http://dx.doi.org/10.1016/j.engappai.2011.09.018>.
- Chen, K.-Y., Tung, P.-C., Tsai, M.-T., Fan, Y.-H., 2009. A self-tuning fuzzy PID-type controller design for unbalance compensation in an active magnetic bearing. *Expert Syst. Appl.* 36, 8560–8570.
- Chung, H.Y., Chen, B.C., Lin, J.J., 1998. A PI-type fuzzy controller with self-tuning scaling factors. *Fuzzy Sets Syst.* 93, 23–28.
- Duan, X.-G., Li, H.-X., Deng, H., 2008. Effective tuning method for fuzzy PID with internal model control. *Ind. Eng. Chem. Res.* 47, 8317–8323.
- Eksin, I., Guzelkaya, M., Gurleyen, F., 2001. A new methodology for deriving the rule-base of a fuzzy logic controller with a new internal structure. *Eng. Appl. Artif. Intell.* 14–5, 617–628.
- Fang, G., Kwok, N.M., Ha, Q., 2008. Automatic membership function tuning using the particle swarm optimization. In: 2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application, pp. 324–328.
- Fuente, M.J., Robles, C., Casado, O., Syafie, S., Tadeo, F., 2006. Fuzzy control of a neutralization process. *Eng. Appl. Artif. Intell.* 19, 905–914.
- Galichet, S., Foulloy, L., 1995. Fuzzy controllers: synthesis and equivalences. *IEEE Trans. Fuzzy Syst.* 3, 140–148.
- Guzelkaya, M., Eksin, I., Gurleyen, F., 2001. A new methodology for designing a fuzzy logic controller and PI, PD blending mechanism. *J. Intell. Fuzzy Syst.* 11, 85–98.
- Guzelkaya, M., Eksin, I., Yesil, E., 2003. Self-tuning of PID-type fuzzy logic controller coefficients via relative rate observer. *Eng. Appl. Artif. Intell.* 16, 227–236.
- Hu, B., Mann, G.K.I., Gasine, R.G., 1999. New methodology for analytical and optimal design of fuzzy PID controllers. *IEEE Trans. Fuzzy Syst.* 7 (5), 521–539.
- Huang, T.T., Chung, H.Y., Lin, J.J., 1999. A fuzzy PID controller being like parameter varying PID. In: IEEE International Fuzzy Systems Conference Proceeding, Seoul, Korea, vol. 1, pp. 269–275.
- Juang, Y.T., Chang, Y.T., Huang, C.P., 2008. Design of fuzzy PID controllers using modified triangular membership functions. *Inform. Sci.* 178, 1325–1333.
- Karasakal, O., Yesil, E., Guzelkaya, M., Eksin, I., 2005. Implementation of a new self-tuning fuzzy PID controller on PLC. *Turk. J. Electr. Eng. Comput. Sci.* 13 (2), 277–286.
- Karasakal, O., Guzelkaya, M., Eksin, I., Yesil, E., 2011. An error-based on-line rule weight adjustment method for fuzzy PID controllers. *Expert Syst. Appl.* 38 (8), 10124–10132.
- Kumbasar, T., Eksin, I., Guzelkaya, M., Yesil, E., 2011a. Adaptive fuzzy model based inverse controller design using BB-BC optimization algorithm. *Expert Syst. Appl.* 38 (10), 12356–12364.
- Kumbasar, T., Eksin, I., Guzelkaya, M., Yesil, E., 2011b. Interval type-2 fuzzy inverse controller design in nonlinear IMC structure. *Eng. Appl. Artif. Intell.* 24 (6), 996–1005.
- Kumbasar, T., Eksin, I., Guzelkaya, M., Yesil, E., 2011c. Type-2 fuzzy model based controller design for neutralization processes. *ISA Trans.*, 10.1016/j.isatra.2011.10.007.
- Li, H.X., Gatland, H.B., 1996. Conventional fuzzy control and its enhancement. *IEEE Trans. Syst. Man Cybern.—Part B* 26 (5), 791–797.
- Mona, S.M., Manju, A., Nigam, M.J., 2011. A study on parameter tuning of weighted fuzzy rule base using genetic algorithm for trajectory control of PUMA560. *Int. J. Eng. Sci. Technol.* 3 (9), 7016–7028.
- Moon, B.S., 1995. Equivalence between fuzzy logic controllers and PI controllers for single input systems. *Fuzzy Sets Syst.* 69, 105–113.
- Mudi, R.K., Pal, N.R., 1999. A robust self-tuning scheme for PI- and PD-type fuzzy controllers. *IEEE Trans. Fuzzy Syst.* 7 (1), 2–16.
- Nauck, D., Kruse, R., 1998. How the learning of rule weights affects the interpretability of fuzzy systems. In: Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZIEEE'98), pp. 1235–1240.
- Oh, S.K., Pedrycz, W., Rho, S.B., Ahn, T.C., 2004. Parameter estimation of fuzzy controller and its application to inverted pendulum. *Eng. Appl. Artif. Intell.* 17, 37–60.
- Oh, S.K., Jang, H.J., Pedrycz, W., 2009. The design of a fuzzy cascade controller for ball and beam system: a study in optimization with the use of parallel genetic algorithms. *Eng. Appl. Artif. Intell.* 22, 261–271.
- Qiao, W.Z., Mizumoto, M., 1996. PID type fuzzy controller and parameters adaptive method. *Fuzzy Sets Syst.* 78, 23–35.
- Saad, N., Arrofiq, M., 2011. A PLC-based modified-fuzzy controller for PWM-driven induction motor drive with constant V/Hz ratio control. *Robot. Comput.-Integr. Manuf.*, doi:10.1016/j.rcim.2011.07.001.
- Sharkawy, A.B., 2010. Genetic fuzzy self-tuning PID controllers for antilock braking systems. *Eng. Appl. Artif. Intell.* 23, 1041–1052.
- Skogestad, S., 2003. Simple analytic rules for model reduction and PID controller tuning. *J. Process Control* 13 (4), 291–309.
- Teng, M., Xiang, F., Wang, R., Wu, Z., 2004. Using genetic algorithm for weighted fuzzy rule-based system. In: Proceedings of the 5th World Congress on Intelligent Control and Automation, pp. 4292–4295.
- Woo, Z.W., Chung, H.Y., Lin, J.J., 2000. A PID-type fuzzy controller with self-tuning scaling factors. *Fuzzy Sets Syst.* 115, 321–326.