Journal of Process Control 19 (2009) 1011–1023

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Online tuning fuzzy PID controller using robust extended Kalman filter

K.K. Ahn [a,*], D.Q. Truong [b]

[a] School of Mechanical and Automotive Engineering, University of Ulsan, San 29, Muger 2-dong, Nam-gu, Ulsan 680-764, Republic of Korea
[b] Graduate School of Mechanical and Automotive Engineering, University of Ulsan, Ulsan, Republic of Korea

ARTICLE INFO

Article history:
Received 14 September 2008
Received in revised form 30 December 2008
Accepted 10 January 2009

Keywords:
Fuzzy
PID controller
Extended Kalman filter
Robust extended Kalman filter (REKF)
Real-time

ABSTRACT

Fuzzy PID controllers have been developed and applied to many fields for over a period of 30 years. However, there is no systematic method to design membership functions (MFs) for inputs and outputs of a fuzzy system. Then optimizing the MFs is considered as a system identification problem for a nonlinear dynamic system which makes control challenges. This paper presents a novel online method using a robust extended Kalman filter to optimize a Mamdani fuzzy PID controller. The robust extended Kalman filter (REKF) is used to adjust the controller parameters automatically during the operation process of any system applying the controller to minimize the control error. The fuzzy PID controller is tuned about the shape of MFs and rules to adapt with the working conditions and the control performance is improved significantly. The proposed method in this research is verified by its application to the force control problem of an electro-hydraulic actuator. Simulations and experimental results show that proposed method is effective for the online optimization of the fuzzy PID controller.

1. Introduction

Nowadays, conventional proportional-integral-derivative (PID) controllers are commonly used in industry due to their simplicity, clear functionality and ease of implementation. Meanwhile, fuzzy control, an intelligent control method imitating the logical thinking of human and being independent on accurate mathematical model of the controlled object, can overcome some shortcomings of the traditional PID. But the fuzzy is a nonlinear control and the output of the controller has the static error [1]. The reason is that when the fuzzy input value is in the stable area defined by a membership function, the fuzzy control output is same which is near or reaches zero command. Then fuzzy PID control which combines the traditional PID control and the fuzzy control algorithm is a solution [2–5]. Fuzzy PID control technique has been applied to many successful applications to a variety of consumer products and industrial systems such as position control of slider crank mechanisms [2], position control of shape memory alloy actuator [3], and speed control for high performance brushless servo drives [4], etc. However, the fuzzy PID controllers proposed in these reaches are experimentally designed based on working conditions of the control systems and their dynamic responses. Consequently, the design of fuzzy rules depends largely on the experience of experts. There is no systematic method to design and examine the number of rules, input space partitions and membership functions [6,7]. Hence, the typical fuzzy PID controllers cannot adapt for a wide range of working environments with large variation of perturbations [13]. As a result, another control technique such as robust control, intelligent theory, or estimation methods is needed to combine with the fuzzy PID to overcome this weakness [8–13]. By using these advanced techniques, the parameters of the fuzzy PID controller will be adjusted about the shape and position of the input/output MFs to minimize the control error. And Kalman estimation technique is an effective solution for controller training purpose.

Kalman filter is a powerful mathematical tool for stochastic estimation from noisy sensor measurements [14–16]. It makes an approximation of the system states, called the priori estimate, which is used to predict the measurement that is about to arrive. It recursively conditions the current estimate on all of the past measurements, and generally converges in a few iterations. A Kalman filter that linearizes about the current mean and covariance is referred to as an extended Kalman filter (EKF) which has been widely applied in many engineering fields and control system designs. However, the conventional Kalman filters are just accurate for problems with small nonlinearities and nearly Gaussian noise statistic. Meanwhile, most physical systems contain large nonlinearities and uncertainties. Moreover, the noise in the measurements is a combination of errors coming from many different sources and generally does not have a Gaussian distribution. Then they can perform very badly due so-called wrong measurements. It is therefore a challenge to find a robust filter, which is able to detect the wrong measurements and to handle them accordingly [17–29]. Finally, the robust filter is applied to optimization purpose of the fuzzy PID controller.

* Corresponding author. Tel.: +82 52 259 2282; fax: +82 52 259 1680.
E-mail addresses: kkahn@ulsan.ac.kr (K.K. Ahn), truongdq@mail.ulsan.ac.kr (D.Q. Truong).

0959-1524/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jprocont.2009.01.005

In order to solve the above control problems, this paper proposes an online tuning fuzzy PID based on a robust extended Kalman filter (REKF) to get better control performance with higher stability. The REKF is a combination of an extended Kalman filter (EKF) and the results in [23–25] which are used to robustify the EKF. Here, a set of the fuzzy PID parameters is similar to a state vector which represents the control ability. Once this vector designed for the system fits completely with the working condition, the performance error becomes zero. In other words, the control system error is caused by the weakness in controller ability and also the environment noise. If the optimization process of the fuzzy PID controller is consider as a filter, then the set of ideal state vectors for the controller is as a process model while the set of state vectors used to control the system is as a measurement model. Consequently, the different between the process and the measurement models, measurement error, is related to the control system error. Therefore, the task of the REKF is to directly estimate the ideal state vector of the fuzzy PID controller for the next step based on the current control error, the current state vector, and previous information. Then the MFs and fuzzy rules are updated online together to minimize the system error function. Consequently, the fuzzy PID inference has higher learning ability and the control qualities are improved significantly even in the case of complicated system and disturbance environment. To verify the overall proposed control system with its advantages, a co-simulation between AMESim [30] and Matlab/Simulink, and also real-time experiments are carried out for a special case like force control of an electro-hydraulic system. Simulation and experimental results show the effectiveness of the hybrid actuator using proposed control method to reach the force control target.

The remainder of this paper is organized as follows: Section 2 is the procedure of designing a robust controller and Section 3 presents the simulation and experimental results. Concluding remarks are presented in Section 4.

2. Robust controller design

2.1. Fuzzy pid controller analysis

In this research, the control problem is considered for systems which have single control input and single output. It is known that, PID controller is the most widely used in modern industry due to its simple control structure and easy design. The control signal for a system using a conventional PID controller can be expressed in the time domain as:

$$u_{\text{PID}}(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \tag{1}$$

where $e(t)$ is the error between desired set point and the system output, $de(t)$ is the derivation of error $e(t)$, $u_{\text{PID}}(t)$ is the control signal for the system and K_p, K_i, and K_d are the proportional gain, integral gain, and the derivative gain, respectively.

But the conventional PID controllers do not yield reasonable performance over a wide range of operating conditions because of the fixed gains used. That is the reason why another control technique needs to be used to tune the parameters of the PID controller. And fuzzy logic is one of the effective solutions.

From (1), three coefficients K_p, K_i and K_d need to be tuned by using fuzzy tuners. Therefore, the detailed fuzzy PID scheme is clearly shown as in Fig. 1.

Through fuzzy logic knowledge, the fuzzy PID tuners which tune PID parameters (K_p,K_i,K_d) can be established by using the following equation:

$$K_a = K_{a0} + U_a \Delta K_a,\ U_a \in [0,1], \quad a \text{ is } p,\ i \text{ or } d \tag{2}$$

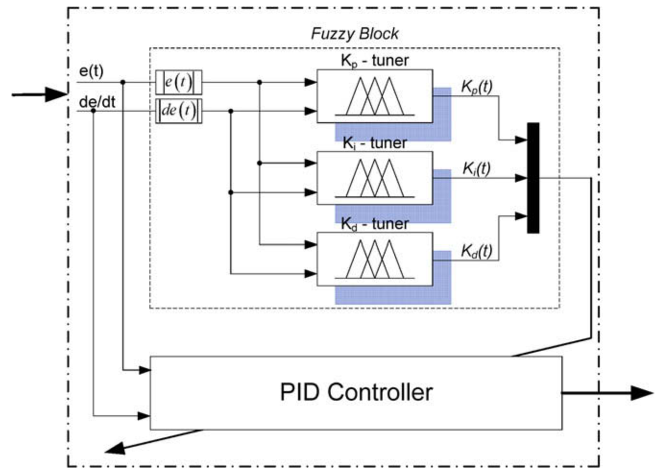


Fig. 1. The configuration of fuzzy PID control block.

where U_a is the parameter obtained from the output of the tuning fuzzy controllers, $\Delta K_a = K_{a1} - K_{a0}$ is the allowable deviation of K_a. K_{a0}, K_{a1} are the minimum and maximum values of K_a determined from experiments, respectively.

From (2) and Fig. 1, three coefficients K_p, K_i and K_d are tuned by using the three independent fuzzy tuners. Consequently, the three separate fuzzy P, I and D controllers are combined to form the overall fuzzy PID controller.

There are two inputs to the fuzzy controllers: absolute error $|e(t)|$ and absolute derivative of error $|de(t)|$. The ranges of these inputs are from 0 to 1, which are obtained from the absolute values of system error and its derivative through scale factors chosen from the specification of the nonlinear system. For each input variables, triangle membership functions (MFs) are requested to use. Because all of the MFs are triangle shapes, so we can express these MFs as follows:

$$f_{ji}(x) = \begin{cases} 1 + \frac{(x-a_{ji})}{b_{ji}^-} & \text{if } (-b_{ji}^-) \le (x - a_{ji}) \le 0 \\ 1 - \frac{(x-a_{ji})}{b_{ji}^+} & \text{if } 0 \le (x - a_{ji}) \le (b_{ji}^+), \quad j = 1,2,\ldots,N \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where x is the input; the a_{ji}, b_{ji}^- and b_{ji}^+ are the centroid, left half-width, and right half-width of the jth triangle membership function of the ith input, respectively. N is the numbers of triangles.

Each of the fuzzy P, I and D controllers has one output which is U_p, U_i and U_d, respectively. In practice, fuzzy control is applied using local inferences. That means each rule is inferred and the results of the inferences of individual rules are then aggregated. The most common inference methods are: the max-min method, the max-product method and the sum-product method, where the aggregation operator is denoted by either max or sum, and the fuzzy implication operator is denoted by either min or prod. Especially the max–min calculus of fuzzy relations offers a computationally nice and expressive setting for constraint propagation. Finally, a defuzzification method is needed to obtain a crisp output from the aggregated fuzzy result. Popular defuzzification methods include maximum matching and centroid defuzzification. The centroid defuzzification is widely used for fuzzy control problems where a crisp output is needed, and maximum matching is often used for pattern matching problems where we need to know the output class. Hence in this study, the fuzzy reasoning results of outputs are gained by aggregation operation of fuzzy sets of inputs and designed fuzzy rules, where max-min aggregation method and centroid defuzzification method are used. In the proposed fuzzy controller, we can compute the control output U_p, U_i or U_d with a pair of inputs:

$$U_a = \frac{\sum_{k=1}^{M} mf(w_k) w_k}{\sum_{k=1}^{M} mf(w_k)}, \quad (a \text{ is } p,\ i \text{ or } d) \tag{4}$$

where w_k is the weight of the control output (the centroid of the kth output fuzzy MF), M is the number of fuzzy output sets and $mf(w_k)$ is the fuzzy output function given by

$$mf(w) = \sum_{i,j} mf_{ij}(w) \tag{5}$$

where $mf_{ij}(w)$ is defined as the consequent fuzzy output function when the first and the second input are in the i and the j class, respectively

$$mf_{ij}(w) = \delta_{ij}\mu_{ij} \tag{6}$$

where δ_{ij} is a activated factor, which is active when the input $|e(t)|$ is in class i, and the input $|de(t)|$ is in class j and μ_{ij} is the height of the consequent fuzzy function obtained from the input class i and j

$$\mu_{ij} = \min[f_{i1}(|e(t)|),\ f_{j2}(|de(t)|)] \tag{7}$$

The output U_a of the tuning fuzzy controller contains single output values. They are initially set at the same intervals.

Generally, the fuzzy rules are dependent on the plant to be controlled and the type of the controller. These rules are determined from the intuition or practical experience. However, there is no systematic method to design and examine the number of rules, input space partitions and MFs. Furthermore, the fuzzy system lacks of the learning ability and adaptive capability, especially in case that the controlled object contains nonlinearities, large uncertainties and noised environment.

Therefore, an adaptive technique is needed to be combined with the traditional fuzzy PID controller to perform a new robust controller. As a result, an online tuning fuzzy PID controller based a robust extended Kalman filter is one solution to achieve the better control performance. Consider an error function given by

$$E = \frac{1}{2}(y - y_r)^2 \tag{8}$$

where y_r and y are the reference input for control task of the system (or named the target value) and the system output, respectively.

The control purpose is how to minimize the error function with respect to the fuzzy MF parameters. As mention above, the fuzzy PID controller is optimized by using Kalman estimation technique. The idea of the proposed controller is using a robust extended Kalman filter to tune the input MFs shape and the weight of the controller outputs during the system operation process. Then, the decisive factors of the fuzzy input MFs a_j, b_j^-, b_j^+ and the weights of the output w_j are automatically updated by using REKF. This novel method is described in details as following sections.

2.2. Extended Kalman filter (EKF)

This section presents the EKF algorithm. The state of the system at time t_k (k=1,2,...) is modeled as a stochastic variable x_k. The evolution of the state in time is expressed by a stochastic different equation

$$\begin{aligned} x_{k+1} &= g_k(x_k, q_k) \\ q_k &\sim N(0, Q_k) \end{aligned} \tag{9}$$

and the measurement vector y_k are related to the system state by\

$$\begin{aligned} y_{k+1} &= h_k(x_k, \upsilon_k) \\ \upsilon_k &\sim N(0, R_k) \end{aligned} \tag{10}$$

where $g(\cdot)$ and $h(\cdot)$ are nonlinear vector functions of the state; q_k and υ_k represent the process and measurement noise, respectively. Q_k is the process noise covariance, and R_k is the measurement noise covariance.

Assume that the initial state x_0 and the noises q_k and υ_k are white zero-mean, Gaussian and independent from each other with

$$\begin{cases} E(x_0) = \bar{x}_0 \\ E[(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T] = P_0 \\ E(q_i) = 0 \\ E(q_i q_j^T) = Q\delta_{ij} \\ E(\upsilon_i) = 0 \\ E(\upsilon_i \upsilon_j^T) = R\delta_{ij} \end{cases} \tag{11}$$

where $E(\cdot)$ is expectation operator; δ_{ij} is the Kronecker delta given by

$$\delta_{ij} = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases} \tag{12}$$

The EKF is an approximate analytic solution if consider that the noises q_k and υ_k are additive Gaussian noise sequences. The state and the measurement functions are linearized according to

$$G_k = \frac{\partial g_k(x_k)}{\partial x_k}\Big|_{x_k = \hat{x}_k^-}; \quad H_k = \frac{\partial h_k(x_k)}{\partial x_k}\Big|_{x_k = \hat{x}_k^-} \tag{13}$$

where consider $\hat{x}_k^-$ and $\hat{x}_k$ are the prior and posterior mean estimates at time step t_k. Then, the system model can be approximated as

$$\begin{aligned} x_{k+1} &= G_k x_k + q_k \\ y_k &= H_k x_k + \upsilon_k \end{aligned} \tag{14}$$

The prior density is now Gaussian with mean and covariance as followings

$$\begin{aligned} E(x_k|y_1 \ldots y_{k-1}) &= \hat{x}_k^- = G_k \hat{x}_{k-1} \\ V(x_k|y_1 \ldots y_{k-1}) &= P_k^- = G_k P_{k-1} G_k^T + Q_k \end{aligned} \tag{15}$$

The posterior density is now Gaussian with mean and covariance computed as

$$\begin{aligned} E(x_k|y_1 \ldots y_k) &= \hat{x}_k = \hat{x}_k^- + K_k(z_k - H_k \hat{x}_k^-) \\ V(x_k|y_1 \ldots y_k) &= P_k = (I - K_k H_k) P_k^- \end{aligned} \tag{16}$$

where K_k is the Kalman gain matrix

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{17}$$

2.3. Robust extended Kalman filter (REKF)

In Section 2.2, the EKF algorithm is based on the assumption that the process noise and measurement noise are nearly Gaussian distribution. As a result, in most physical systems, and in the working environment, the large number of uncertainties and nonlinearities can cause the controller based on the EKF with the strict assumption of Gaussian noises to lose in efficiency. To solve the above problem, an asymptotically min-max robust estimation technique is combined with the EKF to perform a robust extended Kalman filter.

For a simple problem of estimating location, appropriate choices for the nonlinear transformation and gain constant of the algorithm lead to an asymptotically min–max robust estimator with respect to a family F(y_p, p) of symmetrical distributions having the same mass p outside $[-y_p, y_p]$, $0 < p < 1$. From [24], one type of the recursive stochastic approximation (SA), referred to as the p-point estimator (PPE), is proposed. This estimator has the additional striking property that the asymptotic variance is constant over the family F(y_p,p). Consequently, the PPE estimator in [24] was applied to robustify the extended Kalman filter [23]. Based on the PPE estimator [24] and the REKF with the robust Bayesian estimation [23], this paper proposed the use of the REKF to optimize online the characteristic parameters of the fuzzy PID controller with respect to the varied working environment of the control system. The followings present the REKF in details.

2.3.1. p-Point estimator

Consider a family of distribution F

$$\mathscr{F}(\gamma_p, p) = \left\{ F(.) \middle| \int_{-\infty}^{-y_p} F(\theta)d\theta \equiv F(-\gamma_p) = p/2 = \Phi(-\gamma_p),\ \gamma_p > 0, \right.$$
$$\left. 0 < p < 1,\ F(.) \text{ symmetric \& continous at } \pm\ \gamma_p \right\} \tag{18}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function.

Let $\mathscr{T}$ be the family of all regular translation invariant estimators T. Denote the asymptotic variance of the estimator when the sample size tends to infinitive by $V(T,F), T \in \mathscr{T}, F \in \mathscr{F}(\gamma_p,p)$.Therefore, there exits a min-max solution $(T_0,\ F_0)$ $(F_0 \in \mathscr{F}(\gamma_p,p)$ and $T_0 \in \mathscr{T})$ such that

$$\sup_F V(T_0,F) = V(T_0,F_0) = \inf_T V(T,F_0) \tag{19}$$

where the supremum is over all $F \in F(\gamma_p,p)$ and the infimum is over all $T \in$ T

F_0: The least favorable distribution of class F.
T_0: The min–max robust estimator which is actually the maximum likelihood estimator for the least favorable density F_0.

The least favorable density F_0 is explicitly given in term of its density function by

$$f_0(\gamma) = \begin{cases} K\cos^2\left(\frac{\gamma}{2s_m\gamma_p}\right), & |\gamma| \le \gamma_p \\ K\cos^2\left(\frac{1}{2s_m}\right)\exp[-2Kp^{-1}\cos^2\left(\frac{1}{2s_m}\right)(|\gamma|-\gamma_p), & |\gamma| > \gamma_p \end{cases} \tag{20}$$

where K is defined by

$$K = \frac{1-p}{\gamma_p\left[1 + s_m\sin\left(\frac{1}{s_m}\right)\right]} \tag{21}$$

For each p there exists an $s = s_m$ that minimizes the asymptotic variance and s_m does not depend upon y_p. The minimizing value s_m satisfies the following equation

$$2s_m - p\left[1 + \tan^2\left(\frac{1}{2s_m}\right)\right]\left[2s_m + \tan\left(\frac{1}{2s_m}\right)\right] = 0 \tag{22}$$

From (20)–(22), the likelihood score of the least favorable density F_0 of the PPE F_p is computed as

$$s_M(\gamma) = \frac{f_0'(\gamma)}{f_0(\gamma)} = \begin{cases} -\frac{1}{s_m\gamma_p}\tan\left(\frac{\gamma}{2s_m\gamma_p}\right), & |\gamma| \le \gamma_p \\ -\frac{1}{s_m\gamma_p}\tan\left(\frac{1}{2s_m}\right)sign(\gamma_p), |\gamma| > \gamma_p \end{cases} \tag{23}$$

2.3.2. Robustifying Kalman filter

The PPE was applied to the EKF algorithm to perform the robust extended Kalman filter [23]. From the normal vector observation model in (14), consider an innovation transformed version of the measurement model in (14)

$$\begin{cases} z_k = H_k^* x_k + v_k^* \\ \gamma_k = z_k - H_k^* \hat{x}_k^- \end{cases} \tag{24}$$

with

$$\begin{cases} z_k = T_k y_k \\ H_k^* = T_k H_k \\ v_k^* = T_k v_k \end{cases} \tag{25}$$

where T_k is a transformation matrix which is determined below.

From (16) and (17), the posterior can be re-written as

$$E(x_k|y_1\ldots y_k) = \hat{x}_k = \hat{x}_k^- + P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}(z_k - H_k\hat{x}_k^-) \tag{26}$$

The transformation matrix is chosen as

$$T_k = (H_k P_k^- H_k^T + R_k)^{-1/2}$$
$$(T_k^T T_k = (H_k P_k^- H_k^T + R_k)^{-1}) \tag{27}$$

Replace the transformation matrix (27) into (26), we obtain

$$E(x_k|y_1\ldots y_k) = \hat{x}_k = \hat{x}_k^- + P_k^- H_k^T T_k^T T_k (z_k - H_k\hat{x}_k^-) \tag{28}$$

In addition, because the transformation matrix is a standard normal distribution, apply the PPE (Section 2.3.1) into the EKF. If $p(\gamma_k|\gamma_1\ldots\gamma_{k-1})$ is the least favorable density of the PPE $\mathscr{F}_p$ then the posterior mean and posterior covariance in (16) are now modified as following equations. First, the posterior mean in (28) is computed

$$E(x_k|y_1\ldots y_k) = \hat{x}_k = \hat{x}_k^- + P_k^- H_k^T T_k^T \psi(\gamma_k) \tag{29}$$

where $\psi(\gamma_k)$ is the odd symmetric scalar influence function corresponding to the min–max estimate for distribution function F, and $\psi(\gamma_k)$ is defined as

$$\psi(\gamma_k) = -\nabla \ln p(\gamma_k|\gamma_1..\gamma_{k-1})|_{\gamma_k} = \begin{cases} \frac{1}{s_m\gamma_p}\tan\left(\frac{\gamma}{2s_m\gamma_p}\right)|\gamma| \le \gamma_p \\ \frac{1}{s_m\gamma_p}\tan\left(\frac{1}{2s_m}\right)sign(\gamma_p), |\gamma| > \gamma_p \end{cases} \tag{30}$$

where the factor $\gamma_k = T_k(z_k - H_k\hat{x}_k^-)$

Next, the posterior covariance is given by

$$\begin{aligned} V(x_k|y_1\ldots y_k) = P_k &= [I - (P_k^- H_k^T T_k^T T_k)H_k E_{F_0}(\psi'(\gamma_k))]P_k^- \\ &= [I - K_k H_k E_{F_0}(\psi'(\gamma_k))]P_k^- \end{aligned} \tag{31}$$

where the factor $E_{F_0}(\psi'(\gamma_k))$ is defined by PPE

$$E_{F_0}(\psi'(\gamma_k)) = (s_m\gamma_p)^{-2}\left[1 - p\left(1 + \tan^2\left(\frac{1}{2s_m}\right)\right)\right] \tag{32}$$

2.4. Application of the robust extended Kalman filter to optimize the fuzzy PID controller

In this section, the REKF presented in the previous section is used for training the MF parameters of fuzzy inputs a_j, b_j^-, b_j^+ and the weights of the output w_j. The overall structure of the online tuning fuzzy PID controller based a robust extended Kalman filter is shown in Fig. 2.

The optimization of fuzzy controller is known as a weighted least square minimization problem, where the error is the difference between the system output and the target value for that output. The fuzzy PID controller for a nonlinear system contains the three separate fuzzy P, I and D as presented in Section 2.1. Each of the fuzzy controller (P or I or D) have the two inputs ($|e(t)|$, and$|de(t)|$) and one output (k_p or k_i or k_d); and the REKF will tune online each of the fuzzy controller separately. For the training purpose, the state vector of the nonlinear system can be represented as

$$x = [x_P \quad x_I \quad x_D]^T \tag{33}$$

where each of element of the state vector x is a characteristic vector as given

$$\begin{aligned} &x_{(P/I/D)k} \\ &= [a_{11}\ b_{11}^-\ b_{11}^+\ \ldots\ a_{n1}\ b_{n1}^-\ b_{n1}^+\ a_{12}\ b_{12}^-\ b_{12}^+\ \ldots\ a_{m2}\ b_{m2}^-\ b_{m2}^+\ w_1\ \ldots\ w_v]_k \end{aligned} \tag{34}$$

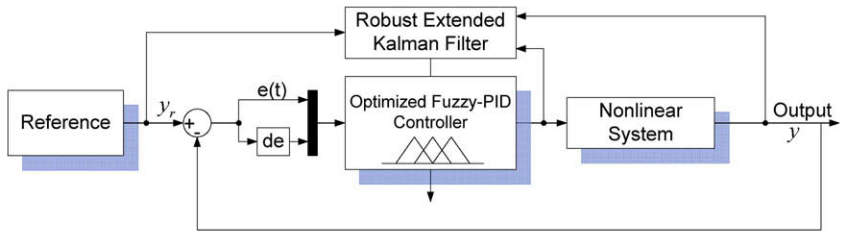


Fig. 2. Structure of the proposed control mechanism applied to a nonlinear system.

where a_{ji}, b^-_{ji} and b^+_{ji} are the centroid, left half-width, and right half-width of the jth triangle MF of the ith input (which has n fuzzy sets if it is the first input or has m fuzzy sets if it is the second input); w_k is the weight of the control output (the centroid of the kth output fuzzy MF, number of weights is υ).

Next, z is used to denote the vector of real system output, and h is used to denote the target vector for the system output. The nonlinear system model to which the REKF can be applied is expressed

$$\begin{aligned} x_{k+1} &= x_k + q_k \\ y_k &= h(x_k) + \upsilon_k \end{aligned} \tag{35}$$

From (35), $h(x_k)$ is the mapping fuzzy PID controller function which represents the system output with respect to a set of the fuzzy parameters. In addition, the process noises, q_k and υ_k, are added into the system model to avoid numerical divergence of the algorithm and poor local minima problems.

The initial estimation vector at $t = 0$ is $\hat{x}^-_0$. Furthermore, because of this prior knowledge about the process, the error associated with our initial estimate is zero. Thus,

$$\hat{P}^-_0 = 0 \tag{36}$$

By using the REKF presented in Sections 2.2 and 2.3 in which Q_k, the tuning fuzzy parameter matrix, is a diagonal covariance matrix

$$Q_k = \text{cov}\begin{bmatrix} q_p & q_I & q_D \end{bmatrix}^T_{0..k} \tag{37}$$

where each of element of the state vector Q_k is a characteristic vector as given

$$q_{(P/I/D)0..k} = \begin{bmatrix} q_{11} & q^-_{11} & q^+_{11} & \cdots & q^+_{n1} & q_{12} & \cdots & q_{m2} & q^-_{m2} & q^+_{m2} & q_1 & \cdots & q_\upsilon \end{bmatrix}_{0..k} \tag{38}$$

The "robust extended Kalman filter" block in Fig. 2 for tuning the fuzzy PID parameters can be expressed as in Fig. 3. For this case, the measurement functions in (13) can be computed by using partial derivatives of the system output with respect to each parameter of fuzzy PID controller: a, b^-, b^+, and w_k and then replace them into equations of the REKF (Section 2.3) to update the controller. For the step k of time

$$H_k = \left.\frac{\partial h_k(x_k)}{\partial x_k}\right|_{x_k = \hat{x}^-_k}$$

Firstly, the partial derivative of the system output with respect to the weight of each fuzzy out MF is given

$$\frac{\partial h_k(w_i)}{\partial w_i} \equiv \frac{\partial y}{\partial w_i} = \frac{\partial y}{\partial u_{PID}} \frac{\partial u_{PID}}{\partial U_a} \frac{\partial U_a}{\partial w_i} \tag{39}$$

where

$$\frac{\partial y}{\partial u_{PID}} = \frac{\Delta y}{\Delta u_{PID}} = \frac{y(t) - y(t-1)}{u_{PID}(t) - u_{PID}(t-1)} \tag{40}$$

$$\frac{\partial u_{PID}}{\partial U_a} : \begin{cases} \frac{\partial u_{PID}}{\partial U_p} = \Delta K_p e(t) \\ \frac{\partial u_{PID}}{\partial U_i} = \Delta K_i \int e(t)dt \\ \frac{\partial u_{PID}}{\partial U_d} = \Delta K_d de(t) \end{cases} \tag{41}$$

$$\frac{\partial U_a}{\partial w_i} = \frac{mf(w_i)}{\sum_{k=1}^{M} mf(w_k)} \tag{42}$$

Secondly, the partial derivative of the system output with respect to the center of each input MF can be computed by

$$\frac{\partial h_k(a_i)}{\partial a_i} \equiv \frac{\partial y}{\partial a_i} = \frac{\partial y}{\partial U_a} \frac{\partial U_a}{\partial \mu_i} \frac{\partial \mu_i}{\partial a_i} = \frac{\partial y}{\partial u_{PID}} \frac{\partial u_{PID}}{\partial U_a} \frac{\partial U_a}{\partial mf(w_i)} \frac{\partial mf(w_i)}{\partial a_i} \tag{43}$$

where $\frac{\partial y}{\partial u_{PID}}$, and $\frac{\partial u_{PID}}{\partial U_a}$ is calculated by using (40) and (41), respectively.

$$\frac{\partial U_a}{\partial mf(w_i)} = \frac{\sum_{k=1}^{M} mf(w_k)(w_i - w_k)}{\left(\sum_{k=1}^{M} mf(w_k)\right)^2} \tag{44}$$

$$\frac{\partial mf(w_i)}{\partial a_i} = \begin{cases} \frac{-1}{b^-_i} & \text{if } (-b^-_i) \le (x - a_i) \le 0 \\ \frac{1}{b^+_i} & \text{if } 0 \le (x - a_i) \le (b^+_i) \\ 0 & \text{otherwise} \end{cases} \tag{45}$$

Thirdly, the partial derivative of the system output with respect to the half widths of each input MF can be computed by

$$\frac{\partial h_k(b^{+/-}_i)}{\partial b^{+/-}_i} \equiv \frac{\partial y}{\partial b^{+/-}_i} = \frac{\partial y}{\partial U_a} \frac{\partial U_a}{\partial mf(w_i)} \frac{\partial mf(w_i)}{\partial b^{+/-}_i} \tag{46}$$

where $\frac{\partial y}{\partial U_a}$ and $\frac{\partial U_a}{\partial mf(w_i)}$ is calculated by using 40, 41, and 44.

$$\frac{\partial mf(w_i)}{\partial b^{+/-}_i} = \begin{cases} -\frac{(x-a_i)}{(b^-_i)^2} & \text{if } (-b^-_i) \le (x - a_i) \le 0 \\ \frac{(x-a_i)}{(b^+_i)^2} & \text{if } 0 \le (x - a_i) \le (b^+_i) \\ 0 & \text{otherwise} \end{cases} \tag{47}$$

After get the measurement functions in (13), the fuzzy PID controller is automatically updated by applying the robust estimator as in Fig. 3.

However, for each separate fuzzy controller (P, or I, or D) with the same two inputs and single output, the more membership functions and rules are, the larger the number components of REKF state vector (33) is. Then the REKF is very complex because of many big size matrixes (such as P, Q) performed from the large state vector x. It causes the calculation time for updating the fuzzy PID parameters to increase so much. This is a big problem for applying the REKF to tune online the fuzzy PID controller.

To solve this problem, the number components of the Kalman state vector needs to be reduced. Firstly, the operation of the fuzzy controller is considered. Each of the fuzzy input/output variables is represented by the MFs which make the overlap input partitions. Hence, for each of the fuzzy input variables ($|e(t)|$ or $|de(t)|$), any value of it will drop commonly into the input partition spaces of MFs of which maximum are two MFs. It is called two active MFs. In addition, one fuzzy rule is designed for a couple of two input MFs which respect to two values of the two input variables. Therefore, for a couple values of the two input variables ($|e(t)|$ and $|de(t)|$), the maximum number of active input MFs as same as active rules, which are used to calculate the output, is only four while another MFs and another rules are not used. Then, the REKF calculations will be used to update the parameters for only four active MFs and four active rules with respect to the system error function. Moreover, each membership function has two parameters,

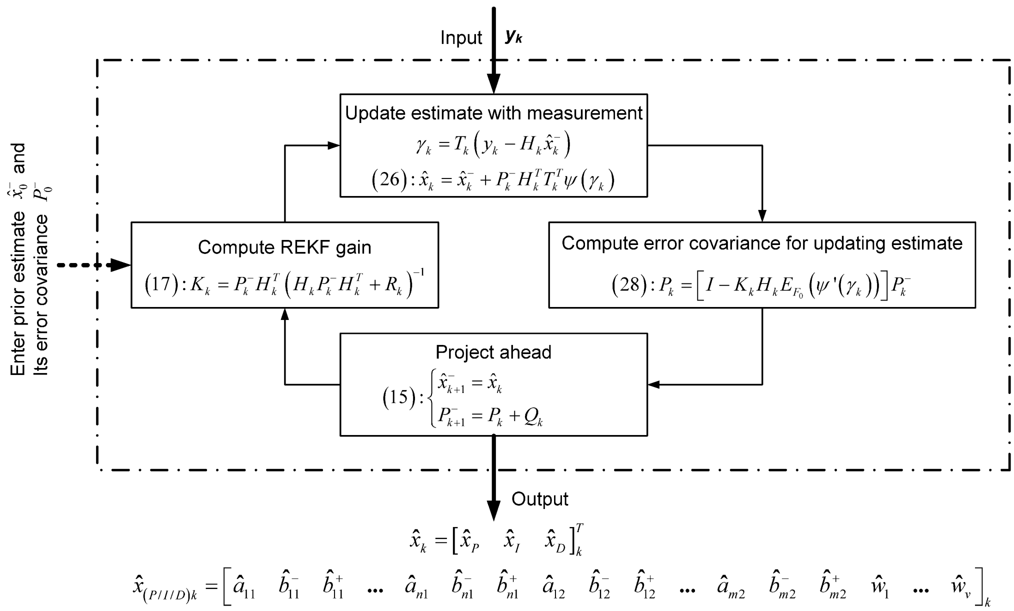


Fig. 3. Internal components of "robust extended Kalman filter" block.

$a, b^-(\text{or}\,b^+)$, and each rule have one parameter, w_k, need to be tuned. Consequently, the size of the representative state vector for each separate fuzzy P, I, or D controller is now shortened

$$\begin{aligned}\text{A state vector size} &= 4 \times (3 \text{ tuned fuzzy parameters}) \\ &= 12(\text{elements})\end{aligned}$$

From the above analysis, the REKF now can be re-written by a reduced form in which the state vector (33) and (34) is only a vector of the all active MF parameters. In other words, the size of state vector for each separate P, I, or D controller is only twelve elements for updating four active MFs. Consequently, the state vector of non-linear system is given as

$$x^* = [x_P^* \quad x_I^* \quad x_D^*]^T \tag{48}$$

where each of element of the state vector x^* is a characteristic vector of the four active MFs (two of the first input, and two of the second input), and four active rules (composed from the active MFs) as given

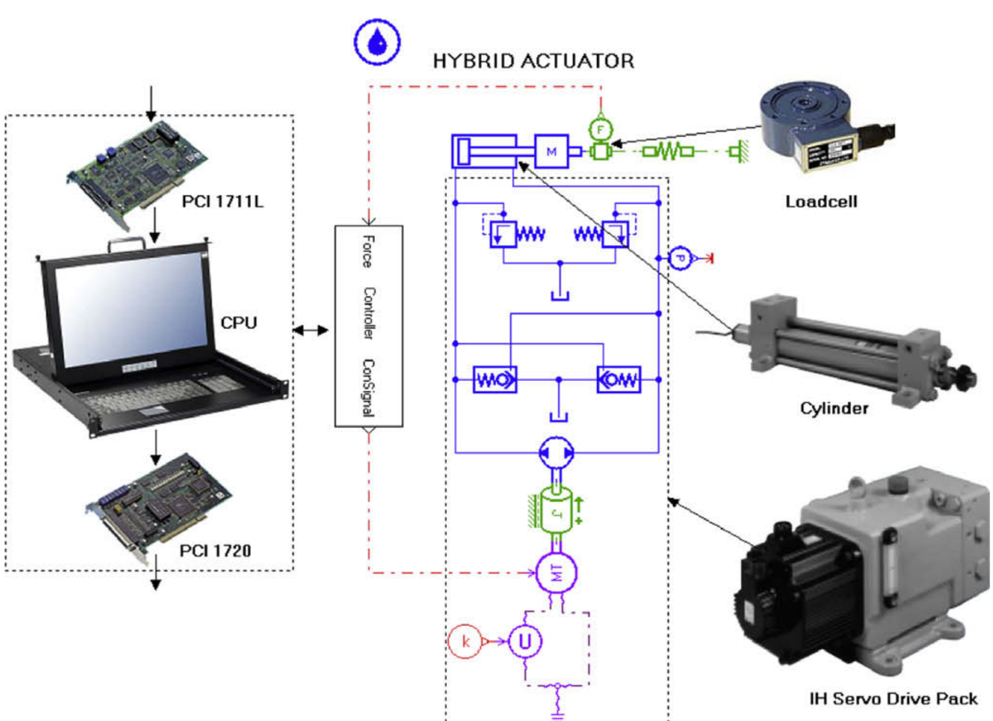
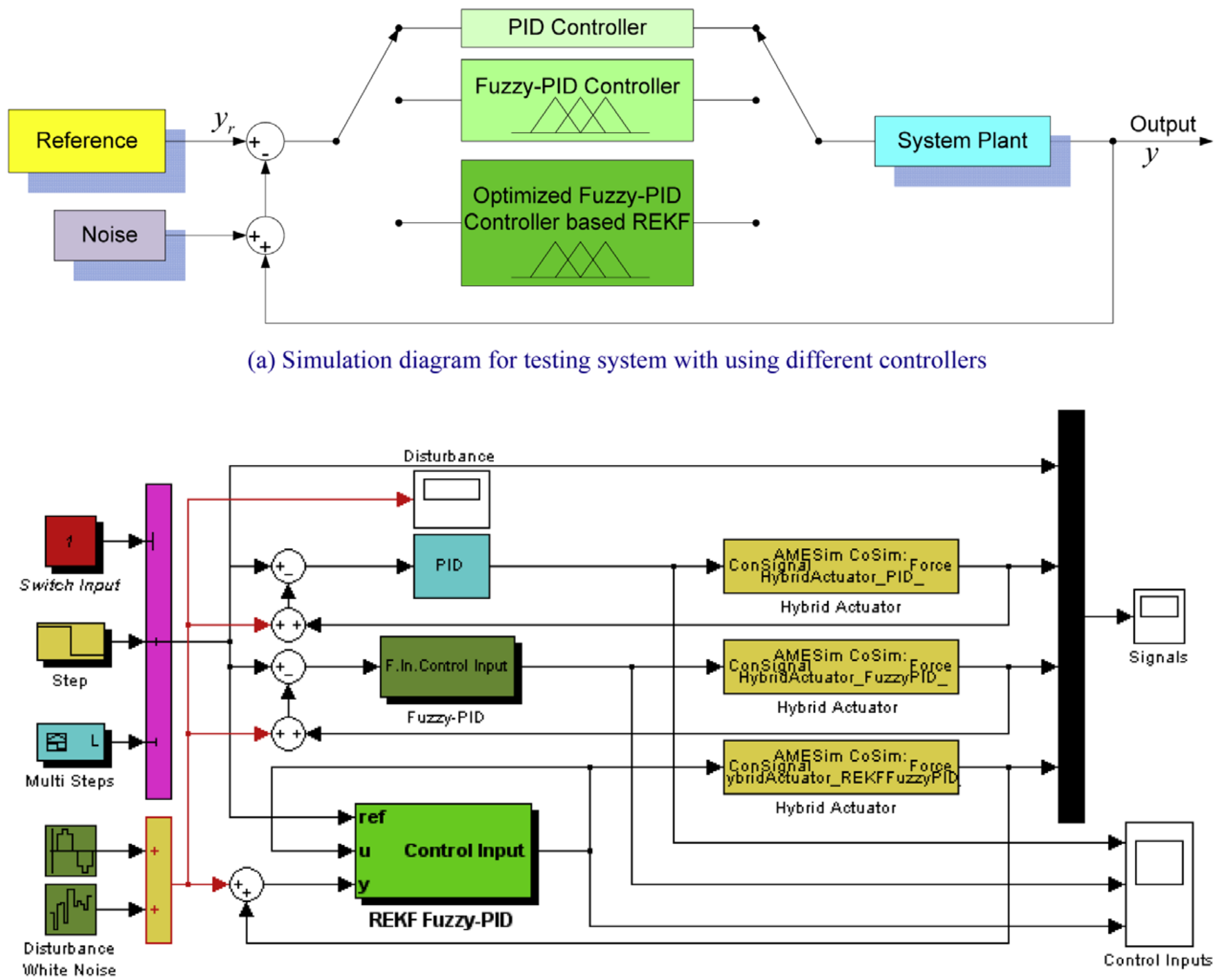


Fig. 4. AMESim model of the hybrid actuator.

$$x^*_{(P/I/D)k} = \begin{bmatrix} a^*_{11} & b^{*-}_{11}(b^{*+}_{11}) & a^*_{21} & b^{*-}_{21}(b^{*+}_{21}) & a^*_{12} & b^{*-}_{12}(b^{*+}_{12}) \\ a^*_{22} & b^{*-}_{22}(b^{*+}_{22}) & w^*_1 & w^*_2 & w^*_3 & w^*_4 \end{bmatrix}_k \tag{49}$$

Next, all of the calculations from (35)–(47) are derived into the reduced forms based on (48) and (49) and then are used to optimize the controller. This way saves much time for calculation and control process and then increases the ability of the controller. With the self tuning ability based on the REKF and then the optimization the error, the fuzzy PID controller works more and more effectively. The simulations and experimental results in the next section will prove for this consideration.



(a) Simulation diagram for testing system with using different controllers

(b) Simulation diagram built in Simulink/Matlab to connect with AMESim

Fig. 5. Simulation model of hybrid systems with controllers.

Table 1
Setting parameters for AMESim – Electro-hydraulic actuator model.

System parameters	Value	Meaning
AC servo motor	200	Power supply(volts AC)
	2.9	Power (kW)
	18.6	Rate torque (Nm)
Pump	2500	Speed (rev/min)
	15	Pump displacement (cc/rev)
Cylinder parameters	$63 \times 35 \times 150$	Piston diameter × Rod diameter × Length of stroke (mm)
	21	Maximum pressure (MPa)
Spring – k	519	Environment stiffness (kN/m)
Relief pressure (bar)	175	Relief valve cracking pressure
Working liquid – Oil	0.87	Specific gravity
	1.5×10^9	Bulk modulus (Pa)
Load – *M*	1000	Loading environment
Sensor gain	3	Force sensor signal

3. Simulations and experiments

3.1. Simulations

3.1.1. Simulation setup

In this section, simulations were carried out to prove the effectiveness of the designed controller. A co-simulation between AMESim [30] and Simulink is chosen to verify the proposed controller applied to a hydraulic circuit of an hybrid electro-hydraulic actuator with force control requirement.

The hybrid actuator is a combination of an AC servo motor, a piston pump, a reservoir and a hydraulic control circuit. The cylinder is controlled by the motor and the bidirectional pump to get the desired performance. In addition, a compression spring is used as a loading environment and a load cell is used for obtaining the feedback force signal. The speed of the servo motor is the control target in this case.

The hydraulic circuit was built in the simulation AMESim software, version 4.3 (Imagine S.A., 2005). AMESim corresponds to Advanced Modeling Environment Simulation software, which allows the simulation of actuator dynamics including electrical motor, hydraulic systems through using several libraries. Fig. 4 shows the AMESim model of the hybrid actuator. AMESim generates C-files for the actuator model and creates a DLL file for the model. The DLL is then used in the simulation model via Simulink by associating with a S-Function block. The AMESim model contains one input (control signal to control the motor) and one output (feedback force signal from the load cell) to communicate with the suggested control system built in Simulink, consequently form a closed-loop feedback control.

Next, Matlab/Simulink is chosen as a common shell for building the simulation model due to its ability to support and interface seamlessly with the different DLLs provided from other tools. The DLL can be included in the Simulink environment in the form of an S-Function. Fig. 5 shows the system with the controller built in Simulink.

3.1.2. Simulation results

In this section, by using the above developed co-simulation platform, the states of the hydraulic system solved in AMESim are fed into the Simulink controller. The control signals from the controller are then fed back into the AMESim hydraulic model and the new states are solved to perform the force control performance. The setting parameters for the hybrid system model are obtained from the real components of test rig as shown in Table 1. The simulations were done with a 0.01 s sampling rate, to check the system responses.

In addition, to prove the effectiveness of the proposed controller, a disturbance scheme was included in the control diagram as shown in Fig. 5. The disturbance generated in this case, a combination of a white noise and a sine wave noise, can be expressed as given

$$Dis(t) = A\sin(\omega t) + Rnd(t) \quad (50)$$

Table 2
Rules table of online self tuning fuzzy PID controller.

(U_p, U_i, U_d)		$\|de(t)\|$				
		Z	VS	S	M	B
$\|e(t)\|$	Z	(VS,B,M)	(VS,B,M)	(Z,B,M)	(Z,B,B)	(Z,B,B)
	VS	(VS,B,S)	(VS,B,M)	(VS,B,M)	(Z,M,M)	(Z,M,B)
	S	(S,M,VS)	(S,M,VS)	(S,M,VS)	(VS,S,S)	(VS,S,S)
	M	(M,Z,Z)	(M,Z,Z)	(M,VS,VS)	(S,VS,VS)	(S,VS,VS)
	B	(B,Z,Z)	(B,Z,Z)	(B,Z,Z)	(B,Z,Z)	(M,Z,Z)

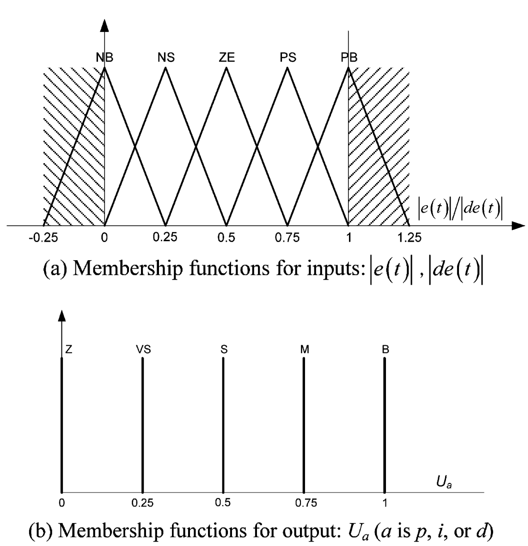


Fig. 6. Initial membership functions of the inputs and output of the self tuning P, I, or D fuzzy.

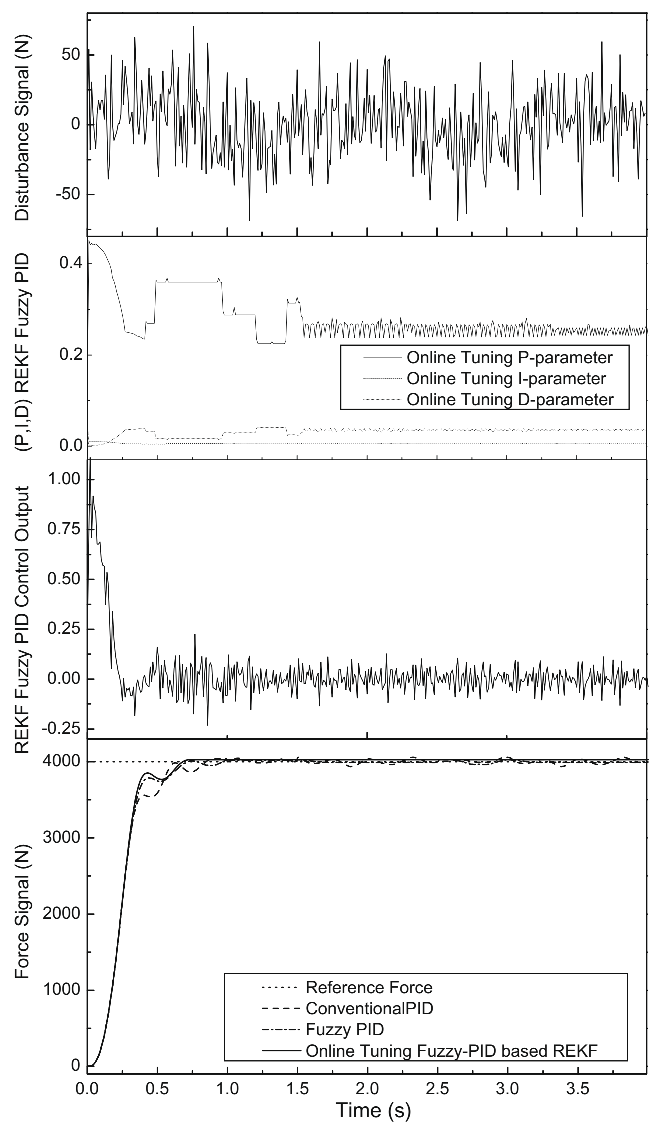
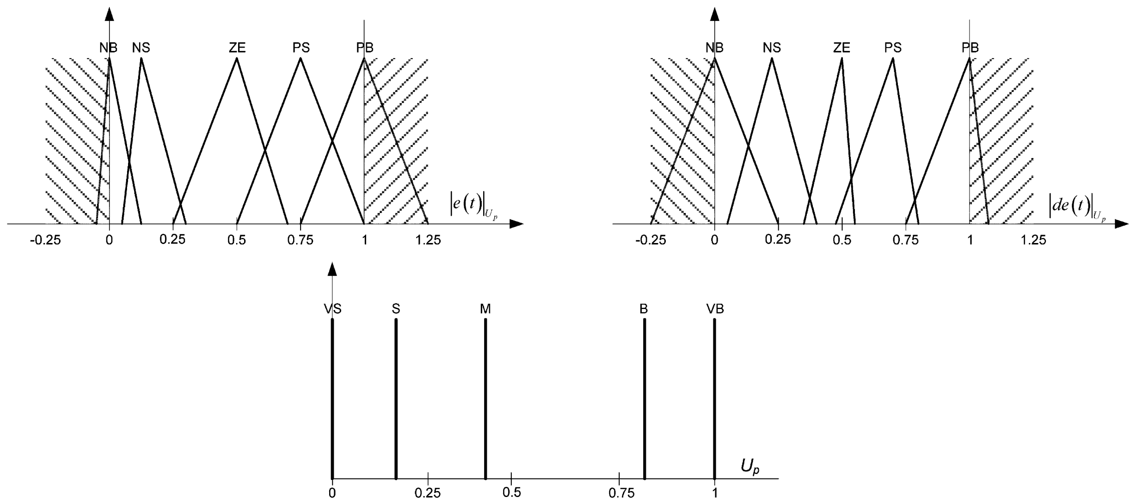


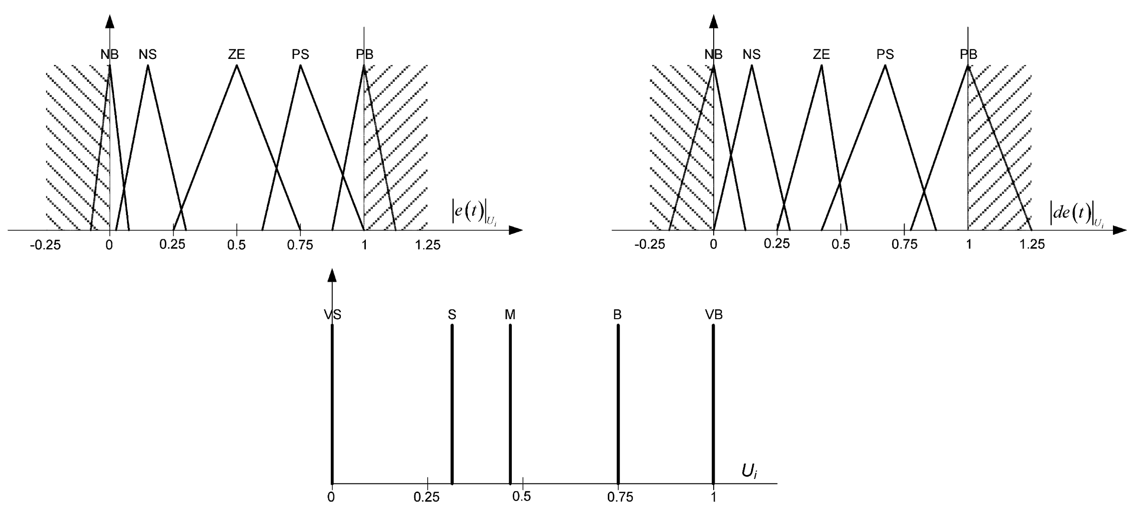
Fig. 7. Simulation results – Comparison of system responses between using different controllers with respect to a step force reference.

where A and ω are amplitude and frequency parameters, respectively, and $Rnd(t)$ is the white noise signal.

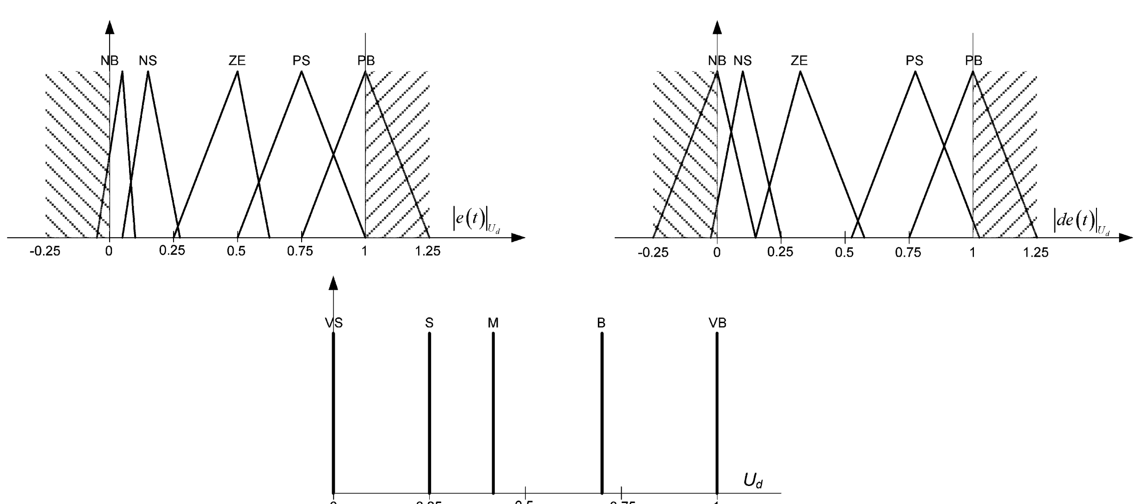
The comparison of a conventional PID controller, a common fuzzy PID controller, and the proposed online tuning fuzzy PID



(a) MFs of the fuzzy P inputs and output after tuning



(b) MFs of the fuzzy I inputs and output after tuning



(c) MFs of the fuzzy D inputs and output after tuning

Fig. 8. Simulation results – Fuzzy PID controller based the REKF after tuning.

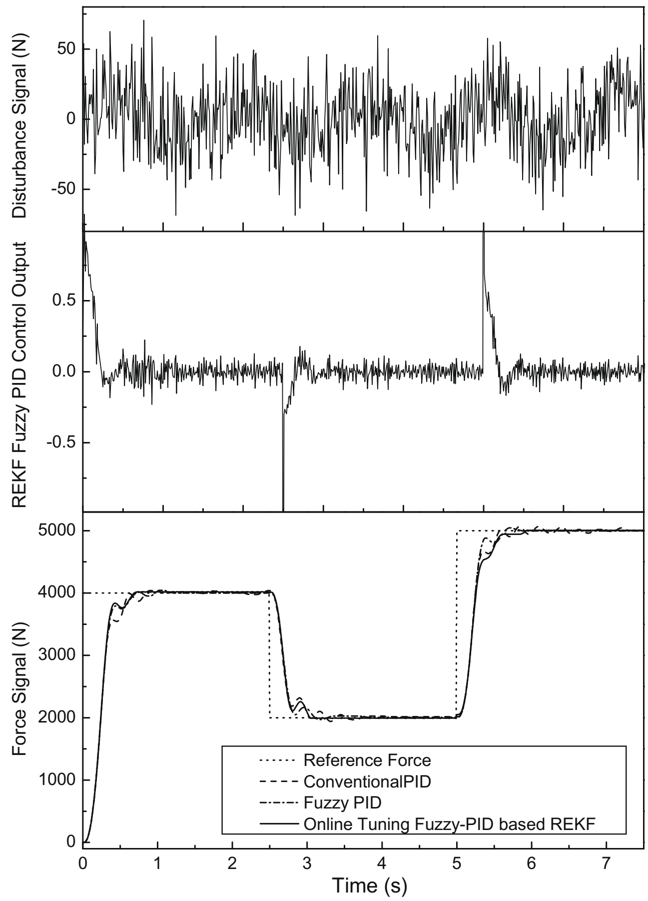


Fig. 9. Simulation results – Comparison of system responses between using different controllers with respect to a multi step force reference.

controller based the REKF applied to the hydraulic system model were performed as shown in Fig. 5. Three similar hydraulic circuits representing for the hybrid actuator were built and inserted into the control system in Simulink. The compared controllers were then applied to the three hydraulic circuits, respectively, to make the comparison of the simulated force responses.

In case of using the proposed controller, the detailed fuzzy PID based the REKF scheme is clearly described in Section 2 and Figs. 1–3. There are two inputs to the fuzzy controllers: absolute error $|e(t)|$ and absolute derivative of error $|de(t)|$.

The ranges of these inputs are from 0 to 1, which are obtained from the absolute values of system error and its derivative through the scale factors chosen from the specification of the load simulator. For each input variable, some MFs are used to device overlap partitions in the variable input range. The more MFs, the more complex control system and the more computation time are. Based on design experience obtained from the previous researches [7,8,13], five triangle MFs are used in this paper for smooth tuning P, I, and D parameters while it does not require much calculating time consumption. Here, "Z", "VS", "S", "M" and "B" are "Zero", "Very Small", "Small", "Medium" and "Big", respectively. The centroids of the MFs are set at the same intervals and the same shape sizes initially as in Fig. 6a.

The output U_a of the tuning fuzzy controller has five MFs: "Z" (Zero), "VS" (Very Small), "S" (Small), "M" (Medium), and "B" (Big). They are initially set at the same intervals as in Fig. 6b.

Based on the above fuzzy sets of the input and output variables, the fuzzy rules for the online tuning fuzzy PID applied to the hydraulic system are described in Table 2. From the output of the three separated fuzzy P, I, and D controllers, the control signal applied for the system to control the servo motor is computed as describing in Section 2.1.

Simulations were carried out for checking the control performances of the different controllers used for the hydraulic actuator in case of perturbation working environment. Fig. 7 shows the simulated step responses of the system in cases of using different controllers. From the results in Fig. 7, the simulated force tracking of the system using the PID controller (dash line) contained oscillations due to the disturbance signal added to the force feedback signal. The performance of system using the fuzzy PID controller (the dash-dot line in Fig. 7) was a little better than that using the conventional PID controller but also remained oscillations. However, when using the online tuning fuzzy PID controller based the REKF, the control quality was the best with a very small steady error about 0.7% of the desired force level. During the simulation pro-

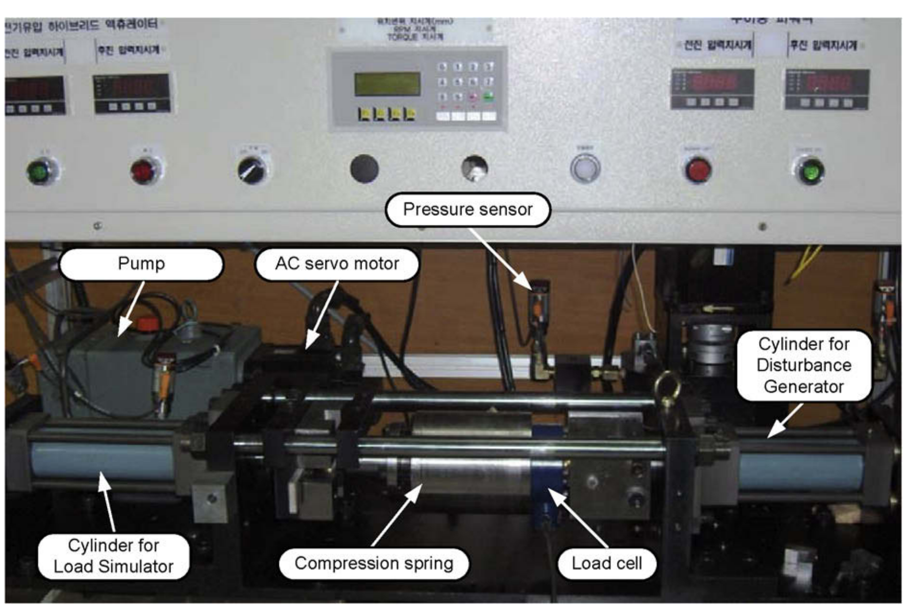


Fig. 10. Photograph of hydraulic testing machine.

cess, the characteristic parameters of the proposed controller were directly adjusted based on the system error and the set of previous data of these parameters to reach the control target with high precision. Fig. 8 is the optimizing results of the fuzzy PID controller using the REKF to obtain the better control quality as in Fig. 7.

In addition, to compare the simulation control results for different set-point forces including noise which is the same condition with the previous simulation, the tracking for a multi step reference signal was investigated by simulation. Fig. 9 displays the simulated output responses of the testing circuits using the PID, the fuzzy PID, and the REKF fuzzy PID controller in comparison. From the simulation results, it is obvious that the hybrid hydraulic circuit using the proposed controller achieved the best tracking force response with the steady state error coming to zero (less than 0.5% of the desired force level) while the simulated responses of the system using the other controllers existed oscillations. From all the simulations, the results proved that the system applied the proposed controller has high stability even in case of the working condition containing large perturbation. Consequently, the proposed control method was applied to the real testing machine in the next section for real-time checking the force control performance.

3.2. Experiments

3.2.1. Experimental setup

The experimental apparatus is shown in Fig. 10. The system hardware consists of a hybrid electro-hydraulic actuator, a computer included PCI-bus multifunction cards and another hydraulic circuit generating disturbances simulating the noises in the hybrid hydraulic systems. The structure of the hybrid actuator is described in Section 3.1.1. A compression spring is used to connect the hybrid actuator and the disturbance generator. Moreover, a load cell is used for obtaining the feedback force signal. The setting parameters for the testing machine are as shown in Table 3.

A compatible PC included two PCI-bus data acquisition & control cards (Advantech cards, PCI 1711 and PCI 1720) is used to receive, process feedback signal and generate the output signal to control the motor of the hybrid actuator and then perform force control performance.

3.2.2. Experimental results

The ability of the proposed REKF fuzzy PID controller was shown through simulation results in Section 3.1.1. In this section, experiments were carried out to prove the effectiveness of the designed controller when applied to the real-time control system, hybrid hydraulic testing machine. The working environment included a large effect of disturbance is applied to verify the tracking performance and the robustness of the online tuning REKF fuzzy PID controller in the comparison with the conventional controllers.

The online tuning fuzzy PID based REKF control algorithm which is used to control the testing machine is built by the combination of Simulink and Real-time Windows Target Toolbox of Matlab and connected to Advantech cards. The sampling time was set to be 0.01 s for all experiments. Furthermore, to make the chal-

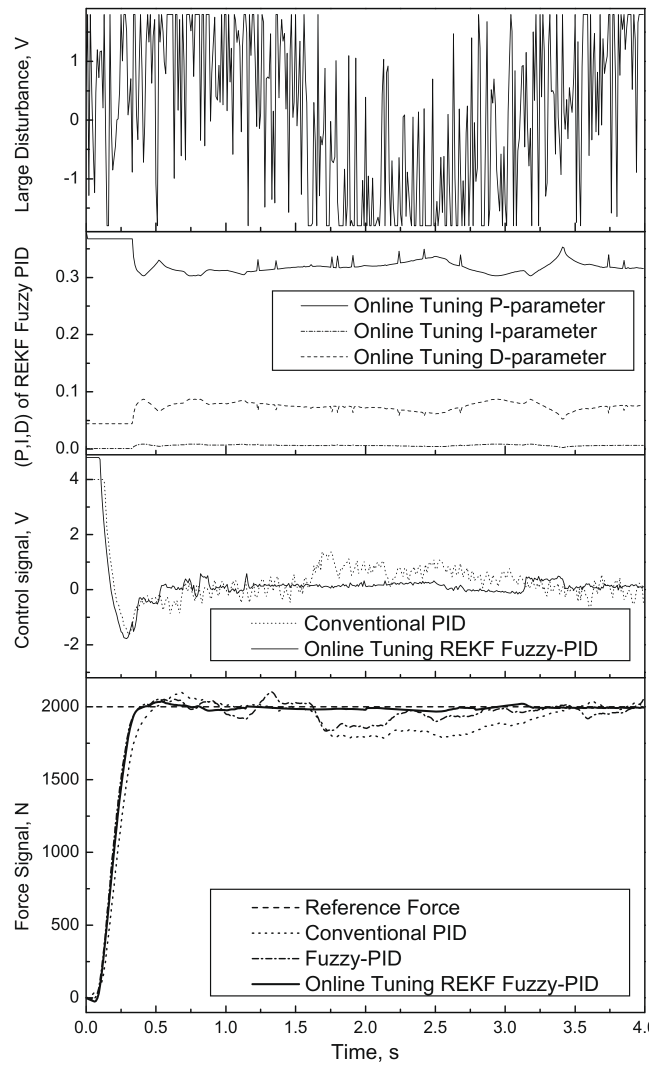


Fig. 11. Experimental results – Comparison of system responses between using different controllers with respect to a step force reference in case of large disturbance environment.

Table 3
Setting parameters for the experimental system.

System parameters	Parts		Meaning
	Hybrid actuator	Disturbance generation	
AC servo motor	YASKAWA SGMGH-30PCA21	OTIS-LG FMA-KN55-EB01	Series No.
	200	200	Power supply(volts AC)
	2.9	2.2	Power (kW)
	18.6	26.18	Rate torque (Nm)
Pump	2500	2000	Speed (rev/min)
	15	10	Pump displacement (cc/rev)
	$63 \times 35 \times 150$	$55 \times 35 \times 100$	Piston diameter × Rod diameter × Length of stroke (mm)
	21	21	Maximum pressure (MPa)
Spring system	519		Environment stiffness 1 (kN/m)
Relief pressure (bar)	175		Relief valve cracking pressure (bar)
Load cell	5		Capacity (tonf)

lenge for the proposed controller, a large disturbance source containing the band-limited white noises and the sine wave noise as (50) is generated real time during the system operation for all experiments. The noise signal performed in (50) is sent from the computer to the AC servo driver of the disturbance generation part by the DA converter (PCI 1720). Then the AC servo motor (FMA-KN55) with the hydraulic circuit are used to create the large perturbation environment for the hybrid system in testing the force control performance.

At first, the conventional PID controller is applied to the hybrid system with step force control target. The force response of the system in this case is plotted as the dot line in Fig. 11. The result shows that the system using the conventional PID controller with fixed gains does not yield reasonable performance over a wide range of operating conditions. Next, the fuzzy PID controller without online self tuning capability is used to improve the control performance. Consequently, the response in this case displayed as the dash-dot line in Fig. 11. It shows that the tracking result in case of using the fuzzy PID is better than in case of the conventional PID. However, it is clear that the control performance is just improved slightly and not stable.

Hence, the online tuning fuzzy PID controller based REKF described in Section 2 is implemented to overcome the above control problems. The force response of the system using the proposed control method with respect to a step force reference is depicted as the black line in Fig. 11. The comparison of experimental results in Fig. 11 proves that the response of the system using the proposed controller was more stable than the other controllers and the performance was significantly improved when the system worked in the large perturbation environment. Table 4 is an analysis of the system responses using the different controllers with respect to a step force reference. From this table, it shows that the steady state error in case of using the proposed controller was small as the simulation results while in case of using the other controllers, the system was not stable.

Table 4
Comparison of system step responses using different controllers.

Controller	2000 N reference force – Large disturbed working environment			
	T_d (Delay time – s)	T_r (Rise time – s)	T_s (Settling time – s)	*SSE* (Steady state error – %)
PID	0.21	0.27	–	–
Fuzzy PID	0.16	0.19	–	–
REKF fuzzy PID	0.16	0.19	0.35	1

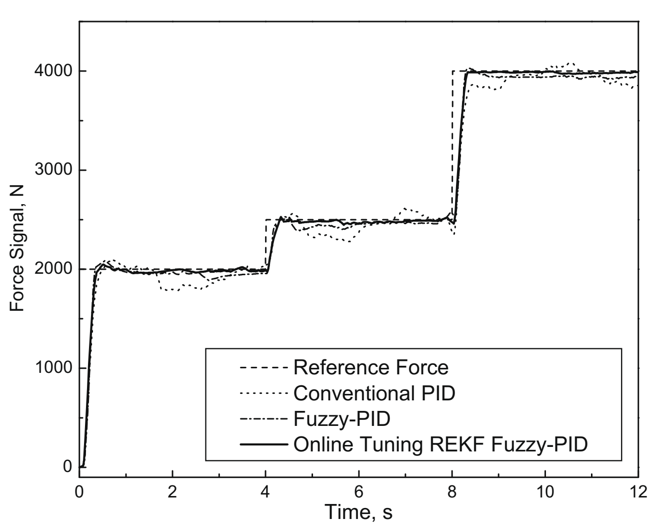


Fig. 12. Experimental results – Comparison of system responses between using different controllers with respect to a multi step force reference in case of large disturbance environment.

In addition, to compare the ability of the REKF fuzzy PID controller with the conventional controllers for different set-point reference inputs, the multiple step signals were investigated as displayed in Fig. 12. It can be seen that the proposed controller achieves the best tracking response when compared with the other controllers. It is clear that a good regulation is realized in the case of using the online tuning fuzzy PID tuners based on the REKF to design a controller.

4. Conclusions

This paper presents a novel control method – online tuning fuzzy PID controller based on a robust extended Kalman filter. The traditional PID control combined with self tuning fuzzy sets whose shapes of MFs are tuned online by using REKF algorithm obtains better performance and higher control precision.

Simulations and experiments were carried out to evaluate the effectiveness of the proposed control method applied for a specific case – force control of an electro-hydraulic system even if the external disturbance varies as the real working environment. The simulation evaluation and experimental results showed that the proposed online tuning fuzzy PID controller could achieve good tracking with respect to different reference input signals and in case of variation of disturbances. In addition, the proposed controller was compared with the conventional controllers to prove convincingly that the controller designed by fuzzy PID methodology and REKF technique could satisfy the robust performance requirement, tracking performance specification, and disturbance attenuation requirement.

Acknowledgements

This research was financially supported by the Ministry of Commerce, Industry and Energy (MCOIE) and Korea Industrial Technology Foundation (KOTEF) through the Human Resource Training Project for Regional Innovation.

References

[1] J. Wang, D. An, C. Lou, Application of fuzzy-PID controller in heating ventilating and air-conditioning system, in: Proceedings of the IEEE International Conference on Mechatronics and Automation, China, 2006, pp. 2217–2222.
[2] C.D. Lee, C.W. Chuang, C.C. Kao, Apply fuzzy PID rule to PDA based control of position control of slider crank mechanisms, in: Proceedings of IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and Robotics, Automation and Mechatronics (RAM), Singapore, 2004, pp. 508–513.
[3] N.B. Kha, K.K. Ahn, Position control of shape memory alloy actuators by using self tuning fuzzy PID controller, in: Proceedings of IEEE International Conference on Industrial Electronics and Applications, Singapore, 2006.
[4] R. Ahmed, O. Abdul, C. Marcel, dSPACE DSP-based rapid prototyping of fuzzy PID controls for high performance brushless servo drives, in: Proceedings of IEEE International Conference, USA, 2006, pp. 1360–1364.
[5] X. Huang, L. Shi, Simulation on a fuzzy-PID position controller on the CNC servo system, in: Proceedings of the IEEE Sixth International Conference on Intelligent Systems Design and Applications, China, 2006, pp. 305–309.
[6] D.Q. Truong, K.K. Ahn, K.J. Soo, J.H. Soo, Application of fuzzy-PID controller in hydraulic load simulator, in: Proceedings of the IEEE International Conference on Mechatronics and Automation, Harbin, China, 2007, pp. 3338–3343.
[7] K.K. Ahn, D.Q. Truong, Y.H. Soo, Self tuning fuzzy PID control for hydraulic load simulator, in: Proceedings of the IEEE International Conference on Control, Automation and Systems, Korea, 2007, pp. 345–349.
[8] M. Guzelkaya, I. Eksin, E. Yesil, Self-tuning of PID-type fuzzy logic controller coefficients via relative rate observer, Engineering Applications of Artificial Intelligence 16 (2003) 227–236.
[9] Y. Yongquan, H. Ying, Z. Bi, The dynamic fuzzy method to tune the weight factors of neural fuzzy PID controller, in: Proceedings of IEEE International

Joint Conference on Neural Networks, Budapest, Hungary, 2004, pp. 2397–2402.
[10] K.K. Tan, S. Huang, R. Ferdous, Robust self-tuning PID controller for nonlinear systems, Journal of Process Control 12 (2002) 753–761.
[11] W.D. Chang, R.C. Hwang, J.G. Hsieh, A self-tuning PID control for a class of nonlinear systems based on the Lyapunov approach, Journal of Process Control 12 (2002) 233–242.
[12] Y.T. Juang, Y.T. Chang, C.P. Huang, Design of fuzzy PID controllers using modified triangular membership functions, Information Sciences 178 (2008) 1325–1333.
[13] K.K. Ahn, D.Q. Truong, T.Q. Thanh, B.R. Lee, Online self-tuning fuzzy proportional–integral–derivative control for hydraulic load simulator, Proceedings of IMechE Part I: Journal of Systems and Control Engineering 222 (2008) 81–95.
[14] F.L. Lewis, Applied Optimal Control and Estimation, Prentice-Hall, New York, 1992.
[15] M.S. Grewal, A.P. Andrews, Kalman Filtering: Theory and Practice, Prentice-Hall, New York, 1993.
[16] R.G. Brown, P.Y.C. Hwang, Introduction to Random Signals and Applied Kalman Filtering, Wiley, New York, 1997.
[17] S.J. Kwon, Robust Kalman filtering with perturbation estimation process for uncertain systems, Proceedings of IEE Control Theory Application 153–155 (2006) 600–606.
[18] J.J. Simon, K.U. Jeffrey, A new extension of the Kalman filter to nonlinear systems, in: Proceedings of International Symposium Aerospace/Defense Sensing, Simulation and Controls, Orlando, 1997.
[19] D.J. Jwo, S.H. Wang, Adaptive fuzzy strong tracking extend Kalman filter for GPS navigation, Proceedings of IEEE Journal of Sensors 7 (5) (2007) 778–789.
[20] K. Kazuyuki, C.C. Ka, W. Kajiro, Estimation of absolute vehicle speed using fuzzy logic rule-based Kalman filter, in: Proceedings of the American Control Conference, Seattle, Washington, 1995, pp. 3086–3090.
[21] W.K. Wong, H.S. Lim, An extended Kalman filter based fuzzy adaptive equalizer for powerline channel, in: Proceedings of International Symposium on Powerline Communication and its Application, Vancouver, Canada, 2005, pp. 250–254.
[22] E. Benazera, S. Narasimhan, An extension to the Kalman filter for an improved detection of unknown behavior, in: Proceedings of American Control Conference, USA, 2005, pp. 1039–1041.
[23] C.J. Masreliez, R.D. Martin, Robust Bayesian estimation for the linear model and robustifying the Kalman filter, Proceedings of IEEE Transactions on Automatic Control 22 (3) (1977) 361–371.
[24] C.J. Masreliez, R.D. Martin, Robust Estimation via Stochastic Approximation, Proceedings of IEEE Transactions of Information Theory 21 (3) (1975) 263–271.
[25] R.D. Martin, Robust Estimation of Signal Amplitude, Proceedings of IEEE Transactions on Information Theory 18 (5) (1972) 596–606.
[26] T. Perala, R. Piche, Robust extended Kalman filtering in hybrid positioning applications, On 4th Workshop on Positioning, Navigation and Communication, Hannover, Germany, 2007, pp. 55–63.
[27] C.S. Rodrigo, C.Z. Aldo, Fuzzy logic based nonlinear Kalman filter applied to mobile robots modelling, in: Proceedings of IEEE International Conference on Fuzzy Systems, Budapest, Hungary, 2004, pp.1485–1490.
[28] G.V. Puskorius, L.A. Feldkamp, Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks, Proceedings of IEEE Transactions on Neural Networks 5 (2) (1994) 279–297.
[29] D. Simon, Fuzzy membership optimization via the extended Kalman filter, in: Proceedings of the 19th International Conference of the North American on Fuzzy Information Processing, Atlanta, GA, USA, 2000, pp. 311–315.
[30] AMESim help & www.amesim.com/support.aspx.