Online tuning fuzzy PID controller using robust extended Kalman filter

K.K. Ahn*†, D.Q. Truong‡

*School of Mechanical and Automotive Engineering, University of Ulsan, San 29, Muger 2-dong, Nam-gu, Ulsan 680-764, Republic of Korea
‡Graduate School of Mechanical and Automotive Engineering, University of Ulsan, Ulsan, Republic of Korea

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.

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

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.
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 considered 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 control system 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 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 electrohydraulic 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:

\[ u_{PID}(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt} \]  \hspace{1cm} (1)

where \( e(t) \) is the error between desired set point and the system output, \( de(t) \) is the derivation of error \( e(t) \), \( u_{PID}(t) \) is the control signal for the system and \( K_p, K_i, K_d \) 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 \( K_p, K_i, K_d \) 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:

\[ K_a = K_{a_0} + U_a AK_{a}, \quad U_a \in [0,1], \quad a = p, i \text{ or } d \]  \hspace{1cm} (2)

where \( U_a \) is the parameter obtained from the output of the tuning fuzzy controllers. \( AK_{a} = K_{a1} - K_{a2} \) is the allowable deviation of \( K_{a} \). \( K_{a0}, K_{a1}, K_{a2} \) are the minimum and maximum values of \( K_{a} \) determined from experiments, respectively.

From (2) and Fig. 1, three coefficients \( K_p, K_i, K_d \) 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 \( |e| \) and absolute derivative of error \( |de| \). 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, we can express these MFs as follows:

\[ f_a(x) = \begin{cases} 
1 + \frac{|x - a_j|}{b_{ji}} & \text{if } (-b_{ji}) \leq (x - a_j) \leq 0 \\
1 - \frac{|x - a_j|}{b_{ji}} & \text{if } 0 \leq (x - a_j) \leq (b_{ji}), \\
0 & \text{otherwise} 
\end{cases} \]

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 \( j \)th triangle membership function of the \( i \)th input, respectively. \( N \) is the number 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:
where \( w_i \) is the weight of the control output (the centroid of the \( k \)th output fuzzy MF), \( M \) is the number of fuzzy output sets and \( m_f(w_i) \) is the fuzzy output function given by

\[
m_f(w) = \sum_{ij} m_{fi}(w)
\]

where \( m_{fi}(w) \) is defined as the consequent fuzzy output function when the first and the second input are in the \( i \) and the \( j \) class, respectively.

\[
m_f(w_i) (w) = \delta_i \mu_j
\]

where \( \delta_i \) is an activated factor, which is active when the input \( |e(t)| \) is in class \( i \), and the input \( |de(t)| \) is in class \( j \) and \( \mu_j \) is the height of the consequent fuzzy function obtained from the input class \( i \) and \( j \)

\[
\mu_j = \min |f_{ij}(|e(t)|)|, f_{ij}(|de(t)|)
\]

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

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 decision factors of the fuzzy input MFs \( a_i, b_i^p, b_i^n \) and the weights of the output \( w_i \) are automatically updated by using REKF. This method is described in details as follows sections.

### 2.2. Extended Kalman filter (EKF)

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

\[
x_{k+1} = g_k(x_k, q_k)
\]

\( q_k \sim N(0, Q_k) \)

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

\[
y_{k+1} = h_k(x_k, v_k)
\]

\( v_k \sim N(0, R_k) \)

where \( g(\cdot) \) and \( h(\cdot) \) are nonlinear vector functions of the state; \( q_k \) and \( v_k \) 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 \( v_k \) are white zero-mean, Gaussian and independent from each other with

\[
\begin{align*}
E(x_0) &= 0 \\
E(|x_0 - \bar{x}_0|^2) &= P_0 \\
E(q_k) &= 0 \\
E(q_k q_l^T) &= Q_{kl} \\
E(v_k) &= 0 \\
E(v_k v_l^T) &= R_{kl}
\end{align*}
\]

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

\[
\delta_{ij} = \begin{cases} 
0, & i \neq j \\
1, & i = j
\end{cases}
\]

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

\[
G_k = \frac{\partial g_k(x_k)}{\partial x_k} |_{x_k = \bar{x}_k} ; \ H_k = \frac{\partial h_k(x_k)}{\partial x_k} |_{x_k = \bar{x}_k}
\]

where consider \( \bar{x}_0 \) and \( \bar{x}_k \) are the prior and posterior mean estimates at time step \( t_k \). Then, the system model can be approximated as

\[
x_{k+1} = G_k x_k + q_k
\]

\( y_k = H_k x_k + v_k \)

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

\[
\begin{align*}
E(x_k | y_1, y_2, \ldots, y_k) &= \bar{x}_k^k = G_k \bar{x}_{k-1}^k \\
V(x_k | y_1, y_2, \ldots, y_k) &= P_k^k = G_k P_{k-1}^k G_k^T + Q_k
\end{align*}
\]

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

\[
\begin{align*}
E(x_k | y_1, y_2, \ldots, y_k) &= \bar{x}_k^p = \bar{x}_k^k + K_k(z_k - H_k x_k^p) \\
V(x_k | y_1, y_2, \ldots, y_k) &= P_k^p = (I - K_k H_k) P_k^k
\end{align*}
\]

where \( K_k \) is the Kalman gain matrix

\[
K_k = P_k^k H_k^T (H_k P_k^k H_k^T + R_k)^{-1}
\]

### 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 opti-
mimize 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 $F$ 

$$\mathcal{F}(\gamma_p, p) = \{ F(\cdot) \mid \int_{-\gamma_p}^{\gamma_p} F(\theta) d\theta = F(-\gamma_p) - p/2 = \Phi(-\gamma_p), \quad \gamma_p > 0, \quad 0 < p < 1, \quad F(\cdot) \text{ symmetric & continuous at } \pm \gamma_p \}$$

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

Let $\mathcal{F}$ 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 $V(T, F)$. $T \in \mathcal{F}, F \in \mathcal{F}(\gamma_p, p)$. Therefore, there exits a min-max solution $(T_0, F_0)$ ($F_0 \in \mathcal{F}(\gamma_p, p)$ and $T_0 \in \mathcal{F}$) such that

$$\sup_T V(T_0, F) = V(T_0, F_0) = \inf_F V(T, F_0)$$

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

$F_0$: 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

$$f_0(\gamma) = \left\{ \begin{array}{ll}
K \cos^2 \left( \frac{\gamma}{2 \tan \left( \frac{1}{2s_m} \right)} \right), & |\gamma| \leq \gamma_p \\
K \cos^2 \left( \frac{1}{2s_m} \right) \exp[-2Kp^{-1} \cos^2 \left( \frac{1}{2s_m} \right)](|\gamma| - \gamma_p), & |\gamma| > \gamma_p
\end{array} \right. \tag{20}$$

where $K$ is defined by

$$K = \frac{1 - p}{\gamma_p \left[ 1 + s_m \sin \left( \frac{1}{2s_m} \right) \right]} \tag{21}$$

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 $F_0$ is computed as

$$s_m(\gamma) = \frac{f_0(\gamma)}{f_0(\gamma_p)} = \left\{ \begin{array}{ll}
- \frac{1}{s_m \gamma_p} \tan \left( \frac{\gamma}{2s_m \gamma_p} \right), & |\gamma| \leq \gamma_p \\
- \frac{1}{s_m \gamma_p} \tan \left( \frac{1}{2s_m} \right) \text{sign}(\gamma_p), & |\gamma| > \gamma_p
\end{array} \right. \tag{23}$$

### 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)

$$\left\{ \begin{array}{l}
\dot{x}_k = H'_{k} x_k + v_k \\
\gamma_k = z_k - H'_{k} x_k
\end{array} \right. \tag{24}$$

with

$$\left\{ \begin{array}{l}
\dot{z}_k = T_k y_k \\
H'_k = T_k H_k \\
v'_k = T_k v_k
\end{array} \right. \tag{25}$$

where $T_k$ is a transformation matrix which is determined below.

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

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

The transformation matrix is chosen as

$$T_k = (H'_k P_k H'_k + R_k)^{-1/2} (T_k^T T_k + (H'_k P_k H'_k + R_k)^{-1}) \tag{27}$$

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

$$E(x_k|y_1 \ldots y_k) = \hat{x}_k + P_k H'_k T_k^T T_k (z_k - H'_k \hat{x}_k) \tag{28}$$

In addition, because the transformation matrix is a standard normal distribution, apply the PPE (Section 2.3.1) into the EKF. If $p(y_k|y_1 \ldots y_{k-1})$ is the least favorable density of the PPE $\mathcal{F}_p$ then the posterior mean and posterior covariance in (16) are now modified as following equations. First, the posterior mean in (28) is computed

$$E(x_k|y_1 \ldots y_k) = \hat{x}_k + P_k H'_k T_k^T \psi(\gamma_k) \tag{29}$$

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

$$\psi(\gamma_k) = -\nabla \log p(y_k|y_1 \ldots y_{k-1}) \bigg|_{y_k} = \left\{ \begin{array}{ll}
\frac{1}{s_m \gamma_p} \tan \left( \frac{\gamma}{2s_m \gamma_p} \right), & |\gamma| \leq \gamma_p \\
\frac{1}{s_m \gamma_p} \tan \left( \frac{1}{2s_m} \right) \text{sign}(\gamma_p), & |\gamma| > \gamma_p
\end{array} \right. \tag{30}$$

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

Next, the posterior covariance is given by

$$V(x_k|y_1 \ldots y_k) = P_k = \left[ I - (P_k H'_k T_k^T T_k H_k E_{p}(\psi(\gamma_k))) P_k \right] P_k = I - K_h H_k E_{p}(\psi(\gamma_k)) P_k$$

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

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

### 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 $a_i, b_i', b_i''$ and the weights of the output $w$. 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 ($[e(t), \text{and} \{d(e(t))\}]$) 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

$$x = [x_p \quad x_l \quad x_D]^T \tag{33}$$

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

$$x_{p,l,D} = [a_1, b_1', b_1'' \ldots a_n, b_n', b_n'', a_{n+1} b_{n+1}', b_{n+1}'', \ldots a_{2n} b_{2n}', b_{2n}'', w_1 \ldots w_c]_a$$

\[34\]
where \(a_i\), \(b_i^+\), and \(b_i^-\) are the centroid, left half-width, and right half-width of the \(j\)th triangle MF of the \(i\)th 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 \(k\)th output fuzzy MF, number of weights is \(n\)).

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

\[
x_{k+1} = x_k + q_k \\
y_k = h(x_k) + v_k
\]

(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 \(v_k\), 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 \(x_0\). Furthermore, because of this prior knowledge about the process, the error associated with our initial estimate is zero. Thus,

\[
P_0 = 0
\]

(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

\[
Q_k = \text{cov} \{ q_0, q_1, q_2 \}_{0:k}
\]

(37)

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

\[
q_{[q_0/q_1/q_2]} = [q_01 q_11 \ldots q_0^n q_{1} \ldots q_{m2} q_{m2} q_1 \ldots q_n]_{0:k}
\]

(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

\[
H_k = \frac{\partial h_k(x_k)}{\partial x_k}|_{x_k=x_k^*}
\]

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

\[
\frac{\partial h_k(w_k)}{\partial w_i} = \frac{\partial y}{\partial w_i} = \frac{\partial y}{\partial u_m} \frac{\partial u_m}{\partial U_a} \frac{\partial U_a}{\partial w_i}
\]

(39)

where

\[
\frac{\partial y}{\partial u_m} = \Delta m + \frac{y(t) - y(t-1)}{\Delta u_m(t) - \Delta u_m(t-1)}
\]

(40)

\[
\frac{\partial u_m}{\partial U_a} = \frac{\Delta u_m}{\Delta K_e \Delta \epsilon(t)}
\]

\[
\frac{\partial U_a}{\partial x_i} = \frac{\Delta K_e}{\Delta \epsilon(t)}
\]

(41)

\[
\frac{\partial U_a}{\partial u_m} = \frac{\Delta K_e \Delta \epsilon(t)}{\Delta \epsilon(t)}
\]

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

\[
\frac{\partial h_k(a_i)}{\partial a_i} = \frac{\partial y}{\partial a_i} \frac{\partial a_i}{\partial u_m} \frac{\partial u_m}{\partial U_a} \frac{\partial U_a}{\partial \epsilon(t)} \frac{\partial \epsilon(t)}{\partial \epsilon(t)}
\]

(42)

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

\[
\frac{\partial m_f(w_k)}{\partial U_a} = \frac{\sum_{k=1}^{n} m_f(w_k)(w_k-w_k)}{(\sum_{k=1}^{n} m_f(w_k))^2}
\]

(43)

(44)

(45)

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

\[
\frac{\partial h_k(b^+_i)}{\partial b^+_i} = \frac{\partial y}{\partial b^+_i} \frac{\partial b^+_i}{\partial u_m} \frac{\partial u_m}{\partial U_a} \frac{\partial U_a}{\partial \epsilon(t)} \frac{\partial \epsilon(t)}{\partial \epsilon(t)}
\]

(46)

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

\[
\frac{\partial m_f(w_k)}{\partial b^+_i} = \frac{\sum_{k=1}^{n} m_f(w_k)(w_k-w_k)}{(\sum_{k=1}^{n} m_f(w_k))^2}
\]

(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 (\(P, \text{ or } I, \text{ or } D\)) 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 matrices (such as \(P, \text{ or } Q\) 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 \([\epsilon(t)] \text{ or } [\delta(t)]\), 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 \([\epsilon(t)] \text{ or } [\delta(t)]\), 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,
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.

A state vector size \( = 4 \times (3 \text{ tuned fuzzy parameters}) = 12 \text{(elements)} \)

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

\[
\hat{x}_{(P1,D9)} = \begin{bmatrix} \hat{a}_1 & \hat{b}_1 & \hat{c}_1 & \ldots & \hat{a}_4 & \hat{b}_4 & \hat{c}_4 & \ldots & \hat{a}_8 & \hat{b}_8 & \hat{c}_8 \end{bmatrix}^T
\]

where each of element of the state vector \( \hat{x} \) 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.

\[ x' = [x_P \ x_I \ x_D]^T \]
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.

![Simulation diagram for testing system with using different controllers](image)

![Simulation diagram built in Simulink/Matlab to connect with AMESim](image)

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

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

<table>
<thead>
<tr>
<th>System parameters</th>
<th>Value</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC servo motor</td>
<td>200</td>
<td>Power supply (volts AC)</td>
</tr>
<tr>
<td></td>
<td>2.9</td>
<td>Power (kW)</td>
</tr>
<tr>
<td></td>
<td>18.6</td>
<td>Rate torque (Nm)</td>
</tr>
<tr>
<td>Pump</td>
<td>2500</td>
<td>Speed (rev/min)</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>Pump displacement (cc/rev)</td>
</tr>
<tr>
<td>Cylinder parameters</td>
<td>63 x 35 x 150</td>
<td>Piston diameter x Rod diameter x Length of stroke (mm)</td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>Maximum pressure (MPa)</td>
</tr>
<tr>
<td>Spring – k</td>
<td>519</td>
<td>Environment stiffness (kN/m)</td>
</tr>
<tr>
<td>Relief pressure (bar)</td>
<td>175</td>
<td>Relief valve cracking pressure</td>
</tr>
<tr>
<td>Working liquid – Oil</td>
<td>0.67</td>
<td>Specific gravity</td>
</tr>
<tr>
<td></td>
<td>1.5 x 10^8</td>
<td>Bulk modulus (Pa)</td>
</tr>
<tr>
<td>Load – M</td>
<td>1000</td>
<td>Loading environment</td>
</tr>
<tr>
<td>Sensor gain</td>
<td>3</td>
<td>Force sensor signal</td>
</tr>
</tbody>
</table>
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

\[ \text{Dis}(t) = A \sin(\omega t) + \text{Rnd}(t) \]  

(50)

Table 2

Rules table of online self tuning fuzzy PID controller.

| \(U_c, U_i, U_d\) | \(|de(t)|\) | 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) |

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 $\omega$ 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

Fig. 8. Simulation results – Fuzzy PID controller based the REKF after tuning.
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 $U_0$ 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 described 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. 9. Simulation results – Comparison of system responses between using different controllers with respect to a multi step force reference.

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-
leng 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.

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 Technol- ogy Foundation (KOTEF) through the Human Resource Training Project for Regional Innovation.

References


