Engineering Applications of Artificial Intelligence 53 (2016) 74–85

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

Modeling of unstructured uncertainties and robust controlling of nonlinear dynamic systems based on type-2 fuzzy basis function networks

Phuong D. Ngo, Yung C. Shin *

School of Mechanical Engineering, Purdue University, 585 Purdue Mall, West Lafayette, IN 47907, USA

ARTICLE INFO

Article history:
Received 4 November 2015
Received in revised form
21 March 2016
Accepted 29 March 2016

Keywords:
Fuzzy control
Robust control
TS fuzzy model
FBFN model
Unstructured uncertainties

ABSTRACT

This paper proposes new methods for modeling unstructured uncertainties and robust controlling of unknown nonlinear dynamic systems by using a novel robust Takagi Sugeno fuzzy controller (RTSFC). First, a new training algorithm for an interval type-2 fuzzy basis function network (FBFN) is proposed. Next, a novel technique is presented to convert the interval type-2 FBFN to an interval type-2 Takagi Sugeno (TS) fuzzy model. Based on the interval type-2 TS and type-2 FBFN models, a robust controller is presented with an adjustable convergence rate. Since the type-2 fuzzy model with its new training technique can effectively capture the unstructured uncertainties and accurately estimate the upper and lower bounds of unknown nonlinear dynamic systems, the stability condition of the proposed control system is much less conservative than other robust control methods that are based on norm bounded uncertainties. Simulation results on an electrohydraulic actuator show that the RTSFC can reduce steady state error under different conditions while maintaining better responses than the other robust sliding mode controllers.

1. Introduction

For unknown dynamic systems, many robust adaptive control techniques have been proposed based on the parameters of a universal approximator (Lee and Tomizuka, 2000; Lee, 2011). Goyal et al. (2015) introduced a robust sliding mode control based on Chebyshev neural networks. Chadli and Guerra (2012) proposed a robust static output feedback for a discrete Takagi–Sugeno (TS) fuzzy system. The stability conditions in their studies are represented in terms of a set of linear matrix inequalities (LMI) conditions. An observer-based output feedback nonlinear robust control of nonlinear systems with parametric uncertainties were introduced by Yao et al. (2014a) to provide a sufficient condition for robust stabilization of the systems when all state variables are not available for measurement. By using a Lyapunov–Krasovskii function (LKF), Hu et al. (2012) introduced a stability condition to stabilize discrete stochastic systems with mixed time delays, randomly occurring uncertainties, and randomly occurring nonlinearities. However, since these methods represented uncertainties as functions of system parameters, they are not applicable for cases where the causes of uncertainties are not known (unstructured uncertainties).

In general, most of the papers in the literature only investigate the stability of fuzzy control systems with structured uncertainties (Lee et al., 2001; Lin et al., 2013; Sato, 2009; Sloth et al., 2009). Unstructured uncertainties, however, represent a much more general class of nonlinear systems and can incorporate model inaccuracies and measurement noise. One method to represent unstructured uncertainties is to model a nonlinear system by a linear system with norm bounded uncertain matrices. Wang et al. (2014) proposed a set of LMIs that need to be solved at each time step to obtain a control solution that satisfies some performance criteria. However, since finding the LMI solution requires special computing tools, real time computation is a challenge in this case especially when the sampling time is relatively small. Furthermore, the solution of the LMIs might not be found because representing a highly nonlinear system by a set of linear systems will lead to large values of uncertainty norms due to linearization error. Another approach that deals with nonlinear systems with unstructured uncertainties is a combination of backstepping and small gain theorem (Li et al., 2014; Liu et al., 2014; Tong et al., 2009). Hsu et al. (2015) proposed the intelligent nonsingular terminal sliding-mode controller and used the Lyapunov theory to prove the stability of the control system. By using the Lyapunov method, Salgado et al. (2014) introduced the proportional derivative fuzzy control supplied with second order sliding mode differentiation. Baghbani et al. (2016) proposed a robust adaptive fuzzy controller

* Corresponding author.
E-mail addresses: pdngo@purdue.edu (P.D. Ngo), shin@purdue.edu (Y.C. Shin).

http://dx.doi.org/10.1016/j.engappai.2016.03.010
0952-1976/

by minimizing the H_2 energy and tracking cost function. However, the above methods can only be applied to a certain class of nonlinear dynamic systems where the input is represented by a linear term in the system's mathematical model. Gao et al. (2012) presents an approach to control general nonlinear systems based on Takagi–Sugeno (T–S) fuzzy dynamic models. The method uses LMI approach to design the TS fuzzy controller to stabilize systems with norm bounded unstructured uncertainties. However, obtaining the bounded norms of uncertain nonlinear systems was not addressed in the paper and the LMI conditions for norm bounded uncertainties are generally conservative.

To capture the uncertainties in systems, type-2 fuzzy systems (Karnik et al., 1999) have been introduced, in which the type-2 fuzzy set is utilized. However, due to the complexity of the rule uncertainties and computational requirements to calculate the output, modeling nonlinear systems by using type-2 fuzzy systems is a very computationally intensive process. This leads to the concept of an interval type-2 fuzzy-logic system, in which the secondary membership functions of either the antecedents or the consequents are simplified to an interval set. Similar to type-1 fuzzy systems, the combination of type-2 fuzzy systems and neural networks brings different intelligent modeling and optimization techniques to obtain rule bases and membership functions without the need of an expert knowledge. Méndez and de los Angeles Hernandez (2009) presented a technique to obtain an interval type-2 fuzzy neural network by the orthogonal least square and back propagation methods. Rubio-Solis and Panoutsos (2015) proposed a modeling framework for an interval type-2 radial basis function neural network via a granular computing and adaptive back propagation approaches. However, the uncertainties represented in type-2 fuzzy neural systems are normally not in the form that can be easily used to design a robust controller. Furthermore, there is a lack of a theoretical stability analysis for type-2 fuzzy neural network based control systems.

Hydraulic positioning systems are important in different industries such as transportation, agriculture and aerospace. The effects of nonlinear frictions are considered as the most important obstacle for improving the precision of hydraulic actuators. Nonlinear friction exists in all hydraulic systems (Wang et al., 2008). The friction uncertainty includes stribeck effect, hysteresis, spring-like characteristics, stiction and varying break-away force (Lin et al., 2013). It has also been known that nonlinear friction is very difficult to model, and hence it is considered as the sources of uncertainties for which many controllers have been implemented to demonstrate their robustness in recent years (Lin et al., 2013; Mandal et al., 2015; Wang et al., 2008; Yao et al., 2014b).

This paper proposes a new method to train an interval type-2 fuzzy basis function network (FBFN) (cf. Section 2). The training algorithm not only further improves the performances of the fuzzy neural network system but also provides a framework to design a robust TS fuzzy controller. FBFNs have been used as models for many nonlinear systems in the literature (Jin and Shin, 2015; Lin, 2007; Ngo and Shin, 2015) since an FBFN was proven to be a universal approximator (Wang and Mendel, 1992). The antecedent of the interval type-2 FBFN in this study is obtained by using the adaptive least square with genetic algorithm (Lee and Shin, 2003) while the interval values of the consequent are obtained by the active set method. A new technique is also proposed to convert an interval type-2 FBFN to an interval type-2 TS fuzzy model (cf. Section 3). Based on the interval type-2 TS model and the interval type-2 FBFN, a robust controller that is not only robust but also produces good transient performances when implemented on nonlinear systems with unstructured uncertainties is presented (cf. Section 4).

2. Training interval type-2 FBFN models by using genetic algorithm and active set method

This section provides a new training method to obtain the type-2 FBFN that can capture unstructured uncertainties within an unknown nonlinear system. Consider a class of nonlinear dynamical system with m inputs and n state variables (m and n are positive integers), which can be represented by the following state space equation:

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k)), \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{1}$$

where $\mathbf{x}(k) = [x_1(k), \ldots, x_n(k)]^T$ is the vector of measurable state variables, $\mathbf{u}(k) = [u_1(k), \ldots, u_m(k)]^T$ is the input vector, k is the time instance, $\mathbf{f}$ is the vector of functions that are locally Lipschitz nonlinear and real continuous in a compact set. The locally Lipschitz property of $\mathbf{f}$ ensures that the solution of the state space equations is existent and unique (Khalil, 2002).

It has been proven by Wang and Mendel (1992) that a linear combination of fuzzy basis functions are capable of uniformly approximating any real continuous function on a compact set to arbitrary accuracy. In this paper, to approximate future states of a nonlinear system, an interval type-2 FBFN model can be constructed from the input and measurable state variable data through a set of J fuzzy rules, in which rule R^j to calculate the future value of the state variable x_p has the following form:

$$\begin{aligned} &\text{Rule } R^j\text{: IF } x_1(k) \text{ is } X_1^j \text{ AND} \ldots x_n(k) \text{ is } X_n^j \text{ AND } u_1(k) \\ &\text{is } U_1^j \text{ AND} \ldots u_m(k) \text{ is } U_m^j \\ &\text{THEN } \tilde{y}(k+1) = \tilde{x}_p(k+1) = \tilde{Y}^j, \quad j = 1, \ldots, J \end{aligned} \tag{2}$$

where $u_1(k) \ldots u_m(k)$ are the inputs at time instance k. $x_1(k) \ldots x_n(k)$ are the measured state variables. $\tilde{y}(k+1)$ is the interval output of the FBFN. $X_1^j \ldots X_n^j$ and $U_1^j \ldots U_m^j$ are type-1 fuzzy sets of rule R^j characterized by Gaussian membership functions $\mu_{X_p^j}(x_i)$ and $\mu_{U_q^j}(u_j)$ $(p = 1, \ldots, n;\ q = 1, \ldots, m)$ with the centers c_{Xp}^j, c_{Uq}^j and standard deviations σ_{Xp}^j, σ_{Uq}^j:

$$X_p^j = \left(x_p, \mu_{X_p^j}(x_p)\right), \quad \mu_{X_p^j}(x_p) = \exp\left[-\frac{1}{2}\left(\frac{x_p - c_{Xp}^j}{\sigma_{Xp}^j}\right)^2\right] \tag{3}$$

$$U_q^j = \left(u_q, \mu_{U_q^j}(u_q)\right), \quad \mu_{U_q^j}(u_q) = \exp\left[-\frac{1}{2}\left(\frac{u_q - c_{Uq}^j}{\sigma_{Uq}^j}\right)^2\right] \tag{4}$$

$\tilde{Y}^j$ is a type-2 interval fuzzy set. $\tilde{Y}^j$ is determined by w_l^j and w_r^j, which are the two end points of its centroid interval set: $\tilde{Y}^j = (x, \mu_{\tilde{Y}^j}(x))$, $\mu_{\tilde{Y}^j}(x) = 1$ when $x \in [w_l^j \quad w_r^j]$.

By assuming that the singleton fuzzier, product inference and centroid defuzzifier are used in the inferencing process, for a crisp input vector

$$\mathbf{z} = (z_1, \ldots, z_{m+n}) = (x_1, \ldots, x_n, u_1, \ldots, u_m)^T \tag{5}$$

the output of the type-2 FBFN described in (2) is an interval number and can be calculated by (Lee and Shin, 2003; Liang and Mendel, 2000):

$$\tilde{y} = [y_l, y_r] = \left[\frac{\sum_{j=1}^{J} w_l^j \left(\prod_{i=1}^{m+n} \mu_i^j(z_i) \right)}{\sum_{j=1}^{J} \left(\prod_{i=1}^{m+n} \mu_i^j(z_i) \right)}, \frac{\sum_{j=1}^{J} w_r^j \left(\prod_{i=1}^{m+n} \mu_i^j(z_i) \right)}{\sum_{j=1}^{J} \left(\prod_{i=1}^{m+n} \mu_i^j(z_i) \right)} \right]$$
$$= \left[\sum_{j=1}^{J} w_l^j p_j(\mathbf{z}), \sum_{j=1}^{J} w_r^j p_j(\mathbf{z}) \right] \quad j = 1, \ldots, J \quad (6)$$

where $p_j(\mathbf{z}) = \frac{\prod_{i=1}^{n} \mu_i^j(z_i)}{\sum_{j=1}^{J} (\prod_{i=1}^{n} \mu_i^j(z_i))}$ is the pseudo fuzzy basis function of rule R^j, z_i is the ith element of the crisp input vector $\mathbf{z}$, J is the number of fuzzy rules.

Assuming that N input-output training pairs $\{\mathbf{z}_t(i), y_t(i)\}$ (with $i = 1, \ldots, N$) are available, the task of training a type-2 FBFN is to determine the pseudo fuzzy basis functions $p_j(\mathbf{z})$ with $j = 1, \ldots, J$ and the output interval fuzzy set characterized by w_l^j and w_r^j in order to minimize the errors $e_l(i)$, $e_r(i)$, $\delta y_l(i)$ and $\delta y_r(i)$ defined by the following equations:

$$y_t(i) = \sum_{j=1}^{J} p_j(\mathbf{z}_t(i)) w_l^j + e_l(i) + \delta y_l(i)$$
$$= \sum_{j=1}^{J} p_j(\mathbf{z}_t(i)) w_r^j - e_r(i) - \delta y_r(i) \quad (7)$$

where $e_l(i)$ and $e_r(i)$ are the training errors, and $\delta y_l(i)$ and $\delta y_r(i)$ are the errors due to system uncertainties. $e_l(i)$, $e_r(i)$, $\delta y_l(i)$ and $\delta y_r(i)$ must be kept positive during the training process to obtain the lower and upper bound of the output interval fuzzy set.

The above equations can be rearranged into matrix forms as follows:

$$\mathbf{y}_t = \mathbf{P}\mathbf{w}_l + \boldsymbol{\varepsilon}_l = \mathbf{P}\mathbf{w}_r - \boldsymbol{\varepsilon}_r \quad (8)$$

where $\mathbf{y}_t = [y_t(1), \ldots, y_t(N)]^T$, $\mathbf{w}_l = [w_{l1}, \ldots, w_{lJ}]^T$, $\mathbf{w}_r = [w_{r1}, \ldots, w_{rJ}]^T$, $\boldsymbol{\varepsilon}_l = [e_l(1) + \delta y_l(1), \ldots, e_l(N) + \delta y_l(1)]^T$, $\boldsymbol{\varepsilon}_r = [e_r(\mathrm{N}) + \delta y_r(\mathrm{N}), \ldots, e_r(N) + \delta y_r(\mathrm{N})]^T$ and

$$\mathbf{P} = \begin{bmatrix} p_1(\mathbf{z}_t(1)) & \ldots & p_J(\mathbf{z}_t(1)) \\ \vdots & \ddots & \vdots \\ p_1(\mathbf{z}_t(N)) & \ldots & p_J(\mathbf{z}_t(N)) \end{bmatrix} \quad (9)$$

The pseudo fuzzy basis functions $p_j(\mathbf{z})$ with $j = 1, \ldots, J$ can be found in a similar way as in the type-1 FBFN (Lee and Shin, 2003). By using the genetic algorithm, the method starts with a preset pseudo fuzzy basis function and sequentially selects basis functions that will decrease the error the most. In other words, each added pseudo fuzzy basis function will maximize the following error reduction measure:

$$[err] = \|\mathbf{P}\mathbf{P}^+\mathbf{y}_t\|^2 \quad (10)$$

where $\mathbf{P}^+$ is the pseudo inverse of the pseudo fuzzy basis function matrix $\mathbf{P}$. The pseudo-fuzzy basis function $p_j(\mathbf{z})$ is characterized by a nonlinear parameter set $\lambda_j = \{\mathbf{c}_j, \boldsymbol{\sigma}_j\}$, where $\mathbf{c}_j = (c_1^j, \ldots, c_{m+n}^j)$ and $\boldsymbol{\sigma}_j = (\sigma_1^j, \ldots, \sigma_{m+n}^j)$ are the vectors of the means and standard deviations of input membership functions. In order to obtain the optimal values of these parameters, the parameters are encoded into binary string and the evolution of the population is conducted through reproduction, cross over and mutation. The fitness of each individual in the population is chosen to be a linear function of the error:

$$g(\lambda) = a[err] + b \quad (11)$$

where a and b are scalar parameters. The use of genetic algorithm for fuzzy basis function network has been proven to be effective for obtaining the pseudo fuzzy basis functions (Lee and Shin, 2001). The training can be done offline based on the input and output data of the nonlinear system. The parameters of the model will be used to design the controller. Hence real time computation with generic algorithms is not required during the implementation of the controller.

Once the response vector matrix $\mathbf{P}$ is determined, finding $\mathbf{w}_l$ and $\mathbf{w}_r$ becomes two constrained linear least-squares problems:

$$\begin{cases} \min_{\mathbf{w}_l} \|\mathbf{P}\mathbf{w}_l - \mathbf{y}_t\|_2, \ \mathbf{P}\mathbf{w}_l \le \mathbf{y}_t \\ \min_{\mathbf{w}_r} \|\mathbf{P}\mathbf{w}_r - \mathbf{y}_t\|_2, \ \mathbf{P}\mathbf{w}_r \ge \mathbf{y}_t \end{cases} \quad (12)$$

In this work, only the first case is considered since the second case can be transformed to the first case by replacing the condition $\mathbf{P}\mathbf{w}_r \ge \mathbf{y}_t$ with an equivalent condition $-\mathbf{P}\mathbf{w}_r \le -\mathbf{y}_t$.

With $\mathbf{H} = \mathbf{P}^T\mathbf{P}$ and $\mathbf{c} = -\mathbf{P}^T\mathbf{y}_t$, the following can be obtained:

$$\frac{1}{2}\|\mathbf{P}\mathbf{w}_l - \mathbf{y}_t\|_2 = \frac{1}{2}(\mathbf{P}\mathbf{w}_l - \mathbf{y}_t)^T(\mathbf{P}\mathbf{w}_l - \mathbf{y}_t)$$
$$= \frac{1}{2}\mathbf{w}_l^T\mathbf{P}^T\mathbf{P}\mathbf{w}_l - \mathbf{y}_t^T\mathbf{P}\mathbf{w}_l + \frac{1}{2}\mathbf{y}_t^T\mathbf{y}_t$$
$$= \frac{1}{2}\mathbf{w}_l^T\mathbf{H}\mathbf{w}_l + \mathbf{c}^T\mathbf{w}_l + \frac{1}{2}\mathbf{y}_t^T\mathbf{y}_t \quad (13)$$

Since $\mathbf{y}_t^T\mathbf{y}_t$ is constant, the first constrained linear least square problem given in Eq. (12) becomes a constrained quadratic programing problem:

$$\min_{\mathbf{w}_l} \|\mathbf{P}\mathbf{w}_l - \mathbf{y}_t\|_2,$$
$$\mathbf{P}\mathbf{w}_l \le \mathbf{y}_t \Leftrightarrow \min_{\mathbf{w}_l}\left\{\mathbf{c}^T\mathbf{w}_l + \frac{1}{2}\mathbf{w}_l^T\mathbf{H}\mathbf{w}_l\right\} \text{subject to } \mathbf{P}\mathbf{w}_l \le \mathbf{y}_t \quad (14)$$

The solution of (14) can be solved by using the active-set method. The active set method is described in (Gill et al., 1991, 1984; MathWorks, 2015) and is available as a commercial package by using the MATLAB optimization toolbox. The steps to obtain $\mathbf{w}_l$ by using the active set method are described as follows:

Step 1: Construct the active constraint matrix $\mathbf{S}_k$ whose rows are taken from the constraints given in matrix $\mathbf{P}$ that are active at the solution point (equality constraint is satisfied). k is the iteration number.

Step 2: Assume that $\mathbf{Q}_k$ and $\mathbf{R}_k$ are the QR decomposition matrices of $\mathbf{S}_k$ ($\mathbf{Q}_k$ is an orthogonal matrix and $\mathbf{R}_k$ is an upper triangular matrix). From the last $N - l$ columns of $\mathbf{Q}_k$, where N is the number of training data and l is the number of active constraints, form matrix $\mathbf{Z}_k$:

$$\mathbf{Z}_k = \mathbf{Q}_k[:, l+1:N] \quad \text{where } \mathbf{Q}_k^T\mathbf{S}_k^T = \mathbf{R}_k \quad (15)$$

Step 3: Calculate the search direction $\mathbf{d}_k$ as a linear combination of the columns of $\mathbf{Z}_k$: $\mathbf{d}_k = \mathbf{Z}_k\mathbf{r}$ for some vector $\mathbf{r}$.

Step 4: Update the value of vector $\mathbf{w}_l$ by the search direction $\mathbf{d}_k$:

$$\mathbf{w}_{l\{k+1\}} = \mathbf{w}_{l\{k\}} + \alpha\mathbf{d}_k \quad (16)$$

where $\alpha = \min_{i=\{1,\ldots,N\}} \frac{-(\mathbf{p}_i\mathbf{w}_{l\{k\}} - y_t(i))}{\mathbf{p}_i\mathbf{d}_k}$ and $\mathbf{p}_i$ is the i^{th} row vector of matrix $\mathbf{P}$.

Step 5: Calculate the Lagrange multiplier vector $\boldsymbol{\lambda}_k$, which satisfies:

$$\mathbf{S}_k^T\boldsymbol{\lambda}_k = \mathbf{c} \quad (17)$$

Step 6: If all the elements of $\boldsymbol{\lambda}_k$ are positive, $\mathbf{w}_{l\{k+1\}}$ is the optimal solution. Otherwise, go to step 1.

3. Obtaining the interval type-2 T–S fuzzy model from the interval type-2 A1-C2 FBFN model

Since type-2 TS fuzzy models have been used extensively to design robust controllers, this section introduces a method to convert an interval type-2 FBFN to an interval type-2 A1-C2 TS

fuzzy model. In the interval type-2 A1-C2 TS fuzzy model, the antecedents are type-1 fuzzy set (A1) while the consequents are type 2 interval numbers (C2). This method will expand the applications of the type-2 FBFN in many areas since existing robust controllers can be easily implemented on nonlinear systems with unstructured uncertainties. Consider a nonlinear system with p state variables where each state variable can be approximated by an interval type-2 FBFN model as described in the previous section. The structure of rule R_p^j of the type-2 FBFN that calculates the state variable x_p, $(p = 1 \dots n)$ has the following form:

$$R_p^j\text{: IF } x_1(k) \text{ is } X_{p,1}^j \text{ and } \dots x_n(k) \text{ is } X_{p,n}^j \text{ and } u_1(k) \text{ is } U_{q,1}^j \text{ and } \dots u_m(k) \text{ is } U_{q,m}^j$$

$$\text{THEN } \tilde{x}_p(k+1) = \tilde{G}_p^j, \quad j = 1, \dots, J \tag{18}$$

where X^j and U^j are type-1 fuzzy sets with Gaussian membership functions. $\tilde{G}_p^j$ is an interval type-2 fuzzy set with its centroid $\tilde{w}_p^j$ as an interval set: $\tilde{w}_p^j = [w_{pl}^j, w_{pr}^j]$. $\tilde{x}_p(k+1)$ is the predicted interval value of the state variable x_p.

Consider $\mathbf{x}(k) = [x_1(k), \ x_2(k), \ \dots x_n(k)]^{\mathrm{T}}$ as the vector of the measured state variables and $\mathbf{u}(k) = [u_1(k), \ u_2(k), \ \dots u_m(k)]^{\mathrm{T}}$ as the input vector. From Eq. (6), $\tilde{x}_p(k+1)$ can be computed by an uncertain nonlinear mapping $\tilde{f}_p\colon \mathbf{u}(k) \subset \Re^m, \mathbf{x}(k) \subset \Re^n \to \tilde{x}_p(k+1) \subset \Re$. The mapping includes $\tilde{w}_p^j$ in the function as the uncertain parameter:

$$\begin{aligned}\tilde{x}_p(k+1) &= \tilde{f}_p(\mathbf{x}(k), \mathbf{u}(k)) \\ &= \frac{\sum_{j=1}^{J} \tilde{w}_p^j \prod_{r=1}^{n} \mu_{X_{p,i}^j}(x_i(k)) \prod_{i=1}^{m} \mu_{U_{q,i}^j}(u_i(k))}{\sum_{j=1}^{J} \prod_{i=1}^{n} \mu_{X_{p,i}^j}(x_i(k)) \prod_{i=1}^{m} \mu_{U_{q,i}^j}(u_i(k))} \\ &= \left[f_{pl}(\mathbf{x}(k), \mathbf{u}(k)), \ f_{pr}(\mathbf{x}(k), \mathbf{u}(k)) \right] \\ &= \left[\frac{\sum_{j=1}^{J} w_{pl}^j \prod_{i=1}^{n} \mu_{X_{p,i}^j}(x_i(k)) \prod_{i=1}^{m} \mu_{U_{q,i}^j}(u_i(k))}{\sum_{j=1}^{J} \prod_{i=1}^{n} \mu_{X_{p,i}^j}(x_i(k)) \prod_{i=1}^{m} \mu_{B_{q,i}^j}(u_i(k))}, \right. \\ &\quad \left. \frac{\sum_{j=1}^{J} w_{pr}^j \prod_{i=1}^{n} \mu_{X_{p,i}^j}(x_i(k)) \prod_{i=1}^{m} \mu_{U_{q,i}^j}(u_i(k))}{\sum_{j=1}^{J} \prod_{i=1}^{n} \mu_{X_{p,i}^j}(x_i(k)) \prod_{i=1}^{m} \mu_{U_{q,i}^j}(u_i(k))} \right] \end{aligned} \tag{19}$$

When the states of the system are around a certain trajectory:

$$\mathbf{x} \approx \boldsymbol{\chi}_i = [\chi_i^{(1)}, \ \dots, \chi_i^{(n)}]^{\mathrm{T}}, \quad \mathbf{u} \approx \boldsymbol{\upsilon}_i = \left[\upsilon_i^{(1)}, \dots, \upsilon_i^{(m)} \right]^{\mathrm{T}} \tag{20}$$

the local linear models of the nonlinear system represented by Eq. (18) can be used to construct fuzzy rules in the interval type-2 TS fuzzy model. By choosing enough operating points, the interval type-2 TS fuzzy model will become a good approximation of the nonlinear dynamic system. At each operating point, the interval type-2 TS fuzzy rule can be obtained as follows:

$$R^i\text{: IF } x_1(k) \text{ is } X_1^i \text{ and } \dots x_n(k) \text{ is } X_n^i \text{ and } u_1(k) \text{ is } U_1^i \text{ and } \dots u_m(k) \text{ is } U_m^i$$

$$\begin{aligned}\tilde{\mathbf{x}}(k+1) &= \boldsymbol{\chi}_i + \tilde{\mathbf{A}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i)\left[\mathbf{x}(k) - \boldsymbol{\chi}_i\right] \\ &\quad + \tilde{\mathbf{B}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i)\left[\mathbf{u}(k) - \boldsymbol{\upsilon}_i\right] \end{aligned} \tag{21}$$

where $X_1, \dots, X_n$ and $U_1 \dots U_m$ are type-1 fuzzy sets with triangular membership functions that describe the operating condition. Each element in the coefficient matrices $\tilde{\mathbf{A}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i)$ and $\tilde{\mathbf{B}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i)$ in Eq. (21) is an interval number. $\tilde{\mathbf{A}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i)$ and $\tilde{\mathbf{B}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i)$ are computed as follows:

$$\begin{aligned}\tilde{\mathbf{A}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i) &= \left[\begin{array}{ccc} \frac{\partial \tilde{f}_1(\mathbf{x}, \mathbf{u})}{\partial x_1} & \dots & \frac{\partial \tilde{f}_1(\mathbf{x}, \mathbf{u})}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial \tilde{f}_n(\mathbf{x}, \mathbf{u})}{\partial x_1} & \dots & \frac{\partial \tilde{f}_n(\mathbf{x}, \mathbf{u})}{\partial x_n} \end{array} \right]_{\mathbf{x}=\boldsymbol{\chi}_i, \mathbf{u}=\boldsymbol{\upsilon}_i} \\ \tilde{\mathbf{B}}_i(\boldsymbol{\chi}_i, \boldsymbol{\upsilon}_i) &= \left[\begin{array}{ccc} \frac{\partial \tilde{f}_1(\mathbf{x}, \mathbf{u})}{\partial u_1} & \dots & \frac{\partial \tilde{f}_1(\mathbf{x}, \mathbf{u})}{\partial u_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial \tilde{f}_n(\mathbf{x}, \mathbf{u})}{\partial u_1} & \dots & \frac{\partial \tilde{f}_n(\mathbf{x}, \mathbf{u})}{\partial u_m} \end{array} \right]_{\mathbf{x}=\boldsymbol{\chi}_i, \mathbf{u}=\boldsymbol{\upsilon}_i} \end{aligned} \tag{22}$$

The partial derivative of the nonlinear mapping f_p with respect to the state variable x_q can be calculated by the following formula:

$$\frac{\partial \tilde{f}_p(\mathbf{x}, \mathbf{u})}{\partial x_q} = \mathbf{a}_{p,q}^{\mathrm{T}}(\mathbf{x}, \mathbf{u}) \cdot \tilde{\mathbf{w}}_p \tag{23}$$

where $\tilde{\mathbf{w}}_p = [\tilde{w}_p^{(1)}, \ \tilde{w}_p^{(2)}, \ \dots \tilde{w}_p^{(J)}]^{\mathrm{T}}$ and $\mathbf{a}_{p,q} = [a_{p,q}^{(1)} \ a_{p,q}^{(2)} \ \dots \ a_{p,q}^{(J)}]^{\mathrm{T}}$. The jth element of vector $\mathbf{a}_{p,q}$ can be calculated as:

$$a_{p,q}^{(j)}(\mathbf{x}, \mathbf{u}) = \frac{\left(-\frac{x_q - c_{X_{p,q}^j}}{\sigma^2_{X_{p,q}^j}} \right) \prod_{r=1}^{n} \mu_{X_{p,r}^j}(x_r) \prod_{r=1}^{m} \mu_{U_{q,r}^j}(u_r)}{\sum_{j=1}^{J} \prod_{r=1}^{n} \mu_{X_{p,r}^j}(x_r) \prod_{r=1}^{m} \mu_{U_{q,r}^j}(u_r)} - \frac{\left[\prod_{r=1}^{n} \mu_{X_{p,r}^j}(x_r) \prod_{r=1}^{m} \mu_{U_{q,r}^j}(u_r) \right] \left[\sum_{j=1}^{J} \left(-\frac{x_q - c_{X_{p,q}^j}}{\sigma^2_{X_{p,q}^j}} \right) \prod_{i=1}^{n} \mu_{X_{p,r}^j}(x_r) \prod_{j=1}^{m} \mu_{U_{q,r}^j}(u_r) \right]}{\left[\sum_{j=1}^{J} \prod_{r=1}^{n} \mu_{X_{p,r}^j}(x_r) \prod_{r=1}^{m} \mu_{U_{q,r}^j}(u_r) \right]^2} \tag{24}$$

Within the rule j of the FBFN model (for the output x_p), $c_{X_{p,q}^j}$ and $\sigma_{X_{p,q}^j}$ are, respectively, the mean and standard deviation of the Gaussian membership function of x_q.

Similarly, the partial derivative of the nonlinear mapping f_p with respect to the state variable u_q can be computed by

$$\frac{\partial \tilde{f}_p(\mathbf{x}, \mathbf{u})}{\partial u_q} = \mathbf{b}_{p,q}^{\mathrm{T}}(\mathbf{x}, \mathbf{u}) \cdot \tilde{\mathbf{w}}_p \tag{25}$$

where $\tilde{\mathbf{w}}_p = [\tilde{w}_p^{(1)},\ \tilde{w}_p^{(2)},\ \ldots \tilde{w}_p^{(M)}]^{\mathrm{T}}$ and $\mathbf{b}_{p,q} = [b_{p,q}^{(1)}\ \ b_{p,q}^{(2)}\ \ \ldots\ \ b_{p,q}^{(M)}]^{\mathrm{T}}$. The j^{th} element of vector $\mathbf{b}_{p,q}$ can be calculated as:

$$b_{q,p}^{(j)}(\mathbf{x},\mathbf{u}) = \frac{\left(-\frac{u_q - c_{U_{p,q}^j}}{\sigma^2_{U_{p,q}^j}}\right)\prod_{r=1}^{n}\mu_{X_{p,r}^j}(x_r)\prod_{r=1}^{m}\mu_{U_{q,r}^j}(u_r)}{\sum_{j=1}^{J}\prod_{r=1}^{n}\mu_{X_{p,r}^j}(x_r)\prod_{r=1}^{m}\mu_{U_{q,r}^j}(u_r)} - \frac{\left[\prod_{r=1}^{n}\mu_{X_{p,r}^j}(x_r)\prod_{r=1}^{m}\mu_{U_{q,r}^j}(u_r)\right]\left[\sum_{j=1}^{J}\left(-\frac{u_q - c_{U_{p,q}^j}}{\sigma^2_{U_{p,q}^j}}\right)\prod_{r=1}^{n}\mu_{X_{p,r}^j}(x_r)\prod_{r=1}^{m}\mu_{U_{q,r}^j}(u_r)\right]}{\left[\sum_{j=1}^{J}\prod_{r=1}^{n}\mu_{X_{p,r}^j}(x_r)\prod_{r=1}^{m}\mu_{U_{q,r}^j}(u_r)\right]^2} \tag{26}$$

Assume that $\mathbf{A}_{i\,\min}$, $\mathbf{A}_{i\,\max}$, $\mathbf{B}_{i\,\min}$ and $\mathbf{B}_{i\,\max}$ are matrices that contain the lower and upper values of each element of matrices $\tilde{\mathbf{A}}_i$ and $\tilde{\mathbf{B}}_i$, respectively. Finding $\mathbf{A}_{i\,\min}$, $\mathbf{A}_{i\,\max}$, $\mathbf{B}_{i\,\min}$ and $\mathbf{B}_{i\,\max}$ becomes the problem of obtaining the maximum and minimum values of $\mathbf{a}_{p,q}^{\mathrm{T}}(\mathbf{x},\mathbf{u})\big|_{\mathbf{x}=\chi_i,\mathbf{u}=\upsilon_i}\cdot\tilde{\mathbf{w}}_p$ and $\mathbf{b}_{p,q}^{\mathrm{T}}(\mathbf{x},\mathbf{u})\big|_{\mathbf{x}=\chi_i,\mathbf{u}=\upsilon_i}\cdot\tilde{\mathbf{w}}_p$, respectively. Since the elements of matrices $\mathbf{a}_{p,q}$ and $\mathbf{b}_{p,q}$ are crisp numbers while the elements of vector $\tilde{\mathbf{w}}_p$ are interval numbers, the solution can be obtained easily by using existing linear programing methods such as the simplex method (Dantzig et al., 1955) or interior-point methods (Mehrotra, 1992; Zhang, 1998).

In addition to $\mathbf{A}_{i\,\min}$, $\mathbf{A}_{i\,\max}$, $\mathbf{B}_{i\,\min}$ and $\mathbf{B}_{i\,\max}$, finding the coefficient matrices of the type-2 TS fuzzy model, which produce the upper and lower bounds of the output is important for the controller design purpose. With $f_{pl}(\mathbf{x}(k),\mathbf{u}(k))$ and $f_{pr}(\mathbf{x}(k),\mathbf{u}(k))$ defined in Eq. (19), the matrices $\mathbf{A}_{il}$, $\mathbf{B}_{il}$ are introduced as the linearized coefficient matrices of $f_{pl}(\mathbf{x}(k),\mathbf{u}(k))$ through the linearization process as given in Eq. (22). Similarly, $\mathbf{A}_{ir}$, $\mathbf{B}_{ir}$ are introduced as the linearized coefficient matrices of $f_{pr}(\mathbf{x}(k),\mathbf{u}(k))$. Then, when $\mathbf{x}(k)\approx\chi_i$, $\mathbf{u}(k)\approx\upsilon_i$ the following approximations can be obtained:

$$\chi_i + \mathbf{A}_{il}(\chi_i,\upsilon_i)\big[\mathbf{x}(k)-\chi_i\big] + \mathbf{B}_{il}(\chi_i,\upsilon_i)\big[\mathbf{u}(k)-\upsilon_i\big] \approx f_{pl}(\mathbf{x}(k),\mathbf{u}(k)) \tag{27}$$

and

$$\chi_i + \mathbf{A}_{ir}(\chi_i,\upsilon_i)\big[\mathbf{x}(k)-\mathbf{x}_0\big] + \mathbf{B}_{ir}(\chi_i,\upsilon_i)\big[\mathbf{u}(k)-\upsilon_i\big] \approx f_{pr}(\mathbf{x}(k),\mathbf{u}(k)) \tag{28}$$

In other words, $\mathbf{A}_{il}$ and $\mathbf{B}_{il}$ are the coefficient matrices of the local linear model, which approximate the lower bounds of the nonlinear system output, $\mathbf{A}_{ir}$ and $\mathbf{B}_{ir}$ are the coefficient matrices that are used to approximate the upper bound of the output. It is noted that the values of $\mathbf{A}_{il}$, $\mathbf{B}_{il}$, $\mathbf{A}_{ir}$ and $\mathbf{B}_{ir}$ are different from the values of $\mathbf{A}_{i\,\min}$, $\mathbf{A}_{i\,\max}$, $\mathbf{B}_{i\,\min}$ and $\mathbf{B}_{i\,\max}$.

With $\mathbf{w}_{pl}$ and $\mathbf{w}_{pr}$ as the lower and upper bounds of $\tilde{\mathbf{w}}_p$, respectively, the element $A_{il\{p,q\}}$ (on the p^{th} row and q^{th} column) of matrix $\mathbf{A}_{il}$ can be calculated by using Eq. (23) as follows:

$$A_{il\{p,q\}} = \mathbf{a}_{p,q}^{\mathrm{T}}(\mathbf{x},\mathbf{u})\big|_{\mathbf{x}=\chi_i,\mathbf{u}=\upsilon_i}\cdot\mathbf{w}_{pl} \tag{29}$$

Similarly:

$$A_{ir\{p,q\}} = \mathbf{a}_{p,q}^{\mathrm{T}}(\mathbf{x},\mathbf{u})\big|_{\mathbf{x}=\chi_i,\mathbf{u}=\upsilon_i}\cdot\mathbf{w}_{pr} \tag{30}$$

$$B_{il\{p,q\}} = \mathbf{b}_{p,q}^{\mathrm{T}}(\mathbf{x},\mathbf{u})\big|_{\mathbf{x}=\chi_i,\mathbf{u}=\upsilon_i}\cdot\mathbf{w}_{pl} \tag{31}$$

$$B_{ir\{p,q\}} = \mathbf{b}_{p,q}^{\mathrm{T}}(\mathbf{x},\mathbf{u})\big|_{\mathbf{x}=\chi_i,\mathbf{u}=\upsilon_i}\cdot\mathbf{w}_{pr} \tag{32}$$

By defining the following matrices:

$$\mathbf{A}_i = \frac{\mathbf{A}_{i\,\max}+\mathbf{A}_{i\,\min}}{2},\quad \mathbf{B}_i = \frac{\mathbf{B}_{i\,\min}+\mathbf{B}_{i\,\max}}{2},\quad \Delta\tilde{\mathbf{A}}_i = \tilde{\mathbf{A}}_i - \mathbf{A}_i,\quad \Delta\tilde{\mathbf{B}}_i = \tilde{\mathbf{B}}_i - \mathbf{B}_i \tag{33}$$

in order to derive the upper bound of the Lyapunov equation proposed in the next section, the matrices $\Delta\mathbf{A}_{im}$ and $\Delta\mathbf{B}_{im}$ are introduced such that

$$\begin{aligned}&\left(\Delta\mathbf{A}_{im}\mathbf{x}+\Delta\mathbf{B}_{im}\mathbf{u}\right)^T\left(\Delta\mathbf{A}_{im}\mathbf{x}+\Delta\mathbf{B}_{im}\mathbf{u}\right)\\ &\quad=\max_{\Delta\mathbf{A}\in\Delta\tilde{\mathbf{A}}_i,\ \Delta\mathbf{B}\in\Delta\tilde{\mathbf{B}}_i}\left[(\Delta\mathbf{A}\mathbf{x}+\Delta\mathbf{B}\mathbf{u})^T(\Delta\mathbf{A}\mathbf{x}+\Delta\mathbf{B}\mathbf{u})\right]\end{aligned} \tag{34}$$

Further introductions of $\boldsymbol{\delta}\mathbf{a}_{im}^p$, $\boldsymbol{\delta}\mathbf{a}_{il}^p$, $\boldsymbol{\delta}\mathbf{a}_{ir}^p$, $\boldsymbol{\delta}\tilde{\mathbf{a}}_i^p$, $\boldsymbol{\delta}\mathbf{b}_{im}^p$, $\boldsymbol{\delta}\mathbf{b}_{il}^p$, $\boldsymbol{\delta}\mathbf{b}_{ir}^p$, $\boldsymbol{\delta}\tilde{\mathbf{b}}_i^p$, $\mathbf{a}_{il}^p$. $\mathbf{a}_{ir}^p$, $\mathbf{b}_{il}^p$, $\mathbf{b}_{ir}^p$, $\mathbf{a}_i^p$, $\mathbf{b}_i^p$ as the pth row of matrices $\Delta\mathbf{A}_{im}$, $\Delta\mathbf{A}_{il}$, $\Delta\mathbf{A}_{ir}$, $\Delta\tilde{\mathbf{A}}_i$, $\Delta\mathbf{B}_{im}$, $\Delta\mathbf{B}_{il}$, $\Delta\mathbf{B}_{ir}$, $\Delta\tilde{\mathbf{B}}_i$, $\mathbf{A}_{il}$, $\mathbf{A}_{ir}$, $\mathbf{B}_{il}$, $\mathbf{B}_{ir}$, $\mathbf{A}_i$ and $\mathbf{B}_i$, respectively, and $\chi_{i\{p\}}$ as the p^{th} row of the operating condition vector χ_i are needed to construct the matrices $\Delta\mathbf{A}_{im}$ and $\Delta\mathbf{B}_{im}$.

If the operating condition $\chi_{i\{p\}}$ is positive, from the definitions of $\mathbf{A}_{ir}$ and $\mathbf{B}_{ir}$, the following can be obtained when $x_p(k)$ is near $\chi_{i\{p\}}$:

$$\begin{aligned}&\left(\boldsymbol{\delta}\mathbf{a}_{ir}^p\mathbf{x}+\boldsymbol{\delta}\mathbf{b}_{ir}^p\mathbf{u}\right)^{\mathrm{T}}\left(\boldsymbol{\delta}\mathbf{a}_{ir}^p\mathbf{x}+\boldsymbol{\delta}\mathbf{b}_{ir}^p\mathbf{u}\right)\\ &\quad=\max_{\boldsymbol{\delta}\mathbf{a}\in\boldsymbol{\delta}\tilde{\mathbf{a}}_i^p,\ \boldsymbol{\delta}\mathbf{b}\in\boldsymbol{\delta}\tilde{\mathbf{b}}_i^p}\left[(\boldsymbol{\delta}\mathbf{a}\mathbf{x}+\boldsymbol{\delta}\mathbf{b}\mathbf{u})^{\mathrm{T}}(\boldsymbol{\delta}\mathbf{a}\mathbf{x}+\boldsymbol{\delta}\mathbf{b}\mathbf{u})\right]\end{aligned} \tag{35}$$

Similarly, if the operating condition $\chi_{i\{p\}}$ is negative, the following can be obtained when $x_p(k)$ is near $\chi_{i\{p\}}$:

$$\begin{aligned}&\left(\boldsymbol{\delta}\mathbf{a}_{il}^p\mathbf{x}+\boldsymbol{\delta}\mathbf{b}_{il}^p\mathbf{u}\right)^{\mathrm{T}}\left(\boldsymbol{\delta}\mathbf{a}_{il}^p\mathbf{x}+\boldsymbol{\delta}\mathbf{b}_{il}^p\mathbf{u}\right)\\ &\quad=\max_{\boldsymbol{\delta}\mathbf{a}\in\boldsymbol{\delta}\tilde{\mathbf{a}}_i^p,\ \boldsymbol{\delta}\mathbf{b}\in\boldsymbol{\delta}\tilde{\mathbf{b}}_i^p}\left[(\boldsymbol{\delta}\mathbf{a}\mathbf{x}+\boldsymbol{\delta}\mathbf{b}\mathbf{u})^{\mathrm{T}}(\boldsymbol{\delta}\mathbf{a}\mathbf{x}+\boldsymbol{\delta}\mathbf{b}\mathbf{u})\right]\end{aligned} \tag{36}$$

Hence, the rows of $\Delta\mathbf{A}_{im}$ and $\Delta\mathbf{B}_{im}$ can be computed by:

$$\begin{aligned}&\text{if } \chi_{i\{p\}}<0:\quad \boldsymbol{\delta}\mathbf{a}_{im}^p=\mathbf{a}_{il}^p-\mathbf{a}_i^p,\quad \boldsymbol{\delta}\mathbf{b}_{im}^p=\mathbf{b}_{il}^p-\mathbf{b}_i^p\\ &\text{if } \chi_{i\{p\}}\geq 0:\quad \boldsymbol{\delta}\mathbf{a}_{im}^p=\mathbf{a}_{ir}^p-\mathbf{a}_i^p,\quad \boldsymbol{\delta}\mathbf{b}_{im}^p=\mathbf{b}_{ir}^p-\mathbf{b}_i^p\end{aligned} \tag{37}$$

4. Robust TS fuzzy controller with integral term

In this section, by using the parameters of the interval type-2 TS and type-2 FBFN models, a robust controller that is based on a relaxed stability condition is presented. Consider a nonlinear system where the state variable vector can be approximated by a type-2 TS fuzzy model with M rules. The structure of rule R^i of the model is described as follows:

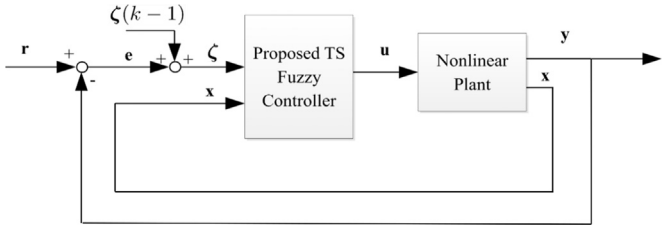

Fig. 1. Schematic diagram of the closed loop control system.

R^i: IF $x_1(k)$ is X_1^i and $\ldots x_n(k)$ is X_n^i and $u_1(k)$ is U_1^i and $\ldots u_m(k)$ is U_m^i

$$\tilde{\mathbf{x}}(k+1) = \tilde{\mathbf{A}}_i\mathbf{x}(k) + \tilde{\mathbf{B}}_i\mathbf{u}(k) \tag{38}$$

where $\tilde{\mathbf{x}}(k+1)$ is the predicted interval value of the state variable vector $\mathbf{x}$. X_1^i, …, X_n^i and U_1^i, …, U_n^i are type-1 fuzzy sets with triangular membership functions. Each element in the coefficient matrices $\tilde{\mathbf{A}}_i$ and $\tilde{\mathbf{B}}_i$ is an interval number. By using the TS fuzzy inference mechanism, the predicted interval output of the fuzzy model can be derived as follows:

$$\tilde{\mathbf{x}}(k+1) = \sum_{i=1}^{M} \bar{\mu}_i(\mathbf{x}(k), \mathbf{u}(k))\cdot\left\{\tilde{\mathbf{A}}_i\mathbf{x}(k) + \tilde{\mathbf{B}}_i\mathbf{u}(k)\right\}$$

$$= \sum_{i=1}^{M} \bar{\mu}_i(\mathbf{x}(k), \mathbf{u}(k))\cdot\left\{(\mathbf{A}_i + \Delta\tilde{\mathbf{A}}_i)\mathbf{x}(k) + (\mathbf{B}_i + \Delta\tilde{\mathbf{B}}_i)\mathbf{u}(k)\right\}$$

$$\tilde{\mathbf{y}}(k) = \mathbf{C}\tilde{\mathbf{x}}(k) \tag{39}$$

where $\mathbf{A}_i$, $\mathbf{B}_i$, $\Delta\tilde{\mathbf{A}}_i$, $\Delta\tilde{\mathbf{B}}_i$ are defined in Eq. (33). $\bar{\mu}_i$ is the normalized weighting function:

$$\bar{\mu}_i(\mathbf{x}(k), \mathbf{u}(k)) = \frac{\prod_{t=1}^{n}\mu_{X_t^i}(x_t)\prod_{t=1}^{m}\mu_{U_t^i}(u_t)}{\sum_{i=1}^{M}\prod_{t=1}^{n}\mu_{X_t^i}(x_t)\prod_{t=1}^{m}\mu_{U_t^i}(u_t)} \tag{40}$$

$\mu_{X_t^i}(x_t)$ and $\mu_{U_t^i}(u_t)$ are the membership functions of x_t and u_t, respectively. $\mathbf{x}(k) = [x_1(k), \ldots x_n(k)]^T \in \mathbb{R}^n$ is the state variable matrix, $\mathbf{u}(k) \in \mathbb{R}^m$ is the control input vector and $\tilde{\mathbf{y}}(k)$ is the output of the system.

A dynamic state feedback robust TS fuzzy controller (RTSFC) (Fig. 1) with N rules is proposed. The structure of rule $R^j$of the controller is described as follows:

R^j: IF $x_1(k)$ is X_1^j and $\ldots x_n(k)$ is X_n^j

THEN $\mathbf{u}(k) = \mathbf{K}_j\mathbf{x}(k) + \mathbf{k}_j\zeta(k)$,

$$\zeta(k) = \zeta(k-1) + \mathbf{e}(k-1), \quad \mathbf{e}(k) = \mathbf{r}(k) - \mathbf{C}\mathbf{x}(k) \tag{41}$$

where ζ is the integral of the error vector $\mathbf{e}$. $\mathbf{K}_j$ is the proportional feedback gain and $\mathbf{k}_j$ is the integral gain of rule j. $\mathbf{r}(k)$ is the reference signal. By using the TS inference mechanism, the output of the controller $\mathbf{u}(k)$ described by Eq. (41) at time instance k can be calculated as:

$$\mathbf{u}(k) = \sum_{j=1}^{N} \bar{v}_j(\mathbf{x}(k))\cdot\left\{\mathbf{K}_j\mathbf{x}(k) + \mathbf{k}_j\zeta(k)\right\} \tag{42}$$

where $\mathbf{x}(k) = [x_1(k), \ldots x_n(k)]^T \in \mathbb{R}^n$ is the state variable matrix, $\bar{v}_j$ is the normalized firing strength of the jth rule:

$$\bar{v}_j(\mathbf{x}(k)) = \frac{\prod_{t=1}^{n}\nu_{X_t^j}(x_t)}{\sum_{j=1}^{N}\prod_{t=1}^{n}\nu_{X_t^j}(x_t)} \tag{43}$$

and $\nu_{X_t^i}(x_t)$ is the membership functions of x_t.

By substituting Eq. (42) into Eq. (39), the closed loop equations can be obtained:

$$\begin{aligned}\tilde{\mathbf{x}}(k+1) &= \sum_{i=1}^{M}\sum_{j=1}^{N}\bar{\mu}_i(\mathbf{x}(k), \mathbf{u}(k))\bar{v}_j(\mathbf{x}(k))\\ &\quad\cdot\left\{\left(\mathbf{A}_i + \Delta\tilde{\mathbf{A}}_i + \mathbf{B}_i\mathbf{K}_j\right)\mathbf{x}(k) + \mathbf{B}_i\mathbf{k}_j\zeta(k) + \Delta\tilde{\mathbf{B}}_i\mathbf{u}(k)\right\}\\ &= \left(\mathbf{A}_0 + \Delta\tilde{\mathbf{A}}_0 + \mathbf{B}_0\mathbf{K}_0\right)\mathbf{x}(k) + \mathbf{B}_0\mathbf{k}_0\zeta(k) + \Delta\tilde{\mathbf{B}}_0\mathbf{u}(k)\zeta(k)\\ &= \zeta(k-1) + \mathbf{r}(k-1) - \mathbf{C}\mathbf{x}(k-1)\end{aligned} \tag{44}$$

where $\mathbf{A}_0 = \sum_{i=1}^{M}\bar{\mu}_i\mathbf{A}_i$, $\Delta\mathbf{A}_0 = \sum_{i=1}^{M}\bar{\mu}_i\Delta\mathbf{A}_i$, $\mathbf{B}_0 = \sum_{i=1}^{M}\bar{\mu}_i\mathbf{B}_i$, $\Delta\mathbf{B}_0 = \sum_{i=1}^{M}\bar{\mu}_i\Delta\mathbf{B}_i$, $\mathbf{K}_0 = \sum_{j=1}^{N}\bar{v}_j\mathbf{K}_j$ and $\mathbf{k}_0 = \sum_{j=1}^{N}\bar{v}_j\mathbf{k}_j$.

With the following vectors and matrices defined: $\mathbf{z}(k) = [\mathbf{x}(k)\ \ \zeta(k)]^T$, $\mathbf{K} = [\mathbf{K}_0\ \ \mathbf{k}_0]$, $\mathbf{A} = \begin{bmatrix}\mathbf{A}_0 & 0\\ -\mathbf{C} & \mathbf{I}\end{bmatrix}$, $\Delta\tilde{\mathbf{A}} = \begin{bmatrix}\Delta\tilde{\mathbf{A}}_0 & 0\\ -\mathbf{C} & \mathbf{I}\end{bmatrix}$, $\mathbf{B} = [\mathbf{B}_0\ \ 0]^T$ and $\Delta\tilde{\mathbf{B}} = [\Delta\tilde{\mathbf{B}}_0\ \ 0]^T$, the closed loop system can be rewritten as

$$\tilde{\mathbf{z}}(k+1) = \left(\mathbf{A} + \Delta\tilde{\mathbf{A}} + (\mathbf{B} + \Delta\tilde{\mathbf{B}})\mathbf{K}\right)\mathbf{z}(k) \tag{45}$$

where $\tilde{\mathbf{z}}(k+1)$is the predicted interval value of the state variable vector $\mathbf{z}$.

The following lemma is an expansion of the lemma provided in (Wang et al., 2014), in which the positive constant α is replaced by a positive definite matrix $\mathbf{Z}$.

Lemma 1. *Given matrices $\mathbf{E}$, $\mathbf{F}$ and a positive definite matrix $\mathbf{Z}$, the following inequality can be obtained:*

$$\begin{bmatrix}0 & \mathbf{E}^T\mathbf{F}\\ \mathbf{F}^T\mathbf{E} & 0\end{bmatrix} \le \begin{bmatrix}\mathbf{E}^T\mathbf{Z}\mathbf{E} & 0\\ 0 & \mathbf{F}^T\mathbf{Z}^{-1}\mathbf{F}\end{bmatrix} \tag{46}$$

Proof. *See* Appendix A.

Based on Lemma 1 and the coefficient matrices of the type 2 TS fuzzy model, a set of LMI is derived in Theorem 1. The feedback gains of the RTSFC can be found from the solution of the LMI.

Theorem 1. *Given a nonlinear control system approximated by a type-2 TS fuzzy model as described in Eq. (38), which is obtained from a type-2 FBFN system as described in Eq. (18). If there exists a matrix $\mathbf{Y}$, a positive symmetric matrix $\mathbf{Q}$, positive definite diagonal matrices $\mathbf{Z}_{ij}$, a positive constantα, and the following LMI is satisfied:*

$$\begin{bmatrix}-(1-\alpha)\mathbf{Q} & \mathbf{Q}\Delta\mathbf{A}_{im}^T + \mathbf{Y}_j^T\Delta\mathbf{B}_{im}^T & \mathbf{Q}\mathbf{A}_i^T + \mathbf{Y}_j^T\mathbf{B}_i^T\\ \Delta\mathbf{A}_{im}\mathbf{Q} + \Delta\mathbf{B}_{im}\mathbf{Y}_j & -\mathbf{Z}_{ij}^{-1} & 0\\ \mathbf{A}_i\mathbf{Q} + \mathbf{B}_i\mathbf{Y}_j & 0 & -\mathbf{Q} + \mathbf{Z}_{ij}^{-1}\end{bmatrix} \le 0, \quad \text{with}$$
$$i = 1, \ldots M, \quad j = 1, \ldots, N \tag{47}$$

then the system with a robust TS fuzzy controller as described in Eq. (41) with $\bar{\mathbf{K}}_j = [\mathbf{K}_j\ \ \mathbf{k}_j] = \mathbf{Y}_j\mathbf{Q}^{-1}$ is quadratic stable with a convergent rate α.

Proof. Define a Lyapunov function $V(\mathbf{z}(k)) = \mathbf{z}(k)^T\mathbf{P}\mathbf{z}(k)$ where $\mathbf{P}$ is a positive definite matrix. The system is stable with a convergent rate α when

$$\Delta V(\mathbf{z}) + \alpha V(\mathbf{z}) \le 0 \tag{48}$$

which is equivalent to

$$\mathbf{z}^{\mathrm{T}}(k+1)\mathbf{P}\mathbf{z}(k+1)-\mathbf{z}^{\mathrm{T}}(k)\mathbf{P}\mathbf{z}(k)+\alpha\mathbf{z}^{\mathrm{T}}(k)\mathbf{P}\mathbf{z}(k)\le 0 \Leftrightarrow \mathbf{z}^{\mathrm{T}}(k)\Big[\big((\mathbf{A}+\Delta\tilde{\mathbf{A}})^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}(\mathbf{B}+\Delta\tilde{\mathbf{B}})^{\mathrm{T}}\big)\mathbf{P}\big((\mathbf{A}+\Delta\tilde{\mathbf{A}})+(\mathbf{B}+\Delta\tilde{\mathbf{B}})\mathbf{K}\big)-(1-\alpha)\mathbf{P}\Big]\mathbf{z}(k)\le 0 \tag{49}$$

The above inequality can be written in the matrix form as:

$$\begin{bmatrix} -(1-\alpha)\mathbf{P} & (\mathbf{A}+\Delta\tilde{\mathbf{A}})^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}(\mathbf{B}+\Delta\tilde{\mathbf{B}})^{\mathrm{T}} \\ (\mathbf{A}+\Delta\tilde{\mathbf{A}})+(\mathbf{B}+\Delta\tilde{\mathbf{B}})\mathbf{K} & -\mathbf{P}^{-1}\end{bmatrix}\le 0 \tag{50}$$

$$\begin{bmatrix} -(1-\alpha)\mathbf{P} & \mathbf{A}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\mathbf{B}^{\mathrm{T}} \\ \mathbf{A}+\mathbf{B}\mathbf{K} & -\mathbf{P}^{-1}\end{bmatrix}+\begin{bmatrix} 0 & \Delta\tilde{\mathbf{A}}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\Delta\tilde{\mathbf{B}}^{\mathrm{T}} \\ \Delta\tilde{\mathbf{A}}+\Delta\tilde{\mathbf{B}}\mathbf{K} & 0\end{bmatrix}\le 0 \tag{51}$$

By applying Lemma 1, the following can be obtained:

$$\begin{bmatrix} 0 & \Delta\tilde{\mathbf{A}}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\Delta\tilde{\mathbf{B}}^{\mathrm{T}} \\ \Delta\tilde{\mathbf{A}}+\Delta\tilde{\mathbf{B}}\mathbf{K} & 0\end{bmatrix}\le\begin{bmatrix} \big(\Delta\tilde{\mathbf{A}}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\Delta\tilde{\mathbf{B}}^{\mathrm{T}}\big)\mathbf{Z}\big(\Delta\tilde{\mathbf{A}}+\Delta\tilde{\mathbf{B}}\mathbf{K}\big) & 0 \\ 0 & \mathbf{Z}^{-1}\end{bmatrix} \tag{52}$$

where $\mathbf{Z}=\sum_{i=1}^{M}\sum_{j=1}^{N}\bar{\mu}_i\bar{\nu}_j\mathbf{Z}_{ij}$, $\mathbf{Z}_{ij}$ is a positive definite diagonal matrix.

From (52), inequality (51) is satisfied if

$$\begin{bmatrix} -(1-\alpha)\mathbf{P} & \mathbf{A}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\mathbf{B}^{\mathrm{T}} \\ \mathbf{A}+\mathbf{B}\mathbf{K} & -\mathbf{P}^{-1}\end{bmatrix}+\begin{bmatrix} \big(\Delta\tilde{\mathbf{A}}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\Delta\tilde{\mathbf{B}}^{\mathrm{T}}\big)\mathbf{Z}\big(\Delta\tilde{\mathbf{A}}+\Delta\tilde{\mathbf{B}}\mathbf{K}\big) & 0 \\ 0 & \mathbf{Z}^{-1}\end{bmatrix}\le 0 \tag{53}$$

$$\begin{bmatrix} -(1-\alpha)\mathbf{P}+\big(\Delta\tilde{\mathbf{A}}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\Delta\tilde{\mathbf{B}}^{\mathrm{T}}\big)\mathbf{Z}\big(\Delta\tilde{\mathbf{A}}+\Delta\tilde{\mathbf{B}}K\big) & \mathbf{A}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\mathbf{B}^{\mathrm{T}} \\ \mathbf{A}+\mathbf{B}\mathbf{K} & -\mathbf{P}^{-1}+\mathbf{Z}^{-1}\end{bmatrix}\le 0 \tag{54}$$

Since $\mathbf{Z}_{ij}$ is a positive definite diagonal matrix, the following inequality can be obtained:

$$\big(\Delta\tilde{\mathbf{A}}_i^{\mathrm{T}}+\mathbf{K}_j^{\mathrm{T}}\Delta\tilde{\mathbf{B}}_i^{T}\big)\mathbf{Z}_{ij}\big(\Delta\tilde{\mathbf{A}}_i+\Delta\tilde{\mathbf{B}}_i\mathbf{K}_j\big)\le\big(\Delta\mathbf{A}_{im}^{\mathrm{T}}+\mathbf{K}_j^{\mathrm{T}}\Delta\mathbf{B}_{im}^{\mathrm{T}}\big)\mathbf{Z}_{ij}\big(\Delta\mathbf{A}_{im}+\Delta\mathbf{B}_{im}\mathbf{K}_j\big) \tag{55}$$

where $\Delta\mathbf{A}_{im}$ and $\Delta\mathbf{B}_{im}$ can be calculated by Eq. (37). Hence, inequality (54) is satisfied if

$$\begin{bmatrix} -(1-\alpha)\mathbf{P}+\big(\Delta\mathbf{A}_M^{T}+\mathbf{K}^{\mathrm{T}}\Delta\mathbf{B}_M^{\mathrm{T}}\big) & \mathbf{A}^{\mathrm{T}}+\mathbf{K}^{\mathrm{T}}\mathbf{B}^{\mathrm{T}} \\ \mathbf{Z}\big(\Delta\mathbf{A}_M+\Delta\mathbf{B}_M\mathbf{K}\big) & \\ \mathbf{A}+\mathbf{B}\mathbf{K} & -\mathbf{P}^{-1}+\mathbf{Z}^{-1}\end{bmatrix}\le 0 \tag{56}$$

where $\Delta\mathbf{A}_M=\begin{bmatrix}\Delta\mathbf{A}_{0M} & 0\\ -\mathbf{C} & \mathbf{I}\end{bmatrix}$, $\Delta\mathbf{B}_M=\begin{bmatrix}\Delta\mathbf{B}_{0M}\\ 0\end{bmatrix}$ with $\Delta\mathbf{A}_{0M}=\sum_{i=1}^{M}\bar{\mu}_i\Delta\mathbf{A}_{im}$ and $\Delta\mathbf{B}_{0M}=\sum_{i=1}^{M}\bar{\mu}_i\Delta\mathbf{B}_{im}$. By replacing matrix $\mathbf{P}$ by $\mathbf{Q}$ such that $\mathbf{Q}=\mathbf{P}^{-1}$, (56) is equivalent to

$$\begin{bmatrix} -(1-\alpha)\mathbf{Q} & \mathbf{Q}\Delta\mathbf{A}_m^{\mathrm{T}}+\mathbf{Q}\mathbf{K}^{\mathrm{T}}\Delta\mathbf{B}_m^{\mathrm{T}} & \mathbf{Q}\mathbf{A}^{\mathrm{T}}+\mathbf{Q}\mathbf{K}^{\mathrm{T}}\mathbf{B}^{\mathrm{T}} \\ \Delta\mathbf{A}_m\mathbf{Q}+\Delta\mathbf{B}_m\mathbf{K}\mathbf{Q} & -\mathbf{Z}^{-1} & 0 \\ \mathbf{A}\mathbf{Q}+\mathbf{B}\mathbf{K}\mathbf{Q} & 0 & -\mathbf{Q}+\mathbf{Z}^{-1}\end{bmatrix}\le 0 \tag{57}$$

The above inequality can be rewritten as

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\bar{\mu}_i\bar{\nu}_j\begin{bmatrix} -(1-\alpha)\mathbf{Q} & \mathbf{Q}\Delta\mathbf{A}_{im}^{\mathrm{T}}+\mathbf{Q}\bar{\mathbf{K}}_j^{\mathrm{T}}\Delta\mathbf{B}_{im}^{T} & \mathbf{Q}\mathbf{A}_i^{\mathrm{T}}+\mathbf{Q}\bar{\mathbf{K}}_j^{\mathrm{T}}\mathbf{B}_i^{\mathrm{T}} \\ \Delta\mathbf{A}_{im}\mathbf{Q}+\Delta\mathbf{B}_{im}\bar{\mathbf{K}}_j\mathbf{Q} & -\mathbf{Z}_{ij}^{-1} & 0 \\ \mathbf{A}_i\mathbf{Q}+\mathbf{B}_i\bar{\mathbf{K}}_j\mathbf{Q} & 0 & -\mathbf{Q}+\mathbf{Z}_{ij}^{-1}\end{bmatrix}\le 0 \tag{58}$$

$$\sum_{i=1}^{M}\sum_{j=1}^{N}\bar{\mu}_i\bar{\nu}_j\begin{bmatrix} -(1-\alpha)\mathbf{Q} & \mathbf{Q}\Delta\mathbf{A}_{im}^{\mathrm{T}}+\mathbf{Y}_j^{\mathrm{T}}\Delta\mathbf{B}_{im}^{\mathrm{T}} & \mathbf{Q}\mathbf{A}_i^{\mathrm{T}}+\mathbf{Y}_j^{\mathrm{T}}\mathbf{B}_i^{\mathrm{T}} \\ \Delta\mathbf{A}_{im}\mathbf{Q}+\Delta\mathbf{B}_{im}\mathbf{Y}_j & -\mathbf{Z}_{ij}^{-1} & 0 \\ \mathbf{A}_i\mathbf{Q}+\mathbf{B}_i\mathbf{Y}_j & 0 & -\mathbf{Q}+\mathbf{Z}_{ij}^{-1}\end{bmatrix}\le 0 \tag{59}$$

where $\mathbf{Y}_j=\bar{\mathbf{K}}_j\mathbf{Q}$, $\bar{\mathbf{K}}_j=[\mathbf{K}_j\ \ \mathbf{k}_j]$. Since $\bar{\mu}_i\bar{\nu}_j\ge 0$, the above inequality is satisfied if each term under the summation is negative semi definite. Hence, the theorem is proven. □

Theorem 1provides a method to obtain a robust TS fuzzy controller that not only can guarantee the system stability but also can achieve good transient performance. The designer can use the convergent rate to adjust how fast the system converges to steady state values. Since the LMI set does not depend on the uncertainty norm but on the linear coefficient matrices of the local linear systems that maximize the Lyapunov function, the stability conditions provided in this paper are much more relaxed than other robust controller's conditions that are based on normed bounded uncertainties. The result is a robust TS controller that can achieve performance as good as a TS controller designed for a system without uncertainty.

5. Simulation results on an electrohydraulic actuator

In this section, performance comparisons on an electrohydraulic actuator (EHA) between the RTSFC, the robust sliding mode controller (Lin et al., 2013) and the H∞ sliding mode controller (Zhang et al., 2014) are presented. The electrohydraulic actuator is driven by a bidirectional fixed displacement gear pump. A special symmetrical actuator is connected with the load and the motion of the load is controlled by varying the speed of the electric motor. In (Wang et al., 2008), a nonlinear model of the hydraulic part of the EHA system was developed as follows:

$$\begin{aligned} x_1(k+1)&=x_1(k)+Tx_2(k)+Tw_1(k)\\ x_2(k+1)&=x_2(k)+Tx_3(k)+Tw_2(k)\\ x_3(k+1)&=\left[1-T\left(\frac{a_2V_0+M\beta_eC_T}{MV_0}\right)\right]x_3(k)-T\frac{(2A_p^2+a_2C_T)\beta_e}{MV_0}x_2(k)\\ &\quad-T\frac{2a_1V_0x_2(k)x_3(k)}{MV_0}\operatorname{sgn}(x_2(k))\\ &\quad-T\frac{\beta_cC_T[a_1(x_2(k))^2+a_3]}{MV_0}\operatorname{sgn}(x_2(k))+T\frac{2A_pD_p\beta_e}{MV_0}u(k)+Tw_3(k)\\ y(k+1)&=x_1(k+1)\end{aligned} \tag{60}$$

where x_1, x_2 and x_3 are the position (m), velocity (m/s) and acceleration of the load (m/s^2), respectively; $u(k)$ represents the rotation speed of the bidirectional hydraulic pump (rpm), which is also the control signal of the system. Other parameters can be found in Table 1.

The uncertainties of the EHA are introduced by time-varying friction effects, which are included in the variations of the coefficients of the nonlinear actuator friction a_1, a_2 and a_3 (Lin et al.,

Table 1
System parameters (Lin et al., 2013; Wang et al., 2008; Zhang et al., 2014).

Symbol	Name	Value
M	Mass of the load	20 kg
A_p	Pressure area in the symmetric actuator	5.05×10^{-4}m^2
D_p	Pump displacement	1.6925×10^{-7}m^3/rad
β_e	Bulk modulus of the hydraulic oil	2.1×10^8Pa
C_T	Lumped leakage coefficient	5×10^{-13}m^3/s·Pa
V_0	Mean volume of the hydraulic actuator	6.85×10^{-5}m^3
ω_1, ω_2, ω_3	Lumped system noises and disturbances	0.01×10^{-3}m

Table 2
Training cases.

Case	Δa_1	Δa_2	Δa_3
1	0.1	0.1	0.1
2	0	0.1	0

2013):

$$a_1 \in \left[a_{1o} - \Delta a_1 \cdot a_{1o} \quad a_{1o} + \Delta a_1 \cdot a_{1o}\right]$$
$$a_2 \in \left[a_{2o} - \Delta a_2 \cdot a_{2o} \quad a_{2o} + \Delta a_2 \cdot a_{2o}\right]$$
$$a_3 \in \left[a_{3o} - \Delta a_3 \cdot a_{3o} \quad a_{3o} + \Delta a_3 \cdot a_{3o}\right] \tag{61}$$

with $a_{1o} = 2.1 \times 10^4$, $a_{2o} = -1450$, $a_{3o} = 46$.

Based on the experimental data, Lin et al. (2013) have shown that the output of the systems lies within the ten-percent variation of the time-varying friction coefficients (a_1, a_2 and a_3). By using the same amount of uncertainties to construct the type-2 FBFN model and design the controller, the performance comparison of the controllers could be made. It is noted that the type-2 FBFN is designed solely based on the input and output data, not on the structure of the uncertainties in the system model. Hence, the model can be applied to other systems with uncertainty or unknown dynamics.

A type-2 FBFN model is used to approximate the state variable x_3 of the nonlinear system. The structure of rule j of the FBFN has the following form:

$$R^j: \text{ IF } x_1(k) \text{ is } X_1^j \text{ and } x_2(k) \text{ is } X_2^j \text{ and } x_3(k) \text{ is } X_3^j \text{ and } u(k) \text{ is } U^j$$
$$\text{THEN } \tilde{x}_3(k+1) = \tilde{G}^j \tag{62}$$

where $\tilde{x}_3(k+1)$ is the predicted interval value of the state variable vector x_3. $\tilde{G}^j$ is an interval type-2 fuzzy set with its centroid $\tilde{w}^j$ as an interval set: $\tilde{w}_p^j = [w_l^j, w_r^j]$.

In order to evaluate the performances of the type-2 FBFN for capturing the uncertainties of the data, the type-2 FBFN is trained with the training data generated from the nonlinear system, then comparisons between the outputs of the type-2 FBFN and the nonlinear system are conducted. During the data generation process, the uncertain parameters in the nonlinear model are assigned with random values within the bounded ranges. In this work, the type-2 FBFN model was obtained two times from the same nonlinear model with different amounts of uncertainties represented by the nonlinear friction coefficients a_1, a_2 and a_3. It has been shown that 10% variations of the parameters a_1, a_2 and a_3 can reasonably capture the real friction in the actual system (Lin et al., 2013). For each training data, the parameters a_1, a_2 and a_3 were chosen as random numbers within the lower and upper bounds as shown in Eq. (61). The values of Δa_1, Δa_2, Δa_3 can be found in Table 2.

The training of the type-2 FBFNs took about 25 hours on a computer with one 800 MHz AMD CPU core. However, the training only needs to be done one time since it can capture the dynamics of the system under the entire operating condition. Fig. 2 shows the non-dimensional error indices (NDEI) during training in two cases. The figure shows that the errors observed during the training processes approach steady state values as the number of hidden nodes is increased. Fig. 3 shows the system responses of the nominal nonlinear system when the input is constant. Figs. 4 and 5 show the response comparison between the type-2 FBFN and the uncertain nonlinear model under two uncertain conditions and input values. It can be seen from the results that the type-2 FBFN models are able to capture all the uncertainties of the nonlinear system very "tightly". The deviations from nominal responses of the type-2 FBFN are also very small, which proves that the type-2 FBFN can approximate accurately the nonlinear system.

From the type-2 FBFN, a type-2 TS fuzzy model was obtained by using the procedure as described in Section 3. The type-2 TS fuzzy model has fours rules in which each rule has the following

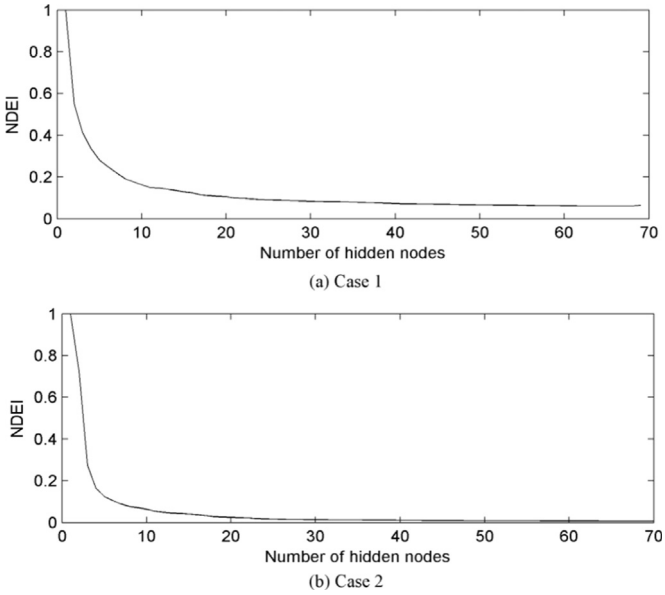

Fig. 2. NDEI during training of type 2 FBFNs.

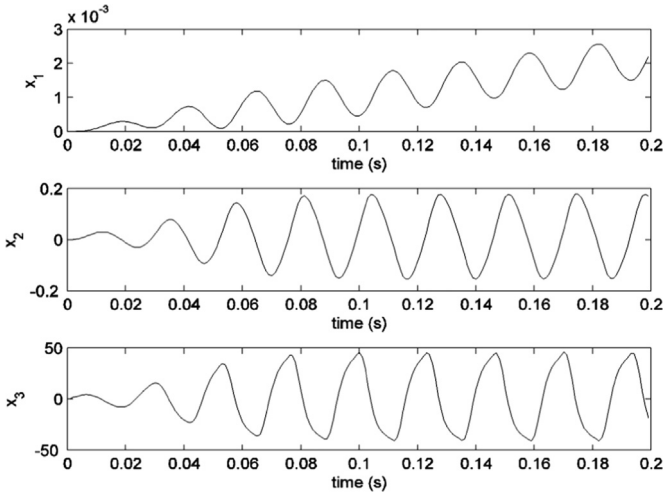

Fig. 3. Nominal system responses (u=30 rpm).

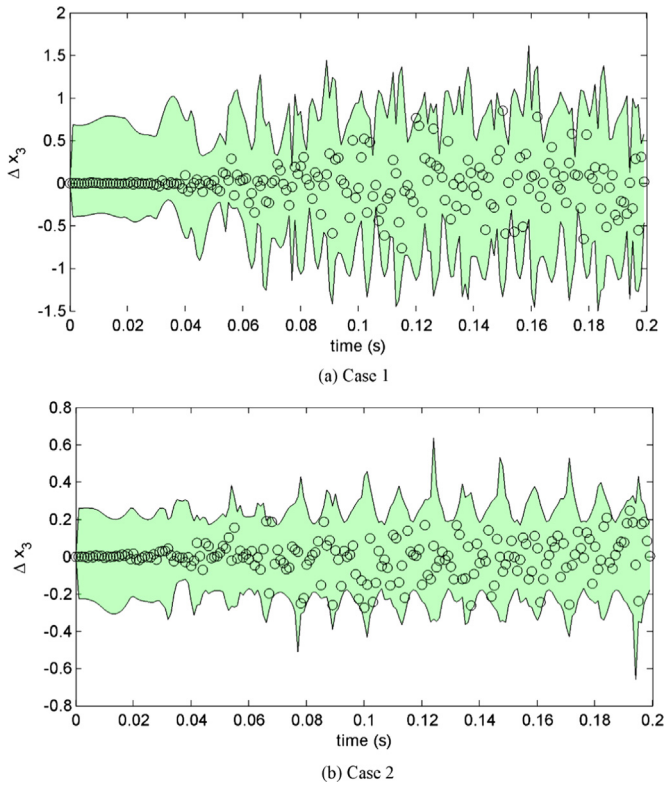

Fig. 4. Deviations from nominal responses with $u=10$ rpm, shaded areas indicate the interval output deviation of the type-2 FBFN model, circle markers represent sampling data measured from the responses of the uncertain nonlinear system.

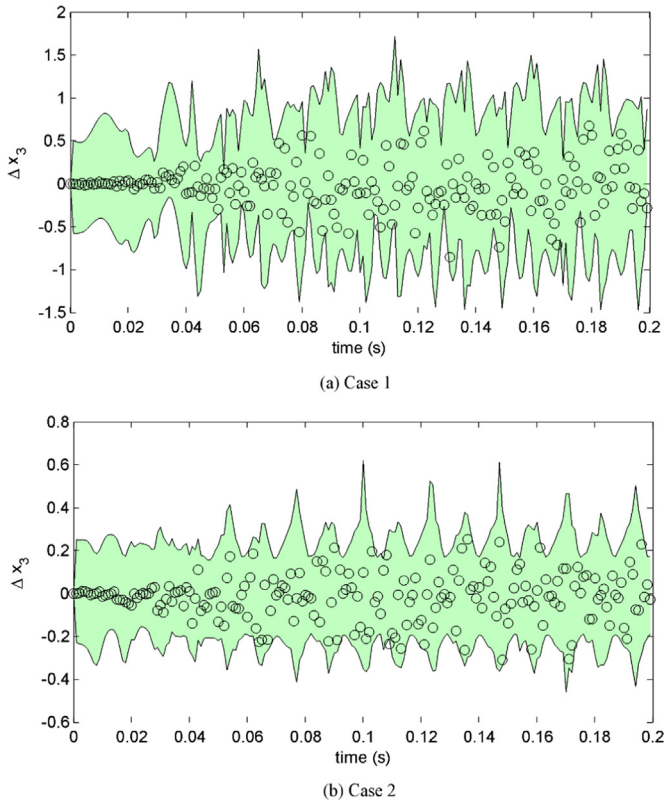

Fig. 5. Deviations from nominal responses with $u=30$ rpm, shaded areas indicate the interval output deviation of the type-2 FBFN model, circle markers represent sampling data measured from the responses of the uncertain nonlinear system.

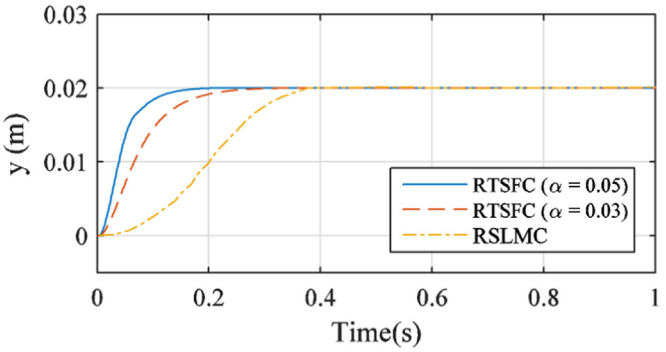

Fig. 6. System response comparisons with a constant reference signal ($r=0.02$ m) between the RTSFC and the RSLMC.

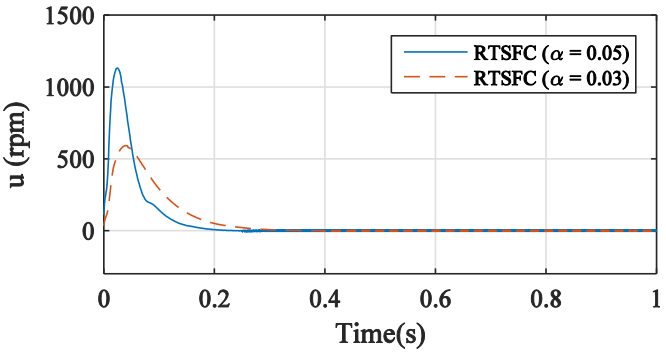

Fig. 7. Control inputs from the RSTSFC under different convergence rate.

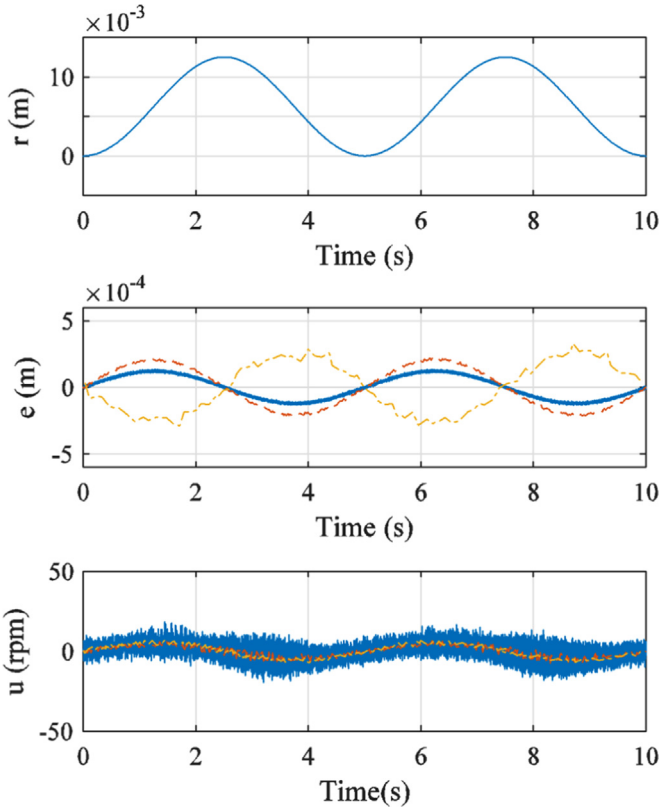

Fig. 8. System response comparisons between the RTSFC and the RH∞SLMC with a sinusoidal reference signal (solid: RTSFC $\alpha=0.2$, dash: RTSFC $\alpha=0.1$, dash-dot: RH∞SLMC).

Table 3
Comparison of mean absolute errors between the RTSFC and the RHSLMC under sinusoidal reference signal.

Controller	**Mean absolute error (m)**
RTSFC α=0.2	7.7352e−05
RTSFC α=0.1	1.3222e−04
RH∞SLMC	1.6707e−04

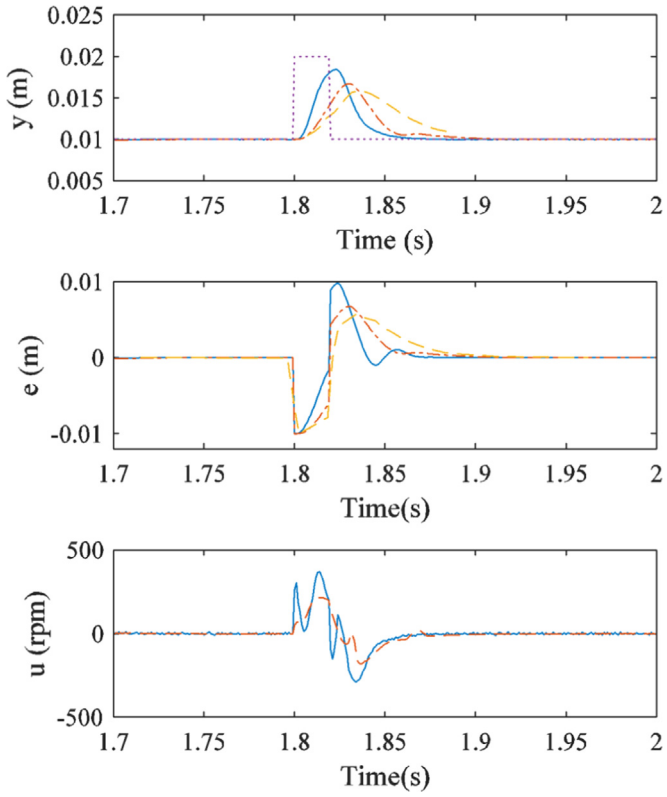

Fig. 9. System response comparisons between the RTSFC and the RH∞SLMC with a spike reference signal (solid: RTSFC α=0.2, dash-dot: RTSFC α=0.1, dash: RH∞SLMC, dot: reference signal).

form:

$$R^i:\ \text{IF } x_2(k) \text{ is } X_2^i \text{ and } x_3(k) \text{ is } X_3^i$$
$$\text{THEN } \tilde{\mathbf{x}}(k+1) = \tilde{\mathbf{A}}_i\mathbf{x}(k) + \tilde{\mathbf{B}}_i\mathbf{u}(k) \tag{63}$$

where $\tilde{\mathbf{x}}(k+1)$ is the predicted interval value of the state variable vector $\mathbf{x}$. The centers of the fuzzy sets X_2^i and X_3^i are chosen as follows:

$$c_{X_2^1} = c_{X_2^2} = -0.015 \text{ m/s}, \quad c_{X_2^3} = c_{X_2^4} = 0.015 \text{ m/s}$$
$$c_{X_3^1} = c_{X_3^3} = -0.015 \text{ m/s}, \quad c_{X_3^2} = c_{X_3^4} = 0.015 \text{ m/s} \tag{64}$$

The minimum and maximum values of matrices $\tilde{\mathbf{A}}_i$ and $\tilde{\mathbf{B}}_i$ are given in Appendix B.

By solving the LMI given in Theorem 1, a robust TS fuzzy controller (RTSFC) which has fours rules can be found. Each rule of the controller has the following form:

Rule$j$: IF $x_2(k)$ is X_2^i and $x_3(k)$ is X_3^i

$$\text{THEN } \mathbf{u}(k) = \mathbf{K}_j\mathbf{x}(k) + \mathbf{k}_j\boldsymbol{\zeta}(k),$$
$$\text{where } \boldsymbol{\zeta}(k) = \boldsymbol{\zeta}(k-1) + \mathbf{e}(k-1), \quad \mathbf{e}(k) = \mathbf{r}(k) - \mathbf{C}\mathbf{x}(k) \tag{65}$$

where the feedback gains of each rule for three different convergent values are given in Appendix C.

To investigate the performances of the RTSFC when implemented on the hydraulic actuator, simulations were conducted in the MATLAB/SIMULINK environment. The computation time to calculate the output of the RTSFC when using the DELL Optilex 960 PC is 0.01 ms. Hence, the RTSFC is very suitable for many real time applications with small sampling time.

Fig. 6 shows the system responses of the hydraulic actuator with the robust sliding mode controller (RSMC) (Lin et al., 2013) and the RTSFC with two different convergent rates used. The objective of the controllers in this simulation is to drive the output from 0 to 0.02 m. The results show that the higher the convergent rate, the faster responses that the RTSFC can achieve. In the first case ($\alpha = 0.1$), the settling time is less than 0.05 s while in the second case ($\alpha = 0.1$), the settling time is about 0.08 s. The control efforts of the RTSFC are shown in Fig. 7.

Fig. 8 shows the system response comparisons between the RTSFC and the robust H∞ sliding mode controller (RH∞SMC) (Zhang et al., 2014) with a sinusoidal reference signal under system noises and disturbances (ω_1, ω_2, ω_3). The values of ω_1, ω_2, ω_3 are shown in Table 1. The mean absolute errors between the controllers' responses and the reference signals are shown in Table 3. From the results, it can be seen that the RTSFC with a convergent rate $\alpha = 0.2$ can reduce the steady state error by almost 50 percent compared to the RH∞SMC.

Fig. 9 shows the system response comparisons between the RTSFC and the robust H∞ sliding mode controller (RH∞SMC) (Zhang et al., 2014) with a spike reference signal under lump system noises and disturbances (Table 1). From the results, it can be seen that the RTSFC can follow the reference signal better than RH∞SMC with very small transient time. The control efforts of the RTSFC can also be found in Fig. 9.

6. Conclusion

A new method of training an interval type-2 FBFN was presented. The antecedents of the FBFN are obtained by using the adaptive least square with the genetic algorithm method, while the interval values of the consequents are obtained by the active set method. Moreover, a new technique was proposed to convert the interval type-2 FBFN to an interval type-2 TS fuzzy model. Based on the proposed methods, a robust controller was designed based on a set of linear matrix inequalities that represent a relaxed stability condition of the closed loop system. The convergence rate allows the controller to be more flexible. Simulation results on an electrohydraulic actuator demonstrate the robustness and better performance of the proposed controller in comparison with the other robust sliding mode controllers.

Appendix A: Proof of Lemma 1

The lemma can be proven by using the following property of the matrix norm:

$$\left(\mathbf{G}\mathbf{Z}^{1/2} - \mathbf{H}\mathbf{Z}^{-1/2}\right)\left(\mathbf{G}\mathbf{Z}^{1/2} - \mathbf{H}\mathbf{Z}^{-1/2}\right)^{\mathrm{T}} \geq 0 \tag{A.1}$$

where $\mathbf{G} = \begin{bmatrix} \mathbf{E}^T \\ 0 \end{bmatrix}$, $\mathbf{H} = \begin{bmatrix} 0 \\ \mathbf{F}^T \end{bmatrix}$. The above inequality is equivalent to

$$\mathbf{G}\mathbf{H}^T + \mathbf{H}\mathbf{G}^T \le \mathbf{G}\mathbf{Z}\mathbf{G}^T + \mathbf{H}\mathbf{Z}^{-1}\mathbf{H}^T \tag{A.2}$$

or

$$\begin{bmatrix} \mathbf{E}^T \\ 0 \end{bmatrix}\begin{bmatrix} 0 & \mathbf{F}^T \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{F}^T \end{bmatrix}\begin{bmatrix} \mathbf{E}^T & 0 \end{bmatrix} \le \begin{bmatrix} \mathbf{E}^T \\ 0 \end{bmatrix}\mathbf{Z}\begin{bmatrix} \mathbf{E}^T & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{F}^T \end{bmatrix}\mathbf{Z}^{-1}\begin{bmatrix} 0 & \mathbf{F}^T \end{bmatrix} \tag{A.3}$$

which is equivalent to inequality (46). Hence, the lemma is proven.

Appendix B: Type 2 TS fuzzy model coefficient matrices

Rule R^1

$$\mathbf{A}_{1\,\min} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -78.366 & 1.0365 \end{bmatrix}$$

$$\mathbf{A}_{1\,\max} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -72.107 & 1.0447 \end{bmatrix} \tag{B.1}$$

$$\mathbf{B}_{1\,\min} = \begin{bmatrix} 0 & 0 & 0.0245 \end{bmatrix}\ \mathbf{B}_{1\,\max} = \begin{bmatrix} 0 & 0 & 0.0249 \end{bmatrix} \tag{B.2}$$

Rule R^2

$$\mathbf{A}_{2\,\min} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -78.3412 & 1.0365 \end{bmatrix}\ \mathbf{A}_{1\,\max} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -72.0821 & 1.0447 \end{bmatrix} \tag{B.3}$$

$$\mathbf{B}_{2\,\min} = \begin{bmatrix} 0 & 0 & 0.0245 \end{bmatrix}\ \mathbf{B}_{2\,\max} = \begin{bmatrix} 0 & 0 & 0.0249 \end{bmatrix} \tag{B.4}$$

Rule R^3

$$\mathbf{A}_{3\,\min} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -89.0027 & 1.0329 \end{bmatrix}$$

$$\mathbf{A}_{3\,\max} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -85.5089 & 1.0427 \end{bmatrix} \tag{B.5}$$

$$\mathbf{B}_{3\,\min} = \begin{bmatrix} 0 & 0 & 0.0265 \end{bmatrix}\ \mathbf{B}_{3\,\max} = \begin{bmatrix} 0 & 0 & 0.0269 \end{bmatrix} \tag{B.6}$$

Rule R^4

$$\mathbf{A}_{4\,\min} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -89.0252 & 1.0329 \end{bmatrix}$$

$$\mathbf{A}_{4\,\max} = \begin{bmatrix} 1 & 0.001 & 0 \\ 0 & 1 & 0 \\ 0 & -83.5317 & 1.0427 \end{bmatrix} \tag{B.7}$$

$$\mathbf{B}_{4\,\min} = \begin{bmatrix} 0 & 0 & 0.0269 \end{bmatrix}\ \mathbf{B}_{4\,\max} = \begin{bmatrix} 0 & 0 & 0.0265 \end{bmatrix} \tag{B.8}$$

Appendix C: Feedback gains of the RTSFC

See Table C.1,C.4

Table C.1
Feedback gains of the RTSFC for $\alpha = 0.03$.

Rule R^j	$\mathbf{K}_j$	$\mathbf{k}_j$
1	$10^5 \cdot [-1.5068 \ \ -0.0036 \ \ -0.0001]$	$0.0187 \cdot 10^5$
2	$10^5 \cdot [-1.4550 \ \ -0.0018 \ \ -0.0001]$	$0.0180 \cdot 10^5$
3	$10^5 \cdot [-1.3753 \ \ -0.0002 \ \ -0.0001]$	$0.0169 \cdot 10^5$
4	$10^5 \cdot [-1.3486 \ \ -0.0010 \ \ -0.0001]$	$0.0166 \cdot 10^5$

Table C.2
Feedback gains of the RTSFC for $\alpha = 0.05$.

Rule R^j	$\mathbf{K}_j$	$\mathbf{k}_j$
1	$10^5 \cdot [-2.8885 \ \ -0.0092 \ \ -0.0001]$	$0.0600 \cdot 10^5$
2	$10^5 \cdot [-2.7527 \ \ -0.0068 \ \ -0.0001]$	$0.0570 \cdot 10^5$
3	$10^5 \cdot [-2.6014 \ \ -0.0049 \ \ -0.0001]$	$0.0536 \cdot 10^5$
4	$10^5 \cdot [-2.5460 \ \ -0.0035 \ \ -0.0001]$	$0.0524 \cdot 10^5$

Table C.3
Feedback gains of the RTSFC for $\alpha = 0.1$.

Rule R^j	$\mathbf{K}_j$	$\mathbf{k}_j$
1	$10^5 \cdot [-6.9056 \ \ -0.0282 \ \ -0.0002]$	$0.2632 \cdot 10^5$
2	$10^5 \cdot [-6.5493 \ \ -0.0249 \ \ -0.0002]$	$0.2487 \cdot 10^5$
3	$10^5 \cdot [-6.0553 \ \ -0.0213 \ \ -0.0002]$	$0.2275 \cdot 10^5$
4	$10^5 \cdot [-5.9091 \ \ -0.0193 \ \ -0.0002]$	$0.2220 \cdot 10^5$

Table C.4
Feedback gains of the RTSFC for $\alpha = 0.2$.

Rule R^j	$\mathbf{K}_j$	$\mathbf{k}_j$
1	$10^6 \cdot [-2.3785 \ \ -0.0103 \ \ -0.0000]$	$0.1550 \cdot 10^6$
2	$10^6 \cdot [2.8299 \ \ -0.0098 \ \ -0.0000]$	$0.1487 \cdot 10^6$
3	$10^6 \cdot [-2.1143 \ \ -0.0089 \ \ -0.0000]$	$0.1359 \cdot 10^6$
4	$10^6 \cdot [-2.0772 \ \ -0.0086 \ \ -0.0000]$	$0.1335 \cdot 10^6$

References

Baghbani, F., Akbarzadeh-T, M., Akbarzadeh, A., Ghaemi, M., 2016. Robust adaptive mixed H2/H∞ interval type-2 fuzzy control of nonlinear uncertain systems with minimal control effort. Eng. Appl. Artif. Intell. 49, 1–26. http://dx.doi.org/10.1016/j.engappai.2015.12.003.

Chadli, M., Guerra, T.M., 2012. LMI solution for robust static output feedback control of discrete Takagi–Sugeno fuzzy models. IEEE Trans. Fuzzy Syst. 20, 1160–1165. http://dx.doi.org/10.1109/TFUZZ.2012.2196048.

Dantzig, G.B., Orden, A., Wolfe, P., 1955. The generalized simplex method for minimizing a linear form under linear inequality restraints. Pac. J. Math. 5, 183–195. http://dx.doi.org/10.2140/pjm.1955.5-2.

Gao, Q., Zeng, X.-J., Feng, G., Wang, Y., Qiu, J., 2012. T–S-fuzzy-model-based approximation and controller design for general nonlinear systems. IEEE Trans. Syst. Man, Cybern. Part B, Cybern. 42, 1143–1154. http://dx.doi.org/10.1109/TSMCB.2012.2187442.

Gill, P.E., Murray, W., Saunders, M. a, Wright, M.H., 1984. Procedures for optimization problems with a mixture of bounds and general linear constraints. ACM Trans. Math. Softw. 10, 282–298. http://dx.doi.org/10.1145/1271.1276.

Gill, P.E., Murray, W., Wright, M.H., 1991. Numerical Linear Algebra and Optimization vol. 1. Addison Wesley.

Goyal, V., Deolia, V.K., Sharma, T.N., 2015. Robust sliding mode control for nonlinear discrete-time delayed systems based on neural network. Intell. Control. Autom. 06, 75–83. http://dx.doi.org/10.4236/ica.2015.61009.

Hsu, C.-F., Lee, T.-T., Tanaka, K., 2015. Intelligent nonsingular terminal sliding-mode control via perturbed fuzzy neural network. Eng. Appl. Artif. Intell. 45, 339–349. http://dx.doi.org/10.1016/j.engappai.2015.07.014.

Hu, J., Wang, Z., Gao, H., Stergioulas, L.K., 2012. Robust sliding mode control for discrete stochastic systems with mixed time delays, randomly occurring uncertainties, and randomly occurring nonlinearities. IEEE Trans. Ind. Electron. 59, 3008–3015. http://dx.doi.org/10.1109/TIE.2011.2168791.

Jin, X., Shin, Y.C., 2015. Nonlinear discrete time optimal control based on Fuzzy Models. Journal of Intelligent & Fuzzy Systems 29 (2), 647–658. http://dx.doi.org/10.3233/IFS-141376.

Karnik, N.N., Mendel, J.M., Liang, Q., 1999. Type-2 fuzzy logic systems. IEEE Trans. Fuzzy Syst. 7, 643–658. http://dx.doi.org/10.1109/91.811231.

Khalil, H., 2002. Nonlinear Systems. Prentice Hall, New Jersey.

Lee, C.W., Shin, Y.C., 2003. Construction of fuzzy systems using least-squares method and genetic algorithm. Fuzzy Sets Syst. 137, 297–323. http://dx.doi.org/10.1016/S0165-0114(02)00344-5.

Lee, C.W., Shin, Y.C., 2001. Construction of fuzzy basis function networks using adaptive least squares method. In: IFSA World Congress and 20th NAFIPS International Conference. IEEE, Vancouver, BC. pp. 2630–2635. doi:10.1109/NAFIPS.2001.943638.

Lee, H., 2011. Robust adaptive fuzzy control by backstepping for a class of MIMO nonlinear systems. IEEE Trans. Fuzzy Syst. 19, 265–275. http://dx.doi.org/10.1109/TFUZZ.2010.2095859.

Lee, H., Tomizuka, M., 2000. Robust adaptive control using a universal approximator for SISO nonlinear systems. IEEE Trans. Fuzzy Syst. 8, 95–106. http://dx.doi.org/10.1109/91.824777.

Lee, H.J., Park, J.B., Chen, G., 2001. Robust fuzzy control of nonlinear systems with parametric uncertainties. IEEE Trans. Fuzzy Syst. 9, 369–379. http://dx.doi.org/10.1109/91.919258.

Li, Y., Tong, S., Liu, Y., Li, T., 2014. Adaptive fuzzy robust output feedback control of nonlinear systems with unknown dead zones based on a small-gain approach. IEEE Trans. Fuzzy Syst. 22, 164–176. http://dx.doi.org/10.1109/TFUZZ.2013.2249585.

Liang, Q., Mendel, J.M., 2000. Interval type-2 fuzzy logic systems: theory and design. IEEE Trans. Fuzzy Syst. 8, 535–550. http://dx.doi.org/10.1109/91.873577.

Lin, C.-K., 2007. Robust adaptive critic control of nonlinear systems using fuzzy basis function networks: an LMI approach. Inf. Sci. 177, 4934–4946. http://dx.doi.org/10.1016/j.ins.2007.06.017.

Lin, Y., Shi, Y., Burton, R., 2013. Modeling and robust discrete-time sliding-mode control design for a fluid power electrohydraulic actuator (EHA) system. IEEE/ASME Trans. Mechatron. 18, 1–10. http://dx.doi.org/10.1109/TMECH.2011.2160959.

Liu, Z., Wang, F., Zhang, Y., Chen, X., Phillip Chen, C.L., 2014. Adaptive fuzzy output-feedback controller design for nonlinear systems via backstepping and small-gain approach. IEEE Trans. Cybern. 44, 1714–1725. http://dx.doi.org/10.1109/TCYB.2013.2292702.

Mandal, P., Sarkar, B.K., Saha, R., Chatterjee, A., Mookherjee, S., Sanyal, D., 2015. Real-time fuzzy-feedforward controller design by bacterial foraging optimization for an electrohydraulic system. Eng. Appl. Artif. Intell. 45, 168–179. http://dx.doi.org/10.1016/j.engappai.2015.06.018.

MathWorks, 2015. MATLAB optimization toolbox: User's Guide (r2015a) [WWW Document]. URL ⟨http://www.mathworks.com/help/pdf_doc/optim/optim_tb.pdf⟩ (accessed 02.06.15).

Mehrotra, S., 1992. On the implementation of a primal-dual interior point method. SIAM J. Optim. 2, 575–601. http://dx.doi.org/10.1137/0802028.

Méndez, G.M., de los Angeles Hernandez, M., 2009. Hybrid learning for interval type-2 fuzzy logic systems based on orthogonal least-squares and back-propagation methods. Inf. Sci. 179, 2146–2157. http://dx.doi.org/10.1016/j.ins.2008.08.008.

Ngo, P.D., Shin, Y.C., 2015. Gain estimation of nonlinear dynamic systems modeled by an FBFN and the maximum output scaling factor of a self-tuning PI fuzzy controller. Eng. Appl. Artif. Intell. 42, 1–15. http://dx.doi.org/10.1016/j.engappai.2015.03.004.

Rubio-Solis, A., Panoutsos, G., 2015. Interval type-2 radial basis function neural network: a modeling framework. IEEE Trans. Fuzzy Syst. 23, 457–473. http://dx.doi.org/10.1109/TFUZZ.2014.2315656.

Salgado, I., Camacho, O., Yáñez, C., Chairez, I., 2014. Proportional derivative fuzzy control supplied with second order sliding mode differentiation. Eng. Appl. Artif. Intell. 35, 84–94. http://dx.doi.org/10.1016/j.engappai.2014.06.005.

Sato, M., 2009. Robust model-following controller design for LTI systems affected by parametric uncertainties: a design example for aircraft motion. Int. J. Control. 82, 689–704. http://dx.doi.org/10.1080/00207170802225948.

Sloth, C., Esbensen, T., Niss, M.O.K., Stoustrup, J., Odgaard, P.F., 2009. Robust LMI-based control of wind turbines with parametric uncertainties. In: Proceedings of the 18th IEEE International Conference on Control Applications. IEEE, Saint Petersburg, Russia. pp. 776–781. http://dx.doi:10.1109/CCA.2009.5281171.

Tong, S.-C., He, X.-L., Zhang, H.-G., 2009. A combined backstepping and small-gain approach to robust adaptive fuzzy output feedback control. IEEE Trans. Fuzzy Syst. 17, 1059–1069. http://dx.doi.org/10.1109/TFUZZ.2009.2021648.

Wang, L.-X., Mendel, J., 1992. Fuzzy basis functions, universal approximation, and orthogonal least-squares learning. IEEE Trans. Neural Netw. 3, 807–814. http://dx.doi.org/10.1109/72.159070.

Wang, S., Habibi, S., Burton, R., 2008. Sliding mode control for an electrohydraulic actuator system with discontinuous non-linear friction. Proc. Inst. Mech. Eng. Part I J. Syst. Control. Eng. 222, 799–815. http://dx.doi.org/10.1243/09596518JSCE637.

Wang, X., Yaz, E.E., Long, J., 2014. Robust and resilient state-dependent control of discrete-time nonlinear systems with general performance criteria. Syst. Sci. Control. Eng. 2, 48–54. http://dx.doi.org/10.1080/21642583.2013.877858.

Yao, J., Jiao, Z., Ma, D., 2014a. Extended-state-observer-based output feedback nonlinear robust control of hydraulic systems with backstepping. IEEE Trans. Ind. Electron. 61, 6285–6293. http://dx.doi.org/10.1109/TIE.2014.2304912.

Yao, J., Jiao, Z., Ma, D., Yan, L., 2014b. High-accuracy tracking control of hydraulic rotary actuators with modeling uncertainties. IEEE/ASME Trans. Mechatron. 19, 633–641. http://dx.doi.org/10.1109/TMECH.2013.2252360.

Zhang, H., Liu, X., Wang, J., Karimi, H.R., 2014. Robust H∞ sliding mode control with pole placement for a fluid power electrohydraulic actuator (EHA) system. Int. J. Adv. Manuf. Technol. 73, 1095–1104. http://dx.doi.org/10.1007/s00170-014-5910-8.

Zhang, Y., 1998. Solving large-scale linear programs by interior-point methods under the Matlab Environment. Optim. Methods Softw. 10, 1–31. http://dx.doi.org/10.1080/10556789808805699.