*International Journal of Innovative Computing, Information and Control*
Volume **9**, Number **10**, October **2013**

ICIC International ©2013 ISSN 1349-4198
pp. 3915–3928

# FAULT-TOLERANT RECONFIGURABLE CONTROL FOR MIMO SYSTEMS USING ONLINE FUZZY IDENTIFICATION

Ruiyun Qi, Limeng Zhu and Bin Jiang

College of Automation Engineering
Nanjing University of Aeronautics and Astronautics
No. 29, Yudao Street, Nanjing 210016, P. R. China
{ ruiyun.qi; binjiang }@nuaa.edu.cn

Received September 2012; revised January 2013

Abstract. *In this paper, a novel fault-tolerant control scheme is proposed for a class of uncertain multiple-input-multiple-output (MIMO) nonlinear systems based on online fuzzy clustering and identification. The unknown nonlinear function of the system is identified online using an evolving Takagi-Sugeno (T-S) fuzzy model and the fault-tolerant control law is designed based on the identified fuzzy model. The proposed method has the following features: (i) both the structure and the parameters of the T-S fuzzy model can evolve online, which makes it capable of representing more parameter and structure uncertainties of the nonlinear system; (ii) an online fuzzy clustering algorithm is employed to determine the generation of a new cluster center (new rule), which also serves as a warning signal of detecting a fault; (iii) a self-structuring fault-tolerant controller is constructed based on the online identified fuzzy model, together with a compensation controller, to guarantee the closed-loop stability under parameter variations and faults. The proposed fault-tolerant control scheme does not rely on a specially designed fault diagnosis module. Simulation studies on an inverted pendulum example have verified the effectiveness of the proposed control scheme and demonstrated the proposed control scheme has the capability to achieve desired tracking performance when there exist parameter changes or faults and the generation of new rules can alert the occurrence of faults in the system.*
**Keywords:** Fault-tolerant control, Online fuzzy identification, Fuzzy clustering, Fault-detection

1. **Introduction.** In recent two decades, fault diagnosis and fault-tolerant control have attracted more and more attention due to increasing demand for reliability and safety of systems. There have been rapidly growing researches in this area and fruitful results have been achieved to protect the system from the effects of system faults, ensuring system stability and regaining desired system performance [1].

Fault-tolerant control design is not an independent control design branch and it utilizes various advanced control theory and techniques, such as robust control, adaptive control, sliding-mode control. Some faults can be treated as some kinds of system uncertainties. Robust control theory can be applied to make the system achieve robustness to faults, which are also called reliable control designs [2]. However, the magnitude of faults is usually higher than disturbances and other system uncertainties. With fixed structure and parameters, the ability of such fault-tolerant controllers is limited especially when it encounters complex fault situations. A fault-tolerant controller that has the ability to adjust its parameters and/or structures is called a reconfigurable controller which is more powerful to deal with faults. There are different ways to design a reconfigurable control system, including employing a fault detection and identification unit in the control system to provide useful fault information for reconfigurable control [3-5], reshaping the output reference and reallocating control [6], sliding-mode control [7, 8], adaptive control [9, 10], and intelligent control [11].

In recent years, approximation based control design for nonlinear system has become a very active area for both theoretical and practical reasons [9, 12, 18, 19]. Most plants are actually nonlinear systems and their nonlinear dynamics are usually unknown or only partially known. Moreover, the occurrence of faults may often lead to more nonlinearities and push the plant from a relatively linear operating point to a more nonlinear region [11]. Neural networks and fuzzy systems have emerged as a powerful tool to represent system nonlinearities due to their proved universal approximation ability to approximate any smooth nonlinear functions with arbitrary accuracy [12, 13]. However, there are relatively fewer researches considering online structure identification in the closed-loop control. Direct adaptive approximation based control schemes are proposed for MIMO nonlinear systems [22] and for SISO nonlinear systems [23]. In [24, 25], self-organizing adaptive fuzzy neural controllers are designed for SISO and MIMO systems, respectively, together with sliding-mode control, to achieve desired tracking performance.

Motivated by the capability of online identification algorithms to capture the changes in system dynamics and provide good evolving approximation for control designs, this paper presents an online identification based reconfigurable control scheme for a class of MIMO nonlinear systems subject to certain system faults. An online clustering based fuzzy identification algorithm is used to generate new cluster centers (correspondingly, new rules) online when it detects significant change happening in the system dynamics. The controller is reconfigurable in the sense that both the structure and parameters of fuzzy systems used in constructing the controller are changed online to accommodate changes in system dynamics due to faults. Such an approach combines the advantages of data-driven online identification methods and model-based adaptive control designs, which provides a flexible controller structure and can effectively deal with uncertainties of large magnitude, including system uncertainties caused by faults.

## 2. System Description and Problem Formulation.

In this section, we describe a basic general architecture of the proposed fault detection and reconfigurable control scheme under consideration in this work. It consists of three parts: the plant, the online identification module and the reconfigurable control module.

**Nominal Plant Dynamics.** To present the main idea without undue complication, we consider the following $n$-th order nonlinear MIMO plant dynamics [19]:

$$\begin{aligned}
\dot{x}_{r_i1} &= x_{r_i2} \\
&\vdots \\
\dot{x}_{r_i(r_i-1)} &= x_{r_ir_i} \\
\dot{x}_{r_ir_i} &= f_i(x) + \sum_{j=1}^{m} g_{ij}(x)u_j + d_i \\
y_i &= x_{r_ii}
\end{aligned} \tag{1}$$

where $r_i$, $i = 1, 2, \dots, m$ are relative degrees, $r_1 + r_2 + \cdots + r_m = n$, $y = [y_1, y_2, \dots, y_m]^T \in R^m$ is a vector of system outputs, $x = [x_{r_11}, \cdots, x_{r_1r_1}, \cdots, x_{r_m1}, \cdots, x_{r_mr_m}]^T \in R^n$, is a vector of system states, $u = [u_1, u_2, \dots, u_m]^T \in R^m$ is a vector of system inputs, and $d_i$, $i = 1, 2, \dots, m$, are external disturbances. $f_i(x)$ and $g_{ij}(x)$, $i, j = 1, 2, \dots, m$, are smooth functions representing the nonlinear dynamics of the plant.

**Remark 2.1.** *The model (1) we consider here is a canonical nonlinear MIMO model which is capable of representing a wide class of real nonlinear MIMO plants such as a* *mass-spring-damper system* [21], *two inverted pendulums connected by a moving spring mounted on two carts* [20], *a two-link robot manipulator* [22].

Introducing $F_0(x) = [f_1(x), \cdots, f_m(x)]^T \in R^m$ and $G(x) = \{G_1(x), \cdots, G_m(x)\} \in R^{m\times m}$ with $G_i(x) = [g_{1i}(x), \cdots, g_{mi}(x)]^T \in R^m$, the system (1) can be formulated into the following form:

$$\begin{aligned} \dot{x} &= Ax + B[F_0(x) + G(x)u + d] \\ y &= Cx, \end{aligned} \tag{2}$$

where $d = [d_1, d_2, \cdots, d_m]^T \in R^m$, $A = \operatorname{diag}\{A_1, \cdots, A_m\} \in R^{n\times n}$, $B = \operatorname{diag}\{B_1, \cdots, B_m\} \in R^{n\times m}$, $C = \operatorname{diag}\{C_1, \cdots, C_m\} \in R^{m\times n}$, and

$$A_i = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{r_i\times r_i}, \quad B_i = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}_{r_i\times 1}, \quad C_i = [\, 1 \quad 0 \quad 0 \quad \cdots \quad 0 \,]_{1\times r_i}. \tag{3}$$

**Plant Dynamics under Faults.** When a fault occurs at uncertain time $t = t_f$, consider the following change of plant dynamics:

$$\begin{aligned} \dot{x} &= Ax + B[F_0(x) + G(x)u + \beta(t - t_f)\rho(\mathbf{x}) + d] \\ y &= Cx, \end{aligned} \tag{4}$$

where $\beta(t-t_f)$ is a step function representing the time profile of faults and $\rho(\mathbf{x})$ represents the abrupt system fault.

**Design Objectives.** In this work, for the system described by (4), we try to achieve two design objectives using both the *model-based* approach and *data-driven* approach: (1) develop a reconfigurable control law which can ensure the closed-loop stability and system performance *without* explicitly knowing the occurrence time and value of the fault; (2) develop a fault detection scheme using the online data collected in the closed-loop system, which can detect the abnormality in the system.

In the above two design objectives, the first one is the most important one since the closed-loop system stability is crucial for safety-critical and performance-critical systems.

## 3. Online Fuzzy Identification Based Fault-Tolerant Control Design.

In this section, we firstly propose the nominal control law for the nominal system (1). Then a fuzzy approximation-based reconfigurable controller is proposed for the system under faults (4).

For the nominal system (1), to ensure its controllability, $G(x)$ is assumed to be nonsingular. Given a bounded smooth reference signal vector $y_r = [y_{r_1}, y_{r_2}, \cdots, y_{r_m}]^T$, we can choose the control law as

$$u = G^{-1}(x)\left[-F_0(x) + y_r^{(r)} + Ke\right] \tag{5}$$

where $e = [e_1, \dot{e}_1, \cdots, e_1^{(r_1-1)}, \cdots, e_m, \dot{e}_m, \cdots, e_m^{(r_m-1)}]^T$, $e_1 = y_{r_1} - y_1, \ldots, e_m = y_{r_m} - y_m$, $K = \operatorname{diag}[K_1, K_2, \cdots, K_m]$, $K_i = [k_{i1}, k_{i2}, \cdots, k_{ir_i}]$. $K$ is chosen to make the matrix $A - BK$ be a Hurwitz matrix.

Substituting (5) into (1) yields the closed-loop system dynamics:

$$\dot{e} = (A - BK)e, \tag{6}$$

which ensures the boundedness of all the closed-loop signals and the asymptotical convergence of the tracking error: $\lim_{t\to\infty} e(t) = 0$.

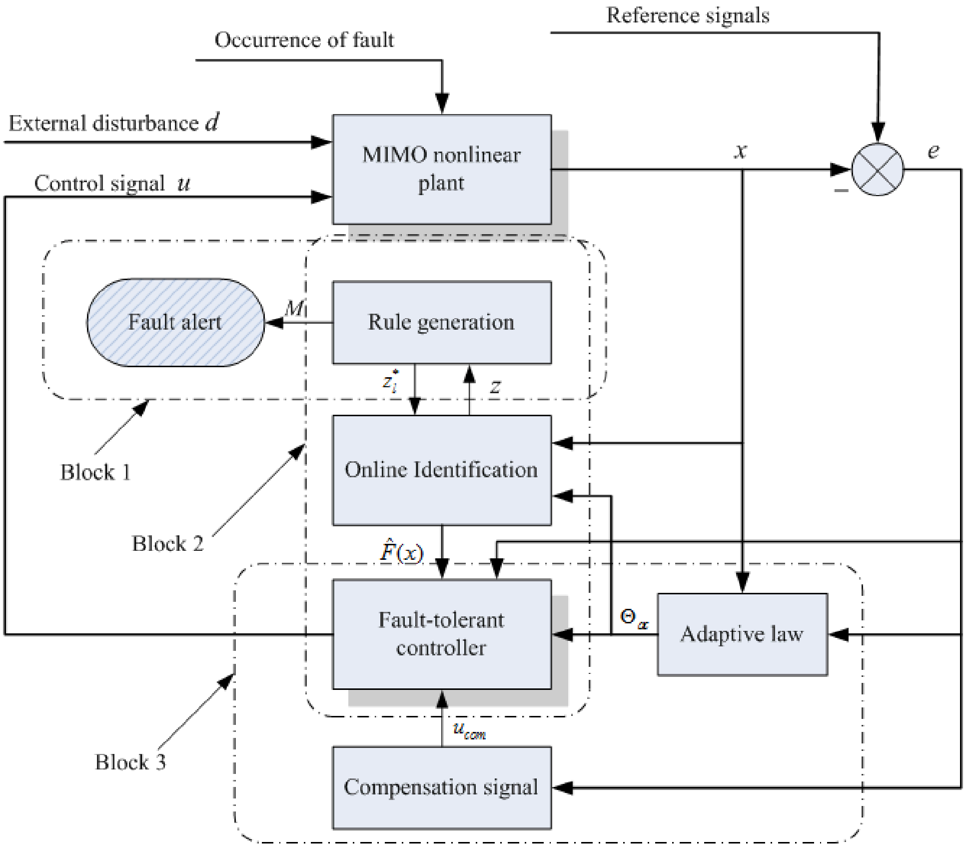


FIGURE 1. Block diagram of the overall control scheme

When faults occur in the plant and the plant dynamics become (4), the nominal control law (5) cannot guarantee the stability and performance of the closed-loop system. Some fault-tolerant mechanisms need to be added.

In this paper, we propose an online fuzzy identification based fault-tolerant control scheme whose overall structure is demonstrated in Figure 1.

In the following sections, we will present the detailed designs for each part in Figure 1.

3.1. **Online fuzzy identification.** In this section, the T-S fuzzy modeling technique with online learning ability is employed to approximate the uncertain nonlinear function of the system. Both the structure and the parameters of the T-S fuzzy model are identified online to accommodate the changes in the system dynamics due to faults.

Defining $F(x) = F_0(x) + \beta(t - t_f)\rho(x)$, the system (4) can be written as

$$\begin{aligned} \dot{x} &= Ax + B[F(x) + G(x)u + d] \\ y &= Cx. \end{aligned} \tag{7}$$

Since there exist uncertain parameters in $F(x)$ and an unexpected fault may also affect $F(x)$, $F(x)$ cannot be directly used for control design. To capture the changing dynamics of $F(x)$, we employ T-S fuzzy modeling technique to identify $F(x)$ online.

**Fuzzy Approximation.** For the $m$ uncertain nonlinear functions $f_j(x)$, $j = 1, 2, \ldots,$ $m$, the following T-S fuzzy model is used with its $i$-th rule represented by:

$$\begin{aligned} R^i: \quad & \text{IF} \quad \xi_1 \text{ is } \aleph_1^i \text{ and } \xi_2 \text{ is } \aleph_2^i \text{ and } \ldots \text{ and } \xi_L \text{ is } \aleph_L^i \\ & \text{THEN} \quad \hat{f}_j^i(x) = \pi_{ij}^T x_e, \quad j = 1, 2, \ldots, m, \end{aligned} \tag{8}$$

where $R^i$, $i = 1, 2, \ldots, M$, denotes the $i$th fuzzy rule defining the $i$th subsystem, $M$ is the number of fuzzy rules, $x_e = [1, x^T]^T \in R^{n+1}$, $\pi_{ij} = [\theta_{ij0}, \theta_{ij1}, \ldots, \theta_{ijn}]^T \in R^{n+1}$, $j = 1, 2, \ldots, m$, are parameters of the $i$th subsystem, and "$\xi_l$ is $\aleph_l^i$", $l = 1, 2, \ldots, L$, is a part of the $i$th fuzzy rule, with the premise variables $\xi = [\xi_1, \ldots, \xi_L]$ being some measurable system signals or their functions and $\aleph_l^i$ being a fuzzy set associated with which there is a membership function $N_l^i(\xi_k)$ to indicate the degree of membership of $\xi_l$ in $N_l^i$. In our paper, we use Gaussian functions as the membership functions:

$$N_l^i(\xi_j) = \exp\left\{-\frac{(\xi_l - c_l^i)^2}{\sigma_l^{i2}}\right\}. \tag{9}$$

Following a standard T-S fuzzy modeling procedure [14], the overall T-S fuzzy model is constructed as

$$\hat{f}_j(x) = \sum_{i=1}^{M} \mu_i \pi_{ij}^T x_e, \tag{10}$$

where $\mu_i$ is the normalized membership function:

$$\mu_i(\xi) = \frac{\lambda_i(\xi)}{\sum_{i=1}^{M} \lambda_i(\xi)}, \quad \lambda_i(\xi) = \prod_{l=1}^{L} N_l^i(\xi_l), \tag{11}$$

such that $\mu_i(\xi) \geq 0$ and $\sum_{i=1}^{M} \mu_i(\xi) = 1$.

Let

$$\begin{aligned}
\Theta_j &= \left[\pi_{1j}^T, \pi_{2j}^T, \ldots, \pi_{Mj}^T\right]^T \in R^{(n+1)M} \\
\Theta &= \left[\Theta_1^T, \Theta_2^T, \ldots, \Theta_m^T\right]^T \in R^{(n+1)Mm} \\
\phi_j(x) &= \left[\mu_1 x_e^T, \mu_2 x_e^T, \ldots, \mu_M x_e^T\right]^T \in R^{(n+1)M} \\
\Phi(x) &= \operatorname{diag}\{\phi_1(x), \phi_2(x), \ldots, \phi_m(x)\} \in R^{(n+1)Mm \times m},
\end{aligned} \tag{12}$$

and we have the following approximation function $\hat{F}(x) = [\hat{f}_1(x), \hat{f}_2(x), \ldots, \hat{f}_m(x)]^T$ for $F(x)$:

$$\hat{F}(x) = \Phi(x)\Theta. \tag{13}$$

In the literature, usually adaptive laws for parameter learning can be developed to tuning the parameters in (13) online with the basis functions $\Phi(x)$ fixed. The basis functions are assumed known or identified through offline training [11]. For a normal system, it is feasible to determine the basis functions in advance. However, when a fault occurs in the system, not only the system parameters but also the system structure may undergo changes. In such a case, it is not adequate to adjust only the parameters of the approximators. In a T-S fuzzy model, each rule represents locally the system dynamics defined in a local operating region. *When a fault occurs, it may be necessary to add some new rules to describe some new system dynamics caused by the fault.* Therefore, different from most adaptive approximation based control design for normal systems without changing the basis functions, we *update both the structure* (i.e., the number of fuzzy rules and membership functions which determines the basis function) *and the parameters* of the T-S fuzzy model to accommodate the effects of the fault on the system dynamics.

**Online Structure Identification and Fault Detection.** Our online structure mechanism serves for two purposes. The main purpose is to identify the structure of the approximation model (10) and the second purpose is to detect the occurrence of faults.

Due to the limitation of offline training data and designers' knowledge and experience, it is difficult to determine an optimal number of fuzzy rules. It is pointed out in [26] that in reality many regimes and process states cannot be practically included into the training data set (such as faulty process behavior), but states close to them could well appear during the process run. Hence, the online data can provide us with more information about the actual system dynamics and behavior. In a T-S fuzzy model, each rule represents the local system dynamics in a subregion. When there is no significant change in the system, the online data will appear within or close to those local regions. When a significant change happens to the system, there may appear a new region of data space that has not been covered by the previous training data.

Next, we will describe the rule generation mechanism. In (8), the premise variables $\xi_k$, $k = 1, 2, \ldots, L$ are used to define the subregion of each fuzzy rule. The premise variables can be selected from system states which play a key role in reflecting system behavior. There are numerous ways for selecting the premise variables, including using expert knowledge and learning from system generated data using a search tree or genetic algorithm [17]. Once the premise variables are determined, the next task is to determine the number of fuzzy rules and the locations of the membership functions. Various approaches have been proposed to identify the number of fuzzy rules, such as mosaic methods, genetic algorithm-based approaches and fuzzy clustering. Compared with other approaches, fuzzy clustering based fuzzy rule identification methods have the advantages that the information of the cluster centers can be directly used to form fuzzy rules. For a particular system to be modeled, each IF-THEN rule specifies an area exemplified by a typical point in the graph of control function that can be identified with the Cartesian product of the membership functions modeling the linguistic terms [16]. Other points around the typical point are less representative and have decreasing membership degrees in that area with increasing distance from the typical point. Similarly, a cluster prototype represents the center of a fuzzy cluster and the data neighboring the prototype have decreasing membership degree with increasing distance to the prototype.

In our approach to identifying the T-S fuzzy model online, the number of fuzzy rules and the parameters of membership functions are determined through online fuzzy clustering algorithms [27] while the consequent parameters are updated through adaptive laws (see Section 3.2) derived from stability analysis using a Lyapunov function to ensure the closed-loop boundedness.

**Remark 3.1.** *The appearance of a cluster center (a new rule) can be viewed as a reaction to a dramatic change of the system behavior. Such a dramatic change could be caused by a new operating mode of the system or the occurrence of a fault. Usually, it can be judged by operators whether the system enters a new operating mode which can be identified by observing the system behavior and some key system states. If it is not the case, the generation of a new rule can be seen as a sign of detecting a fault, at least a warning of abnormal system behavior.*

3.2. **Reconfigurable control design.** Based on the online identified T-S fuzzy model which approximates of the system dynamics under faults, the nominal control law (5) can be replaced by the fault-tolerant reconfigurable control law:

$$u = G^{-1}(x)(-\hat{F}(x) + y_r^{(r)} + k^T e - u_{com}), \tag{14}$$

where $y_r^{(r)} = [y_{r_1}^{(r_1)}, y_{r_2}^{(r_2)}, \cdots, y_{r_m}^{(r_m)}]^T$, $u_{com}$ is a compensation signal which will be designed later.

Substituting (14) into (7) yields

$$\dot{e} = (A - BK)e + B[\hat{F}(x) - F(x)] + Bu_{com} - Bd. \tag{15}$$

We assume there exist optimal parameters $\Theta^*$ and the optimal number of rules $N^*$ satisfying

$$\{\Theta^*, N^*\} \stackrel{\Delta}{=} \arg \min_{\Theta \in \Omega_\Theta, N \in \Omega_N} \left[ \sup_{x \in U} \left\| \hat{F}(x|\Theta, N) - F(x) \right\| \right], \tag{16}$$

where

$$\Omega_\Theta = \{\Theta| \, \|\Theta\| \leq M_\Theta\}, \quad \Omega_N = \{N|N \leq N_{rule}\} \tag{17}$$

are the constraint sets for $\Theta$ and $\Omega_N$, respectively. $N_{rule}$ is the upper bound of the required number of rules that can achieve the desired approximation accuracy. In practice, it can be estimated through experience or fuzzy clustering algorithms. A most conservative choice is the number that does not make the control signal calculation exceed the hardware computation capability. With the definition on $\Theta^*$ and $N^*$, we have the minimum fuzzy approximation error:

$$w_e = \hat{F}(x|\Theta^*, N^*) - F(x). \tag{18}$$

Since the fuzzy rule base is not fixed in our proposed control scheme, both the number of rules and parameters change during the system operation. Let $N$, $N \leq N_{rule}$, be the final rule number, $\Theta_j = [\pi_{j1}^T, \pi_{j2}^T, \cdots, \pi_{jN}^T]^T$ the final parameter vector, and $\phi_j(x) = [\mu_1 x_e^T, \mu_2 x_e^T, \cdots, \mu_N x_e^T]^T$ the final regressor. When the number of rule $M \leq N$, the parameters in $\Theta_j$ can be divided into two groups $\Theta_j = [\Theta_{ac}^{jT}, \Theta_{in}^{jT}]^T$, where $\Theta_{ac}^j = [\pi_{j1}^T, \pi_{j2}^T, \cdots, \pi_{jM}^T]^T \in R^{M(n+1)}$ is the activated parameter vector and $\Theta_{in}^j = [\pi_{j(M+1)}^T, \pi_{j(M+2)}^T, \cdots, \pi_{jN}^T]^T \in R^{(N-M)(n+1)}$ is the inactivated parameter vector. Similarly, the regressor $\phi_j(x)$ can be divided into two parts:

$$\phi_{ac}^j(x) = [\mu_1 x_e^T, \mu_2 x_e^T, \cdots, \mu_M x_e^T]^T, \quad \phi_{in}^j(x) = [\mu_{M+1} x_e^T, \mu_{M+2} x_e^T, \cdots, \mu_N x_e^T]^T. \tag{19}$$

With the above definition, we have

$$\hat{F}(x|\Theta, N) = \Phi_{ac}(x)\Theta_{ac} + \Phi_{in}(x)\Theta_{in}, \tag{20}$$

where

$$\begin{aligned} \Phi_{ac}(x) &= \mathrm{diag}\{\phi_{ac}^1(x), \phi_{ac}^2(x), \ldots, \phi_{ac}^m(x)\}, \quad \Theta_{ac} = [\Theta_{ac}^{1T}, \Theta_{ac}^{2T}, \ldots, \Theta_{ac}^{mT}]^T \\ \Phi_{in}(x) &= \mathrm{diag}\{\phi_{in}^1(x), \phi_{in}^2(x), \ldots, \phi_{in}^m(x)\}, \quad \Theta_{in} = [\Theta_{in}^{1T}, \Theta_{in}^{2T}, \ldots, \Theta_{in}^{mT}]^T. \end{aligned} \tag{21}$$

Let $\hat{F}(x|\Theta_{ac}^*, M) = \Phi_{ac}(x)\Theta_{ac}^*$, where $\Theta_{ac}^* = [\Theta_{ac}^{1*T}, \Theta_{ac}^{2*T}, \ldots, \Theta_{ac}^{m*T}]^T$, the error Equation (15) can be formulated as

$$\begin{aligned} \dot{e} &= (A - BK)e + B[\hat{F}(x|\Theta_{ac}, M) - \hat{F}(x|\Theta_{ac}^*, M) + \hat{F}(x|\Theta_{ac}^*, M) - \hat{F}(x|\Theta^*, N^*) \\ &\quad + \hat{F}(x|\Theta^*, N^*) - F(x)] + Bu_{com} - Bd \\ &= (A - BK)e + B[\hat{F}(x|\Theta_{ac}, M) - \hat{F}(x|\Theta_{ac}^*, M) + w'_e + w_e] + Bu_{com} - Bd \\ &= (A - BK)e + B[\hat{F}(x|\Theta_{ac}, M) - \hat{F}(x|\Theta_{ac}^*, M)] + Bu_{com} + B(w'_e + w_e - d) \\ &= (A - BK)e + B[\Phi_{ac}(x)(\Theta_{ac} - \Theta_{ac}^*)] + Bu_{com} + Bw \\ &= (A - BK)e + B\Phi_{ac}(x)\tilde{\Theta}_{ac} + Bu_{com} + Bw \end{aligned} \tag{22}$$

where $w'_e = \hat{F}(x|\Theta_{ac}^*, M) - \hat{F}(x|\Theta^*, N^*)$, $\tilde{\Theta}_{ac} = \Theta_{ac} - \Theta_{ac}^*$ and $w = w'_e + w_e - d$.

To ensure the closed-loop stability and desired tracking performance, we design the compensation signal $u_{com}$ as

$$u_{com} = -\frac{1}{\alpha} B^T P e \tag{23}$$

and the parameter adaptive law as

$$\dot{\Theta}_{ac} = -\eta\Phi_{ac}(x)B^T Pe, \tag{24}$$

where $\eta > 0$ is a design parameter and the symmetrical positive definite matrix $P$ is given by the following Riccati equation:

$$(A - BK)^T P + P(A - BK) + Q - \frac{2}{\alpha}PBB^T P + \frac{1}{\rho^2}PBB^T P = 0 \tag{25}$$

with $2\rho^2 \geq \alpha > 0$ and $Q$ being a symmetrical positive definite matrix.

To prevent parameter drift due to disturbance and approximation error and ensure the boundedness of parameters, a parameter projection algorithm is applied. With the parameter constraint $\Omega_\Theta$ defined in (17), the adaptive law (24) can be modified as follows:

$$\dot{\Theta}_{ac} = \begin{cases} -\eta\Phi_{ac}(x)B^T Pe & \text{if } \|\Theta_{ac}\| < M_\Theta \\ & \text{or } \|\Theta_{ac}\| = M_\Theta \text{ and } \Theta_{ac}^T\Phi_{ac}(x)B^T Pe \geq 0 \\ -\eta\left(I - \frac{\Theta_{ac}\Theta_{ac}^T}{\|\Theta_{ac}\|^2}\right)\Phi_{ac}(x)B^T Pe & \text{if } \|\Theta_{ac}\| = M_\Theta \text{ and } \Theta_{ac}^T\Phi_{ac}(x)B^T Pe < 0. \end{cases} \tag{26}$$

**Lemma 3.1.** *The adaptive law (26) ensures the boundedness of $\Theta_{ac}$ ($\Theta_{ac} \in \Omega_\Theta$) under the condition $\Theta_{ac}(0) \in \Omega_\Theta$.*

**Proof:** Consider the following Lyapunov function candidate:

$$V_\Theta = \frac{1}{2}\Theta_{ac}^T\Theta_{ac}. \tag{27}$$

Differentiating $V_\Theta$ with respect to time, we have

$$\dot{V}_\Theta = \Theta_{ac}^T\dot{\Theta}_{ac}. \tag{28}$$

With (26), when $\|\Theta_{ac}\| = M_\Theta$ and $\Theta_{ac}^T\Phi_{ac}(x)B^T Pe \geq 0$, we have

$$\dot{V}_\Theta = -\eta\Theta_{ac}^T\Phi_{ac}(x)B^T Pe \leq 0, \tag{29}$$

which guarantees $\|\Theta_{ac}\| \leq \|\Theta_{ac}(0)\|$.

When $\|\Theta_{ac}\| = M_\Theta$ and $\Theta_{ac}^T\Phi_{ac}(x)B^T Pe < 0$, we have

$$\dot{V}_\Theta = -\eta\Theta_{ac}^T\left(I - \frac{\Theta_{ac}\Theta_{ac}^T}{\|\Theta_{ac}\|^2}\right)\Phi_{ac}(x)B^T Pe = 0, \tag{30}$$

which also guarantees $\|\Theta_{ac}\| \leq \|\Theta_{ac}(0)\|$.

Since the initial values of $\Theta_{ac}$ are set satisfying $\Theta_{ac}(0) \in \Omega_\Theta$ by our online identification algorithm, we can conclude $\Theta_{ac} \in \Omega_\Theta$ from the above analysis. □

**Remark 3.2.** *Due to the online identification mechanism, the size of the parameter vector $\Theta_{ac}$ is not fixed. When a new rule is generated at $t = t_1$, the parameter vector is expanded from $\Theta_{ac}(t_1^-)$ to $\Theta_{ac}(t_1^+)$. Since the initial parameter values of the new rule are set to be zero, $\|\Theta_{ac}(t_1^-)\| = \|\Theta_{ac}(t_1^+)\|$. Then all the activated parameters will be tuned by (26), which guarantees the boundedness of the newly expanded $\Theta_{ac}$.*

3.3. **Stability analysis.** The closed-loop system stability is summarized in the following theorem.

**Theorem 3.1.** *Consider the MIMO system (7), the control law (14) with the compensation signal (23) and the parameter adaptive law (26) ensures the closed-loop system has the following properties:*

1. *all the signals in the closed-loop system are bounded;*
2. *there exist positive constants $a_1$ and $a_2$ such that $\int_0^T \|e\|^2 dt \leq a_1 + a_2 \int_0^T \|w\|^2 dt$;*
3. *if $w \in L_2$, $\lim\limits_{t\to\infty} \|e(t)\| = 0$.*

**Proof:** Consider the following Lyapunov candidate:

$$
\begin{aligned}
V = V(e,\tilde{\Theta}) &= \frac{1}{2}e^T P e + \frac{1}{2\eta}\tilde{\Theta}^T\tilde{\Theta} \\
&= \frac{1}{2}e^T P e + \frac{1}{2\eta}\tilde{\Theta}_{ac}^T\tilde{\Theta}_{ac} + \frac{1}{2\eta}\tilde{\Theta}_{in}^T\tilde{\Theta}_{in},
\end{aligned} \tag{31}
$$

where $\eta$ is a positive constant.

The derivative of (31) is

$$
\dot{V} = \frac{1}{2}\dot{e}^T P e + \frac{1}{2}e^T P\dot{e} + \frac{1}{\eta}\tilde{\Theta}_{ac}^T\dot{\tilde{\Theta}}_{ac} + \frac{1}{\eta}\tilde{\Theta}_{in}^T\dot{\tilde{\Theta}}_{in}. \tag{32}
$$

Since $\Theta_{in}$ contains inactivated parameters, whose values are fixed before they are activated (we do not need to know their exact values and just use them here for theoretical analysis), we have $\dot{\Theta}_{in} = 0$. Hence, $\dot{\tilde{\Theta}}_{in} = \dot{\Theta}_{in} - \dot{\Theta}^* = 0$.

Applying (22) and (23) into (32), we have

$$
\begin{aligned}
\dot{V} = &\frac{1}{2}e^T[(A-BK)^T P + P(A-BK)]e + \tilde{\Theta}_{ac}^T\left[\Phi_{ac}(x)B^T P e + \frac{1}{\eta}\dot{\tilde{\Theta}}_{ac}\right] \\
&+\frac{1}{2}(e^T PBw + w^T B^T P e).
\end{aligned} \tag{33}
$$

Applying (25) and the first condition of (26) into (33) yields

$$
\begin{aligned}
\dot{V} &= -\frac{1}{2}e^T Q e - \frac{1}{2\rho^2}e^T PBB^T P e + \frac{1}{2}(e^T PBw + w^T B^T P e) \\
&= -\frac{1}{2}e^T Q e - \frac{1}{2}\left(\frac{1}{\rho}B^T P e - \rho w\right)^T\left(\frac{1}{\rho}B^T P e - \rho w\right) + \frac{1}{2}\rho^2 w^T w \\
&\le -\frac{1}{2}e^T Q e + \frac{1}{2}\rho^2 w^T w \\
&\le -\frac{1}{2}\lambda_{\min}(Q)\|e\|^2 + \frac{1}{2}\rho^2\|w\|^2.
\end{aligned} \tag{34}
$$

With the second condition of (26), we have

$$
\begin{aligned}
\dot{V} &= \frac{1}{2}e^T[(A-BK)^T P + P(A-BK)]e + \frac{1}{2}(e^T PBw + w^T B^T P e) \\
&\quad +\tilde{\Theta}_{ac}^T\left[\Phi_{ac}(x)B^T P e - \left(I - \frac{\Theta_{ac}\Theta_{ac}^T}{\|\Theta_{ac}\|^2}\right)\Phi_{ac}(x)B^T P e\right] \\
&\le -\frac{1}{2}e^T Q e + \frac{1}{2}\rho^2 w^T w + \left(1 - \frac{\Theta_{ac}^{*T}\Theta_{ac}}{\|\Theta_{ac}\|^2}\right)\Theta_{ac}^T\Phi_{ac}(x)B^T P e.
\end{aligned} \tag{35}
$$

Since $\Theta_{ac}^* \in \Omega_\Theta$, recalling $\|\Theta_{ac}\| = M_\Theta$ under the second condition of (26), we have $1 - \frac{\Theta_{ac}^{*T}\Theta_{ac}}{\|\Theta_{ac}\|^2} \ge 0$. Thus,

$$
\dot{V} \le -\frac{1}{2}\lambda_{\min}(Q)\|e\|^2 + \frac{1}{2}\rho^2\|w\|^2. \tag{36}
$$

Therefore, $\dot{V} \le 0$ when $\|e\| \ge \rho\|w\| \Big/ \sqrt{\lambda_{\min}(Q)}$. For a bounded $w$, we have bounded tracking error $e$ and parameter estimates $\Theta_{ac}$. From (1), we have the boundedness of $x$. Hence, all the closed-loop signals are bounded.

Integrating both sides of (35) on $[0,T]$ results

$$\begin{aligned}\int_0^T \|e\|^2 dt &\le \frac{2}{\lambda_{\min}(Q)}(V(0)-V(T)) + \frac{\rho^2}{\lambda_{\min}(Q)}\int_0^T \|w\|^2 dt \\ &\le \frac{2}{\lambda_{\min}(Q)}V(0) + \frac{\rho^2}{\lambda_{\min}(Q)}\int_0^T \|w\|^2 dt. \end{aligned} \tag{37}$$

Let $a_1 = \frac{2}{\lambda_{\min}(Q)}V(0)$ and $a_2 = \frac{\rho^2}{\lambda_{\min}(Q)}$, the equation above can be written into

$$\int_0^T \|e\|^2 dt \le a_1 + a_2 \int_0^T \|w\|^2 dt. \tag{38}$$

If $w \in L_2$, from (38), we have $e \in L_2$. From (22), we have $\dot{e} \in L_\infty$. With $e \in L_2 \cap L_\infty$ and $\dot{e} \in L_\infty$, we have $\lim_{t\to\infty} \|e(t)\| = 0$.

When new rules are added to the fuzzy approximator, the size of the activated parameter vector $\Theta_{ac}$ grows. The Lyapunov function we choose for stability analysis includes both activated parameter errors $\tilde{\Theta}_{ac}$ and inactivated parameter errors $\tilde{\Theta}_{in}$. Before the inactivated parameters $\Theta_{in}$ are activated, their values are kept constant (that is, $\dot{\Theta}_{in} = 0$) and would not lead to the increase of $V(t)$. When some of the inactivated parameters are activated due to the generation of new rules, they become a part of the activated parameters, which are updated by the parameter adaptive law (24), and used to form the control law (14). It has been proven that under the control law (14), the adaptive law (24) and the compensation signal (23), all the closed-loop signals are bounded. □

**Remark 3.3.** *Compared with the control design with fixed rules, our approach does not require the designers are able to construct a fuzzy rule base that achieves the desired approximation accuracy offline. The error dynamics (22) is related to three kinds of errors:* $\tilde{\Theta}_{ac}$, $w'_e = \hat{F}(x|\Theta^*_{ac}, M) - \hat{F}(x|\Theta^*, N^*)$ *and* $w = \hat{F}(x|\Theta^*, N^*) - F(x)$. *Through our proposed online identification based control scheme, both the structure and the parameters of the fuzzy approximator are updated online, which has the advantage to reduce* $\tilde{\Theta}_{ac}$ *and* $w'_e$ *simultaneously. Thus, the effects of* $\hat{F}(x) - F(x)$ *on the tracking error can be further attenuated as compared with the fixed-rule approaches.*

4. **Simulation Study.** In this section, the proposed online fuzzy identification based control scheme is tested on an inverted pendulum system.

The dynamics of an inverted pendulum system is described by [22]:

$$\ddot{q} = \frac{g\sin q - \frac{ml\dot{q}\cos q \sin q}{m_c+m}}{l\left(\frac{4}{3} - \frac{m\cos^2 q}{m_c+m}\right)} + \frac{\frac{\cos q}{m_c+m}}{l\left(\frac{4}{3} - \frac{m\cos^2 q}{m_c+m}\right)} + d, \tag{39}$$

where $q$ and $\dot{q}$ are the angular position and angular velocity of the pendulum, respectively, $u$ is the control input, the mass of the cart $m_c = 1$ kg, the mass of the lever $m = 0.1$ kg, the length of the half lever $l = 1$ m, the acceleration of gravity $g = 9.8$ m/s$^2$, and the external disturbance $d = 0.05\sin(2\pi t)$ N$\cdot$ m. The nonlinear system (39) is first transformed into the standard form with $x = [x_1, x_2]^T = [q, \dot{q}]^T$:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x) + g(x)u + d, \\ y &= x_1 \end{aligned} \tag{40}$$

where

$$f(x) = \frac{g \sin x_1 - \frac{m l x_2 \cos x_1 \sin x_1}{m_c + m}}{l\left(\frac{4}{3} - \frac{m \cos^2 x_1}{m_c + m}\right)}, \quad g(x) = \frac{\frac{\cos x_1}{m_c + m}}{l\left(\frac{4}{3} - \frac{m \cos^2 x_1}{m_c + m}\right)}. \tag{41}$$

The control objective is to make the output $y(t)$ track a trajectory command $y_m(t) = 0.2 \sin(2t)$. The initial conditions are $x_1(0) = 0.2$ rad and $x_2(0) = 0.2$ rad/s. Other parameters in the simulation are set as follows: the parameter defining the influence area of a fuzzy rule $r = 0.4$, the learning rate of adaptive law (24) $\eta = 0.1$, the parameters used in solving the Riccati Equation (25) $\alpha = 0.08$ and $\rho = 0.2$, the controller parameters $k_1 = 2$ and $k_2 = 1$. With $Q = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$, solving (25) yields $P = \begin{bmatrix} 15 & 5 \\ 5 & 5 \end{bmatrix}$.

Consider the following two cases:

**Case 1**: the cart mass $m_c$ changes from 1 kg to 2 kg at $t = 8$ sec.

The tracking results are shown in Figure 2, including the tracking response, the control input and the number of rules. It can be observed that the proposed control scheme is capable of tracking the desired trajectory, even with unknown system dynamics, approximation error, or external disturbance. The online clustering algorithm with parameter adaptive laws is capable of online estimation of system dynamics satisfactorily. The online identification based controller works together with the compensation controller to achieve the boundedness of all the closed-loop signals and desired tracking performance. It can be seen that when the plant parameter $m_c$ changes from 1 kg to 2 kg at $t = 8$ sec, there is no new rule generated. That means the exiting rules have covered all the needed operating

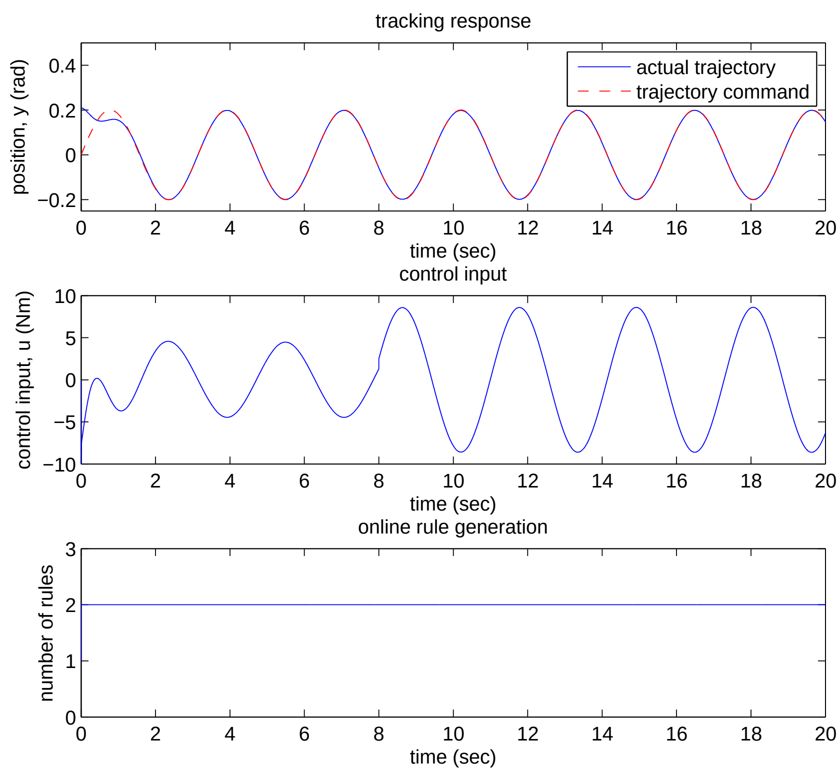


FIGURE 2. Tracking response under parameter change

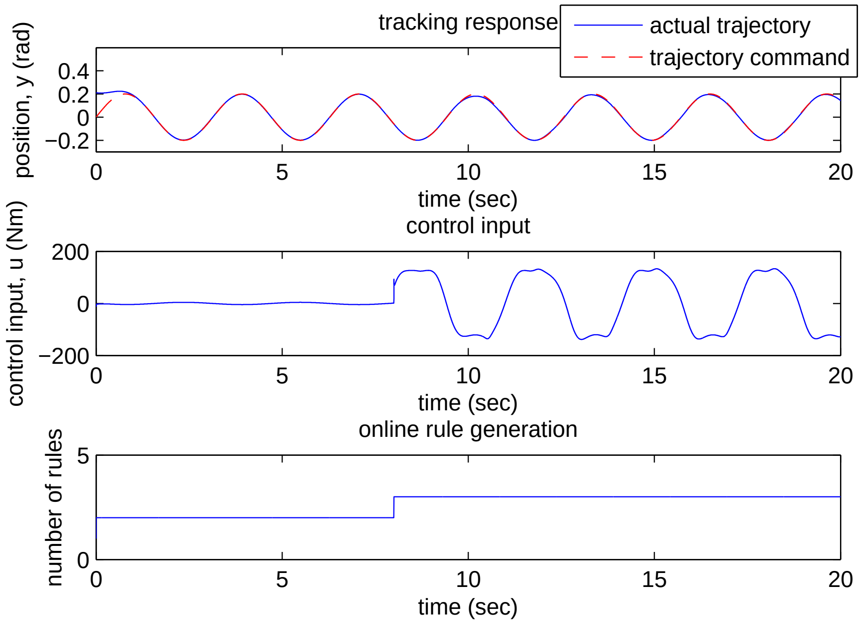


FIGURE 3. Adaptive fault-tolerant control with online rule generation

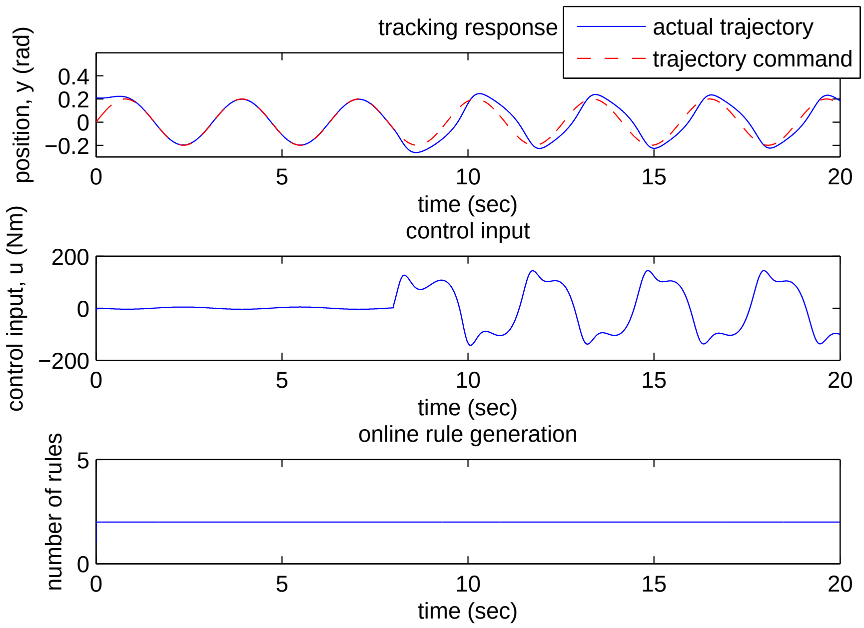


FIGURE 4. Adaptive fault-tolerant control without online rule generation

regions and it is enough to regain satisfactory control performance just by tuning the parameters of the controller.

**Case 2**: the structure of the nonlinear function $f(x)$ changes at $t = 8$ sec.

The tracking results are shown in Figure 3. In this case, there is a dramatic change happening in $f(x)$ which is replaced by another nonlinear function at $t = 8$ sec. This can be viewed as a serious fault occurred in the system to change both the structure and parameters of $f(x)$. With the proposed online identification based fault-tolerant control scheme, satisfactory tracking performance still can be achieved after some transients due to the dramatic change of $f(x)$. It can be seen that the number of rule increases from 2 to 3 at $t = 8$ sec, which helps achieve better approximation accuracy and issue a warning of a serious change in the system (detecting a fault). To illustrate the advantages of our approach, we have compared the fault-tolerant control performance between our approach and the method in [28], where only the model parameters are adapted online to accommodate the system changes. Figure 4 shows the adaptive fault-tolerant tracking control results without online rule generation and we can observe a big tracking error after the occurrence of a fault after 8 sec, which means that the model without adding new rules cannot approximate the changed system dynamics well. Compared with Figure 4, the tracking results in Figure 3 are much more desirable with adding a new rule.

5. **Conclusions.** In this paper, we propose an online fuzzy identification based adaptive fault-tolerant control scheme for a class of nonlinear systems. The rule base of the fuzzy systems evolves online to provide a more flexible structure and bigger capability to approximate more uncertainties in the system dynamics, not only the parameter variation, but also structure change due to a fault. An online clustering algorithm is used to generate new rules based on online data and the cluster centers are used to build the membership functions. With the online clustering algorithm, a new rule will be added to the fuzzy rule base only when significant change happens in the system dynamics. At the same time, the parameters of both old and new rules are updated online using adaptive laws derived in the sense of Lyanpunov stability theory. Both detailed design procedures and closed-loop stability proof are provided in this paper. Finally, the proposed scheme is applied to an inverted pendulum system under different scenarios including parameter variations and function changes due to a fault. Simulation results show satisfactory tracking performance can be achieved with the proposed control scheme.

**Acknowledgments.** This work was supported by the National Natural Science Foundation of China (60904042), the National Aerospace Science Foundation of China (2011ZA5 2009) and the Fundamental Research Funds for the Central Universities (NO. NS2012040).

## REFERENCES


[1] Y. Zhang and J. Jiang, Bibliographical review on reconfigurable fault-tolerant control systems, *Annual Reviews in Control*, vol.32, pp.229-252, 2008.

[2] Y.-W. Liang, D.-C. Liaw and T.-C. Lee, Reliable control of nonlinear systems, *IEEE Transactions on Automatic Control*, vol.45, no.4, pp.706-710, 2000.

[3] H. Alwi and C. Edwards, Fault detection and fault-tolerant control of a civil aircraft using a sliding-mode-based scheme, *IEEE Transactions on Control Systems Technology*, vol.16, no.3, pp.499-510, 2008.

[4] B. Jiang, M. Staroswiecki and V. Cocquempot, Fault accommodation for nonlinear dynamic systems, *IEEE Transactions on Automatic Control*, vol.51, no.9, pp.1578-1583, 2006.

[5] S. N. Huang and K. K. Tan, Fault detection, isolation, and accommodation control in robotic systems, *IEEE Transactions on Automation Science and Engineering*, vol.5, no.3, pp.480-489, 2008.

[6] M. Benosman and K.-Y. Lum, Online references reshaping and control reallocation for nonlinear fault tolerant control, *IEEE Transactions on Control Systems Technology*, vol.17, no.2, pp.366-379, 2009.

[7] Q. Hu, Robust adaptive sliding-mode fault-tolerant control with L2-gain performance for flexible spacecraft using redundant reaction wheels, *IET Control Theory and Application*, vol.4, no.6, pp.1055-1070, 2010.

[8] M. T. Hamayun, C. Edwards and H. Alwi, Design and analysis of an integral sliding mode fault-tolerant control scheme, *IEEE Transactions on Automatic Control*, vol.57, no.7, pp.1783-1789, 2012.

[9] S. Tong, Y. Li and T. Wang, Adaptive fuzzy backstepping fault-tolerant control for uncertain nonlinear systems based on dynamic surface, *International Journal of Innovative Computing, Information and Control*, vol.5, no.10(A), pp.3249-3261, 2009.

[10] R. Qi, G. Tao, B. Jiang and C. Tan, Adaptive control schemes for discrete-time T-S fuzzy systems with unknown parameters and actuator failures, *IEEE Transactions on Fuzzy System*, vol.20, no.3, pp.471-486, 2012.

[11] Y. Diao and K. M. Passino, Intelligent fault-tolerant control using adaptive and learning methods, *Control Engineering Practice*, vol.10, pp.801-817, 2002.

[12] J. A. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control: Unifying Neural, Fuzzy and Traditional Adaptive Approximation Approaches*, John Wiley & Sons, 2006.

[13] H. Ying, Sufficient conditions on uniform approximation of multivariate functions by general Takagi-Sugeno fuzzy systems with linear rule consequent, *IEEE Transactions on Syst., Man and Cybern. – Part A: Systems and Humans*, vol.28, no.4, pp.515-520, 1998.

[14] G. Feng, *Analysis and Synthesis of Fuzzy Control Systems: A Model-Based Approach*, CRC Press, Boca Raton, FL, 2010.

[15] P. P. Angelov and D. P. Filev, An approach to online identification of Takagi-Sugeno fuzzy models, *IEEE Transactions on Syst. Man and Cybern. – Part B: Cybernetics*, vol.34, no.1, pp.484-498, 2004.

[16] R. D. Baruah and P. P. Angelov, Clustering as a tool for self-generation of intelligent systems: A survey, *Evolving Intelligent Systems*, Leicester, UK, 2010.

[17] S. Barada and H. Singh, Generating optimal adaptive fuzzy-neural models of dynamical systems with applications to control, *IEEE Transactions on Syst. Man and Cybern. – Part C: Applications and Review*, vol.28, no.3, pp.297-313, 1998.

[18] R. Qi and M. Brdys, Stable indirect adaptive control based on discrete-time T-S fuzzy model, *Fuzzy Sets and Systems*, vol.159, no.8, pp.900-925, 2008.

[19] S. Tong, B. Chen and Y. Wang, Fuzzy adaptive output feedback control for MIMO nonlinear systems, *Fuzzy Sets and Systems*, vol.156, pp.285-299, 2005.

[20] Q. Wu, L. Jiang and J. Wen, Decentralized adaptive control of interconnected non-linear systems using high gain observer, *International Journal of Control*, vol.77, pp.703-712, 2004.

[21] Y. Chang, Robust tracking control for nonlinear MIMO systems via fuzzy approaches, *Automatica*, vol.36, no.10, pp.1535-1545, 2000.

[22] Y. Gao and M. J. Er, Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems, *IEEE Transactions on Fuzzy System*, vol.11, no.4, pp.462-477, 2003.

[23] P. A. Phan and T. J. Gale, Direct adaptive fuzzy control with a self-structuring algorithm, *Fuzzy Sets Systems*, vol.159, pp.871-899, 2008.

[24] C.-F. Hsu, Self-organizing adaptive fuzzy neural control for a class of nonlinear systems, *IEEE Transactions on Neural Networks*, vol.18, no.4, pp.1232-1241, 2007.

[25] C.-S. Chen, Robust self-organizing neural-fuzzy control with uncertainty observer for MIMO nonlinear systems, *IEEE Transactions on Fuzzy Systems*, vol.19, no.4, pp.694-706, 2011.

[26] G. G. Yen and P. Meesad, An effective neuro-fuzzy paradigm for machinery condition health monitoring, *Proc. of IEEE Int. Joint Conf. IJCNN'99*, Washington, DC, pp.1567-1572, 1999.

[27] R. Qi and M. A. Brdys, Indirect adaptive controller based on self-structuring fuzzy system for nonlinear identification and control, *Int. J. Appl. Math. Comput. Sci.*, vol.19, no.4, pp.619-630, 2009.

[28] Y. Diao and K. M. Passino, Stable adaptive control of feedback linearizable time-varying non-linear systems with application to fault-tolerant engine control, *Int. J. Control*, vol.77, no.17, pp.1463-1480, 2004.