

An Observer-Based Direct Adaptive Fuzzy Sliding Control with Adjustable Membership Functions

Alireza Gholami, Amir H. D. Markazi

Abstract—In this paper, an observer-based direct adaptive fuzzy sliding mode (OAFSM) algorithm is proposed. In the proposed algorithm, the zero-input dynamics of the plant could be unknown. The input connection matrix is used to combine the sliding surfaces of individual subsystems, and an adaptive fuzzy algorithm is used to estimate an equivalent sliding mode control input directly. The fuzzy membership functions, which were determined by time consuming try and error processes in previous works, are adjusted by adaptive algorithms. The other advantage of the proposed controller is that the input gain matrix is not limited to be diagonal, i.e. the plant could be over/under actuated provided that controllability and observability are preserved. An observer is constructed to directly estimate the state tracking error, and the nonlinear part of the observer is constructed by an adaptive fuzzy algorithm. The main advantage of the proposed observer is that, the measured outputs is not limited to the first entry of a canonical-form state vector. The closed-loop stability of the proposed method is proved using a Lyapunov-based approach. The proposed method is applied numerically on a multi-link robot manipulator, which verifies the performance of the closed-loop control. Moreover, the performance of the proposed algorithm is compared with some conventional control algorithms.

Keywords—Adaptive algorithm, fuzzy systems, membership functions, observer.

I. INTRODUCTION

VARIOUS methods exist for control of nonlinear systems. Many of the existing methods are model-based. Recent developments consider less model based methods, for those practical situations where the plant model is not known accurately. One such idea is to combine the concept of fuzzy control with other nonlinear control methodologies [1]-[5]. In particular, a well-known algorithm is the, so called, adaptive fuzzy sliding mode (AFSM) method. This approach exploits the advantages of both sliding mode and fuzzy logic control schemes.

The idea of AFSM was first proposed by Lin and Chen [6]. The main concept was to use the fuzzy inference to estimate the required control signal such that the system states converge towards a predefined sliding surface. An indirect AFSM algorithm was proposed by Yoo and Ham [7], where the unknown model was initially estimated by an adaptive fuzzy approach and the control signal was then designed based on the sliding mode theory. Generally, direct AFSM algorithms are more effective than the indirect ones, because

there are multiple estimation algorithms in the indirect approaches, while in the direct methods the control signal is approximated by a single estimation algorithm, and hence they may converge faster [8].

More recent articles have developed the AFSM method. Wang et al. [9] used the indirect AFSM algorithm to estimate the unknown system parts to design an ideal sliding mode control signal, and also used the direct AFSM approach to approximate the switching control signal [8]. Wai et al. [10] used the direct AFSM algorithm to approximate the bound of the estimation error. Wai [11] used the direct AFSM algorithm to approximate the switching part of the control signal in such a way that the chattering phenomenon could be eliminated [8]. In some articles, such as [12], the AFSM algorithm was used to estimate the plant uncertainties. Wai et al. [13] developed two cascaded AFSM methods, one for estimating the set point and another for estimating the control signal. Later on, Hwang et al. [14] developed a direct AFSM algorithm by a type-2 fuzzy system for control of unknown chaotic systems. Haghighi and Markazi [15] applied an AFSM method to control the chaotic motion of the MEMS resonators. Other researches extend the AFSM method for control of MIMO nonlinear systems. For example, Tong and Li [16] divided the nonlinear system into some canonical subsystems, and the indirect AFSM approach was employed. Aloui et al. [17] used indirect AFSM for MIMO nonlinear systems, and employed an adaptive PD term to eliminate the chatter. Poursamad and Markazi [18] proposed a direct AFSM approach for control of MIMO chaotic system. In their method, the input gain matrix is limited to be diagonal. This is a limitation which appeared in many of the papers proposing the direct AFSM controller for MIMO systems. This limitation means that the nonlinear sub-systems could not have any interconnections through their inputs. Furthermore, determining the fuzzy membership functions is based on a try and error process, which could be a time consuming procedure.

The main assumption of the AFSM method is that all the states are measurable, a matter which may not be the case for many practical situations [8]. Several papers considered the observer based AFSM approach for SISO systems [19]-[21] and also for MIMO plants [22]-[24]. The main limitation of those methods is that the outputs of the plant must be the same as the first entries of some canonical-form state vectors, a matter which restricts the application of the algorithm considerably [8].

In this article, new direct AFSM controller and observer are proposed for MIMO systems. In the proposed direct AFSM controller, it is not necessary for input gain matrix to be

Alireza Gholami PhD graduated from Iran University of Science and Technology and now is a researcher in electromechanical group of JDEVS, Iran, Islamic Republic Of (corresponding author, phone: +98-912-350-1990; e-mail: alirezagholami@gmail.com).

Ami H. D. Markazi is professor of Iran University of Science and Technology, Iran, Islamic Republic Of (e-mail: markazi@iust.ac.ir).

specified positive constant, and

$$\tilde{z}_i = z_{di} - z_i, \quad (4)$$

where z_{di} is the desired state trajectory. The vector of sliding surfaces is then defined as

$$\mathbf{s} = [s_1, \dots, s_m]^T. \quad (5)$$

The ideal control signal, \mathbf{u}^{eq} , is derived such that $\dot{\mathbf{s}} = 0$, i.e.,

$$\mathbf{u}^{eq} = \mathbf{G}(\mathbf{x})^{-1} [-\mathbf{F}(\mathbf{x}) + \mathbf{z}_d^{(r)} + \mathbf{H}(\mathbf{x})], \quad (6)$$

where $\mathbf{z}_d(t) = [z_{d1}, \dots, z_{dm}]^T$, $\mathbf{u}^{eq} = [u_1^{eq}, \dots, u_m^{eq}]^T$ and

$$\mathbf{H}(\mathbf{x}) = \begin{bmatrix} \sum_{j=1}^{r_1-1} C_j^{r_1-1} D^{r_1-1} \lambda_1^j \tilde{z}_1 \\ \vdots \\ \sum_{j=1}^{r_m-1} C_j^{r_m-1} D^{r_m-1} \lambda_m^j \tilde{z}_m \end{bmatrix} \quad \text{in which } C_j^r = \frac{j!}{r!(j-r)!}.$$

The time differentiation of \mathbf{s} is

$$\dot{\mathbf{s}} = \begin{bmatrix} D(D + \lambda_1)^{r_1-1} \tilde{z}_1 \\ \vdots \\ D(D + \lambda_m)^{r_m-1} \tilde{z}_m \end{bmatrix} =$$

$$\begin{bmatrix} \left(D^{r_1} + \sum_{j=1}^{r_1-1} C_j^{r_1-1} D^{r_1-j} \lambda_1^j \right) \tilde{z}_1 \\ \vdots \\ \left(D^{r_m} + \sum_{j=1}^{r_m-1} C_j^{r_m-1} D^{r_m-j} \lambda_m^j \right) \tilde{z}_m \end{bmatrix} = \mathbf{z}_d^{(r)} - \mathbf{z}^{(r)} + \mathbf{H}(\mathbf{x}) \quad (7)$$

Now consider the case, when $\mathbf{F}(\mathbf{x})$ in (1) is not known exactly. In order to follow the behavior of the equivalent sliding mode control, a model-free fuzzy algorithm is proposed to estimate the ideal control signal [8]. Consider m parallel fuzzy systems, with separate inputs, \hat{s}_i , $i = 1, 2, \dots, m$, and using three IF-THEN rules; i.e.,

Rule 1: IF \hat{s}_i is A_i^1 THEN $u_i^{fuz} = k_i^1$,

Rule 2: IF \hat{s}_i is A_i^2 THEN $u_i^{fuz} = k_i^2$,

Rule 3: IF \hat{s}_i is A_i^3 THEN $u_i^{fuz} = k_i^3$,

where, k_i^q $q = 1, 2, 3$, are fuzzy singleton outputs. Because it is assumed the system states are not measurable, therefore the input of fuzzy system is \hat{s}_i instead of s_i which is

$$\hat{s}_i = (D + \lambda_i)^{r_i-1} \tilde{z}_i, \quad (8)$$

where \tilde{z}_i is the estimated value of \tilde{z}_i . A_i^q is a fuzzy set defined through Gaussian membership functions as

$$\mu_{A_i^1}(\hat{s}_i) = \frac{1}{1 + e^{+a_i(\hat{s}_i + c_i)}}, \quad (9)$$

$$\mu_{A_i^2}(\hat{s}_i) = \exp\left[-\left(\frac{\hat{s}_i}{\sigma_i}\right)^2\right], \quad (10)$$

$$\mu_{A_i^3}(\hat{s}_i) = \frac{1}{1 + e^{-a_i(\hat{s}_i - c_i)}}, \quad (11)$$

where a_i , c_i and σ_i are membership functions parameters which will be adjusted adaptively in Lyapunov sense in Section IV. By singleton fuzzification, product inference system and center average defuzzification, the output of the fuzzy algorithm yields as

$$u_i^{fuz} = \frac{\sum_{q=1}^3 k_i^q \mu_{A_i^q}(\hat{s}_i)}{\sum_{q=1}^3 \mu_{A_i^q}(\hat{s}_i)}. \quad (12)$$

Defining the firing strength as

$$w_i^q = \frac{\mu_{A_i^q}(\hat{s}_i)}{\sum_{q=1}^3 \mu_{A_i^q}(\hat{s}_i)}, \quad (13)$$

the output of fuzzy system can be rewritten as

$$u_i^{fuz} = \mathbf{k}_i^T \mathbf{w}_i, \quad (14)$$

where, $\mathbf{k}_i = [k_i^1, k_i^2, k_i^3]^T$ is the vector of fuzzy output singletons, and $\mathbf{w}_i = [w_i^1, w_i^2, w_i^3]^T$ is the vector of firing strengths.

It is well-known that a fuzzy inference system of the form (14) is a universal approximator [8], [25], i.e. it can estimate any real continuous function to any degree of accuracy [8].

Now, define block matrixes \mathbf{K} and \mathbf{W} as $\mathbf{K} = [\mathbf{k}_1^T, \mathbf{k}_2^T, \dots, \mathbf{k}_m^T]^T$ and $\mathbf{W} = \text{diag}[\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_m]$, respectively. The fuzzy output singletons and the membership functions parameters are adjusted by adaptive laws which are designed in the Lyapunov sense. Therefore, the output of the adaptive fuzzy algorithm is written as

where, $\hat{\mathbf{s}} = [\hat{s}_1, \dots, \hat{s}_m]^T$, $\alpha_1 = \text{diag}[\alpha_{1,1}, \alpha_{1,2}, \dots, \alpha_{1,m}]$ and $\alpha_{1,i}$ is positive adaptation rate and

$$\tilde{\mathbf{K}} = \mathbf{K} - \hat{\mathbf{K}}. \quad (24)$$

2) membership function parameters vectors, i.e., σ , \mathbf{c} and \mathbf{a} are tuned adaptively by

$$\dot{\hat{\sigma}} = -\dot{\tilde{\sigma}} = \beta_1 \mathbf{G} \hat{\mathbf{s}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\sigma}} \hat{\mathbf{K}}, \quad (25)$$

$$\dot{\hat{\mathbf{c}}} = -\dot{\tilde{\mathbf{c}}} = \beta_2 \mathbf{G} \hat{\mathbf{s}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{c}}} \hat{\mathbf{K}}, \quad (26)$$

and

$$\dot{\hat{\mathbf{a}}} = -\dot{\tilde{\mathbf{a}}} = \beta_3 \mathbf{G} \hat{\mathbf{s}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{a}}} \hat{\mathbf{K}}, \quad (27)$$

respectively, where $\beta_1 = \text{diag}[\beta_{1,1}, \dots, \beta_{1,m}]$, $\beta_2 = \text{diag}[\beta_{2,1}, \dots, \beta_{2,m}]$ and $\beta_3 = \text{diag}[\beta_{3,1}, \dots, \beta_{3,m}]$ include the adaptation rates, $\mathbf{J} = [1, \dots, 1]^T \in R^{m \times 1}$ is a constant vector and

$$\tilde{\sigma} = \sigma - \hat{\sigma}, \quad (28)$$

$$\tilde{\mathbf{c}} = \mathbf{c} - \hat{\mathbf{c}}, \quad (29)$$

$$\tilde{\mathbf{a}} = \mathbf{a} - \hat{\mathbf{a}}. \quad (30)$$

3) \mathbf{u}^{rb} is robust controller which is designed according to

$$\mathbf{u}^{rb} = \mathbf{G} \text{sgn}(\hat{\mathbf{s}}) \hat{\Psi}, \quad (31)$$

where $\text{sgn}(\hat{\mathbf{s}}) = \text{diag}[\text{sgn}(\hat{s}_1), \dots, \text{sgn}(\hat{s}_m)]$ and $\hat{\Psi} = [\hat{\psi}_1, \dots, \hat{\psi}_m]^T$ is the approximated value of the error bounds vector which is adjusted adaptively by

$$\dot{\hat{\Psi}} = -\dot{\tilde{\Psi}} = \alpha_2 \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}}, \quad (32)$$

where $\alpha_2 = \text{diag}[\alpha_{2,1}, \alpha_{2,2}, \dots, \alpha_{2,m}]$ and $\alpha_{2,i}$ is a user defined positive constant and

$$\tilde{\Psi}(t) = \Psi - \hat{\Psi}(t). \quad (33)$$

then the stability of algorithm is guaranteed.

Proof. A positive definite candidate Lyapunov function is defined as

$$V_c(\hat{\mathbf{s}}, \tilde{\mathbf{K}}, \tilde{\Psi}, \tilde{\sigma}, \tilde{\mathbf{c}}, \tilde{\mathbf{a}}) = \frac{1}{2} \hat{\mathbf{s}}^T \hat{\mathbf{s}} + \frac{1}{2} \tilde{\mathbf{K}}^T \alpha_1^{-1} \tilde{\mathbf{K}} + \frac{1}{2} \tilde{\Psi}^T \alpha_2^{-1} \tilde{\Psi} + \frac{1}{2} \tilde{\sigma}^T \beta_1^{-1} \tilde{\sigma} + \frac{1}{2} \tilde{\mathbf{c}}^T \beta_2^{-1} \tilde{\mathbf{c}} + \frac{1}{2} \tilde{\mathbf{a}}^T \beta_3^{-1} \tilde{\mathbf{a}}. \quad (34)$$

By differentiating (34) with respect to time and using (22)-(33), it is obtained that

$$\begin{aligned} \dot{V}_c &= \hat{\mathbf{s}}^T \dot{\hat{\mathbf{s}}} + \tilde{\mathbf{K}}^T \alpha_1^{-1} \dot{\tilde{\mathbf{K}}} + \tilde{\Psi}^T \alpha_2^{-1} \dot{\tilde{\Psi}} + \tilde{\sigma}^T \beta_1^{-1} \dot{\tilde{\sigma}} + \tilde{\mathbf{c}}^T \beta_2^{-1} \dot{\tilde{\mathbf{c}}} + \tilde{\mathbf{a}}^T \beta_3^{-1} \dot{\tilde{\mathbf{a}}} \\ &= \hat{\mathbf{s}}^T [\mathbf{G}(\hat{\mathbf{W}}^T \tilde{\mathbf{K}}) + \mathbf{G}(\tilde{\mathbf{W}}^T \hat{\mathbf{K}}) + \mathbf{G}\psi - \mathbf{G}\mathbf{u}^{rb}] + \tilde{\mathbf{K}}^T \alpha_1^{-1} (-\alpha_1 \mathbf{W} \hat{\mathbf{G}} \hat{\mathbf{s}}) \\ &\quad + \tilde{\Psi}^T \alpha_2^{-1} \dot{\tilde{\Psi}} + \tilde{\sigma}^T \beta_1^{-1} \dot{\tilde{\sigma}} + \tilde{\mathbf{c}}^T \beta_2^{-1} \dot{\tilde{\mathbf{c}}} + \tilde{\mathbf{a}}^T \beta_3^{-1} \dot{\tilde{\mathbf{a}}} \\ &= \tilde{\mathbf{K}}^T \mathbf{G} \hat{\mathbf{W}}^T \hat{\mathbf{s}} - \tilde{\mathbf{K}}^T \mathbf{G} \tilde{\mathbf{W}}^T \hat{\mathbf{s}} + \hat{\mathbf{s}}^T \mathbf{G} (\psi - \mathbf{u}^{rb}) + \tilde{\Psi}^T \alpha_2^{-1} \dot{\tilde{\Psi}} \\ &\quad + \hat{\mathbf{s}}^T \mathbf{G} (\tilde{\sigma} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\sigma}} + \tilde{\mathbf{c}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{c}}} + \tilde{\mathbf{a}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{a}}}) \hat{\mathbf{K}} + \tilde{\sigma}^T \beta_1^{-1} \dot{\tilde{\sigma}} + \tilde{\mathbf{c}}^T \beta_2^{-1} \dot{\tilde{\mathbf{c}}} + \tilde{\mathbf{a}}^T \beta_3^{-1} \dot{\tilde{\mathbf{a}}} \\ &= \psi^T \mathbf{G} \hat{\mathbf{s}} - \hat{\mathbf{s}}^T \mathbf{G} [\mathbf{G} \text{sgn}(\hat{\mathbf{s}}) \hat{\Psi}] + \tilde{\Psi}^T \alpha_2^{-1} [-\alpha_2 \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}}] \\ &\quad + \hat{\mathbf{s}}^T \mathbf{G} \tilde{\sigma} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\sigma}} \hat{\mathbf{K}} - \hat{\mathbf{s}}^T \mathbf{G} \tilde{\mathbf{c}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{c}}} \hat{\mathbf{K}} + \hat{\mathbf{s}}^T \mathbf{G} \tilde{\mathbf{a}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{a}}} \hat{\mathbf{K}} - \hat{\mathbf{s}}^T \mathbf{G} \tilde{\mathbf{c}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{c}}} \hat{\mathbf{K}} \\ &\quad + \hat{\mathbf{s}}^T \mathbf{G} \tilde{\mathbf{a}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{a}}} \hat{\mathbf{K}} - \hat{\mathbf{s}}^T \mathbf{G} \tilde{\mathbf{a}} \mathbf{J}^T \frac{\partial \hat{\mathbf{W}}^T}{\partial \hat{\mathbf{a}}} \hat{\mathbf{K}} \\ &= \psi^T \mathbf{G} \hat{\mathbf{s}} - \tilde{\Psi}^T \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}} \leq \psi^T \text{sgn}(\psi) \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}} - \tilde{\Psi}^T \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}} \\ &= (\psi^T \text{sgn}(\psi) - \tilde{\Psi}^T) \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}} \leq 0, \end{aligned} \quad (35)$$

where $\text{sgn}(\psi) = \text{diag}[\text{sgn}(\psi_1), \dots, \text{sgn}(\psi_m)]$. Therefore, \dot{V}_c is negative semidefinite, which shows the algorithm stability. In order to prove the asymptotic stability, bring in

$$\Xi(t) = (\psi^T \text{sgn}(\psi) - \tilde{\Psi}^T) \mathbf{G}^2 \text{sgn}(\hat{\mathbf{s}}) \hat{\mathbf{s}} \leq -\dot{V}_c, \quad (36)$$

and integrate $\Xi(t)$ with respect to time, then it can be shown that

$$\int_0^t \Xi(\tau) d\tau \leq V_c(0) - V_c(t). \quad (37)$$

Since $V_c(0)$ is bounded and $V_c(t)$ is nonincreasing and bounded, it implies that [17]:

$$\lim_{t \rightarrow \infty} \int_0^t \Xi(\tau) d\tau \leq \infty. \quad (38)$$

Furthermore, $\dot{\Xi}$ is bounded, so by Barbalat's Lemma [26], it can be shown that $\lim_{t \rightarrow \infty} \Xi(t) = 0$. That is $\lim_{t \rightarrow \infty} \hat{\mathbf{s}}(t) \rightarrow 0$.

Therefore, the closed-loop control method is asymptotically stable.

V. NEW ADAPTIVE FUZZY OBSERVER DESIGN

In this section, a new adaptive fuzzy observer is proposed to estimate the state tracking error vector for the MIMO nonlinear system (1). The state tracking error dynamic is explained in the form of

$$\dot{\tilde{\mathbf{x}}} = \mathbf{A} \tilde{\mathbf{x}} + \mathbf{B} \mathbf{v}(\tilde{\mathbf{x}}, \mathbf{u}), \quad (39)$$

where $\mathbf{v}(\tilde{\mathbf{x}}, \mathbf{u}) = [v_1(\tilde{\mathbf{x}}, \mathbf{u}), \dots, v_m(\tilde{\mathbf{x}}, \mathbf{u})]^T$ includes all of the observer nonlinear terms and

$$\tilde{\mathbf{x}} = \mathbf{x}_d - \mathbf{x}. \quad (40)$$

If the model (1) was known exactly, the observer could be defined as

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\mathbf{v}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{Q}(\mathbf{e} - \hat{\mathbf{e}}), \\ \mathbf{e} &= \mathbf{C}\hat{\mathbf{x}}, \\ \hat{\mathbf{e}} &= \mathbf{C}\hat{\mathbf{x}}, \end{aligned} \quad (41)$$

where \mathbf{Q} is the observer gain and $\hat{\mathbf{x}}$ is the estimated value of the state tracking error vector. The observation error dynamic is obtained by subtracting (41) from (39), i.e.,

$$\dot{\tilde{\mathbf{x}}} = (\mathbf{A} - \mathbf{Q}\mathbf{C})\tilde{\mathbf{x}} + \mathbf{B}\tilde{\mathbf{v}}(\tilde{\mathbf{x}}, \hat{\mathbf{x}}, \mathbf{u}), \quad (42)$$

where

$$\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}} \quad (43)$$

and

$$\tilde{\mathbf{v}}(\tilde{\mathbf{x}}, \hat{\mathbf{x}}, \mathbf{u}) = \mathbf{v}(\tilde{\mathbf{x}}, \mathbf{u}) - \mathbf{v}(\hat{\mathbf{x}}, \mathbf{u}). \quad (44)$$

The gain \mathbf{Q} is determined such that the characteristic polynomial of $\mathbf{A} - \mathbf{Q}\mathbf{C}$ is strictly Hurwitz.

Now consider the case, when the system (1) is not known exactly. In order to mimic the behavior of the nonlinear part of the observer, $\mathbf{v}(\hat{\mathbf{x}}, \mathbf{u})$, a model-free adaptive fuzzy algorithm is used to approximate the $v_i(\hat{\mathbf{x}}, \mathbf{u})$, $i=1,2,\dots,m$ in the real-time. Consider m parallel fuzzy systems, with separate inputs, $\tilde{e}_i = e_i - \hat{e}_i$, $i=1,2,\dots,m$, and using n_l IF-THEN rules; i.e.,

Rule 1: IF \tilde{e}_i is A_i^l THEN $v_i^{fuz,l} = k_{o,i}^l$, $l=1,\dots,n_l$,

where, $k_{o,i}^l$ is a fuzzy singleton output and A_i^l is a fuzzy set defined through constant Gaussian membership functions. The inference system is the same as one used in the controller fuzzy system. The output of the fuzzy algorithm yields as

$$v_i^{fuz} = \mathbf{k}_{o,i}^T \mathbf{w}_{o,i}, \quad (45)$$

where, $\mathbf{k}_{o,i} = [k_{o,i}^1, \dots, k_{o,i}^{n_l}]^T$ is the vector of output fuzzy singletons, and $\mathbf{w}_{o,i} = [w_{o,i}^1, \dots, w_{o,i}^{n_l}]^T$ is the vector of firing strengths of fuzzy rules.

Now, define block matrixes \mathbf{K}_o and \mathbf{W}_o as $\mathbf{K}_o = [\mathbf{k}_{o,1}^T, \mathbf{k}_{o,2}^T, \dots, \mathbf{k}_{o,m}^T]^T$ and $\mathbf{W}_o = \text{diag}[\mathbf{w}_{o,1}, \mathbf{w}_{o,2}, \dots, \mathbf{w}_{o,m}]$, respectively. Then, the vectors of fuzzy output singletons are tuned by an adaptive law as

$$\dot{\hat{\mathbf{K}}}_o = -\hat{\mathbf{K}}_o = \alpha_3 \mathbf{C}\mathbf{B}\mathbf{W}_o^T \tilde{\mathbf{e}}, \quad (46)$$

where, $\alpha_3 = \text{diag}[\alpha_{3,1}, \alpha_{3,2}, \dots, \alpha_{3,m}]$ and $\alpha_{3,i}$ is positive adaptation rate, $\hat{\mathbf{K}}_o$ is the estimated value of \mathbf{K}_o ,

$$\tilde{\mathbf{K}}_o = \mathbf{K}_o - \hat{\mathbf{K}}_o, \quad (47)$$

and

$$\tilde{\mathbf{e}} = \mathbf{e} - \hat{\mathbf{e}}. \quad (48)$$

Therefore, the adaptive fuzzy system output can be written as

$$\hat{\mathbf{v}}^{fuz} = \mathbf{W}_o^T \hat{\mathbf{K}}_o. \quad (49)$$

Now, the proposed model-free adaptive fuzzy observer is formulated in the form of

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\hat{\mathbf{v}}^{fuz}(\hat{\mathbf{x}}, \mathbf{u}) + \mathbf{Q}(\mathbf{e} - \hat{\mathbf{e}}), \\ \hat{\mathbf{e}} &= \mathbf{C}\hat{\mathbf{x}}. \end{aligned} \quad (50)$$

By use of (50) and (2) the AFSM observer error dynamic is resulted as

$$\mathbf{C}(\mathbf{A} - \mathbf{Q}\mathbf{C})(\tilde{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) + \mathbf{C}\mathbf{B}(\mathbf{v} - \hat{\mathbf{v}}^{fuz}) = \dot{\tilde{\mathbf{e}}}. \quad (51)$$

Fig. 2 shows the observer functional diagram.

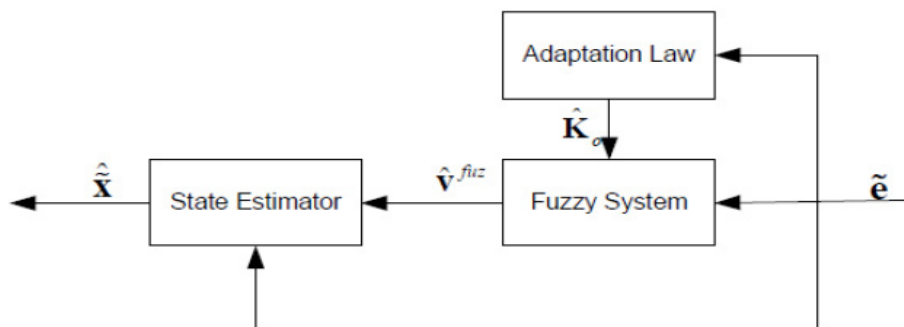


Fig. 2 The structure of the observer

VI. STABILITY ANALYSIS

Theorem 2. Consider the unknown nonlinear system described by (1). With the controller input described in Theorem 1 and the observer designed according to (50), where $\hat{\mathbf{v}}^{fuz}(\hat{\mathbf{x}}, \mathbf{u})$ is approximated by adaptive fuzzy system (49) and $\hat{\mathbf{K}}_o$ is adjusted by (46), then the stability of algorithm is guaranteed.

Proof. A positive definite candidate Lyapunov function is defined as

$$V = V_c + V_o, \quad (52)$$

where, V_c was defined by (34) and V_o is considered as

$$V_o(\tilde{\mathbf{e}}, \tilde{\mathbf{K}}_o) = \frac{1}{2} \tilde{\mathbf{e}}^T \tilde{\mathbf{e}} + \frac{1}{2} \tilde{\mathbf{K}}_o^T \alpha_3^{-1} \tilde{\mathbf{K}}_o. \quad (53)$$

By differentiating (52) with respect to time we have

$$\dot{V} = \dot{V}_c + \dot{V}_o, \quad (54)$$

where, \dot{V}_c was described by (35) and by use of (46) and (51)

\dot{V}_o is obtained as

$$\begin{aligned} \dot{V}_o &= \tilde{\mathbf{e}}^T \dot{\tilde{\mathbf{e}}} + \tilde{\mathbf{K}}_o^T \alpha_3^{-1} \dot{\tilde{\mathbf{K}}}_o \\ &= \tilde{\mathbf{e}}^T [\mathbf{C}(\mathbf{A} - \mathbf{Q}\mathbf{C})(\tilde{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) + \mathbf{C}\mathbf{B}(\mathbf{W}_o^T \tilde{\mathbf{K}}_o)] \\ &\quad + \tilde{\mathbf{K}}_o^T \alpha_3^{-1} (-\alpha_3 \mathbf{C}\mathbf{B}\mathbf{W}_o^T \tilde{\mathbf{e}}) \\ &= \tilde{\mathbf{e}}^T \mathbf{C}(\mathbf{A} - \mathbf{Q}\mathbf{C})(\tilde{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) + \tilde{\mathbf{K}}_o^T \mathbf{C}\mathbf{B}\mathbf{W}_o^T \tilde{\mathbf{e}} - \tilde{\mathbf{K}}_o^T \mathbf{C}\mathbf{B}\mathbf{W}_o^T \tilde{\mathbf{e}}. \end{aligned} \quad (55)$$

Because $(\mathbf{A} - \mathbf{Q}\mathbf{C})$ is strictly Herwitz, therefore,

$$\tilde{\mathbf{e}}^T \mathbf{C}(\mathbf{A} - \mathbf{Q}\mathbf{C})(\tilde{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) = (\tilde{\mathbf{x}} - \hat{\tilde{\mathbf{x}}})^T \mathbf{C}^T \mathbf{C}(\mathbf{A} - \mathbf{Q}\mathbf{C})(\tilde{\mathbf{x}} - \hat{\tilde{\mathbf{x}}}) < 0. \quad (56)$$

Therefore, by (35), (55) and (56) it is obtained that \dot{V} is negative definite, i.e, based on the Lyapunov theory the algorithm error goes to zero asymptotically.

VII. NUMERICAL SIMULATION

In this section, numerical investigations on the effectiveness of the proposed control scheme are carried out. For this purpose, application of the method on a modular and reconfigurable robot (MRR) system, shown in Fig. 3, is considered [27]. Also, an adaptive PID controller with a Luenberger observer is applied on the MRR to compare its performance with the proposed algorithm.

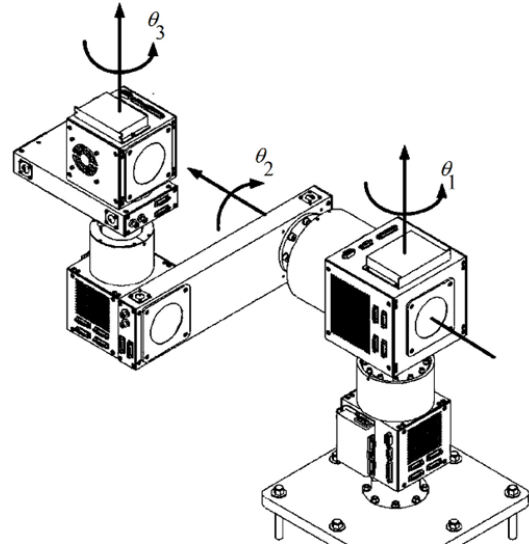


Fig. 3 MRR coordinates

A. MRR Manipulator

Although the proposed control method does not rely on the detailed plant information, it is necessary to make sure that the general model structure is compatible with the class of MIMO nonlinear systems described in (1). Furthermore, it is needed to simulate the plant for the numerical analysis. The general form of the dynamic equation of MRR is [28]

$$\mathbf{M}(\boldsymbol{\Theta})\ddot{\boldsymbol{\Theta}} + \mathbf{V}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})\dot{\boldsymbol{\Theta}} + \mathbf{T}(\dot{\boldsymbol{\Theta}}) + \mathbf{Z}(\boldsymbol{\Theta}) + \boldsymbol{\tau}_d = \boldsymbol{\tau} \quad (57)$$

where $\mathbf{M}(\boldsymbol{\Theta}) \in R^{3 \times 3}$ is the inertia matrix, $\mathbf{V}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) \in R^{3 \times 3}$ is the Coriolis matrix, $\mathbf{T}(\dot{\boldsymbol{\Theta}}) \in R^3$ contains the friction terms, $\mathbf{Z}(\boldsymbol{\Theta}) \in R^3$ is the gravity vector, $\boldsymbol{\tau}_d \in R^3$ represents disturbances, $\boldsymbol{\tau} \in R^3$ is the torque control input vector, and $\boldsymbol{\Theta} = [\theta_1, \theta_2, \theta_3]^T \in R^3$ is the vector of joint angles (see Fig.3)

The MRR dynamic equation (57) can be equivalently written in the form of

$$\begin{aligned} \ddot{\mathbf{x}} &= \mathbf{F}(\mathbf{x}) + \mathbf{G}\boldsymbol{\tau}, \\ \mathbf{y} &= \mathbf{C}\mathbf{x}, \end{aligned} \quad (58)$$

where, $\mathbf{x} = [x_1, x_2, \dots, x_6]^T = [\theta_1, \dot{\theta}_1, \theta_2, \dot{\theta}_2, \theta_3, \dot{\theta}_3]^T \in R^6$ is the state vector, $\mathbf{F}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x})]^T = -\mathbf{M}^{-1}\mathbf{V} - \mathbf{M}^{-1}\mathbf{T} - \mathbf{M}^{-1}\mathbf{Z} - \mathbf{M}^{-1}\boldsymbol{\tau}_d \in R^3$ is a smooth and unknown function vector of \mathbf{x} and $\mathbf{G}(\mathbf{x}) = \mathbf{M}^{-1}$.

As mentioned before, in the proposed observer, the measured outputs do not need to be considered as the first entries of canonical-form state vector. To show this contribution, the output of each joint is considered as the summation of its position and velocity, i.e., the matrix \mathbf{C} is selected as

$$\mathbf{C} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}. \quad \text{The desired output trajectory}$$

for each joint is in the form of

$$y_{di} = A_i \sin\left(\frac{2\pi}{T} * \left(\frac{j}{f}\right)\right), \quad i = 1,2,3, \quad (59)$$

where, $A_1 = A_3 = 90\text{deg}$ and $A_2 = 45\text{deg}$ are the trajectory amplitudes, $T = 40\text{s}$ is the trajectory period, $f = 50\text{Hz}$ is the control frequency, $j = 1,2,\dots$ is the control signal index and i refers to the i th joint.

B. Control and Observer Parameters

The sliding surface constants, fuzzy output adaptation rates and bound estimation adaptation rates, which are needed for designing the controller, are selected as shown in Table I.

TABLE I
 CONTROLLER PARAMETERS

Joint	Sliding surfaces constants λ_i	Fuzzy output adaptation rates $\alpha_{1,i}$	Error estimation adaptation rates $\alpha_{2,i}$
11	15	1.7	0.05
22	15	0.6	0.01
33	9	0.5	0.01

Also, the adaptation rates for adjusting the fuzzy membership functions are shown in Table II.

TABLE II
 MEMBERSHIP FUNCTION ADAPTATION RATES

Joint	Adaptation Rate for σ_i ($\beta_{1,i}$)	Adaptation Rate for c_i ($\beta_{2,i}$)	Adaptation Rate for a_i ($\beta_{3,i}$)
1	35	45	1
2	20	30	1
3	20	40	10

The observer adaptation rates are shown in Table III.

TABLE III
 OBSERVER PARAMETERS

Joint	Fuzzy output adaptation rates $\alpha_{3,i}$
1	3
2	3
3	3

Furthermore, the fuzzy membership functions for observer design are shown in Fig. 4.

The experimental tracking performance of proposed OAFSM controller is depicted in Figs. 5-7. Considering the fact that the information of the plant is not used for controller design except the gain matrix of input signal vector, the proposed algorithm has an acceptable tracking performance. Also, the observer shows the effective estimation of the states tracking errors, while the robot information is not used to design it. Moreover, to show the effect of membership function adaptive algorithms, the output tracking error is compared by the AFSM algorithm with constant membership functions in Fig. 5.

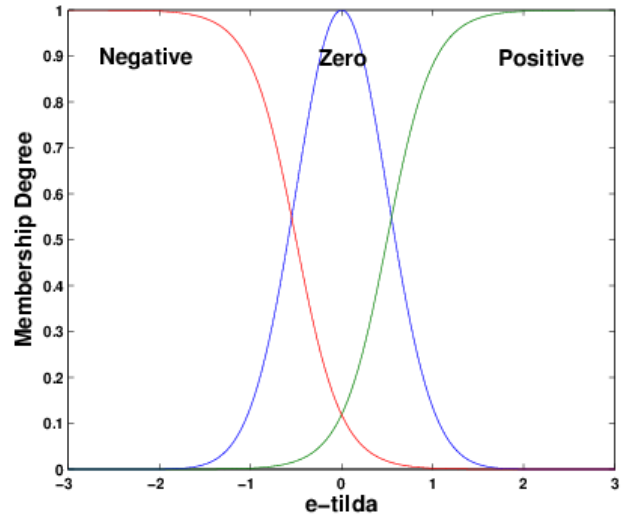
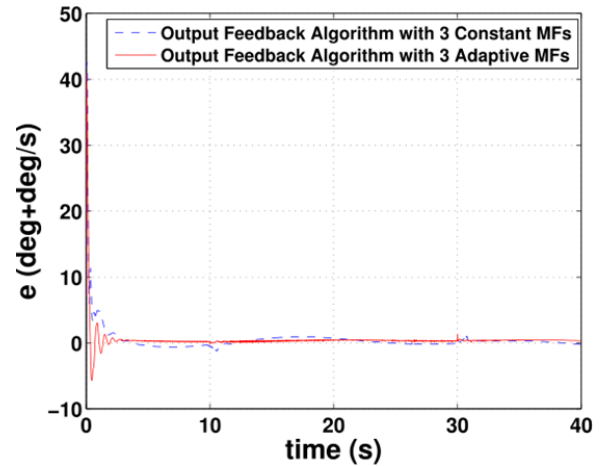
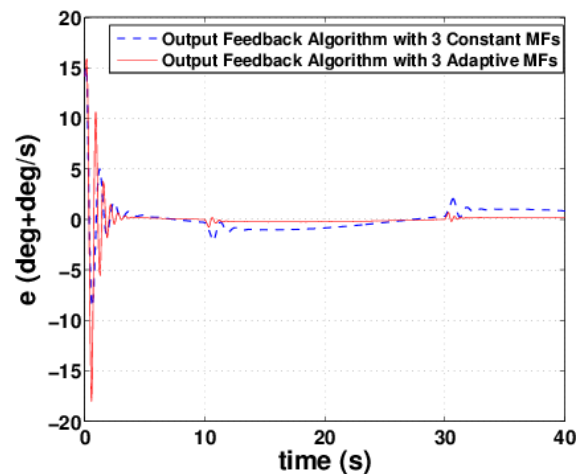


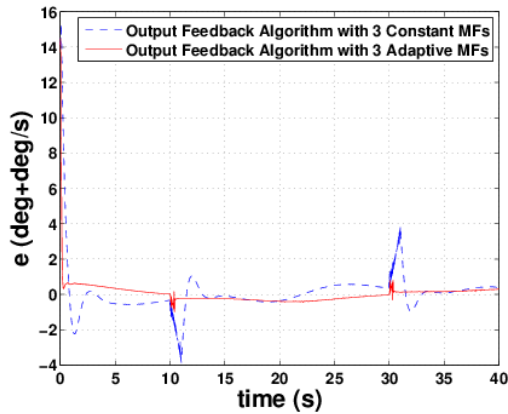
Fig. 4 Membership functions for observer fuzzy systems



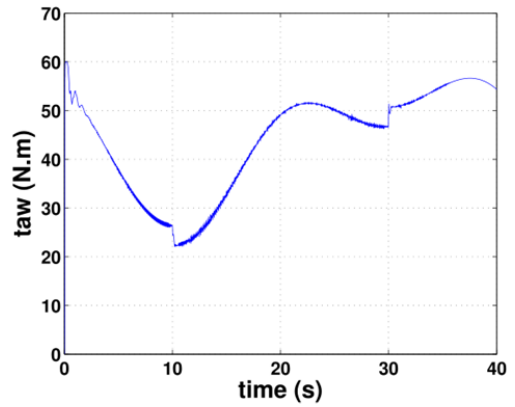
(a)



(b)

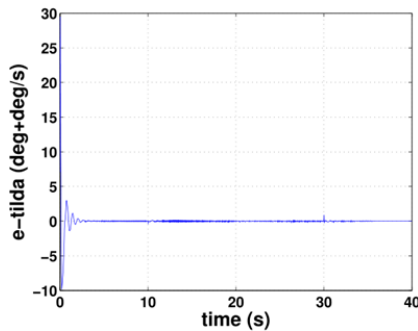


(c)

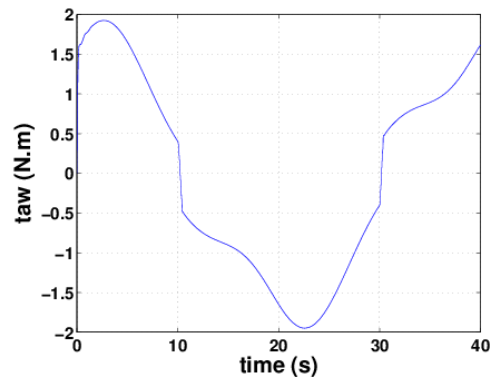


(a)

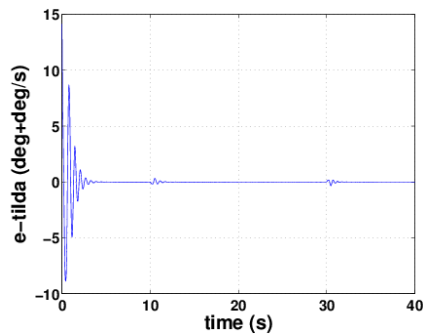
Fig. 5 Comparison of output tracking error between the AFSM algorithms with adaptive membership functions and constant membership functions for (a) Joint 1, (b) Joint 2, (c) Joint 3



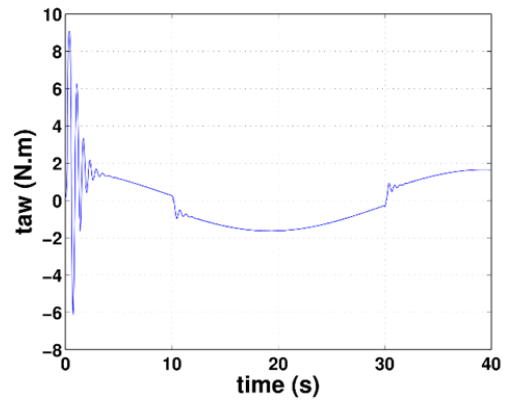
(a)



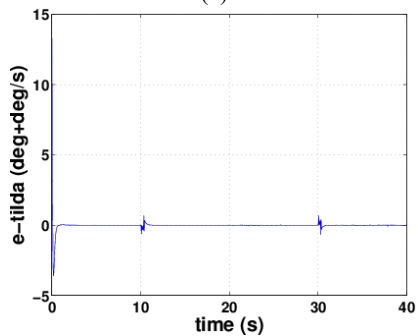
(b)



(b)



(c)



(c)

Fig. 6 Observer error for (a) Joint 1, (b) Joint 2, (c) Joint 3

Fig. 7 Control Inputs for (a) Joint 1 (b) Joint 2 (c) Joint 3

C. Comparison Study

Here, an adaptive PID controller and a conventional Luenberger observer are applied on the robot manipulator. Based on the adaptive PID controller proposed by [29], the PID gains are adjusted by adaptive equations as

$$\dot{K}_{p,j} = \eta_{1,j} e^2 \quad (60)$$

$$\dot{K}_{i,j} = \eta_{2,j} e \int e dt \quad (61)$$

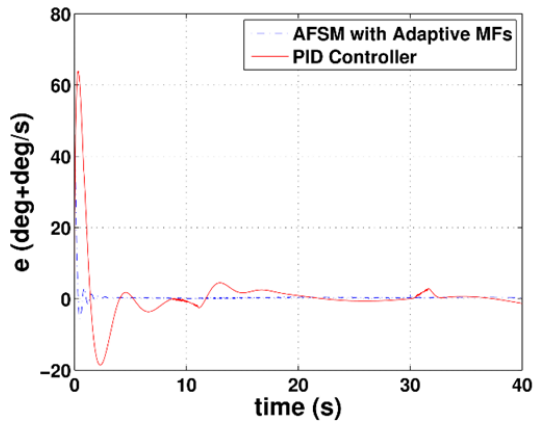
and

$$\dot{K}_{d,j} = \eta_{3,j} e \dot{e} \quad (62)$$

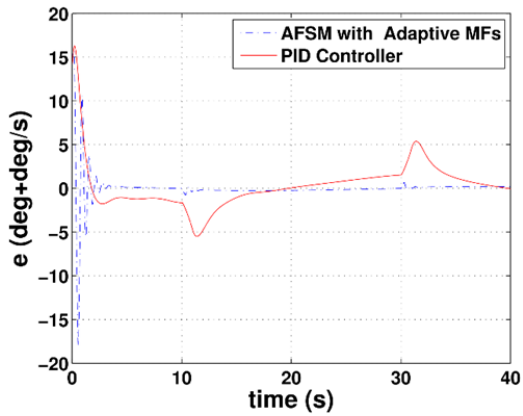
where, $K_{p,j}$, $K_{i,j}$ and $K_{d,j}$ are proportional gain, integral gain and derivative gain for j th joint, respectively. Also, $\eta_{1,j}$, $\eta_{2,j}$ and $\eta_{3,j}$ are adaptation rates.

To design the Luenberger observer, the nonlinear dynamic equation (57) should be converted to the linear equation firstly. Here, the feedback linearization method is used to linearize the robot equations. Then, the Luenberger observer is simply designed based on the linearized model.

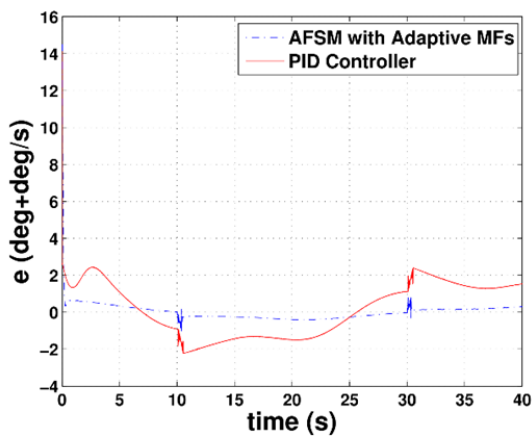
The output tracking error of the adaptive PID controller and the error of the model-based Luengerger observer is compared with the proposed observer-based AFSM algorithm in Figs. 8 and 9. Although all plant information is used to design the PID controller and Luenberger observer, the performance of proposed algorithm which designed based on the partially unknown plant is more acceptable.



(a)

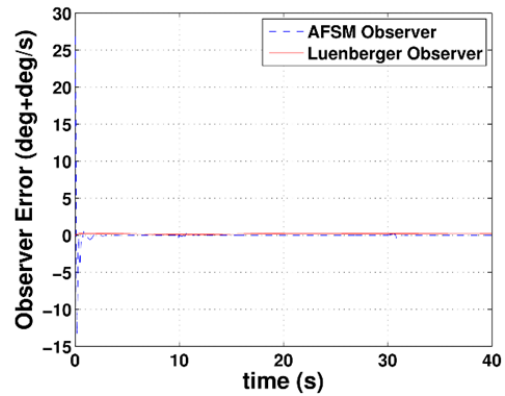


(b)

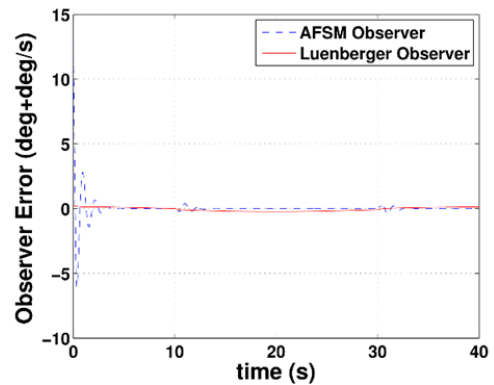


(c)

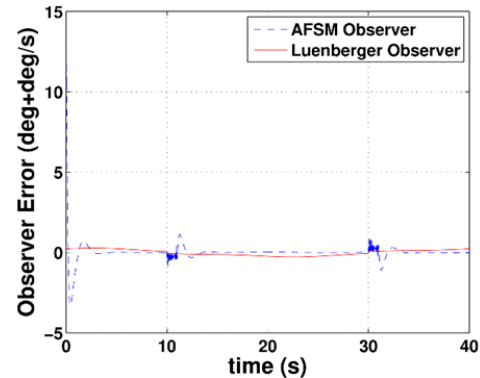
Fig. 8 Comparison of output tracking error between AFSM algorithm and PID controller for (a) Joint 1, (b) Joint 2, (c) Joint 3



(a)



(b)



(c)

Fig. 9 Comparison of observer error between adaptive fuzzy observer and Luenberger observer for (a) Joint 1, (b) Joint 2, (c) Joint 3

VIII. CONCLUSION

An observer-based direct AFSM control methodology is proposed for a class of MIMO nonlinear systems. In the controller design, the input gain matrix is used to combine the sliding surfaces. Then, the direct AFSM is designed based on the combined sliding surfaces. Therefore, in the proposed controller, there is no need to consider the matrix \mathbf{G} as a diagonal matrix. Also, a robust controller is designed to compensate the fuzzy estimation error. Also, the membership functions of the controller fuzzy system are adjusted adaptively, which was defined by the try and error procedure in the previous works.

The observer estimates the state tracking error. The advantage is that the measured outputs do not need to be considered as the first entries of a canonical-form state vectors; a limitation which exists in many of conventional output feedback AFSM methods. In the proposed observer, the adaptive fuzzy algorithm is used for approximation of nonlinear part and the observer error is used as the input of fuzzy system.

The chattering phenomenon which occurred because of discontinues terms in the control law, was eliminated by defining the boundary layers in the neighborhood of the sliding surfaces. Finally, the stability is guaranteed based on a Lyapunov-based approach.

REFERENCES

- [1] Mohsen Farahani, Soheil Ganjefar, Intelligent power system stabilizer design using adaptive fuzzy sliding mode controller, *Neurocomputing*, 226, (2017), 135-144.
- [2] A. Al-khazraji, N. Essounbouli, A. Hamzaoui, F. Nollet, J. Zaytoon, Type-2 fuzzy sliding mode control without reaching phase for nonlinear system, *Engineering Applications of Artificial Intelligence* 24 (1) (2011) 23 – 38.
- [3] O. Cerman, P. Huek, Adaptive fuzzy sliding mode control for electrohydraulic servo mechanism, *Expert Systems with Applications* (0) (2012).
- [4] Samir Zeghlache, Tarak Benslimane, Abderrahmen Bouguerra, Active fault tolerant control based on interval type-2 fuzzy sliding mode controller and nonlinear adaptive observer for 3-DOF laboratory helicopter, *ISA Transactions*, (2017).
- [5] Lanwei Zhou, Guoping Chen, Fuzzy sliding mode control of flexible spinning beam using a wireless piezoelectric stack actuator, *Applied Acoustics*, 128, (2017), 40-44.
- [6] S.-C. Lin, Y.-Y. Chen, Design of adaptive fuzzy sliding mode for nonlinear system control, in: *IEEE International Conference on Fuzzy Systems*, Vol. 1, 1994, pp. 35–39.
- [7] B. Yoo, W. Ham, Adaptive fuzzy sliding mode control of nonlinear system, *IEEE Transactions on Fuzzy Systems* 6 (2) (1998) 315–321.
- [8] A. Gholami, Amir H.D. Markazi, Direct adaptive fuzzy sliding observation and control, *Transactions of the Canadian Society for Mechanical Engineering*, Vol. 36, No. 4, 2012.
- [9] J. Wang, A. Rad, P. Chan, Indirect adaptive fuzzy sliding mode control: Part i: Fuzzy switching, *Fuzzy Sets and Systems* 122 (1) (2001) 21–30.
- [10] R.Wai, C. Lin, C. Hsu, Adaptive fuzzy sliding-mode control for electrical servo drive, *Fuzzy Sets and Systems* 143 (2) (2004) 295–310.
- [11] R. Wai, Fuzzy sliding-mode control using adaptive tuning technique, *IEEE Transactions on Industrial Electronics* 54 (1) (2007) 586–594.
- [12] Z. Chen, C. Shan, H. Zhu, Adaptive fuzzy sliding mode control algorithm for a non-affine nonlinear system, *IEEE Transactions on Industrial Informatics* 3 (4) (2007) 302–311.
- [13] R. Wai, M. Kuo, J. Lee, Design of cascade adaptive fuzzy sliding mode control for nonlinear two-axis inverted-pendulum servomechanism, *IEEE Transactions on Fuzzy Systems* 16 (5) (2008) 1232–1244.
- [14] J. Hwang, H. Kwak, G. Park, Adaptive interval type-2 fuzzy sliding mode control for unknown chaotic system, *Nonlinear Dynamics* (2010) 1–12.
- [15] H. S. Haghghi, A. H. Markazi, Chaos prediction and control in mems resonators, *Communications in Nonlinear Science and Numerical Simulation* 15 (10) (2010) 3091 – 3099.
- [16] S. Tong, H.-X. Li, Fuzzy adaptive sliding-mode control for mimo nonlinear systems, *IEEE Transactions on Fuzzy Systems* 11 (3) (2003) 354–360.
- [17] S. Aloui, O. Pages, A. El Hajjaji, A. Chaari, Y. Koubaa, Improved fuzzy sliding mode control for a class of mimo nonlinear uncertain and perturbed systems, *Applied Soft Computing Journal* 11 (1) (2011) 820–826.
- [18] A. Poursamad, A. Markazi, Adaptive fuzzy sliding-mode control for multi-input multi-output chaotic systems, *Chaos, Solitons and Fractals* 42 (5) (2009) 3100–3109.
- [19] J. Park, G. Park, Adaptive fuzzy observer with minimal dynamic order for uncertain nonlinear systems, *IEE Proceedings: Control Theory and Applications* 150 (2) (2003) 189–197.
- [20] Atta Oveisi, Tamara Nestorović, Robust observer-based adaptive fuzzy sliding mode controller, *Mechanical Systems and Signal Processing*, 76–77, (2016), 58-71.
- [21] Jun He, Minzhou Luo, Qingqing Zhang, Jianghai Zhao, Linsen Xu, Adaptive Fuzzy Sliding Mode Controller with Nonlinear Observer for Redundant Manipulators Handling Varying External Force, *Journal of Bionic Engineering*, 13 (4), (2016), 600-611.
- [22] T.-H. S. Li, Y.-C. Huang, MIMO adaptive fuzzy terminal sliding-mode controller for robotic manipulators, *Information Sciences* 180 (2010) 4641–4660.
- [23] Ramy Rashad, Ayman El-Badawy, Ahmed Aboudonia, Sliding mode disturbance observer-based control of a twin rotor MIMO system, *ISA Transactions*, 69, (2017), 166-174.
- [24] Donghong Ning, Shuaishuai Sun, Lidui Wei, Bangji Zhang, Haiping Du, Weihua Li, Vibration reduction of seat suspension using observer based terminal sliding mode control with acceleration data fusion, *Mechatronics*, (44), (2017), 71-83.
- [25] S. Tong, H. Li, W. Wang, Observer-based adaptive fuzzy control for siso nonlinear systems, *Fuzzy Sets and Systems* 148 (3) (2004) 355–376.
- [26] S. Sastry, M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*, Prentice-Hall, New-Jersey, 1989.
- [27] M. Biglarbegian, W. Melek, J. Mendel, Design of novel interval type- 2 fuzzy controllers for modular and reconfigurable robots: Theory and experiments, *IEEE Transactions on Industrial Electronics* 58 (4) (2011) 1371–1384.
- [28] F. Lewis, L. K., A. Yesildirek, Neural net robot controller with guaranteed tracking performance, *IEEE Transactions on Neural Networks* 6 (3) (1995) 703–715.
- [29] Z. Iwai, I. Mizumoto, L. Liu, S. Shah, H. Jiang, Adaptive stable pid controller with parallel feedforward compensator, in: *9th International Conference on Control, Automation, Robotics and Vision*, no. pt 1, 2007, pp. 1–6.