Journal of AI and Data Mining
Vol 4, No 2, 2016, 169-176

10.5829/idosi.JAIDM.2016.04.02.05

# Adaptive fuzzy pole placement for stabilization of non-linear systems

A. Karami-Mollaee

*Department of Electrical & Robotic Engineering, Shahrood University of Technology, Shahrood, Iran.*

Received 13 January 2015; Accepted 08 December 2016
**Corresponding author: akarami@shahroodut.ac.ir (A. Karami-Mollaee).*

**Abstract**
A new approach is proposed for the pole placement of non-linear systems using the state feedback and fuzzy system. We use a new online fuzzy training method in order to identify and obtain a fuzzy model for an unknown non-linear system using only the system input and output. Then we linearize this identified model at each sampling time to have an approximate linear time-varying system. In order to stabilize the linear system obtained, we first choose the desired time-invariant closed-loop matrix, and then a time-varying state feedback is used. The behavior of the closed-loop non-linear system is regarded as a linear time-invariant (LTI) system. Therefore, the advantage of the proposed method is the global asymptotical exponential stability of unknown non-linear systems. Due to the high speed convergence of the proposed adaptive fuzzy training method, the closed-loop system is robust against uncertainty in system parameters. Finally, a comparison is made with the boundary layer sliding mode control (SMC).

**Keywords:** *Fuzzy Identification, Pole Placement, Non-linear Control, Switch Reluctance Motor, Sliding Mode Control.*

## 1. Introduction

Control of non-linear systems is still a challenging area in the literature of control system theory, and some efforts have been made to study this subject [1]. However, most of them can be only applied to a certain class of non-linear systems. For instance, feedback linearization is only applicable to a class of non-linear systems that meet the involutivity condition and can be transformed to the companion form [1]. Many other methods have some limitations. For example, chattering is the most important problem in the sliding mode control (SMC) [2]. An intelligent approach such as the fuzzy systems and neural networks can help us solve these problems and limitations [3-6].

In addition, many efforts have been made to extend the linear control schemes to non-linear systems. On such a method is gain-scheduling control, which is designed based on a finite number of linearized models at each operating point [7,8], also called multiple-model adaptive control (MMAC) [8]. In [7], a MMAC neural network method is used to control non-linear systems. This method is expensive in terms of training and computation, and moreover, its stability has not been proved [17]. Another simple method in linear controller design is pole placement. When all of the state variables of a system are completely controllable, the closed-loop poles of the system can be placed in arbitrary locations on the phase plane using the state feedback with appropriate gains [1,9]. Some efforts have been directed toward computational methods of finding a feedback gain, and many numerical algorithms have been proposed [10-13]. In these methods, the minimal numerical operations are at least proportional to the cube of the system dimension [13]. To eliminate these time-consuming computations, neural networks have been proposed [14-16]. For example, in [10,15], the Sylvester equation has been solved using the recurrent neural networks. In all of these methods, stability of the closed-loop system has not been proved. In [17], a method for pole placement of non-linear systems has been presented based on the recurrent neural network, and the stability of the closed-loop system has been proved. However, the plant model has been assumed to be known. In [19], a method has been presented using the Takagi–Sugeno (TS) fuzzy systems based on the linear matrix inequality (LMI). Thus this approach cannot be online, i.e. first of all, LMI should be solved. The proposed approach in [20] is based upon solution of the Diophantine equations. However, the stability of the closed-loop non-linear system has not been proved. The approach mentioned in [20] is only for discrete-time dynamic plants. Other methods have been suggested in [18-20].

In this work, we proposed a method for pole placement of non-linear systems using the fuzzy systems, which eliminate the time-consuming computations of MMAC. We assumed that the non-linear system model was unknown, and that the system states were not accessible. Closed-loop stability was also proved. Since the non-linear plant model was unknown, we first identified its model using a fuzzy system, and then this identified model was linearized at any time to obtain a linear time-varying system. As shown in [1,17], the eigenvalues were not the stability criteria for the linear time-varying systems. Thus we applied a time-varying state feedback to this time-varying linear system such that the closed-loop linear system was time-invariant at any time.

The rest of this paper has been organized into six sections. The system model and problem formulation are described in section 2. In section 3, we present the system identification procedure. In section 4, linearization of the non-linear system and state feedback are explained. Finally, in section 5, we discuss the simulation and comparison results to verify the theoretical concepts presented in the previous sections. The conclusion is given in section 6.

## 2. Problem formulation

The eigenvalues are not the criteria used for the stability of linear time-varying systems [1,17]. For example, consider the following matrix:

$$A(t)=\begin{bmatrix} -1+1.5\cos^2(t) & 1-1.5\sin(t)\cos(t) \\ -1-1.5\sin(t)\cos(t) & -1+1.5\sin^2(t) \end{bmatrix} \tag{1}$$

For any $t$, the eigenvalues are $\lambda_{1,2}=-0.25(1\pm j\sqrt{7})$. However, the linear system $\dot{x}=A(t)x$ is not stable. To overcome this problem, we proposed a new approach, which was depicted in figure 1. We first chose a fixed closed-loop matrix $A_{cl}$, and then calculated the feedback gain at each sampling time for the linearized identified model of non-linear system. Consider the following single input non-linear system:

$$\begin{aligned} \dot{x}_i &= x_{i+1} : i=1,2,\ldots,n-1 \\ \dot{x}_n &= f(x,u) \\ y &= x_1 \end{aligned} \tag{2}$$

such that $y$ is the measurable system output, $x=[x_1,x_2\ldots,x_n]^T$ is the inaccessible vector state, and $u$ is the input control signal. Note that the function $f(x,u)$ is unknown. The other form of this equation is as follows:

$$\begin{aligned} \dot{x} &= Ax+Bg(x,u) \\ y &= C^T x \end{aligned} \tag{3}$$

where:

$$g(x,u)=f(x,u)+\sum_{i=1}^{n} a_i x_i \tag{4}$$

and:

$$A=\begin{bmatrix} 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \ldots & 0 & 1 \\ -a_1 & -a_2 & \ldots & -a_{n-1} & -a_n \end{bmatrix}, B=\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, C=\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \tag{5}$$

Assume that $a_i$ is such that $A$ is a Hurwitz matrix, and that the pair $(A,B)$ is controllable, and $(A,C)$ is observable.

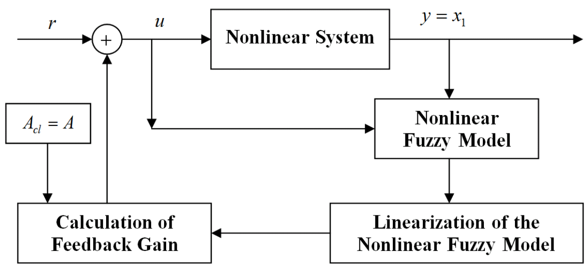


**Figure 1. Structure of the proposed approach.**

## 3. System identification

According to the fuzzy theorems, the Gaussian fuzzy basis functions (GFBF) can approximate any real continuous function with arbitrary accuracy. This means that GFBF has a universal approximation property [4]. Due to the approximation capability of GFBF, there exists an ideal weight vector $w$ with arbitrary large enough dimension $m$ such that the system (3) can be written as follows:

$$\begin{aligned} \dot{x} &= Ax+Bg \\ g &= w^T\xi(x,u)+\varepsilon_x \\ y &= C^T x \end{aligned} \tag{6}$$

where, $\varepsilon_x$ is an arbitrary small reconstruction error with bound $B_\varepsilon$, i.e. $|\varepsilon_x| < B_\varepsilon$.Moreover, $\hat{w} \in R^m$ is the weight vector estimate of fuzzy rules, and $\xi(.): R^{n+1} \to R^m$ is the Gaussian membership function (GMF) vector. Based on (6), to estimate the non-linear function $g(x,u)$, a singleton fuzzifier with product inference engine and a defuzzifier as weight sum of each output rule is used.

Now, the following estimator could be proposed:

$$\dot{\hat{x}} = A\hat{x} + B\left(\hat{w}^T \xi(\hat{x},u) + k_x(y-\hat{y})\right) + T(y-\hat{y}) \quad (7)$$

where, $\hat{x} = [\hat{x}_1, \hat{x}_2 \ldots, \hat{x}_n]^T$ is the identified model state vector, and matrix $A$ and observer gain vector $T \in R^{n\times 1}$ are chosen such that $A_s = A - TC^T$ is stable, i.e. for any symmetric positive definite matrix $Q$, there exists a symmetric positive definite matrix $P$ satisfying the following Lyapunov equation:

$$A_s^T P + PA_s = -Q \quad (8)$$

By subtracting (7) from (6), we obtain:

$$\begin{aligned}\dot{\tilde{x}} &= A\tilde{x} + B\left(w^T \xi(x,u) + \varepsilon_x\right) \\ &\quad - B\left(\hat{w}^T \xi(\hat{x},u) + k_x(y-\hat{y})\right) - T(y-\hat{y})\end{aligned} \quad (9)$$

in which $\tilde{x}(t) = x(t) - \hat{x}(t)$ and $\tilde{w} = w - \hat{w}$ are the state and parameter estimation errors.

$$\begin{aligned}\dot{\tilde{x}} &= A_s\tilde{x} + B\big(w^T \xi(x,u) - \hat{w}^T \xi(\hat{x},u) \\ &\qquad - k_x(y-\hat{y}) + \varepsilon_x\big) = \\ &A_s\tilde{x} + B\big(-k_x(y-\hat{y}) + \varepsilon_x - w^T \xi(\hat{x},u) \\ &\qquad + w^T \xi(\hat{x},u) - k_x(y-\hat{y}) + \varepsilon_x\big) = \\ &A_s\tilde{x} + B\left(\tilde{w}^T \hat{\xi} + w^T \tilde{\xi} - k_x(y-\hat{y}) + \varepsilon_x\right)\end{aligned} \quad (10)$$

such that: $\tilde{\xi} = \xi - \hat{\xi}$, $\xi = \xi(x,u)$, and $\hat{\xi} = \xi(\hat{x},u)$.

Theorem 1: Using the following adaptive weight law:

$$\dot{\hat{w}} = k_w \hat{\xi}\, \tilde{y} - 4k_e k_w |\tilde{y}|\, \hat{w} \quad (11)$$

the estimation error $\tilde{x}(t)$ converges to zero if $k_x \to \infty$. $k_w$ and $k_e$ are the arbitrary positive scalar constants, and $\tilde{y} = y - \hat{y}$.

Proof: Consider the following Lyapunov function:

$$V(t) = \frac{1}{2}\tilde{x}^T P \tilde{x} + \frac{1}{2k_w}\tilde{w}^T \tilde{w} \quad (12)$$

Taking the derivative of $V(t)$ yields:

$$\dot{V}(t) = \frac{1}{2}\dot{\tilde{x}}^T P \tilde{x} + \frac{1}{2}\tilde{x}^T P \dot{\tilde{x}} + \frac{1}{k_w}\tilde{w}^T \dot{\tilde{w}} \quad (13)$$

Substituting (8) and (10) in the above equation follows that:

$$\begin{aligned}\dot{V}(t) &= -\frac{1}{2}\tilde{x}^T Q\tilde{x} + \tilde{x}^T PB(w^T \tilde{\xi} + \varepsilon_x) \\ &\quad - k_x\tilde{x}^T CC^T\tilde{x} + \tilde{w}^T\left(\frac{1}{k_w}\dot{\tilde{w}} + \hat{\xi} B^T P\tilde{x}\right)\end{aligned} \quad (14)$$

Using the equality $\dot{\tilde{w}} = -\dot{\hat{w}}$, and tuning law (11) in the above equation leads to:

$$\begin{aligned}\dot{V}(t) &= -\frac{1}{2}\tilde{x}^T Q\tilde{x} + \tilde{x}^T PB(w^T \tilde{\xi} + \varepsilon_x) \\ &\quad - k_x\tilde{x}^T CC^T\tilde{x} + 4k_e|\tilde{y}|\tilde{w}^T\hat{w}\end{aligned} \quad (15)$$

Now consider the logical assumptions that the actual weight $w$ is norm bounded, and, moreover, the activation functions $\xi$ and $\hat{\xi}$ in (7) are chosen such that their norm be bounded also (see (32)) i.e. $\|w\| \le B_w$ and $\|\xi\| \le B_\xi$ or $\|\hat{\xi}\| \le B_\xi$. Therefore, we can write: $\left|w^T\tilde{\xi} + \varepsilon\right| \le 2B_w B_\xi + B_\varepsilon$. Here, considering the properties of the positive definite matrices $Q$ and $P$, and using $\hat{w} = w - \tilde{w}$, the above equation yields:

$$\begin{aligned}\dot{V}(t) &\le -\left(0.5\underline{\sigma}(Q) + k_x\right)\|\tilde{x}\|^2 \\ &\quad + \bar{\sigma}(PB)(2B_w B_\xi + B_\varepsilon)\|\tilde{x}\| \\ &\quad - 4k_e\left(\|\tilde{w}\|^2 - B_w\|\tilde{w}\|\right)|\tilde{y}|\end{aligned} \quad (16)$$

Now, we define $B_{\tilde{x}}$ as follows:

$$\begin{aligned}B_{\tilde{x}} &= \frac{\bar{\sigma}(PB)(2B_w B_\xi + B_\varepsilon) + k_e B_w^2\|C\|}{0.5\underline{\sigma}(Q) + k_x} \\ &= \frac{\bar{\sigma}(PB)(2B_w B_\xi + B_\varepsilon) + k_e B_w^2}{0.5\underline{\sigma}(Q) + k_x}\end{aligned} \quad (17)$$

where, $\bar{\sigma}$ and $\underline{\sigma}$ denote the maximum and minimum singular values, respectively. Therefore:

$$\begin{aligned}\dot{V}(t) &\le -\left(0.5\underline{\sigma}(Q) + k_x\right)\left(\|\tilde{x}\| - B_{\tilde{x}}\right)\|\tilde{x}\| \\ &\quad - 4k_e\left(\|\tilde{w}\| - \tfrac{1}{2}B_w\right)^2|\tilde{y}|\end{aligned} \quad (18)$$

or:

$$\dot{V}(t) \le -\left(0.5\underline{\sigma}(Q) + k_x\right)\left(\|\tilde{x}\| - B_{\tilde{x}}\right)\|\tilde{x}\| \quad (19)$$

Take: $\omega(t) = \left(0.5\underline{\sigma}(Q) + k_x\right)\left(\|\tilde{x}\| - B_{\tilde{x}}\right)\|\tilde{x}\|$, and suppose: $\|\tilde{x}\| > B_{\tilde{x}}$. Then one can write: $\dot{V} \le -\omega(t) \le 0$, and from (12), one can write $V(t) > V(0)$. Therefore, $\tilde{x}$, $\tilde{\xi}$, and $\tilde{w}$ are bounded when $\|\tilde{x}\| > B_{\tilde{x}}$. Moreover, it is easy to show that $\ddot{V}$ is bounded when $\|\tilde{x}\| > B_{\tilde{x}}$ because it is clear that $\ddot{V}$ is also dependent on $\tilde{x}$, $\tilde{\xi}$ and

$\tilde{w}$ i.e. $\dot{V}(t)$ is uniformly continuous. Integration of $\dot{V} \le -\omega(t) \le 0$ from zero to $t$ yields:

$$0 \le \int_0^t \omega(\tau)d\tau \le \int_0^t \omega(\tau)d\tau + V(t) \le V(0) \quad (20)$$

when $t \to \infty$, the above integral exists, and is less than or equal to $V(0)$. Since $V(0)$ is positive and finite, according to the Barbalat's lemma [1], we have:

$$\lim_{t\to\infty} \omega(t) = \lim_{t\to\infty} \left(0.5\underline{\sigma}(Q) + k_x\right)\left(\|\tilde{x}\| - B_{\tilde{x}}\right)\|\tilde{x}\| = 0 \quad (21)$$

Since $\left(0.5\underline{\sigma}(Q) + k_x\right)$ is greater than zero, (21) implies decreasing $\|\tilde{x}\|$ until it reaches $B_{\tilde{x}}$, whose result is $\lim_{t\to\infty} \|\tilde{x}\| = B_{\tilde{x}}$.

This guarantees that $B_{\tilde{x}}$ is the lower bound of $\|\tilde{x}\|$, and it is clear that $\lim_{k_x\to\infty} B_{\tilde{x}} = 0$. Then $\|\tilde{x}\|$ or $\tilde{x}$ will converge to zero if $k_x \to \infty$. The result of this theorem can be written as:

$$\lim_{\substack{k_x\to\infty \\ t\to\infty}} \tilde{x} = 0 \quad (22)$$

## 4. State feedback

According to (7) and due to the convergence of fuzzy system based on (22), we have:

$$\dot{\hat{x}} = A\hat{x} + B\,\hat{w}^T \xi(\hat{x},u) \quad (23)$$

Using (4) and (23), we can write:

$$\dot{\hat{x}}_n = \hat{w}^T \xi(\hat{x},u) - \sum_{i=1}^{n} a_i x_i \quad (24)$$

Then an approximate instantaneous linear model is as follows [17]:

$$\dot{\hat{x}}_n = \sum_{i=1}^{n}\left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_i} - a_i\right)\hat{x}_i + \left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u}\right)u + H.O.T \quad (25)$$

where, $H.O.T$ is the higher order terms in the Taylor series, which can be considered as the perturbation, and can be neglected [1,17]. Note that this linear system is not time-invariant because the gain vector $w$ varies with time, and as we aforementioned, the eigenvalues are not the criteria for stability.

Then we should apply a state feedback such that the closed system is linear time-invariant (LTI). To this end, we used the following state feedback:

$$u(t) = k(t)\hat{x}(t) + r(t) = \sum_{i=1}^{n} k_i(t)\hat{x}_i(t) + r(t)$$
$$k(t) = [k_1(t), k_2(t), \ldots, k_n(t)] \quad (26)$$

where, $r(t)$ is the new input control signal. Then:

$$\dot{\hat{x}}_n = \sum_{i=1}^{n}\left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_i} - a_i\right)\hat{x}_i + \left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u}\right)\left(\sum_{i=1}^{n} k_i(t)\hat{x}_i(t) + r(t)\right) =$$
$$\sum_{i=1}^{n}\left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_i} - a_i + \hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u} k_i(t)\right)\hat{x}_i + \left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u}\right) r(t) \quad (27)$$

We choose:

$$k_i(t) = -\left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_i}\right) \Big/ \left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u}\right) \quad (28)$$

Then:

$$\dot{\hat{x}}_n = \sum_{i=1}^{n}(-a_i)\hat{x}_i + \left(\hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u}\right) r(t) \quad (29)$$

or:

$$\dot{\hat{x}} = A\hat{x} + Gr \quad (30)$$

where, $A$ is as (5), and:

$$G = \begin{bmatrix} 0 & 0 & \cdots & 0 & \hat{w}^T \frac{\partial\xi(\hat{x},u)}{\partial u} \end{bmatrix}^T \in R^n \quad (31)$$

## 5. Simulation and comparison results

In the following examples, the proposed method is applied to a non-linear non-affine system to show the effectiveness of this approach. Consider the following one-phase model of switch reluctance motor (SRM) [21].

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = \frac{N_r^2 \psi_s}{J\,h^2(x_1)} \frac{dh(x_1)}{dx_1}\left\{1 - \left[1 + u\,h(x_1)\right]e^{-u\,h(x_1)}\right\} - \frac{T}{J} \quad (32)$$
$$h(x_1) = L_a + L_u \sin(x_1)$$

where, $x_1$ is the electrical angular position, $x_2$ is the mechanical angular velocity, $u$ is the stator current (input control signal), $T$ is the load torque, $\psi_s$ is the flux linkage, $J$ is the total rotor and load inertia, $N_r$ is the number of rotor poles, and $L_u$ and $L_u$ are the values for inductance at the aligned and un-aligned positions, respectively. In this work, the SRM parameters werechosen as: $N_r = 4$, $J = 0.07\,Kg.m^2$, $\psi_s = 0.1\,Wb$, $L_a = 180\,mH$, $L_u = 8\,mH$, and $T = 0.5\,N.M$.

The simulations were performed using MATLAB, with a sample time of 0.001. For the fuzzy system, we chose a GMF vector with three inputs $(\hat{x}_1, \hat{x}_2, u)$ and eleven rules as follow:

$$\xi_i(\hat{x}_1, \hat{x}_2, u) = \exp\left(-\left(\sqrt{\hat{x}_1^2 + \hat{x}_2^2 + u^2} - (6-i)\right)^2 \Big/ 6\right) \quad (33)$$

where, $i = 1,2,\ldots,11$. The output of defuzzifier is $\hat{g}$, and the fuzzy network tuning parameters were chosen as $k_w = 5$, $k_x = 100$ and $k_e = 30$. The other parameters were chosen as:

$$A = \begin{bmatrix} 0 & 1 \\ -9 & -8 \end{bmatrix}, T = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{34}$$

The initial conditions for the weight vector were chosen as $w(0) = [0,0,\ldots,0]^T$, and, moreover, $x(0) = [x_1(0), x_2(0)]^T = [2,1]^T$, and $r(t)$ is as a pulse function shown in figure 2. Figures 3 till 13 show the simulation results. We can see that the behavior of the closed-loop systems is as a linear system. Figures 3, 4 and 5 show the non-linear model of $f(x,u)$ and its fuzzy estimation. From figure 5, we can see the accuracy and precision of the proposed adaptive fuzzy system in estimation of the non-linear systems. Figures 6, 7 and 8 shows the input control signal of SRM and as we can see, its initial value is not large and has no oscillation. Figures 9 and 10 show the system states and their estimation. Finally, figures 11 and 12 demonstrate the behavior of the fuzzy systems. In these figures, the adaptive weight vectors are shown, i.e. the outputs of (11). In figure 13, the outputs of fuzzy membership are shown, i.e. the outputs of (33).

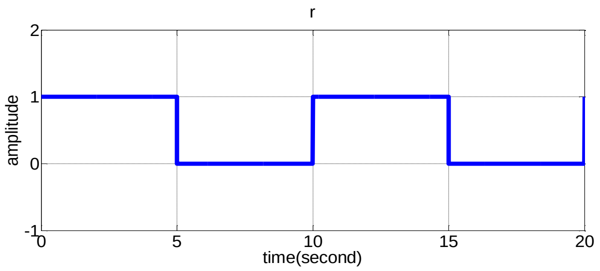


**Figure 2. Reference input.**

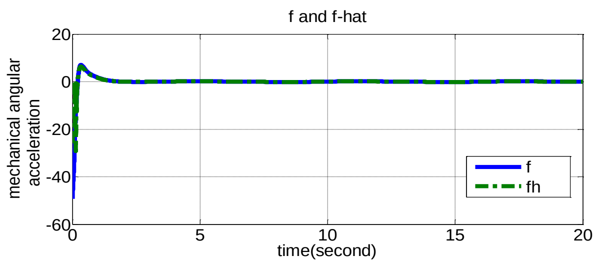


**Figure 3. Non-linear system and its fuzzy estimation.**

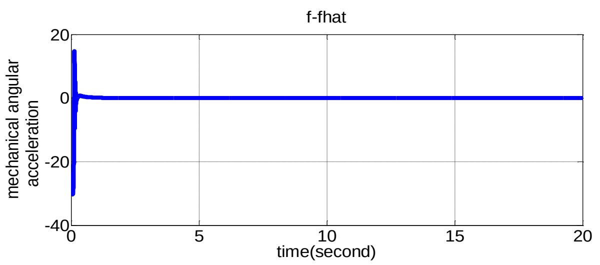


**Figure 4. Difference between non-linear system and its fuzzy estimation.**

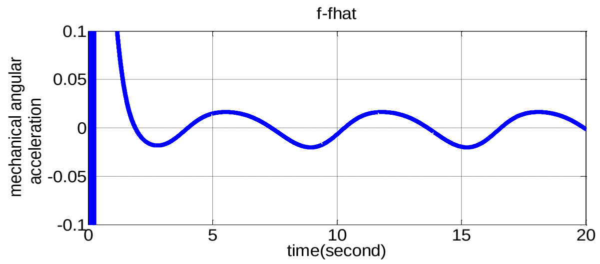


**Figure 5. Difference between non-linear system and its fuzzy estimation.**

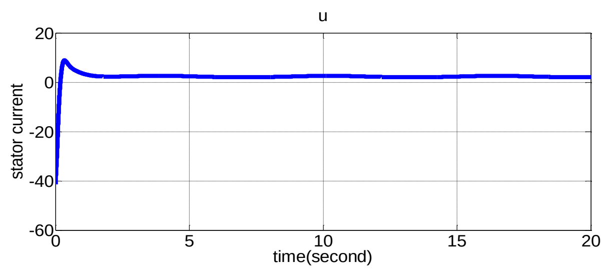


**Figure 6. Input control signal of non-linear system.**

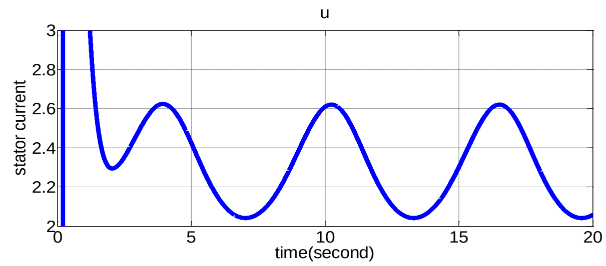


**Figure 7. Input control signal of non-linear system.**

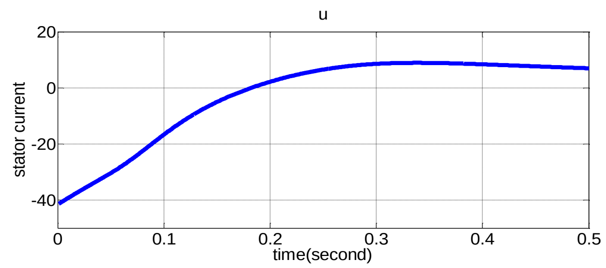


**Figure 8. Input control signal of non-linear system.**

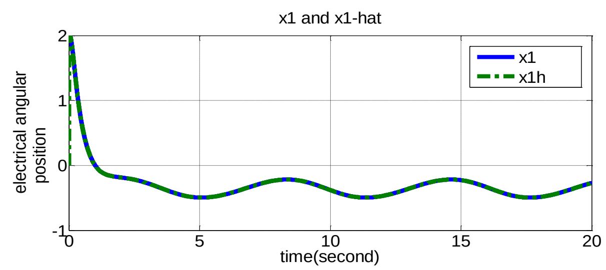


**Figure 9. First state and its estimation.**

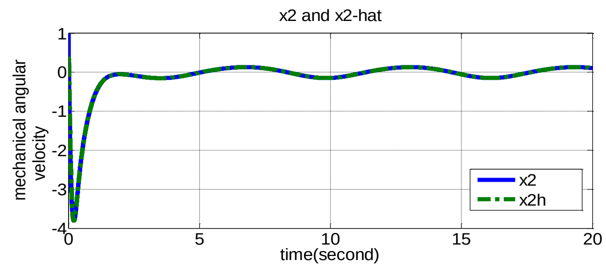


**Figure 10. Second state and its estimation.**

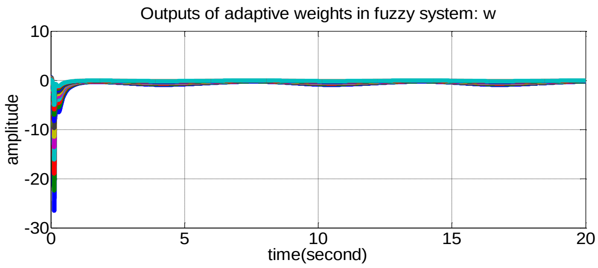


**Figure 11. Adaptive weight of fuzzy system.**

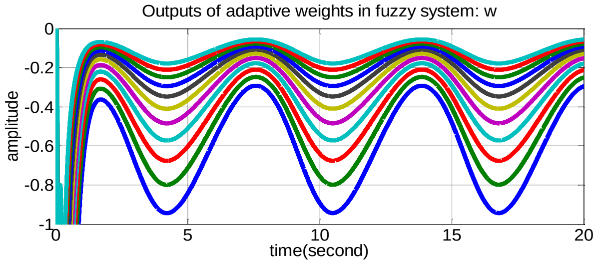


**Figure 12. Adaptive weight of fuzzy system.**

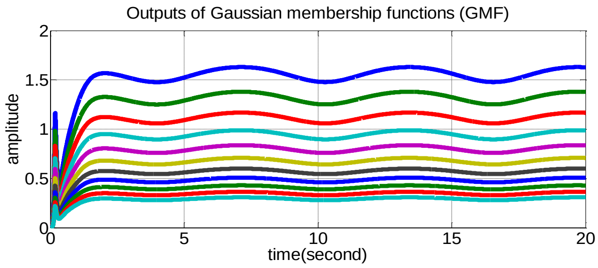


**Figure 13. Output of Gaussian membership functions.**

In order to compare the proposed method with the boundary layer sliding mode control (SMC), the following estimation of the sliding surface was defined:

$$\hat{s}=\sum_{i=1}^{n}\lambda_i\hat{x}_i,\lambda_n=1 \tag{35}$$

The coefficients $\lambda_i : i=1,2,\ldots,n$ should be chosen so that the following polynomial is Hurwitz:

$$P^{n-1}+\lambda_{n-1}P^{n-2}+\lambda_{n-2}P^{n-3}+\ldots+\lambda_2 P+\lambda_1=0 \tag{36}$$

The input control signal can be calculated using the following reaching law [1,2]:

$$\dot{\hat{s}}=-k\,sat(\hat{s}/\varphi) \tag{37}$$

whose result is:

$$u=\left(\hat{w}^T\frac{\partial\xi(\hat{x},u)}{\partial u}\right)^{-1}\left(-k\,sat(\hat{s}/\varphi)-\sum_{i=1}^{n-1}\lambda_i\hat{x}_i -\sum_{i=1}^{n}\left(\hat{w}^T\frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_i}-a_i\right)\hat{x}_i\right) \tag{38}$$

For the stability of the closed-loop system, the inequality $k>|H.O.T|$ should be satisfied [1,2,22]. However, the problem is that the terms $H.O.T$ are unknown. To solve this problem, a large $k$ should be chosen, which leads to chattering [2,22].

For SRM, (38) leads to:

$$u=\left(\hat{w}^T\frac{\partial\xi(\hat{x},u)}{\partial u}\right)^{-1}\left(-k\,sat(\hat{s}/\varphi)-\lambda_1\hat{x}_1-\hat{x}_2 -\left(\hat{w}^T\frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_1}-a_1\right)\hat{x}_1-\left(\hat{w}^T\frac{\partial\xi(\hat{x},u)}{\partial\hat{x}_2}-a_2\right)\hat{x}_2\right) \tag{39}$$

We chose the parameters as $\lambda_1=2,k=5,\varphi=0.1$. Figures 14 to 20 show the simulation results.

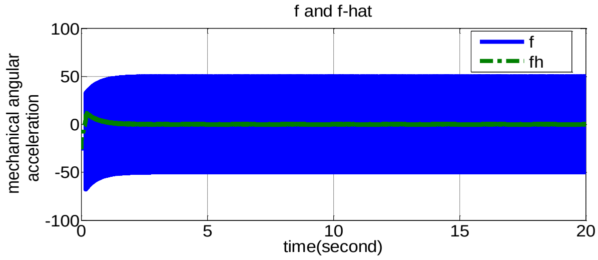


**Figure 14. Non-linear system and its fuzzy estimation.**

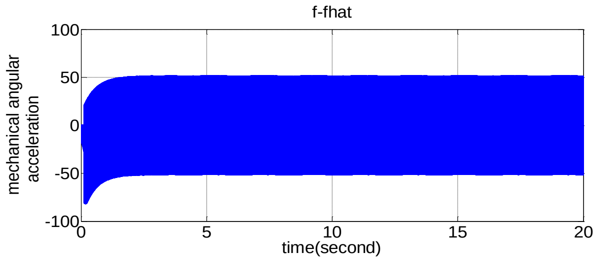


**Figure 15. Difference between non-linear system and its fuzzy estimation.**

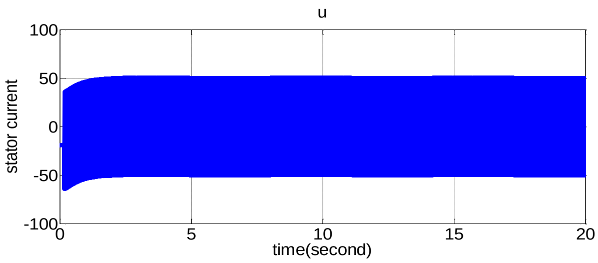


**Figure 16. Input control signal of non-linear system.**

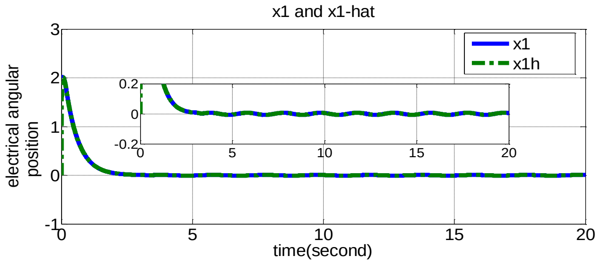


**Figure 17. First state and its estimation.**

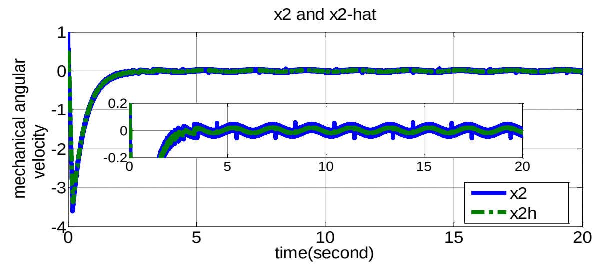


**Figure 18. Second state and its estimation.**

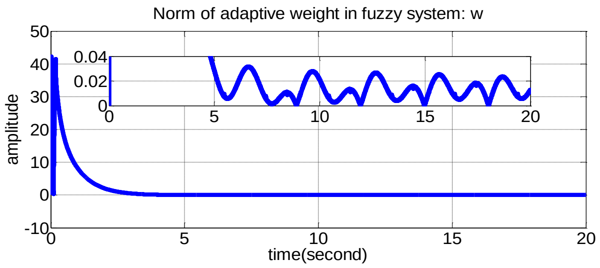


**Figure 19. Norm of adaptive weights of fuzzy system.**

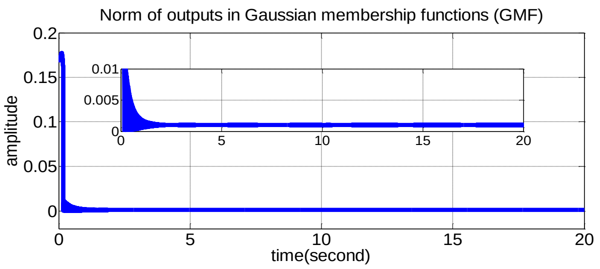


**Figure 20. Norm of output of Gaussian membership functions.**

We can see the effect of chattering in SMC, while the results in the state feedback are smooth. Another SMC drawback is its large value of input control signal, while the input control signal in state feedback is small and without chattering (compare Figures 6 and 16). We know that the convergence in both SMC and state feedback is asymptotical [1]. From figures 9, 10 and 17, 18, we can see that the only advantage of SMC is its faster convergence. In this example, the convergence time in SMC is about 2 seconds but in state feedback, it is about 4 seconds.

## 6. Conclusion

In this work, a new approach was proposed for the state feedback of unknown non-linear systems, which could lead to global asymptotical exponential stability. To have the unknown plant model, a fuzzy system was used, and an online adaptive training method was proposed using only the output system. In comparison with the existing approach, we first chose the closed-loop matrix, and then calculated the state feedback. Then the behavior of the closed-loop non-linear system is as a linear time invariant (LTI) system. Another advantage of the proposed method is its robustness against uncertainty in system parameters because the gain vector is computed online, and any drift and perturbation in parameters of the system affect this gain directly. This approach is simple in concept and realization. Finally, the proposed state feedback was compared with the boundary layer sliding mode control.

## References

[1] Slotine, J.-J. E. & Li, W. (1991). Applied nonlinear control. Prentice-Hall.

[2] Lee, H. & Utkin, V.-I. (2007). Chattering suppression methods in sliding mode control systems. Elsevier, Annual Review in Control, vol. 31, pp. 179-188.

[3] Norgaard, M., Ravn, O., Poulsen, N. K. & Hansen, L. K. (2001). Neural network for modeling and control of dynamic systems. Springer, New York.

[4] Tanaka, K. & Wang, H. O. (2001). Fuzzy control systems design and analysis. John Wily, Canada.

[5] Alibakhshi, F., Teshnehlab, M., Alibakhshi, M. & Mansouri, M. (2015). Journal of AI and Data Mining (JAIDM), vol. 3, no. 2, pp. 141-147.

[6] Fateh, M. M., Ahmadi, S. M. & Khorashadizadeh, S. (2014). Journal of AI and Data Mining (JAIDM), vol.2, no. 2, pp. 159-166.

[7] Ahmed, M. S. & Tasaddug, I. A. (1994). Neural-net control for nonlinear plants: design approach through linearization. IEE Proceeding of Control Theory Application, vol. 141, no. 5, pp. 315-332.

[8] Watanabe, K. (1992). Adaptive estimation and control, partitioning approach. Prentice-Hall, Englewood Cliffs, Nj.

[9] Chi-Tsong, C. (1999). Linear system theory and design. Oxford University Press, Oxford.

[10] Bhattacharyya, S. P. & de Souza, E. (1982). Pole assignment via Sylvester's equation. Elsevier, System and Control Letters, vol. 1, no. 4, pp. 261-263.

[11] Carotenuto, L., Franze, C. & Muraca, P. (2001). New computational procedure to the pole placement problem by static output feedback. IEE Preceding 20010771, pp. 466-471.

[12] Lee, T. H., Wang, Q. G. & Koh, E. K. (1994). An iterative algorithm for pole placement by output feedback. IEEE Transaction on Automatic Control, vol. 39, no. 3, pp. 565-568.

[13] Shafai, B. & Bhattacharyya, S. P. (1988). An algorithm for pole placement in high order multivariable systems. IEEE Transaction on Automatic Control, vol. 33, pp. 870-876.

[14] Wang, J. & Wu, G. (1995). Recurrent neural network for synthesizing linear control systems via pole assignment. International Journal of System Science, vol. 26, no. 12, pp. 2369-2382.

[15] Zhang, Z., Jiang, D. & Wang, J. (2002). A recurrent neural network for solving Sylvester equation with time-varying coefficient. IEEE Transaction on Neural Network, vol. 13, no. 5, pp. 1053-1063.

[16] Zhang, Z. & Wang, J. (2002). Global exponential stability of recurrent neural network for synthesizing linear feedback control systems via pole assignment.

IEEE Transaction on Neural Network, vol. 13, no. 3, pp. 633-644.

[17] Karami-Mollaee, A. & Karami-Mollaee, M. R. (2006). A new approach for instantaneous pole placement with recurrent neural networks and its application in control of nonlinear time-varying systems. Elsevier, System and Control Letters, vol. 55, no. 5, pp. 385-395.

[18] Zhu, Q. M. & Guo, L. Z. (2002). A pole placement controller for non-linear dynamic plants. Journal of Systems and Control Engineering, vol. 216, no. 6, pp. 467-476.

[19] Toulottea, P.-F., Delprata, S., Guerraa, T.-M. & Boonaert, J. (2008). Vehicle spacing control using robust fuzzy control with pole placement in LMI region. Elsevier, Engineering Applications of Artificial Intelligence, vol. 21, no. 5, pp. 756–768.

[20] Marinescu, B. (2010). Output feedback pole placement for linear time-varying systems with application to the control of nonlinear systems. Elsevier, Automatica, vol. 46, no. 9, pp. 1524–1530.

[21] Xu, J.-X. & Tan, Y. (2002). On the P-type and Newton-type ILC schemes for dynamic systems with non-affine-in-input factors. Elsevier, Automatica, vol. 38, pp. 1237-1242.

[22] Karami-Mollaee, A., Pariz, N. & Shanechi, H. M. (2011). Position control of servomotors using neural dynamic sliding mode. Transactions of the IEEE/ASME (American Society of Mechanical Engineering), Journal of Dynamic Systems, Measurement and Control, vol. 133, no. 6, pp. 141-150.

نشریه هوش مصنوعی و داده کاوی

# جایابی قطب تطبیقی فازی برای پایدارسازی سیستم‌های غیرخطی

**علی کرمی ملائی**

**دانشکده برق و رباتیک، دانشگاه صنعتی شاهرود، شاهرود، ایران**

ارسال ۲۰۱۵/۰۱/۱۳؛ پذیرش۲۰۱۵/۱۲/۰۸

**چکیده:**

در این مقاله، روشی جدید برای جایابی قطب در سیستم‌های غیرخطی با استفاده از فیدبک حالت و سیستم‌های فازی پیشنهاد می‌شود. در این روش، از رویکردی نو برای آموزش روی خط سیستم فازی به منظور شناسایی و بدست آوردن مدل نامشخص سیستم غیرخطی استفاده شده است. در روش پیشنهادی، فقط از ورودی و خروجی سیستم برای تخمین مدل فازی سیستم استفاده می‌شود. سپس مدل شناسایی شده را در هر زمان نمونه برداری، خطی نموده تا یک مدل تقریبی خطی متغیر با زمان بدست آوریم. برای پایدار سازی سیستم خطی بدست آمده، ابتدا ماتریس مورد نظر نامتغیر با زمان حلقه بسته را انتخاب نموده و سپس بهره فیدبک حالت متغیر با زمان را تعیین می‌کنیم. بنابراین رفتار سیستم حلقه بسته، مانند یک سیستم خطی نامتغیر با زمان خواهد بود. از مزایای روش پیشنهادی، پایداری نمایی مجانبی سراسری سیستم نامشخص غیرخطی است. به دلیل سرعت بالای همگرایی آموزش تطبیقی سیستم فازی، سیستم حلقه بسته نسبت به نامعینی‌های موجود در پارامترهای سیستم، مقاوم است. در نهایت روش پیشنهادی با کنترل حالت لغزشی لایه مرزی مقایسه شده است.

**کلمات کلیدی:** شناسایی فازی، جایابی قطب، کنترل غیرخطی، موتور مقاومت متغیر،کنترل حالت لغزشی.