2nd International Conference on Control, Instrumentation, and Automation
December 27-29, 2011 in Shiraz (IRAN)

Optimal Design of Adaptive Interval Type-2 Fuzzy Sliding Mode Control Using Genetic Algorithm

Mostafa Ghaemi, Mohammad-R. Akbarzadeh-T., Mohsen Jalaeian-F.

***Abstract*— in this paper, a stable indirect adaptive interval type-2 fuzzy sliding mode control (AIT2-FSMC) is introduced for a class of nonlinear systems. In the presence of uncertainties, especially under noisy and external disturbances, interval type-2 fuzzy system can be helpful in approximating unknown nonlinear system functions. To achieve more efficiency, the proposed controller is designed to use the targeted combination of sliding mode as a robust controller, interval type-2 fuzzy system as a universal approximator, and adaptive control law as an online parameter's tuner. The interval type-2 adaptation law is derived using Lyapunov approach, and mathematical analysis proves the closed loop system to be asymptotically stable. Although stability of the controller is provided via Lyapunov approach, optimization is required for performance improvement. Genetic algorithm (GA), as a population-based approach, is then used to optimize the parameters of the interval Type-2 fuzzy sets. Simulation analysis shows that the optimized IT2FSMC can reach improved performance.**

I. INTRODUCTION

Uncertainty is considered to be one of the biggest obstacles in controlling nonlinear systems. Generally there are two kinds of uncertainties, one caused by shortage of information about structure and parameters of the system and the other is about internal and external disturbances. One of the methods which can address uncertainty is sliding mode control (SMC) that is a robust nonlinear discontinuous feedback control technique with the drawback of chattering [1, 2]. The other method which can handle uncertainty in plant dynamics is adaptive control. In adaptive control, an adaptation law is introduced that adjusts the parameters of the controller against system uncertainties and disturbances. In the recent years, some techniques have been emerged for control of nonlinear systems, especially techniques based on fuzzy logic systems (FLS) [3-5]. Fuzzy logic provides an important tool for utilizing human expert knowledge in addition to mathematical knowledge. This is mainly due to the possibility of making use of fuzzy knowledge-based control to deal with systems whose dynamics are not so well understood and whose models can not be so conveniently established[6]. Many recent researches have utilized an adaptive fuzzy sliding mode control approach (AFSMC) to handle uncertainties and disturbances, such as in [7-10] which they use type-1 FLS. Type-1 FLSs are unable to directly handle rule uncertainties, since their membership functions are type-1 fuzzy sets. Whereas, type-2 FLSs involved in this paper, have capability to handle rule uncertainties. Therefore, hybrid combinations of the SMC, type-2 fuzzy logic, and adaptive control are an attractive approach for designing robust control systems with high degrees of non-linearities and uncertainties. In this paper, an AIT2 FSMC is introduced which use advantage of IT2 FLS and adaptive sliding mode controller; the stability analysis is proved under Lyapunov stability criteria. However using uniform MFs keep the stability, reaching better performance require optimizing the MFs. One of the best optimization methods is GA which introduced in 1975 by Holland [20]. Hence, we use GA to optimize the MF parameters of this attractive controller, therefore stability and good performance achieved and the simulation verifies these advantages.

Manuscript received September 30, 2011.
Mostafa Ghaemi, Mohammad-R. Akbarzadeh-T and Mohsen Jalaeian-F. are with Cognitive Computing Lab, Center for Applied Research on Intelligent Systems and Soft Computing, Dept. of Electrical Eng., Ferdowsi University of Mashhad, Iran (E-mails: Mo.ghaemi@yahoo.com, akbarzadeh@ieee.org, m.jalaeian@yahoo.com).

II. TYPE-2 FUZZY LOGIC SYSTEMS

A general type-2 fuzzy set, denoted $\tilde{A}$, is characterized by a type-2 membership function:

$$\tilde{A} = \int_{x \in X} \mu_{\tilde{A}}(x) / x \tag{1}$$

$$\mu_{\tilde{A}}(x) = \int_{u \epsilon J_x} f_x(u) / u \quad , \quad J_x \epsilon [0,1] \tag{2}$$

Where $\mu_{\tilde{A}}(x)$ is called a secondary membership function (MF) and $f_x(u)$ is called secondary grade. The domain of a secondary MF is called the primary membership of x noted J_x where $J_x \epsilon [0,1]$ for $\forall x \epsilon X$; u is a fuzzy set in [0, 1]. When $f_x(u) = 1$ for $\forall u \epsilon J_x$, the general type-2 MF in (1, 2) is called interval type-2 MF and rewrite as:

$$\tilde{A} = \int_{x \in X} \mu_{\tilde{A}}(x) / x = \int_{x \epsilon X} [\int_{u \epsilon J_x} 1/u] / x \ , J_x \epsilon [0,1] \tag{3}$$

Because of calculation simplicity, especially in the type reduction, many researchers use interval type-2 fuzzy sets [11, 12] instead of general type-2 fuzzy sets.

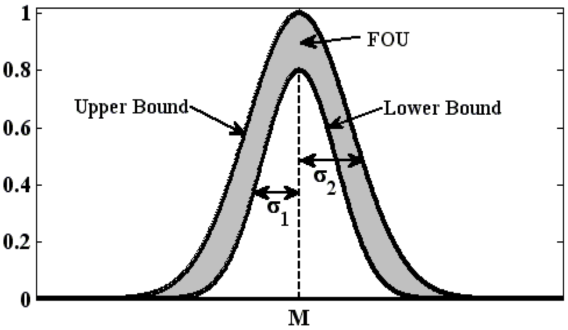

Fig. 1. Interval Type-2 fuzzy membership function(MF)

An IT2 FS is described by its lower $\underline{\mu}_{\tilde{A}}(x)$ and upper $\overline{\mu}_{\tilde{A}}(x)$ MFs. The footprint of uncertainty (FOU) for an IT2 FS is described in terms of these MFs, as

$$FOU(\tilde{A}) = \bigcup_{x \in X} [\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)] \quad (4)$$

Fig.1 illustrates a type-2 fuzzy MF with its details such as upper bound, lower bound, standard divisions (δ_1, δ_2), mean (m) and footprint of uncertainties (FOU).

Generally, an IT2 FLS consists of fuzzifier; fuzzy rule base; fuzzy inference engine; type reducer and defuzzifier; Fig.2 shows the general structure of an IT2 FLS, and in following sections we will introduce each block of IT2 FLS.

A. Fuzzifier

The fuzzifier maps a crisp input into fuzzy sets, these fuzzy sets, in general, can be type-1 or type-2 fuzzy input sets, $\tilde{A}$. In this paper, we use only singleton fuzzifier which its output will be a single point of a nonzero membership.

B. Rule Base

The knowledge of experts will be placed in this part. The l^{th} rule in the IT2 FLS can be written as bellow

R^l : If x_1 is $\tilde{F}_1^l$ and x_2 is $\tilde{F}_2^l$ and … x_p is $\tilde{F}_p^l$ then y is $\tilde{G}^l$ l =1,2,…,M (5)

Where x_i $(i = 1, 2, \ldots, p)$ and y are input and output of the IT2 FLS respectively and $\tilde{F}_p^i$ and $\tilde{G}^i$ are the antecedent and the consequent sets respectively, which both can be IT2 FSs.

C. Inference Engine

The inference engine combines rules and maps input vector $\underline{x} = (x_1, x_2, \ldots, x_n)^T$ to an output scalar y. by performing input and antecedent operations, the firing set will be obtain as following

$$F^i(\underline{x}) = \prod_{j=1}^{n} \mu_{\tilde{F}_j^i}(x_j) \quad (6)$$

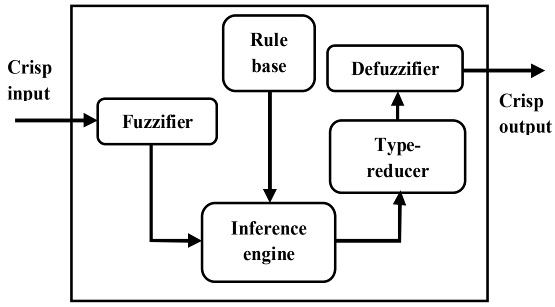

Fig. 2. Structure of an interval type-2 fuzzy logic system

Whereas, in this paper only IT2 FSs are used, depend on upper and lower MFs, the firing set will be as below

$$F^i(\underline{x}) = [\underline{f}^i(\underline{x}), \overline{f}^i(\underline{x})] \quad (7)$$

$$\underline{f}^i(\underline{x}) = \underline{\mu}_{\tilde{F}_1^i} * \underline{\mu}_{\tilde{F}_2^i} * \ldots * \underline{\mu}_{\tilde{F}_n^i} \quad (8)$$

$$\overline{f}^i(\underline{x}) = \overline{\mu}_{\tilde{F}_1^i} * \overline{\mu}_{\tilde{F}_2^i} * \ldots * \overline{\mu}_{\tilde{F}_n^i} \quad (9)$$

Where $\underline{f}^i(\underline{x})$ and $\overline{f}^i(\underline{x})$ are the i^{th} lower and upper membership functions, respectively, and * is product t-norm.

D. Type Reduction and Defuzzification

Due to output of the inference engine is type-2 fuzzy sets, before defuzzification, a type-reducer is needed to convert IT2 FSs into type-1 sets. Type reduction was proposed by Karnik and Mendel [13, 14]. In [15] five different type reduction methods are described which are based on computing the centroid of an IT2 FS. The center of sets (COS) type-reduction is widely used because Eq. (10) may be computed easily using the Karnik-Mendel iterative method [16]. The center of sets (COS) type reduction can be defined as

$$Y_{cos} = \left(Y^1, \ldots, Y^M, F^1, \ldots, F^M\right) = \int_{y^1} \cdots \int_{y^M} \int_{f^1} \cdots \int_{f^M} 1 / \frac{\sum_{i=1}^{M} f^i y^i}{\sum_{i=1}^{M} f^i} \quad (10)$$

Whereas, in this paper we only use IT2 FS then Y_{cos} is determined with its left end point, y_l, and its right end point y_r. In [17] Karnik and Mendel (KM) developed two algorithms for calculating these two end points. We defuzzify the output of COS type reducer by using the average of y_l and y_r [11], therefore crisp output is

$$y(x) = \frac{y_l(x) + y_r(x)}{2} \quad (11)$$

If we use singleton fuzzifier, product inference engine and COS type reducer, y_l and y_r can expressed as

$$y_l = \frac{\sum_{i=1}^{M} f^i y_l^i}{\sum_{i=1}^{M} f^i} = \theta_l^T \xi_l \quad (12)$$

$$y_r = \frac{\sum_{i=1}^{M} f^i y_r^i}{\sum_{i=1}^{M} f^i} = \theta_r^T \xi_r \quad (13)$$

where $\theta_l^i = y_l^i, \theta_r^i = y_r^i, \theta_l = [\theta_l^1, \ldots, \theta_l^M]^T$,

$\xi_l^i = \frac{f^i}{\sum_{i=1}^{M} f^i}$ and $\xi_l = [\xi_l^1, \ldots, \xi_l^M]^T$.

now the defuzzified crisp output obtain as

$$y(x) = \frac{y_r + y_l}{2} = \frac{1}{2}(\theta_l^T \xi_l + \theta_r^T \xi_r) \quad (14)$$

III. Sliding Mode Control

Consider a general class of n'th order SISO nonlinear system as

$$x^{(n)} = f(\underline{x}, t) + g(\underline{x}, t)u + d(t) \quad , \quad y = x \quad (15)$$

Where f and g are unknown bounded nonlinear functions , where the bounds need not be known, $u \epsilon R$ and $y \epsilon R$ are input and output of the system, respectively. $\underline{x} = [x, \dot{x}, \ldots, x^{(n-1)}] \epsilon R^n$ is the state vector of the system which is assumed to be available for measurement, $d(t)$ is the unknown external disturbance which is bounded by a known constant D, i.e.

$$d(t) \le D \quad (16)$$

And assume, System (15) is controllable, $g(\underline{x}, t) \neq 0$ .Without loss of generality, we assume $g(\underline{x}, t) > 0$, i.e. can be negative and the control can be similarly derived. The control objective is to determine a feedback control $u = u(\underline{x})$ such that the state $\underline{x}$ of the system follows the desired state vector $\underline{x}_d = [x_d, \dot{x}_d, \ldots, x_d^{(n-1)}]$ in the presence of disturbances and uncertainties, therefore the tracking error is

$$\underline{e} = \underline{x} - \underline{x}_d = [e, \dot{e}, \ldots, e^{(n-1)}]^T \quad (17)$$

error should converge zero. Then a sliding surface in the space of the error state can be defined as

$$s(\underline{e}) = \underline{c}^T \underline{e} = e^{(n-1)} + c_{n-1} e^{(n-2)} + \ldots + c_1 e \quad (18)$$

Where $\underline{c} = [c_1, \ldots, c_{(n-2)}, c_{(n-1)}, 1]^T \epsilon R^n$ are the coefficients of the Hurwitz polynomial $h(r) = \lambda^{(n-1)} + c_{(n-1)} \lambda^{(n-2)} + \ldots + c_1$, i.e. all the roots are in the open left-hand (λ is a Laplace operator). If the initial condition $\underline{e}(0) = 0$ then the tracking problem can be considered as the state error vector remaining on the sliding surface $s(\underline{e}) = 0$ for $\forall t \ge 0$. A sufficient condition that the system controlled is stable is given in [1] as:

$$\frac{1}{2} \frac{d}{dt} s^2(\underline{e}) \le -\eta |s| \ , \ \eta > 0 \quad (19)$$

Where η is a constant design parameter. Then sliding condition of (19) can be rewritten as:

$$s\dot{s} \le -\eta |s| \quad \text{or} \quad \dot{s} \le -\eta sgn(s) \quad (20)$$

$$sgn(s) = \begin{cases} 1 & for\ s > 0 \\ 0 & for\ s = 0 \\ -1 & for\ s < 0 \end{cases} \quad (21)$$

By taking the time derivative of both sides of (18), we obtain:

$$\dot{s} = \sum_{i=1}^{n-1} c_i e^i + x^{(n)} - x_d^{(n)} = \sum_{i=1}^{n-1} c_i e^i + f(\underline{x}, t) + g(\underline{x}, t)u + d(t) - x_d^{(n)} \quad (22)$$

Substituting (22) into (20), sliding condition can be re-expressed as:

$$\left(\sum_{i=1}^{n-1} c_i e^i + f(\underline{x}, t) + g(\underline{x}, t)u + d(t) - x_d^{(n)} \right) \le -\eta sgn(s) \quad (23)$$

The control problem is to obtain the optimal control input u^* which guarantees the sliding condition (23). If $f(\underline{x}, t), g(\underline{x}, t)$ and $d(t)$ are known, we can design the optimal sliding mode control law as below:

$$u^* = \frac{1}{g(\underline{x}, t)} [-\sum_{i=1}^{n-1} c_i e^i - f(\underline{x}, t) - d(t) + x_d^{(n)} - \eta_\Delta sgn(s)] \quad (24)$$

Where $\eta_\Delta \ge \eta > 0$.

IV. Adaptive Sliding Mode Control Based On IT2 FLS

Since $f(\underline{x}, t)$, $g(\underline{x}, t)$ and $d(t)$ in (24) are unknown, we replace $f(\underline{x}, t)$ and $g(\underline{x}, t)$ by the IT-2 FLS $\hat{f}(\underline{x} | \theta_f)$ and $\hat{g}(\underline{x} | \theta_g)$ which are in the form (14). The resulting control input signal is as

$$u_I = u = \frac{1}{\hat{g}\left(\underline{x} \mid \theta_g\right)}[-\sum_{i=1}^{n-1} c_i e^i - \hat{f}\left(\underline{x} \mid \theta_f\right) + x_d^{(n)} - (\eta_\Delta + D) sgn(s)] \quad (25)$$

where D obtain from (16) and

$$\hat{f}\left(\underline{x} \mid \theta_f\right) = \frac{\hat{f}_l + \hat{f}_r}{2} = \frac{1}{2}\left(\theta_{fl}^T \xi_l + \theta_{fr}^T \xi_r\right) = \theta_f^T \xi_f \quad (26)$$

$$\hat{g}\left(\underline{x} \mid \theta_g\right) = \frac{\hat{g}_l + \hat{g}_r}{2} = \frac{1}{2}\left(\theta_{gl}^T \xi_l + \theta_{gr}^T \xi_r\right) = \theta_g^T \xi_g \quad (27)$$

Theorem1. Consider the nonlinear system (15) and the control input u in (25) if the adaptive laws are chosen as

$$\dot{\theta}_{fl} = \gamma_1 s \xi_{fl} \quad (28)$$

$$\dot{\theta}_{fr} = \gamma_1 s \xi_{fr} \quad (29)$$

$$\dot{\theta}_{gl} = \gamma_2 s \xi_{gl} u \quad (30)$$

$$\dot{\theta}_{gr} = \gamma_2 s \xi_{gr} u \quad (31)$$

The closed loop system signals are uniformly bounded and the tracking error will converge to zero asymptotically.

Proof. Define the optimal parameters of fuzzy system as [18]:

$$\theta_f^* = \arg\min_{\theta_f \epsilon \Omega_f}[\sup_{x \epsilon R^n} |\hat{f}\left(\underline{x} \mid \theta_f\right) - f\left(\underline{x}, t\right)|] \quad (32)$$

$$\theta_g^* = \arg\min_{\theta_g \epsilon \Omega_g}[\sup_{x \epsilon R^n} | \hat{g}\left(\underline{x} \mid \theta_g\right) - g\left(\underline{x}, t\right)] \quad (33)$$

Where Ω_f and Ω_g are constant sets for θ_f and θ_g respectively, and they are defined as $\Omega_f = \{\theta_f \epsilon R^n \| \theta_f | \le M_f\}$ and $\Omega_g = \{\theta_g \epsilon R^n | 0 < |\theta_g| \le M_g\}$, where M_f and M_g are positive constant. The minimum approximation error is defined as:

$$\omega = \left[f\left(\underline{x}, t\right) - \hat{f}\left(\underline{x} \mid \theta_f^*\right)\right] + \left[g\left(\underline{x}, t\right) - \hat{g}\left(\underline{x} \mid \theta_g^*\right)\right] \quad (34)$$

Then from substituting (25) and (34) into (22), derivative of sliding surface is

$$\dot{s} = \sum_{i=1}^{n-1} c_i e^i + f\left(\underline{x}, t\right) + g\left(\underline{x}, t\right) u + d\left(t\right) - x_d^{(n)}$$

$$= \sum_{i=1}^{n-1} c_i e^i + f\left(\underline{x}, t\right) + \left(g\left(\underline{x}, t\right) - \hat{g}\left(\underline{x} \mid \theta_g\right)\right) u + \hat{g}\left(\underline{x} \mid \theta_g\right) u + d\left(t\right) - x_d^{(n)}$$

$$= f\left(\underline{x}, t\right) - \hat{f}\left(\underline{x} \mid \theta_f\right) + \left(g\left(\underline{x}, t\right) - \hat{g}\left(\underline{x} \mid \theta_g\right)\right) u + d\left(t\right) - (\eta_\Delta + D) sgn(s)$$

$$= \hat{f}\left(\underline{x} \mid \theta_f^*\right) - \hat{f}\left(\underline{x} \mid \theta_f\right) + (\hat{g}\left(\underline{x} \mid \theta_g^*\right) - \hat{g}\left(\underline{x} \mid \theta_g\right)) u + d\left(t\right) - \left(\eta_\Delta + D\right) sgn\left(s\right) + \omega$$

$$= \left(\theta_f^{*T} \xi_f - \theta_f^T \xi_f\right) + \left(\theta_g^{*T} \xi_g - \theta_g^T \xi_g\right) u + d(t) - \left(\eta_\Delta + D\right) sgn\left(s\right) + \omega$$

$$= \varphi_f^T \xi_f + \varphi_g^T \xi_f u + d(t) - \left(\eta_\Delta + D\right) sgn\left(s\right) + \omega$$

$$= \frac{1}{2}(\varphi_{fl}^T \xi_{fl} + \varphi_{fr}^T \xi_{fr}) + \frac{1}{2}(\varphi_{gl}^T \xi_{gl} + \varphi_{gr}^T \xi_{gr}) u + d(t) - \left(\eta_\Delta + D\right) sgn\left(s\right) + \omega \quad (35)$$

Where $\varphi_f = \theta_f^* - \theta_f$ and $\varphi_g = \theta_g^* - \theta_g$

Now the Lyapunov function is defined as:

$$V = \frac{1}{2} s^2 + \frac{1}{4\gamma_1} \varphi_{fl}^T \varphi_{fl} + \frac{1}{4\gamma_1} \varphi_{fr}^T \varphi_{fr} + \frac{1}{4\gamma_2} \varphi_{gl}^T \varphi_{gl} + \frac{1}{4\gamma_2} \varphi_{gr}^T \varphi_{gr} \quad (36)$$

Where γ_1 and γ_2 are positive constant. The time derivative of V is:

$$\dot{V} = s\dot{s} + \frac{1}{2\gamma_1}(\varphi_{fl}^T \dot{\varphi}_{fl} + \varphi_{fr}^T \dot{\varphi}_{fr}) + \frac{1}{2\gamma_2}(\varphi_{gl}^T \dot{\varphi}_{gl} + \varphi_{gr}^T \dot{\varphi}_{gr})$$

$$= s(\frac{1}{2}(\varphi_{fl}^T \xi_{fl} + \varphi_{fr}^T \xi_{fr}) + \frac{1}{2}(\varphi_{gl}^T \xi_{gl} + \varphi_{gr}^T \xi_{gr}) u + d(t) - \left(\eta_\Delta + D\right) sgn\left(s\right) + \omega) + \frac{1}{2\gamma_1}(\varphi_{fl}^T \dot{\varphi}_{fl} + \varphi_{fr}^T \dot{\varphi}_{fr}) + \frac{1}{2\gamma_2}(\varphi_{gl}^T \dot{\varphi}_{gl} + \varphi_{gr}^T \dot{\varphi}_{gr})$$

$$= \frac{1}{2\gamma_1} \varphi_{fl}^T \left(\dot{\varphi}_{fl} + \gamma_1 s \xi_{fl}\right) + \frac{1}{2\gamma_1} \varphi_{fr}^T \left(\dot{\varphi}_{fr} + \gamma_1 s \xi_{fr}\right) + \frac{1}{2\gamma_2} \varphi_{gl}^T \left(\dot{\varphi}_{gl} + \gamma_2 s \xi_{gl} u\right) + \frac{1}{2\gamma_2} \varphi_{gr}^T \left(\dot{\varphi}_{gr} + \gamma_2 s \xi_{gr} u\right) + sd\left(t\right) - s\left(\eta_\Delta + D\right) sgn\left(s\right) + s\omega \quad (37)$$

Where $\dot{\varphi}_{fr} = -\dot{\theta}_{fr}$, $\dot{\varphi}_{fl} = -\dot{\theta}_{fl}$, $\dot{\varphi}_{gr} = -\dot{\theta}_{gr}$ and $\dot{\varphi}_{gl} = -\dot{\theta}_{gl}$, substituting (28), (29), (30) and (31) into (37), then we have

$$\dot{V} = sd\left(t\right) - s\left(\eta_\Delta + D\right) sgn\left(s\right) + s\omega = sd\left(t\right) - |s|\left(\eta_\Delta + D\right) + s\omega \le -|s|\left(\eta_\Delta\right) + s\omega$$

(38)

to be based on the approximation theorem [19], it can be anticipated that the term $s\omega$ should be very small if it not equals to zero in the IT-2 FLS, we obtain

$$\dot{V} = -|s|\left(\eta_\Delta\right) \leq 0 \tag{39}$$

Since the c_i's are all real and are chosen such that $h\left(\lambda\right)=\sum_{i=1}^{n} c_i \lambda^{(i-1)}$ is a Hurwitz polynomial, and using corollary of Barbalat's lemma [24], we have $\lim_{t\to\infty}|s(e)|=0$, therefore $\lim_{t\to\infty}|e(t)|=0$, the proof is completed.

V. The Genetic Algorithm Optimization Procedure

In this study, Genetic algorithm (GA) is used to optimize the MF parameters of IT FSs. GA was first introduced by Holland in 1975 [20]. GA is a population based parallel search algorithm; based on the mechanism of natural genetic and selection. GA operates without knowledge of the task domain and utilizes only the fitness of evaluated individuals. Generally, three basic operators of GA are reproduction, crossover and mutation. They can be considered as a general purpose optimization method and have been successfully applied to search and optimization problems [21-23]. Here binary GA method is used as the aim of optimization that is described below.

Let us consider θ, a vector of parameters that is target of the optimization, so determined θ contains standard division (δ_{1i}, δ_{2i}), mean (m_i) and amplitudes of lower MF (Γ_i) see Fig.1. It means $\theta=[m_i, \delta_{i1}, \delta_{i2}, \Gamma_i]$, $i=1,\ldots,n$ where the number of MFs is n and θ is a vector of as many parameters as quadruple of the number of determined MFs.

GA operates on chromosomes, which are binary strings, but the main problem must work out with real numbers, hence artificial genetic algorithm has two spaces, genotype and phenotype. Genotype, the space GA work on, consists in chromosomes and phenotype consists in real number; which each number decodes from one gene. Here, the fitness of each chromosome is generated by monitoring the overall mean squared error of the close loop control system, made by the individual. Also the mutation and crossover operate on individuals of each generation and generate new child for constitute the new generations.

Some constraints are considered based on GA-convergence speed:

$$\begin{aligned} &\text{C1: } 0.01<\delta_{1i}<\delta_{2i}<2 \quad , i=1,2,\ldots,n \\ &\text{C2: } -3<m_1<m_2<\ldots<m_n<3 \\ &\text{C3: } 0.1<\Gamma_i<1 \end{aligned} \tag{40}$$

In next section we apply the optimized MFs to a nonlinear system. Also, uniform MFs is used in the second run, and simulation results verify the advantages of the optimized MFs in the comparison of uniform ones.

VI. Simulation Example

The example is a regulation problem to let the output of a first order nonlinear system to track a constant trajectory, so consider a first order system as:

$$\dot{x} = \frac{1-e^{-x}}{1+e^{x}} + u\left(t\right) + d\left(t\right), \; d(t)=\cos(5t)$$
$$y(t)=x(t) \tag{41}$$

Where $u\left(t\right)$ is the input control signal and $d\left(t\right)$ is a bounded disturbance with known bound D=2, the initial state $x\left(0\right)=1$. We defined $s=e$, the desired trajectory $x_d=0$, step size is 0.01s and $\eta_\Delta=0.5$. Three optimized memberships over interval [-3, 3] are as follow (fig.):

$$\begin{aligned} &\bar{\mu}_1=\frac{1}{1+\exp((x+0.56)/0.13)}, \; \underline{\mu}_1=\frac{0.48}{1+\exp((x+0.56)/0.13)} \\ &\bar{\mu}_2=\exp(\frac{-(x+0.08)^2}{0.07}), \; \underline{\mu}_2=0.46\times\exp(-\frac{(x+0.08)^2}{0.05}) \\ &\bar{\mu}_3=\frac{1}{1+\exp(-(x-1.24)/0.88)}, \; \underline{\mu}_3=\frac{0.87}{1+\exp(-(x-1.24)/0.88)} \end{aligned} \tag{42}$$

Where $\bar{\mu}_i$ and $\underline{\mu}_i$ are upper bound and lower bound membership functions, respectively. The initial consequent parameters $\theta_{fl}(0)$ and $\theta_{fr}(0)$ are chosen uniformly over interval [-2.2, 1.8] and [-1.2, 2.8], respectively. Let the learning rate $\gamma_1=25$. Fig.3 shows optimized MFs using GA after 400 generation. Fig.4 (a) and Fig.4 (b) show that the controller achieves good control performance; Simulation results show effectiveness of optimized method. From Fig.4 (b) we can see the integral squared error (ISE) of the optimized AIT2 FSMC is less than uniform one.

VII. Conclusions

In this paper, an adaptive interval type-2 fuzzy sliding mode controller for a class of nonlinear systems is designed, and its MFs are optimized using GA. We introduced the type-2 fuzzy logic system to approximate the unknown nonlinear terms. Simulation results show that the controller achieves good control performance and guarantees system stability and optimized type-2 controller has better performance.

Refrences

[1] J. E. Slotine and W. Li "Applied Nonlinear Control," *Englewood Cliffs, NJ: Prentice Hall,* 1991.

[2] J. J. Slotine and S. S. Sastry, "Tracking Control of Non-Linear Systems Using Sliding Surfaces, with Application To Robot Manipulators," *International Journal of Control,* vol. 38, pp. 465-492, 1983.

[3] T. Yu, "Adaptive robust fuzzy control for output tracking," in *Proceedings of the 2004 American Control Conference*, 2004, pp. 1788-1793 vol.2.
[4] D. Velez-Diaz and T. Yu, "Adaptive robust fuzzy control of nonlinear systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* vol. 34, pp. 1596-1601, 2004.
[5] M. Wang, B. Chen, and S.-L. Dai, "Direct adaptive fuzzy tracking control for a class of perturbed strict-feedback nonlinear systems," *Fuzzy Sets and Systems,* vol. 158, pp. 2655-2670, 2007.
[6] Y. K. Kim and G. J. Jeon, "Error Reduction of Sliding Mode Control Using Sigmoid-Type Nonlinear Interpolation in the Boundary Layer," *International Journal of Control, Automation, and Systems,* vol. 2, pp. 523-529, December 2004.
[7] H. Shiuh-Jer and L. Wei-Cheng, "Adaptive fuzzy controller with sliding surface for vehicle suspension control," *IEEE Transactions on Fuzzy Systems,* vol. 11, pp. 550-559, 2003.
[8] Z.-M. Chen, J.-G. Zhang, W.-C. Zhao, and J.-C. Zeng, "Adaptive Fuzzy Sliding Mode Control for Uncertain Nonlinear Systems," in *Proceedings of the Second International Conference on Machine Learning and Cybernetics*, Xi'an, 2-5 November 2003.
[9] R. Shahnazi and M. R. Akbarzadeh-T, "PI Adaptive Fuzzy Control With Large and Fast Disturbance Rejection for a Class of Uncertain Nonlinear Systems," *IEEE Transactions on Fuzzy Systems,* vol. 16, pp. 187-197, 2008.
[10] C.-C. Chiang and C.-C. Yang, "Robust Adaptive Fuzzy Sliding Mode Control for A Class of Uncertain Nonlinear Systems with Unknown Dead-Zone," in *IEEE International Conference on Fuzzy Systems* , Vancouver, July 16-21, 2006.
[11] L. Qilian and J. M. Mendel, "Interval type-2 fuzzy logic systems: theory and design," *IEEE Transactions on Fuzzy Systems,* vol. 8, pp. 535-550, 2000.
[12] L. Qilian and J. M. Mendel, "Interval type-2 fuzzy logic systems," in *The Ninth IEEE International Conference on Fuzzy Systems, 2000. FUZZ IEEE 2000*, 2000, pp. 328-333 vol.1.
[13] N. N. Karnik and J. M. Mendel, "Type-2 fuzzy logic systems: type-reduction," in *IEEE International Conference on Systems, Man, and Cybernetics*, 1998, pp. 2046-2051 vol.2.
[14] N. N. Karnik, J. M. Mendel, and L. Qilian, "Type-2 fuzzy logic systems," *IEEE Transactions on Fuzzy Systems,* vol. 7, pp. 643-658, 1999.
[15] N. N. Karnik and J. M. Mendel, "Type-2 fuzzy logic systems: type-reduction," in *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, 1998, pp. 2046-2051 vol.2.
[16] N. N. Karnik, J. M. Mendel, and L. Qilian, "Type-2 fuzzy logic systems," *Fuzzy Systems, IEEE Transactions on,* vol. 7, pp. 643-658, 1999.
[17] N. N. Karnik and J. M. Mendel, "Centroid of a type-2 fuzzy set," *Information Sciences,* vol. 132, pp. 195-220, 2001.
[18] C.-C. Chiang and C.-C. Yang "Robust Adaptive Fuzzy Sliding Mode Control for a Class of Uncertain Nonlinear Systems with Unknown Dead-Zone," in *Fuzzy Systems, 2006 IEEE International Conference on*, 2006, pp. 492-497.
[19] H. Ying "General interval type-2 Mamdani fuzzy systems are universal approximators," in *Fuzzy Information Processing Society, 2008. NAFIPS 2008. Annual Meeting of the North American*, 2008, pp. 1-6.
[20] J.H. Holland,"Adaptation in Natural and Artificial Systems", University of Michigan Press, MI, 1975.
[21] D. Hidalgo, P. Melin, and O. Castillo, "Optimal Design of Type-2 Fuzzy Membership Functions Using Genetic Algorithms in a Partitioned Search Space," in *Granular Computing (GrC), 2010 IEEE International Conference on*, pp. 212-216.
[22] S.-C. Lin and Y.-Y. Chen, "Design of self-learning fuzzy sliding mode controllers based on genetic algorithms," *Fuzzy Sets and Systems,* vol. 86, pp. 139-153, 1997.
[23] P. Melin, D. Sanchez, and L. Cervantes, "Hierarchical genetic algorithms for optimal type-2 fuzzy system design," in *Fuzzy Information Processing Society (NAFIPS), 2011 Annual Meeting of the North American*, pp. 1-6.
[24] J. Wang , A.B. Rad, and P.T. Chan, Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching, *Fuzzy Sets and Systems*; 122:21–30, 2001.

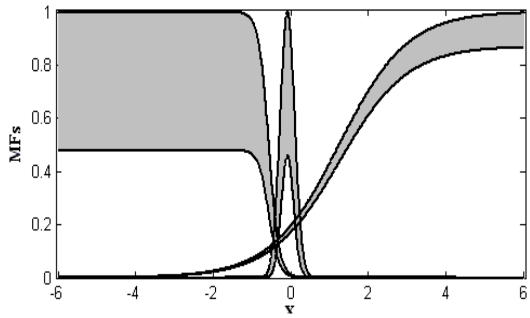

Fig 3. Optimized Membership Functions.

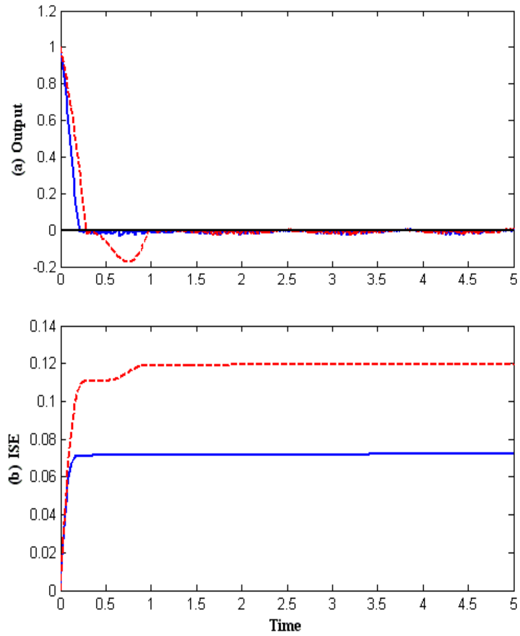

Fig 4. (a) System output for optimized MFs (solid) and uniform MFs (dashed); (b)The tracking performance $\int_{t=0}^{5} e^2 dt$ for optimized MFs (solid) and uniform MFs (dashed).