Controller Design of Nonlinear TITO Systems with Uncertain Delays via Neural Networks and Error Entropy Minimization

J. H. Zhang, A. P. Wang*, H. Wang**

Department of Automation, North China Electric Power University, Beijing, 102206, P. R. China,
(e-mail: zhangwu@ public.bta.net.cn).
**Institute of Computer Sciences, Anhui University, Anhui, P. R. China*
*** Control Systems Centre, The University of Manchester, PO Box 88, Manchester, M60, 1QD, UK*
(e-mail: hong.wang@manchester.ac.uk)

Abstract: In this paper, a novel control algorithm for nonlinear two input and two output (TITO) systems with random input and output delays is presented. Due to the stochastic characteristics induced by uncertain time delays, TITO feedback control systems are cast into a general framework, where the controllers are designed based upon minimizing the entropies of tracking errors. The controllers that have been implemented by BP neural networks are obtained without decoupling. The convergence in the mean square sense is analyzed. Simulation results show the effectiveness of the proposed approaches.

1. INTRODUCTION

Nonlinear dynamic multivariable systems with random input and output time delays exist widely in practical industrial control processes. Due to the existence of nonlinearity, multivariable and random delays, it is very difficult to design controllers for such complicated closed-loop systems. Up till now, little research has been performed to investigate this kind of systems. Fuzzy control method was developed in (Zhang et al., 2007), where Takagi-Sugeno (T-S) fuzzy model with a time-delay term was used to deal with the stabilization problems of nonlinear system with time delays. In this context, a guaranteed cost controller was designed via a state feedback and the stability condition was proposed by linear matrix inequality (LMI) techniques. Simulation results of two input and single output nonlinear systems with certain delays were given.

In this paper, a stochastic control method based on the tracking error entropy minimization is proposed. Because of the stochastic characteristics induced by uncertain delays, the controllers are designed under a general framework. Following the recent developments on the modeling and control via minimizing error entropy (Wang, 2000; Wang and Zhang, 2001; Wang, 2002; Guo and Wang, 2005a; Guo and Wang, 2005b; Yue and Wang, 2003; Erdogmus and Pricipe, 2002a; Erdogmus and Pricipe, 2002b), a feedback control strategy is proposed by neuro-PID controllers without decoupling. The innovation of this scheme can be summarized as follows: 1) the controllers are designed directly utilizing a new performance index constructed by entropies of tracking errors; 2) the stability condition of the feedback control system is given.

The remainder of this paper is organized as follows: Section 2 formulates the problem of designing controller for nonlinear TITO systems with random input/output delays. Section 3 develops the control algorithm; Section 4 presents a simulation example to illustrate the efficiency and feasibility of the proposed approach. The last section concludes this paper.

2. PROBLEM FORMULATION

Consider a nonlinear TITO system with uncertain input and output delays shown in Fig. 1, in which a nonlinear TITO plant are controlled by two neural controllers. In this figure, τ_{sc}^1 and τ_{sc}^2 are random delays from the sensor to the controller, τ_{ca}^1 and τ_{ca}^2 are random delays from the controller to the actuator. Moreover, r_1 and r_2 are the set-points, and ε_1 and ε_2 are the closed loop tracking errors.

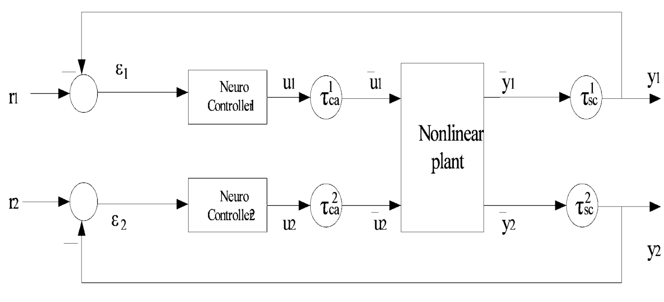

Fig.1. A nonlinear TITO system with uncertain delays

In this paper, the nonlinear ARMAX plant can be described as

$$\begin{cases} \bar{y}_{1k} = f_1(\bar{y}_{1,k-1}, \bar{y}_{1,k-2}, ..., \bar{y}_{1,k-n_1}, \bar{y}_{2,k-1}, \bar{y}_{2,k-2}, ..., \bar{y}_{2,k-\bar{n}_1}, \\ \quad \bar{u}_{1,k}, \bar{u}_{1,k-1}, ..., \bar{u}_{1,k-m_1}, \bar{u}_{2,k}, \bar{u}_{2,k-1}, ..., \bar{u}_{2,k-\bar{m}_1}) \\ \bar{y}_{2k} = f_2(\bar{y}_{1,k-1}, \bar{y}_{1,k-2}, ..., \bar{y}_{1,k-n_2}, \bar{y}_{2,k-1}, \bar{y}_{2,k-2}, ..., \bar{y}_{2,k-\bar{n}_2}, \\ \quad \bar{u}_{1,k}, \bar{u}_{1,k-1}, ..., \bar{u}_{1,k-m_2}, \bar{u}_{2,k}, \bar{u}_{2,k-1}, ..., \bar{u}_{2,k-\bar{m}_2}) \end{cases} \quad (1)$$

As such, the tracking errors are defined by

$$\varepsilon_{i,k} = r_{i,k} - y_{i,k} \quad i = 1, 2 .$$

Because these uncertain delays may not obey Gaussian distributions, it is necessary to consider higher order statistics rather than the mean and variance of tracking error when designing feedback controllers. Recent papers have addressed this issue both in control literature (Wang, 2000; Wang and Zhang, 2001; Wang, 2002; Guo and Wang, 2005a; Guo and Wang, 2005b; Yue and Wang, 2003) and in the signal processing and machine learning literature (Erdogmus and Pricipe, 2002a; Erdogmus and Pricipe, 2002b). It has been pointed out that in certain cases minimizing the error entropy is equivalent to minimizing the distance between the probability distributions of the desired and system outputs (Erdogmus and Pricipe, 2002a).

The controllers can be solved by minimizing the entropies of the two tracking errors in figure 1 as follows:

$$J_k(\Delta u_k) = H(\varepsilon_{1,k}) + H(\varepsilon_{2,k}) + H(\varepsilon_{1,k}, \varepsilon_{2,k}) \to \min \quad (2)$$

where Δu_k is the incremental output of the controller, $H(\varepsilon_{1,k})$ and $H(\varepsilon_{2,k})$ are the entropy of the tracking error $\varepsilon_{1,k}$ and $\varepsilon_{2,k}$ at k instant, respectively. Moreover, $H(\varepsilon_{1,k}, \varepsilon_{2,k})$ in (2) denotes the joint entropy of the two tracking errors.

The objective of the performance index is to make the shape of the probability density function (pdf) of each tracking error to be as narrow as possible. Moreover, the controller in each loop needs an integral operator in order to ensure that the steady state tracking error approaches to zero. Hence, in this paper neuro-PID controllers are utilized in each control loop.

Renyi extended the concept of entropy given by Shannon and the defined Renyi's α entropy is given by

$$H_\alpha(\varepsilon) = \frac{1}{1-\alpha} \log \int_{-\infty}^{\infty} \gamma_\varepsilon^\alpha(\varepsilon) de \quad (3)$$

The joint entropy of two random variables is calculated from

$$H_\alpha(x, y) = \frac{1}{1-\alpha} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \gamma_{x,y}^\alpha(x, y) dxdy \cdot$$

Renyi quadratic entropy ($\alpha = 2$) can be calculated in a non-parametric, practical and efficient way when the probability density function (PDF) is estimated by Parzen window with Gaussian kernels. Therefore, Renyi quadratic entropy in conjunction with Parzen window with Gaussian kernel is utilized here to read

$$H_2(\varepsilon) = -\log V(\varepsilon) \quad (4)$$

$$V(\varepsilon) = \int (\frac{1}{N} \sum_{i=1}^{N} \kappa(\xi - e_i, \sigma^2))^2 d\xi \quad (5)$$

where $V(\varepsilon)$ is named as information potential in (Erdogmus and Pricipe, 2002b), which can be calculated from the samples $\{e_1, e_2, ...e_N\}$ using Gaussian kernel in the following way.

$$V(\varepsilon) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \kappa(e_i - e_j, 2\sigma^2) \quad (6)$$

Since Renyi's quadratic entropy is a monotonic function of the information potential $V(\varepsilon)$, we can equivalently maximize the information potential $V(\varepsilon)$ instead of minimizing Renyi's entropy. As such, we present the following novel performance index shown in (2) so as to design the required controller

$$J_k(\Delta u_k) = \frac{1}{V(\varepsilon_{1,k}) + V(\varepsilon_{2,k}) + V(\varepsilon_{1,k}, \varepsilon_{2,k})} \to \min \quad (7)$$

i.e., $J_k(\Delta u_k) = -(V(\varepsilon_{1,k}) + V(\varepsilon_{2,k}) + V(\varepsilon_{1,k}, \varepsilon_{2,k})) \to \min$. The joint information potential $V(\varepsilon_{1,k}, \varepsilon_{2,k})$ can also be calculated from the samples of two tracking errors.

3. DESIGN OF NEURAL CONTROLLERS

The neural controller in each closed loop is served as an adaptive PID controller and its algorithm is structured in the following

$$u_k = u_{k-1} + \Delta u_k \quad (8)$$

$$\Delta u_k = k_p(\varepsilon_k - \varepsilon_{k-1}) + k_i \varepsilon_k + k_d(\varepsilon_k - 2\varepsilon_{k-1} + \varepsilon_{k-2}) \quad (9)$$

In this paper, the BP neural networks are utilized to implement the PID controller. Taking the neural controller 1 as an example, the control algorithm can be obtained.

For such a neural network, the input layer is defined as

$$O_q^{(1)} = \bar{x}(q) \quad q = 1, 2, ..., Q \quad (10)$$

where the tracking error ε_1 is a primary node in the input layer, the other nodes in the input layer may include the tracking error ε_2, the set-point r_1 and the transmitted output of plant y_1.

The input and output of the hidden layer are denoted by

$$net_p^{(2)}(k) = \sum_{q=0}^{Q} w_{pq}^{(2)} O_q^{(1)}(k) \quad (11)$$

$$O_p^{(2)}(k) = f(net_p^{(2)}(k)) \quad p = 1, 2...P \quad (12)$$

The input and output of the output layer are given by

$$net_l^{(3)}(k) = \sum_{p=0}^{P} w_{lp}^{(3)} O_p^{(2)}(k) \tag{13}$$

$$O_l^{(3)}(k) = g(net_l^{(3)}(k)) \quad l = 1,2,3 \tag{14}$$

$$O_1^{(3)}(k) = k_p \tag{15}$$

$$O_2^{(3)}(k) = k_i \tag{16}$$

$$O_3^{(3)}(k) = k_d \tag{17}$$

where $w_{pq}^{(2)}$ and $w_{lp}^{(3)}$ are weights trained in BP neural networks, and the activation functions have been defined that

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \qquad g(x) = \frac{e^x}{e^x + e^{-x}}.$$

In this paper, the adaptive criterion of the BP neural networks based controller is to minimize the sum of Renyi's entropies defined in (2) (or to maximize the sum of information potentials) due to the reason stated before. Using the steepest descent approach, the training algorithm can be obtained.

At k instant, the statistical information of the tracking error $\varepsilon_{1,k}$ and $\varepsilon_{2,k}$ is estimated by samples $\{e_1^1, e_2^1, ... e_N^1\}$ and $\{e_1^2, e_2^2, ... e_N^2\}$ respectively. The joint information potential is calculated from

$$V(\varepsilon_{1,k}, \varepsilon_{2,k}) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \kappa(e_i^1 - e_j^1, 2\sigma^2) \kappa(e_i^2 - e_j^2, 2\sigma^2).$$

For the output layer, the training of the weights are formulated as follows

$$\begin{aligned} \Delta w_{lp}(k) &= -\eta_1 \frac{\partial J_k(\Delta u_k)}{\partial w_{lp}} \\ &= \eta \frac{\partial [\, V(\varepsilon_{1,k}) + V(\varepsilon_{2,k}) + V(\varepsilon_{1,k}, \varepsilon_{2,k}) \,]}{\partial w_{lp}} \end{aligned} \tag{18}$$

where η_1 and η are the pre-specified learning factors. At this stage, it can be shown that

$$\begin{aligned} \frac{\partial [V(\varepsilon_{1,k})]}{\partial w_{lp}} &= \frac{1}{2N^2\sigma^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (e_j^1 - e_i^1) \cdot \\ &\quad \kappa(e_i^1 - e_j^1, 2\sigma^2)[\frac{\partial e_i^1}{\partial w_{lp}} - \frac{\partial e_j^1}{\partial w_{lp}}] \end{aligned} \tag{19}$$

where it can be further seen that

$$\begin{aligned} \frac{\partial e_i^1(k)}{\partial w_{lp}(k)} &= -\frac{\partial y_i^1(k)}{\partial w_{lp}(k)} \\ &= -\frac{\partial y_i^1(k)}{\partial u_1(k)} \frac{\partial u_1(k)}{\partial O_l^{(3)}(k)} \frac{\partial O_l^{(3)}(k)}{\partial net_l^{(3)}(k)} \frac{\partial net_l^{(3)}(k)}{\partial w_{lp}(k)} \\ & i = 1,2,...,N \end{aligned}$$

Similarly, it can be obtained $\frac{\partial [V(\varepsilon_{2,k})]}{\partial w_{lp}}$, $\frac{\partial [V(\varepsilon_{1,k}, \varepsilon_{2,k})]}{\partial w_{lp}}$ and $\frac{\partial e_i^2(k)}{\partial w_{lp}(k)}$.

The Jacobian information $\frac{\partial y_i^1(k)}{\partial u_1(k)}$ and $\frac{\partial y_i^2(k)}{\partial u_1(k)}$ in the above formulation can be replaced by $\mathrm{sgn}\frac{\partial y_i^1(k)}{\partial u_1(k)}$ and $\mathrm{sgn}\frac{\partial y_i^2(k)}{\partial u_1(k)}$ respectively. In addition, they can be calculated by using model prediction algorithm of the plant under following assumption.

$$\begin{aligned} \frac{\partial y_i^m(k)}{\partial u_1(k)} &= \frac{\partial y_i^m(k)}{\partial u_1(k - \tau_{ca}^k)} \frac{\partial u_1(k - \tau_{ca}^k)}{\partial u_1(k)} \\ &= \frac{\partial y_i^m(k)}{\partial u_1(k - \tau_{ca}^k)} \frac{\partial u_1(k-1)}{\partial u_1(k)} \frac{\partial u_1(k-2)}{\partial u_1(k-1)} \cdots \frac{\partial u_1(k - \tau_{ca}^k)}{\partial u_1(k - \tau_{ca}^k + 1)} \\ &\approx \frac{\partial y_i^m(k)}{\partial u_1(k - \tau_{ca}^k)} \\ m &= 1,2 \end{aligned} \tag{20}$$

For a hidden layer, the training of the relevant neural network weights can be formulated to give

$$\Delta w_{pq}(k) = \eta \frac{\partial [\, V(\varepsilon_{1,k}) + V(\varepsilon_{2,k}) + V(\varepsilon_{1,k}, \varepsilon_{2,k}) \,]}{\partial w_{pq}} \tag{21}$$

For the three terms in the right-hand-side of equation (21), the following calculations can be made

$$\begin{aligned} \frac{\partial [V(\varepsilon_{1,k})]}{\partial w_{pq}} &= \frac{1}{2N^2\sigma^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (e_j^1 - e_i^1). \\ &\quad \kappa(e_i^1 - e_j^1, 2\sigma^2)[\frac{\partial e_i^1}{\partial w_{pq}} - \frac{\partial e_j^1}{\partial w_{pq}}] \end{aligned} \tag{22}$$

where it can be shown that

$$\begin{aligned} \frac{\partial e_i^1(k)}{\partial w_{pq}(k)} &= -\frac{\partial y_i^1(k)}{\partial w_{pq}(k)} \\ &= -\frac{\partial y_i^1(k)}{\partial u_1(k)} \frac{\partial u_1(k)}{\partial O_l^{(3)}(k)} \frac{\partial O_l^{(3)}(k)}{\partial net_l^{(3)}(k)} \cdot \\ &\quad \frac{\partial net_l^{(3)}(k)}{\partial O_p^{(2)}(k)} \frac{\partial O_p^{(2)}(k)}{\partial net_p^{(2)}(k)} \frac{\partial net_p^{(2)}(k)}{\partial w_{pq}(k)} \\ & i = 1,2,...,N \end{aligned} \tag{23}$$

$\frac{\partial[V(\varepsilon_{2,k})]}{\partial w_{pq}}$ and $\frac{\partial[V(\varepsilon_{1,k},\varepsilon_{2,k})]}{\partial w_{pq}}$ can be obtained like equation (22). Moreover, $\frac{\partial e_i^2(k)}{\partial w_{pq}(k)}$ can be obtained like equation (23).

Obviously, another neural controller 2 can be designed in a similar way using the same performance index shown as (7).

The algorithm can be summarized as follows:

Step 1: Perform Monte Carlo test at instant k and obtain the entropies. A set of stochastic time delays are imposed on the system for calculating the samples of tracking errors $\{e_1^1(k), e_2^1(k), \ldots, e_N^1(k)\}$ and $\{e_1^2(k), e_2^2(k), \ldots, e_N^2(k)\}$. The entropies of the tracking errors $H(\varepsilon_{1,k})$, $H(\varepsilon_{2,k})$ and $H(\varepsilon_{1,k},\varepsilon_{2,k})$ can be obtained by (3),

Step 2: Train the two neural networks by updating the weights of BP neural networks using equations (18)-(23).

Step 3: Formulate the control input u_k using (8) and (9) after obtaining k_p, k_i, k_d via the BP neural networks.

Step 4: Apply control input u_k to actuate the plant, set $k \leftarrow k+1$, and go back to step 1.

The use of variable step-size gradient algorithm or adding momentum term can enable improved training of the BP neural networks. In addition, the kernel size (the window width of Parzen estimator) can be set experimentally after a preliminary analysis of the dynamic range of the tracking error.

Once the control algorithms are obtained, it is important to further analyze the stability of closed loop systems.

Theorem: If the parameter satisfies

$$0<\eta\le\frac{1}{2[\Gamma(t)+\bar{\varepsilon}]^2},$$

where $\bar{\varepsilon}$ is an arbitrarily small positive constant, $\Gamma(t)=V(\varepsilon_1(t))+V(\varepsilon_2(t))+V(\varepsilon_1(t),\varepsilon_2(t))$, then the control algorithm given by (18)-(23) will be convergent.

Proof: Select a Lyapunov function as follows

$$\pi=\frac{1}{\Gamma(t)+\bar{\varepsilon}}+(\Delta u(t))^2,$$

then it can be obtained that

$$\begin{aligned}\frac{\partial\pi}{\partial t}&=-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}\frac{\partial\Gamma(t)}{\partial u(t)}\frac{\partial u(t)}{\partial t}+2\Delta u(t)\frac{\partial u(t)}{\partial t}\\&=(-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}\frac{\partial\Gamma(t)}{\partial u(t)}+2\Delta u(t))\frac{\partial u(t)}{\partial t}\end{aligned} \tag{24}$$

Approximately, we have the following

$$\frac{\Delta\pi}{\Delta t}=(-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}\frac{\partial\Gamma(t))}{\partial u(t)}+2\Delta u(t))\frac{\Delta u(t)}{\Delta t} \tag{25}$$

Since the neural controller is designed by minimizing the entropies (i.e. maximizing the information potential) of the tracking errors with steepest descent approach, the above approach to design controller is in fact of the following form

$$\Delta u(t)=\eta\frac{\partial\Gamma(t)}{\partial u(t)} \tag{26}$$

From (25) and (26), it can be formulated that

$$\begin{aligned}\Delta\pi&=[-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}\frac{\partial\Gamma(t)}{\partial u(t)}+2\eta\frac{\partial\Gamma(t)}{\partial u(t)}]\eta\frac{\partial\Gamma(t)}{\partial u(t)}\\&=\eta(-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}+2\eta)(\frac{\partial\Gamma(t)}{\partial u(t)})^2\end{aligned} \tag{27}$$

and

$$E(\Delta\pi)=E\left\{\eta(\frac{\partial\Gamma(t)}{\partial u(t)})^2(-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}+2\eta)\right\} \tag{28}$$

To ensure that the algorithm is convergent, (28) should be non-increasing, (i.e., the right-hand side of (28) should be non-positive.

Since $\eta>0$ and $(\frac{\partial\Gamma(t)}{\partial u(t)})^2\ge 0$ hold, $-\frac{1}{[\Gamma(t)+\bar{\varepsilon}]^2}+2\eta$ should be non-positive so as to guarantee that (27) is not greater than zero. At this stage, we can obtain the condition for convergence as follows:

$$0<\eta\le\frac{1}{2[\Gamma(t)+\bar{\varepsilon}]^2} \tag{29}$$

The proposed neural controller is very simple. The design of the controller is straightforward based on minimizing entropy of the tracking error or maximizing the information potential of the tracking error.

4. SIMULATIONS

In order to show the applicability of the proposed control algorithm for nonlinear TITO systems with input and output uncertain delays, let us consider the following feedback control system, which consists of the plant represented as follows

$$\begin{cases}\bar{y}_1(k)=\frac{1}{(1+\bar{y}_1(k-1))^2}(0.8\bar{y}_1(k-1)+\\\qquad\bar{u}_1(k-2)+0.2\bar{u}_2(k-3))\\\bar{y}_2(k)=\frac{1}{(1+\bar{y}_2(k-1))^2}(0.9\bar{y}_2(k-1)+\\\qquad 0.3\bar{u}_1(k-3)+\bar{u}_2(k-2))\end{cases} \tag{30}$$

In this simulation, the PDFs of these stochastic delays τ_{ca}^1, τ_{ca}^2, τ_{sc}^1 and τ_{sc}^2 are given in order to produce stochastic tracking errors. These PDFs obey the following β-distribution,

$$\beta(\tilde{\alpha}+1,\tilde{\lambda}+1)=\int_0^1 x^{\tilde{\alpha}}(1-x)^{\tilde{\lambda}}dx \quad \tilde{\lambda},\tilde{\alpha}\geq -1 \cdot$$

Therefore it can be seen that

$$\gamma_{\tau_{ca}^1}(x)=\begin{cases}[3^{\alpha_1+\lambda_1+1}\beta(\alpha_1+1,\lambda_1+1)]^{-1}x^{\alpha_1}(3-x)^{\lambda_1}, & x\in(0,3)\\ 0, & otherwise\end{cases} \quad (31)$$

$$\gamma_{\tau_{ca}^2}(x)=\begin{cases}[4^{\alpha_2+\lambda_2+1}\beta(\alpha_2+1,\lambda_2+1)]^{-1}x^{\alpha_2}(4-x)^{\lambda_2}, & x\in(0,4)\\ 0, & otherwise\end{cases} \quad (32)$$

$$\gamma_{\tau_{sc}^1}(x)=\begin{cases}[4^{\alpha_3+\lambda_3+1}\beta(\alpha_3+1,\lambda_3+1)]^{-1}x^{\alpha_3}(4-x)^{\lambda_3}, & x\in(0,4)\\ 0, & otherwise\end{cases} \quad (33)$$

$$\gamma_{\tau_{sc}^2}(x)=\begin{cases}[3^{\alpha_4+\lambda_4+1}\beta(\alpha_4+1,\lambda_4+1)]^{-1}x^{\alpha_4}(3-x)^{\lambda_4}, & x\in(0,3)\\ 0, & otherwise\end{cases} \quad (34)$$

where $\alpha_1=1$, $\lambda_1=4$, $\alpha_2=1$, $\lambda_2=2$, $\alpha_3=3$, $\lambda_3=5$, $\alpha_4=3$, $\lambda_4=1$. In this application, the set-points are set to $r_1=r_2=1$, the sampling period $T=1$. For the simulation, the nodes in input layer of each neural controller consists of two tracking errors, the number of nodes in hidden layer of the neural controllers is 5, the learning factor η=0.001, the kernel size used to estimate the entropy is experimentally set at σ = 0.5, and the samples for neural controllers are trained with a segment of N =200 at each instant. The simulation results are shown in Figs. 2-7.

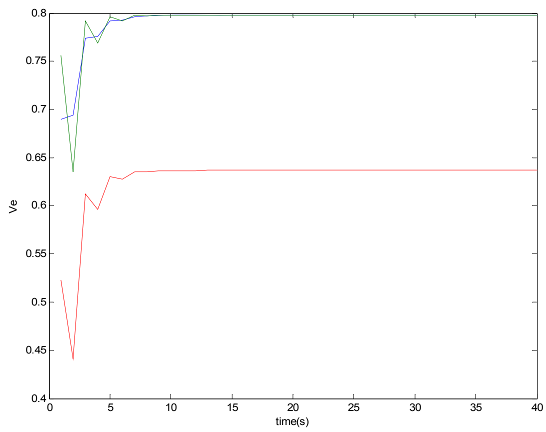

Fig. 2 The information potentials of tracking errors

Fig. 2 shows the increasing the information potentials of tracking errors, which illustrates the entropies of tracking errors decease with time. The tracking errors are shown in Fig. 3. The variances of PID parameters tuned by neural controller 1 and 2 are illustrated in Fig. 4 and Fig. 5. In Fig. 6 and Fig. 7, both the range of tracking error and the PDF of tracking error are given, where it can be seen that the shape of PDF of two tracking errors become narrower along with the increasing time. Figs. 3, 6 and 7 indicate that the control system has a small uncertainty in its closed loop operation.

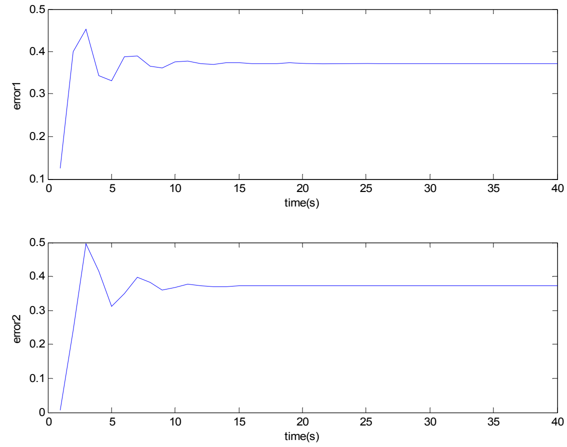

Fig. 3 The tracking errors

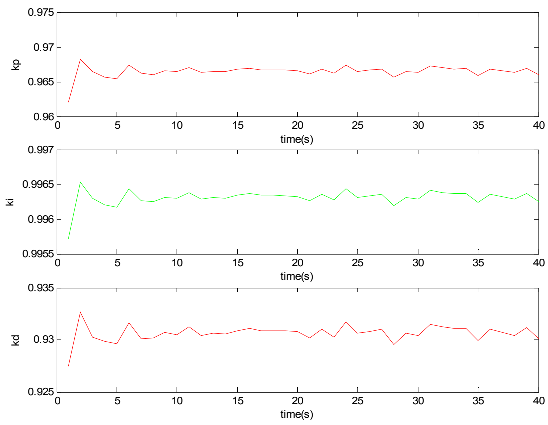

Fig. 4 Parameters of controller 1

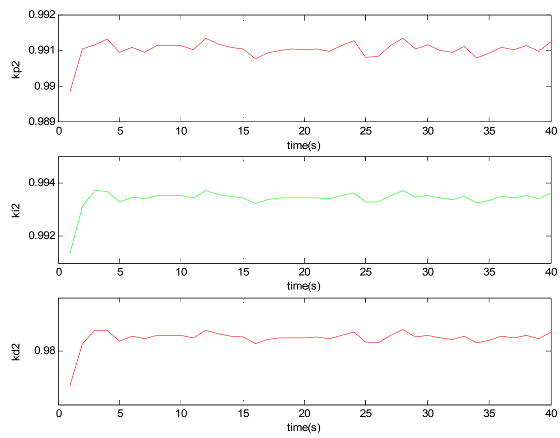

Fig. 5 Parameters of controller 2

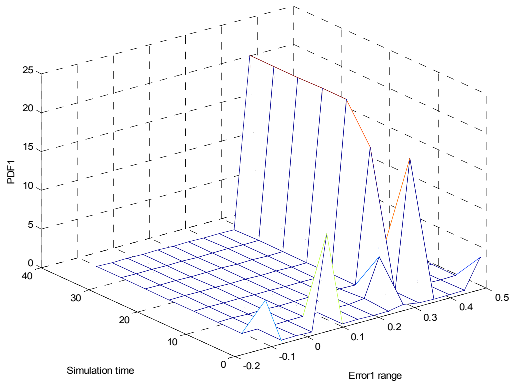

Fig. 6 PDF of the tracking error1

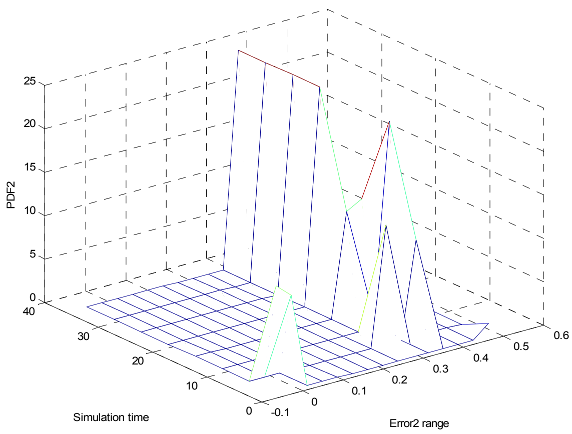

Fig. 7 PDF of the tracking error2

5. CONCLUSIONS

An approach to control nonlinear TITO systems with uncertain input/output time delays is presented based on minimizing total entropies which consists of entropy of each tracking error and their joint entropy. For sake of efficient computation and easy realization, Renyi quadratic entropy in conjunction with Parzen window with Gaussian kernel is employed to construct a novel performance index so as to obtain control algorithm. The controllers are carried out by BP neural networks. The main results have following feature: 1) the approach can be easily extended to generally nonlinear stochastic multivariable control systems with uncertain input/output delays; 2) the control algorithm can deal with stochastic multivariable systems without decoupling; 3) the performance index is established based on entropies of each tracking error besides the joint entropy of tracking errors, the controlled system can approach zero steady errors because controllers include integral strategy; 4) the convergent condition of the proposed control algorithm is given. Finally, a simulation example is given to show the effectiveness of the proposed algorithm.

ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China under grants (No. 60674051, 60534010), Beijing Natural Science Foundation under grant (No. 4072022). These are gratefully acknowledged.

REFERENCES

Zhang H., Yang D. and Chai T. (2007). Guaranteed cost networked control for T-S fuzzy systems with time delays, *IEEE Trans. Syst. Man. Cybern. C*, **37**, 160-172

Wang, H. (2000). *Bounded Dynamic Stochastic Systems: Modelling and Control*, Springer-Verlag London Ltd.

Wang, H. and Zhang, J. H. (2001). Bounded stochastic distribution control for pseudo ARMAX systems, IEEE *Transactions on Automatic Control*, **46**, 486-490.

Wang, H. (2002). Minimum entropy control of non-Gaussian dynamic stochastic systems, *IEEE Transactions on Automatic Control*, **47**, 398 – 403.

Guo, L., and Wang, H. (2005). PID controller design for output PDFs of stochastic systems using linear matrix inequalities, *IEEE Trans. Sys. Man Cyb.*, **35**, 65-71.

Guo, L. and Wang, H. (2005). Generalized discrete-time PI control of output PDFs using square root B-spline expansion. *Automatica*, **41**, 159–162.

Yue, H. and Wang, H. (2003). Minimum entropy control of closed-loop tracking errors for dynamic stochastic systems, *IEEE Trans. on Automatic Control*, **48**, 118-122.

Erdogmus, D. and Pricipe, J.C. (2002a). An error–entropy minimization algorithm for supervised training of nonlinear adaptive systems, *IEEE Transactions on Signal Processing*, **50**, 1780-1786.

Erdogmus, D. and Pricipe, J.C. (2002b) Generalized information potential criterion for adaptive system training, *IEEE Transacton on Neural Networks*, **13**, 1035-1043.