Adaptive Control based Particle Swarm Optimization and Chebyshev Neural Network for Chaotic Systems

Zhen Hong*
Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou, China
Email: zhong@zstu.edu.cn

Xile Li
Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou, China
Email: yizhikuaileo@126.com

Bo Chen
College of Information Engineering, Zhejiang University of Technology, Hangzhou, China
Email: bchen@aliyun.com

***Abstract*—The control approach for chaotic systems is one of the hottest research topics in nonlinear area. This paper is concerned with the controller design problem for chaotic systems. The particle swarm optimization (PSO) algorithm is firstly proposed to search for the weights of the Chebyshev neural networks (CNNs), and then an adaptive controller for the chaotic systems is designed based on the PSO and CNNs. Moreover, it is proved that the designed controller can guarantee the stability of chaotic systems. Numeral simulation shows the effectiveness of the proposed method in the Logistic chaotic system.**

***Index Terms*—adaptive control, particle swarm optimization, Chebyshev neural networks, chaotic systems**

I. Introduction

Since the control approach for the chaotic systems was firstly proposed in [1], controlling chaotic system has become a hot research topic in nonlinear areas. Therefore, there are many approaches solving the control problem of chaotic systems, such as feedback control of chaotic system, adaptive control of chaotic system, state feedback law [2-4]. These methods are required to control all states of the system. However, in the actual engineering system, some state variables cannot be controlled directly. To overcome these drawbacks, it is interesting and important for finding the suitable practical control method in engineering application.

The neural network can learn and approach any nonlinear and uncertain system dynamics model with arbitrary precision, thus it provides new ideas and methods to solve the control problem for chaotic systems. In this case, the chaotic control methods designed by using neural networks have made some achievements [5-12]. On the other hand, the studies in [13] show that the neural network with the orthogonal polynomial function has global approximation properties for approaching continuous function on any compact set with arbitrary precision. Particularly, when the orthogonal polynomial function is taken as Chebyshev polynomial, the performance of the designed neural networks is optimal. The reason is that the connection weights of Chebyshev neural networks (CNNs) is determined by the unidirectional gradient method, which is easy to make the objective function into local optimal impacting the efficiency of such neural network. Additionally, the particle swarm optimization (POS) adopts the speed-displacement search model, where the computational complexity is low, and the optimal solution is obtained by the cooperation and competition between particles. In this sense, the weights of the neural networks (NNs) are trained by using POS, which can give full play to the global optimization capability and rapid local convergence advantages for the PSO. Moreover, the PSO algorithm can also improve the generalization and learning capability of neural network [14]. These advantages of the PSO algorithm and CNNs motivate us to develop a control approach based on the PSO and CNNs for chaotic systems. Furthermore, to be best of the author’s knowledge, few results have been reported on this issue.

Motivated by the aforementioned analysis, the PSO algorithm is utilized to determine the connection weights of the CNNs, thus a novel CNN algorithm based on the PSO is proposed. In this case, we use the proposed algorithm to design an adaptive controller for the chaotic system. Since the convergence interval of Chebyshev basis function is in [-1, 1], an *S*-type function is introduced to extend its input range $[-\infty, +\infty]$, which

Manuscript received November 16, 2013; revised December 25, 2013; accepted January 5, 2014.
* Corresponding author: Zhen Hong (zhong@zstu.edu.cn).

© 2014 ACADEMY PUBLISHER
doi:10.4304/jcp.9.6.1385-1390

expends the scope of application of such neural network. It is proved that the model of neural networks has good approximation performance for the multivariate polynomial. Finally, the one-dimensional Logistic chaotic system is given to demonstrate the effectiveness of the proposed method.

II. CNN Learning Algorithm from PSO

A. Improved CNNs

First, the Chebyshev orthogonal polynomials [15] can be expressed as:

$$T_n(x) = \cos(n \arccos(x)), \quad |x| \le 1 \tag{1}$$

where $T_0(x)=1$, $T_1(x)=x$, and the recurrence formula is:

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \tag{2}$$

Since the range of x is $[-1,1]$, and this condition will restrict the applications of CNNs, we introduce the following S-type function:

$$g(x) = \frac{2}{1+e^{-\alpha x}} - 1 \tag{3}$$

where the domain of the Eq.(3) is $[-\infty, +\infty]$, but the range of $g(x)$ is $[-1,1]$. Meanwhile, the variable α in the function $g(x)$ is a tunable parameter. Then substituting Eq.(3) into Eq.(1) yields:

$$\begin{aligned} C_n(x) &= T_n(g(x)) = \cos(\arccos g(x)) \\ &= \cos\left(n \arccos\left(\frac{2}{1+e^{-\alpha x}} - 1\right)\right) \end{aligned} \tag{4}$$

where $C_0(x) = 1,\ C_1(x) = \frac{2}{1+e^{-\alpha x}} - 1$, and $C_n(x)$ is orthogonal polynomial satisfying

$$C_{n+1}(x) = 2\left(\frac{2}{1+e^{-\alpha x}} - 1\right)C_n(x) - C_{n-1}(x) \tag{5}$$

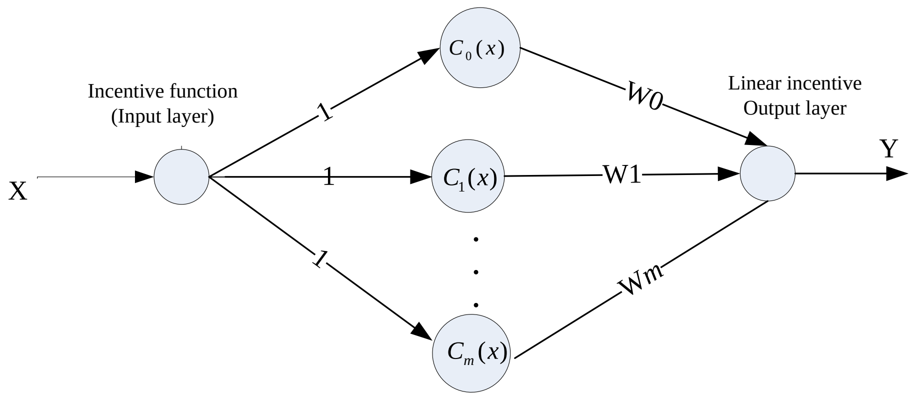

Figure 1 A neural network model of Chebyshev basis function

It can be seen from Fig.1 that the output of the model is:

$$y = \sum_{j=0}^{m} \mathrm{W}_j C_j(x) \tag{6}$$

where the improved Chebyshev function $C_j(x)$ is determined by (5). It is assumed that the desired output is y_d, then the error function is defined by $e_l \triangleq y_d - y$. Under this condition, the objective function of optimization problem is:

$$\varepsilon = \frac{1}{2}\sum_{l=1}^{r} e_l^2 = \frac{1}{2}\sum_{l=1}^{r}\left(y_d - y\right)^2 \tag{7}$$

where r denotes the number of training samples. It follows from the gradient descent method that the learning rule of W_j is:

$$\frac{\partial \varepsilon}{\partial \mathrm{W}_j} = \frac{\partial \varepsilon}{\partial e_l}\frac{\partial e_l}{\partial y}\frac{\partial y}{\partial \mathrm{W}_j} = e_l C_j(x)$$

As mentioned before, one has

$$\mathrm{W}_j(t+1) = \mathrm{W}_j(t) + \eta e_l C_j(x) \tag{8}$$

where η is the learning rate, and $0 < \eta < 1$.

B. Optimized Connection Parameters based on PSO

Since the connection weights of CNNS from section II(A) is determined by single-point gradient method, it is easy to fall into local minimum value. When the learning rate in formula (8) is hard to accurately given, it is hard to make the neural network algorithm converge under certain conditions. In the optimization process, each particle of the PSO algorithm updates themselves through their own experience and group experience, and the convergence rate is fast. In this sense, adopting the PSO algorithm for optimizing network connection weights can

© 2014 ACADEMY PUBLISHER

improve the efficiency of neural network algorithm, and the convergence performance can be improved. In what follows, we firstly give the detailed mathematical description of the PSO:

Suppose that $\Omega \subset \mathrm{R}^n$ is a target search space of *n*-dimension, and a group $X = \{x_1, x_2, \cdots, x_n\}$ is composed by *n* particles. Then the velocity and position of the i_{th} particle is defined by:

$$v_i(k) \triangleq [v_{i1}(k) \quad v_{i2}(k) \quad \cdots \quad v_{in}(t)]^{\mathrm{T}},$$
$$x_i(k) \triangleq [x_{i1}(k) \quad x_{i2}(k) \quad \cdots \quad x_{in}(k)]^{\mathrm{T}}$$

And the current individual optimal solution of the i_{th} particle is:

$$\mathrm{pbest}_i(k) \triangleq (p_{i1}(k) \quad p_{i2}(k) \quad \cdots \quad p_{in}(k))$$

The current group optimal solution is:

$$\mathrm{gbest}(k) \triangleq (g_{g_1}(k), g_{g_2}(k), \cdots, g_{g_n}(k))$$

where *k* is the number of current evolution generation.

According to the theory of optimal particle tracking, the particle x_i updates its velocity and position according to the following formula:

$$\begin{aligned} v_{id}(k+1) &= w(k)v_{id}(k) \\ &+ c_1 \cdot \mathrm{rand}(0,1) \cdot (\mathrm{pbest}_i(k) - x_{id}(k)) \\ &+ c_2 \cdot \mathrm{rand}(0,1) \cdot (\mathrm{gbest}_g(k) - x_{id}(k)) \ \ (i = 1, 2, \cdots, n) \end{aligned} \tag{9}$$

$$x_{id}(k+1) = x_{id}(k) + v_{id}(k+1) \tag{10}$$

where the Eq.(9) and Eq.(10) describe the update mode of particles' velocity and position, respectively. The parameters c_1 and c_2 are the accelerated constant, and their selected values are in [0, 2]. $w(k)$ is a linear inertia weight index. If the value of $w(k)$ is relatively large, the global convergence will be better. Otherwise, the local convergence is better. $w(k)$ is taken as:

$$w(k) = 0.9 - \frac{k}{\mathrm{MaxNumber}} \cdot 0.5$$

Where the "MaxNumber" is the maximum iterations.

For the neural network model in Section II(A), we make the connection weights W_j of CNNs as the position vector *x* in the PSO algorithm. We also determine the fitness function in particle swarm optimization according to the Eq.(7). Given a set of initial velocity randomly, and using the PSO algorithm to conduct iteration training, when the fitness function is less than the given error range, $f_i < \varepsilon_0$, the algorithm is suspended.

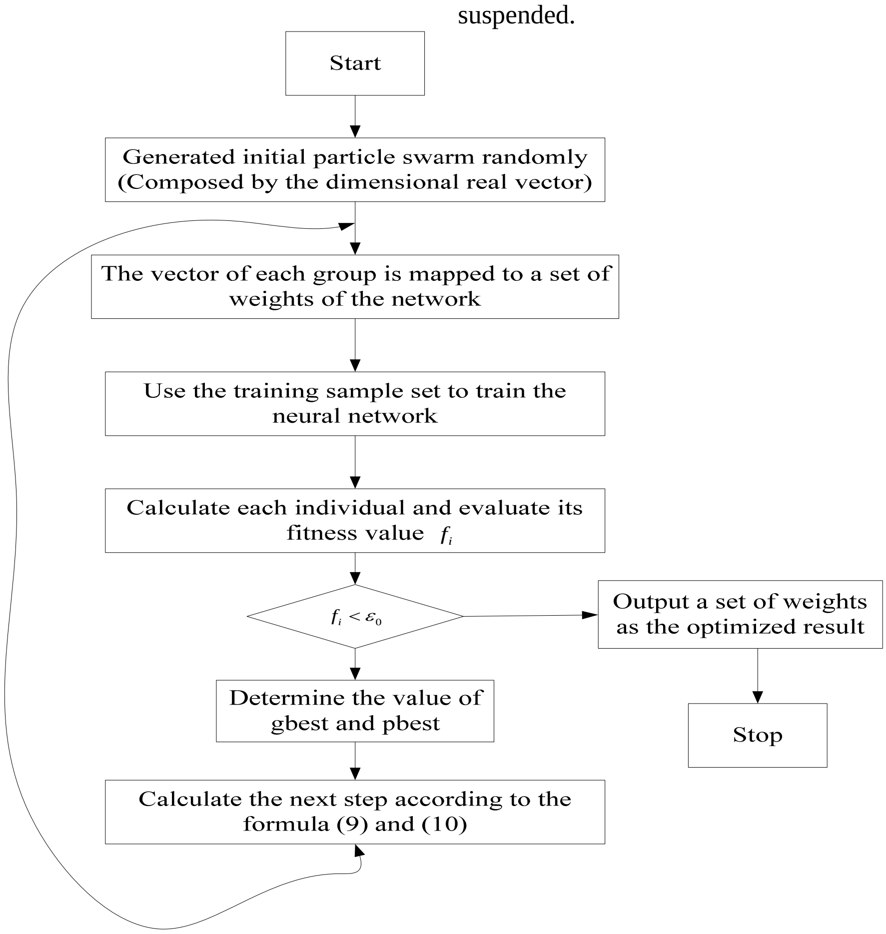

Figure 2 A optimized connection weights of the PSO

© 2014 ACADEMY PUBLISHER

III THE CONTROLLER DESIGN BASED ON CNNS AND PSO

Considering the following chaotic system:

$$x(k+1) = f(x(k), p) \tag{11}$$

where $x \in \mathrm{R}^n$ is the system status, and p is the system parameter.

Using the neural network learning algorithm based on the PSO to simulate the input-output relationship of chaotic model (11), one has

$$\hat{x}(k+1) = \hat{f}(x(k), p) \tag{12}$$

According to the system stability theory, the controller can be taken as:

$$u(k) = -\hat{f}(x(k), p) + x_d(k+1) + \beta(x(k) - x_d(k)) \tag{13}$$

where x_d is the desired target track, β is the parameter. Then combining Eq.(12) and Eq.(13) yields:

$$\begin{aligned} x(k+1) &= f(x(k), p) - \hat{f}(x(k), p) \\ &\quad + x_d(k+1) + \beta(x(k) - x_d(k)) \end{aligned} \tag{14}$$

If the proposed neural network can approach the system (11), then the error system is described by

$$e(k+1) = \beta(x(k) - x_d(k)) = \beta e(k) \tag{15}$$

where $e(k+1) = x(k+1) - x_d(k+1)$.

According to the system stability theory, we assume $|\beta| < 1$, which guarantees the error system (15) is asymptotically stable. It is easy to prove that when the system (11) is applied to control, it can track the controlled objective. So the approximation performance of the neural network is particularly important. In what follows, we will discuss the demands of the fuzzy Chebyshev basis function neural network to the chaotic system control accuracy.

Definition 1 Suppose $\forall X \in R^n$, $x^n \in [a,b]$, $g(X)$ is an n-polynomial and $\| g(X) \|$ is bounded or integral in the domain, and if $g(X)$ satisfies the following inner product relationship:

$$\begin{aligned} (g_l(X), g_k(X)) &= \int_{[a,b]}^{n} \rho(X) g_l(X) g_k(X) dX \\ &= \begin{cases} 0, & l = k \\ \int_{[a,b]}^{n} \rho(X) g_l^2(X) dX, & l \neq k \end{cases} \end{aligned} \tag{16}$$

Then $g_n(X)$ sequence is called multivariate orthogonal polynomial, and the weights are $\rho(X)$.

For the n-dimensional vector $X = [x_1, x_2, \cdots, x_n]^{\mathrm{T}}$, n-arid function $f(X)$ and multiple integral symbols, we make the convention as:

$\forall x_i \in X, x_i \in [a,b]$ is recorded as $x^n \in [a,b]$, $\underbrace{\int_a^b \int_a^b \cdots \int_a^b}_{n个}$ is recorded as $\int_{[a,b]}^{n}$, $dx_1 \cdots dx_n$ is recorded as dX.

Considering the multivariate polynomial composed by orthogonal polynomial, one has

$$g_i(X) = \prod_{j=1}^{n} C_i(X) \tag{17}$$

On the other hand, Chebyshev orthogonal polynomial satisfies the following inner product relationship:

$$\begin{aligned} (C_l(x), C_k(x)) &= \int_{-1}^{1} \rho(x) T_l(x) T_k(x) dx \\ &= \begin{cases} 0, & l = k \\ \int_{-1}^{1} \rho(x) T_l^2(x) dx, & l \neq k \end{cases} \end{aligned} \tag{18}$$

where $x \in R$.

Substituting Eq.(17) into Eq.(18) yields:

$$\begin{aligned} (g_l(X), g_k(X)) &= \int_{[-1,1]}^{n} \rho(X) g_l(X) g_k(X) dX \\ &= \int_{[-1,1]}^{n} \rho(X) \prod_{j=1}^{n} C_l(x_j) \prod_{j=1}^{n} C_k(x_j) dX \end{aligned} \tag{19}$$

Since the orthogonal polynomial $C_0(x)$, $C_1(x)$, …, $C_n(x)$ is linearly independent, the integral order of the Eq.(19) can be changed, then:

$$\begin{aligned} (g_l(X), g_k(X)) &= \int_{[-1,1]}^{n} \rho(X) g_l(X) g_k(X) dX \\ &= \begin{cases} 0, & l = k \\ \int_{[-1,1]}^{n} \rho(X) g_l^2(X) dX, & l \neq k \end{cases} \end{aligned} \tag{20}$$

Obviously, Eq.(20) satisfies the condition in Definition 1, which is a multivariate orthogonal polynomial, and $x^n \in [-1,1]$.

Definition 2 For a given function $f(X)$ which the range is $x^n \in [-1,1]$, using the polynomial $y(X) = \sum_{k=1}^{n} W_k g_k(X)$ to make the optimization mean square approximation and find out the value $W_i (i = 1, 2, \cdots n)$ to make the function $\| f(X) - y(X) \|_2^2 = \int_{[-1,1]}^{n} \rho(X) [f(X) - y(X)] dX$ is the minimum.

Lemma 1 [13] A neural network based on the orthogonal polynomial possesses the global approximation property for arbitrary precision approaching continuous function on any compact set.

Theorem 1 According to Eq.(11), utilizing the input-output relationship of the CNN learning algorithm

© 2014 ACADEMY PUBLISHER

based on the PSO, we can get the neural network model (12) for the chaotic system. Using the controller designed by Eq.(13) to chaotic system (11), then there exists a positive constant $\sigma(\sigma > 1)$ such that $|e(k)| < \sigma |f_e(k)|$, where $e(k)$ and $f_e(k)$ denotes the tracking error and model error, respectively.

Proof: Let us denote $\gamma_1 = \sup_k |e(k)|$ and $\gamma_2 = \sup_k |f_e(k)|$. If we want to prove $|e(k)| < \sigma |f_e(k)|$, we just need to prove $\gamma_1 \le \sigma\gamma_2$.

It follows from Eq.(14) that

$$e(k+1) = f_e(k) + \beta e(k).$$

Taking the absolute value on both sides of the equation above, one has:

$$|e(k+1)| = |f_e(k) + \beta e(k)| \le |f_e(k)| + |\beta e(k)| \qquad (21)$$

Taking the maximum on both sides of the Eq.(21), one has

$$\gamma_1 \le \gamma_2 + |\beta| \gamma_1$$

where $|\beta| < 1$.

Consolidating the above equations yields

$$\gamma_1 \le \frac{1}{1-|\beta|}\gamma_2$$

Then we denote $\sigma = \frac{1}{1-|\beta|}$, one has

$$\gamma_1 \le \sigma\gamma_2$$

where it is obvious that $\sigma > 1$, which implies that $|e(k)| < \sigma |f_e(k)|$.

Remark**:** Theorem 1 denotes that the tracking error cannot be less than the model error. Therefore, the model error on the stability of the system is an extremely important role that the higher accuracy of the model, the higher accuracy of the control.

IV THE NUMERICAL SIMULATION

We use the Logistic chaotic system to check the effectiveness of the chaos control method. The details are as follows.

Logistic mapping system is:

$$x(k+1) = \lambda x(k)(1 - x(k)) \qquad (22)$$

where $x \in [0,1]$ and λ is a positive constant. When $\lambda = 4$, the system is in a chaotic state. Supposing $\eta = 0.05$, $\alpha = 0.01$, and using the fuzzy Chebyshev neural network learning algorithm (22), we can get a model as follows:

$$\hat{x}(k+1) = \hat{f}(x(k))$$

According to the Eq.(13), a desired controller is as follows:

$$u(k) = -\hat{f}(x(k)) + x_d(k+1) + \beta(x(k) - x_d(k))$$

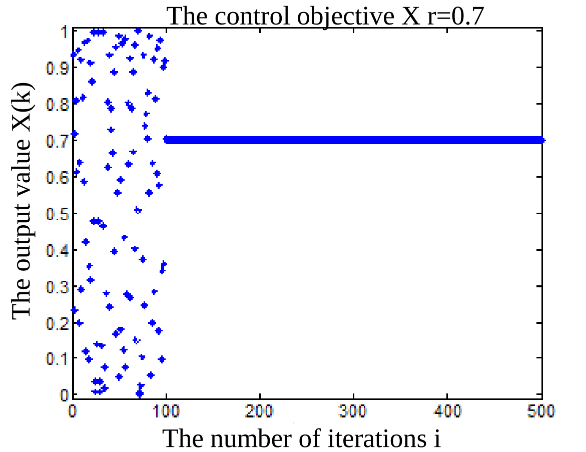

Figure 3

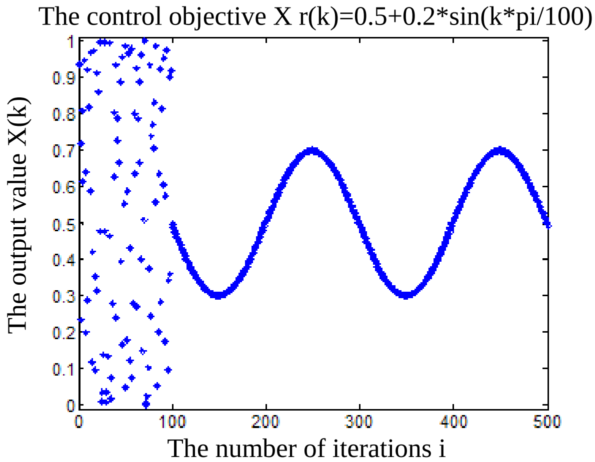

Figure 4

When the control is exerted, the system (22) is changed as:

$$x(k+1) = \lambda x(k)(1 - x(k)) + u(k) \qquad (23)$$

When $\lambda = 4, \beta = 0.02$, the designed control targets are $x_r(k) = 0.7$ and $x_r(k) = 0.5 + 0.2\sin(k\pi/100)$.

When the control is exerted at the 100th step, the control results are showed in figure 3 and figure 4, respectively.

V CONCLUSION

An adaptive control method based PSO and CNNs is proposed for chaotic systems. The PSO algorithm is mainly used to search for the weights of the CNNs. Then, an adaptive controller for the chaotic systems is designed using the approach which is combined with PSO and CNNs. Furthermore, we prove that the designed controller can guarantee the stability of chaotic systems.

© 2014 ACADEMY PUBLISHER

Finally, the Logistic chaotic system is given to demonstrate the effectiveness of the proposed method.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (61304256), Zhejiang Provincial Natural Science Foundation of China (LQ13F030013), project of Education Department of Zhejiang Province (Y20132700), Ningbo Natural Science Foundation (2012A610016) and Science Foundation of Zhejiang Sci-Tech University (1202815-Y).

REFERENCES

[1] E. Ott, C. Grebogi, and J. A. Yorke, “Chaos control,” *Physical Review Letters*, vol. 64, no. 11, pp. 1196-1199, 1990.

[2] G. Chen, “Controlling chaos and bifuractions in engineering systems”, *CRC press*, 2000.

[3] W. Sanum, and B. Srisuchinwong, “Highly complex chaotic system with piecewise linear nonlinearity and compound structures,” *Journal of Computers*, vol. 7, no. 4, pp. 1041-1047, 2012.

[4] X J Li, W X Xiao, Z Liu, W L Wan, and T S Hu, “A new fractional order chaotic system and its compound structure," *Journal of Software*, vol. 8, no. 1, pp. 126-133, 2013.

[5] P. M. Alsing, A. Gavrielides, “Using neural networks for controlling chaos,” *Physical Review E*, vol. 49, pp. 1225-1231, 1994.

[6] C. T Lin, “Controlling chaos by GA-based reinforcement learning neural networks,” *IEEE Transactions on Neural Networks*, vol. 10, no. 4, pp. 846-859, 1999.

[7] C. F Hsu, “Intelligent control of chaotic systems via self-organizing Hermite-polynomial-based neural network,” *Neurocomputing*, vol. 123, pp. 197-206, 2014.

[8] P. Yadmellat, S. K. Y. Nikravesh, “A recursive delayed output-feedback control to stabilize chaotic systems using linear-in-parameter neural networks,” *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 1, pp. 383-394, 2011.

[9] S. C. Jeong, D. H. Ji, J. H. Park, S. C. Won, “Adaptive synchronization for uncertain chaotic neural networks with mixed time delays using fuzzy disturbance observer,” *Applied Mathematics and Computation*, vol. 219, no. 11, pp. 5984-5995, 2013.

[10] L S Yin, Y G He, X P Dong, and Z Q Lu. “Multi-step prediction algorithm of traffic flow chaotic time series based on volterra neural network," *Journal of Computers*, vol. 8, no. 6, pp. 1480-1487, 2013.

[11] W Tan, Y N Wang, Z R Liu,and S W Zhou, “Controlling chaotic system by RBF neural networks nonlinear compensator,” *Control Theory & Applications*, vol. 20, no. 6, 951-954, 2003. (In Chinese)

[12] D Liu, H P Ren, and Z Q Kong, “Control of chaos solely based on RBF neural network without an analytical model,” *Acta Physica Sinica*, vol. 52, no. 3, pp. 533-535, 2003. (In Chinese)

[13] X J Wu, S T Wang, J Y Yang, and Q Y Cao, “The study on the orthogonal polynomials-based neural networks and its properties,” *Computer Engineering and Application*, vol. 39, no. 9, 25-27, 2002. (In Chinese)

[14] J. R Zhang, J Zhang, T. M Lock, and M. R. Lyu, “A hybrid particle swarm optimization—back-propagation algorithm for feedforward neural network training,” *Applied Mathematics and Computation*, vol. 185, pp. 1026-1037, 2007.

[15] V Basios, A. Yu. Bonushkina, and V. V. Ivanov, “On a method for approximatin one-dimensiona functions,” *Computer and Mathematics with Applications*, vol. 34, pp. 687-693, 1997.

Zhen Hong received both B.S. degrees from Computer Science & Technology at Zhejiang University of Technology, Hangzhou, China and Computing at University of Tasmania, Australia, respectively. He received Ph. D. degree in Control Theory and Control Engineering at the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China, in 2012. He is currently working as a lecturer at Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou, China.

His main research interests include wireless sensor networks, optimization, adaptive control and intelligent transportation.

Xile Li received the B.S. degree in Measurement Technology and Instrument at Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou, China, in 2012. She is currently a Master student in Faculty of Mechanical Engineering & Automation, Zhejiang Sci-Tech University, Hangzhou, China.

Her main research interests include wireless sensor networks, topology control.

Bo Chen received the B.S. degree in Information and Computing Science from Jiangxi University of Science and Technology, Ganzhou, China, in 2008. He is currently working toward the Ph.D. degree in Control Theory and Control Engineering at the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China.

His main research interests include information fusion, distributed estimation, and time-delay systems.

© 2014 ACADEMY PUBLISHER