Vol 17 No 5, May 2008
1674-1056/2008/17(05)/1652-12

Chinese Physics B

© 2008 Chin. Phys. Soc.
and IOP Publishing Ltd

Synchronization of different chaotic systems via active radial basis functions sliding mode controller

Guo Hui-Jun(郭会军)†, Yin You-Wei(尹有为), and Wang Hua-Min(王华民)

Department of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, *China*

(Received 10 July 2007; revised manuscript received 22 September 2007)

This paper presents a new method to synchronize different chaotic systems with disturbances via an active radial basis function (RBF) sliding controller. This method incorporates the advantages of active control, neural network and sliding mode control. The main part of the controller is given based on the output of the RBF neural networks and the weights of these single layer networks are tuned on-line based on the sliding mode reaching law. Only several radial basis functions are required for this controller which takes the sliding mode variable as the only input. The proposed controller can make the synchronization error converge to zero quickly and can overcome external disturbances. Analysis of the stability for the controller is carried out based on the Lyapunov stability theorem. Finally, five examples are given to illustrate the robustness and effectiveness of the proposed synchronization control strategy.

Keywords: chaos synchronization, active control, sliding mode control, RBF networks
PACC: 0545

1. Introduction

Since the pioneering work by Pecora and Carroll in the 1990s,[1] chaotic synchronization has received much attention from various research fields due to its wide applicability in physical systems.[2−43] Quite a few synchronization methods have been developed in master–slave configurations, which can be classified into two main streams: synchronization of identical systems[3−17] and synchronization of non-identical systems.[18−36] The synchronization problems of two identical chaotic systems with various constraints have been studied intensively. However, most physical systems have a certain non-linearity, time varying behaviour and various uncertainties, and it is rarely the case that the components are assumed to be identical.[28] In recent years, intensive researches and applications of chaotic synchronization in secure communication show synchronizing two different chaotic systems to be very important to make the whole system more practical. Compared with the synchronization of identical systems, the synchronization of two different chaotic systems is difficult and complex. Quite a few methods have been proposed to deal with the synchronization of a master–slave configuration, which mainly include linear and nonlinear feedback control,[18−21] backstepping control,[22] active control,[23−27] sliding control,[28] adaptive control,[20,25,29−31] tracking control,[32] H_∞ control,[33] RBF neural network[34,35] and linear coupling control.[36]

Sliding mode control, based on the theory of variable structure systems, has been widely applied to the robust control of nonlinear systems due to many attractive advantages, e.g. insensitivity to variation in plant parameters, external disturbances and unmodelled dynamics.[44] Considering the fact that for many chaotic systems variations in parameter and external disturbances are unknown, the sliding mode control method is suitable for designing various synchronization controllers. In Ref.[9], the synchronization of chaotic systems with a parametric uncertainty via the active sliding mode control was studied. In Ref.[15] a sliding mode control scheme was developed to realize the robust synchronization of unified chaotic systems. In Ref.[28] an active sliding mode controller for synchronizing different chaotic systems was proposed. But the synchronization of chaotic systems with mismatching parameters and external disturbances is still a challenge.

†Corresponding author. E-mail: realghj@yahoo.com.cn

In this paper, we consider the synchronization of different chaotic systems with external disturbances. An active RBF sliding mode controller is proposed to achieve the goal. This method incorporates the advantages of the active control, the RBF neural networks and the sliding mode control to provide immunity to external disturbances. The control strategy requires two stages: one is to choose an appropriate active controller and the other is to design an RBF sliding mode controller to realize the synchronization goal. Moreover, we give very simple conditions necessary and sufficient for the realization of the propose method.

2. Synchronization problem description

Consider a chaotic master system described by the following nonlinear differential equation:

$$\dot{\boldsymbol{x}} = A_1\boldsymbol{x} + f_1(\boldsymbol{x}), \tag{1}$$

where $\boldsymbol{x}(t) \in R^3$ denotes the three-dimensional state vector of the system, $A_1 \in R^{3\times3}$ and $f_1 : R^3 \to R^3$ represent the linear and nonlinear parts of the system dynamics, respectively.

A slave system is described as follows:

$$\dot{\boldsymbol{y}} = A_2\boldsymbol{y} + f_2(\boldsymbol{y}) + \boldsymbol{u}(t), \tag{2}$$

where $\boldsymbol{y}(t) \in R^3$ is the three-dimensional state vector of the slave system; $A_2 \in R^{3\times3}$ and $f_2 : R^3 \to R^3$ play the roles similar to those of A_1 and f_1 in the master system. The control input $\boldsymbol{u}(t) \in R^3$ has been added to the slave system in order to synchronize these two systems. In this paper, we study the case where $A_1 \neq A_2$ and $f_1(\boldsymbol{x}) \neq f_2(\boldsymbol{x})$, i.e. the chaotic systems are different.

Defining the synchronization error as $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{x}$ and using Eqs.(1) and (2), the synchronization error dynamics is determined to be

$$\dot{\boldsymbol{e}} = A_2\boldsymbol{y} + f_2(\boldsymbol{y}) - A_1\boldsymbol{x} - f_1(\boldsymbol{x}) + \boldsymbol{u}. \tag{3}$$

The goal is to design the controller $\boldsymbol{u}(t) \in R^3$ such that

$$\lim_{t\to\infty} \|\boldsymbol{e}(t)\| = 0.$$

3. Active RBF sliding mode controller design

3.1. Active controller design

Let

$$F(\boldsymbol{x},\boldsymbol{y}) = (A_2 - rI)\boldsymbol{y} - (A_1 - rI)\boldsymbol{x} + f_2(\boldsymbol{y}) - f_1(\boldsymbol{x}),$$

$$H(t) = F(\boldsymbol{x},\boldsymbol{y}) + \boldsymbol{u}(t),$$

where parameter r is a positive constant. According to the active control design procedure, one uses the control input $\boldsymbol{u}(t)$ to eliminate the nonlinear part of the error dynamics. In other words, the input vector is considered as

$$\boldsymbol{u}(t) = H(t) - F(\boldsymbol{x},\boldsymbol{y}). \tag{4}$$

Substitute expression (4) into Eq.(3), then the error system will be rewritten as

$$\dot{\boldsymbol{e}} = r\boldsymbol{e} + H(t). \tag{5}$$

The control input $H(t)$ is defined as

$$H(t) = \boldsymbol{K}w(t), \tag{6}$$

where $\boldsymbol{K} = [k_1, k_2, k_3]^{\mathrm{T}}$ is a constant gain vector and $w(t) \in R$ is the sliding mode control input.

3.2. Sliding mode controller based on radial basis functions

Sliding mode control system is time invariant when reaching the sliding surface, and thus the motion is independent of parameters and disturbances.[44] In this paper, the sliding surface can be chosen as

$$s(\boldsymbol{e}) = \boldsymbol{C}\boldsymbol{e} = [c_1, c_2, c_3]\boldsymbol{e}. \tag{7}$$

Then

$$\begin{aligned}\dot{s} =& \boldsymbol{C}\dot{\boldsymbol{e}}\\ =& \boldsymbol{C}r\boldsymbol{e} + \boldsymbol{C}H(t)\\ =& \boldsymbol{C}r\boldsymbol{e} + \boldsymbol{C}\boldsymbol{K}w(t).\end{aligned} \tag{8}$$

Sliding variable s will be used as a single input signal for establishing an RBF neural network model to calculate the control law $w(t)$.

Artificial neural networks, originally inspired by the functionality of biological neural networks, can learn complex functional relations by being generalized from a limited number of training data. Neural networks can thus serve as black-box models of

nonlinear, multivariable static and dynamic systems. RBF neural network is one of the most popular neural networks in the modelling and the controlling of nonlinear systems. Compared with the standard feedforward networks, RBF networks have the advantages of faster learning speed and less chance of falling into local minimal convergence.[45,46] The RBF networks each consist of two layers: an input nonlinear layer and an output linear layer. The output of the nonlinear layer for an RBF network is

$$\phi_i(\boldsymbol{x}) = \exp\left(-\frac{\|\boldsymbol{x}-c_i\|^2}{2\sigma_i^2}\right), \quad i=1,2,\cdots,n, \quad (9)$$

the output of the linear layer is

$$y_j = f(\boldsymbol{x}) = \sum_{i=1}^{n} m_{ji}\phi_i(\boldsymbol{x}) = \boldsymbol{M}_j^{\mathrm{T}}\boldsymbol{\phi}, \quad j=1,2,\cdots,m, \quad (10)$$

where $\boldsymbol{x} \in R^n$ and $\boldsymbol{y} \in R^m$ are input vector and output vector of the network, respectively; $\boldsymbol{\phi} = [\phi_1,\cdots,\phi_n]^{\mathrm{T}}$ is the hidden output vector; n is the number of hidden neurons; $\boldsymbol{M}_j = [m_{j1},\cdots,m_{jn}]^{\mathrm{T}}$ is the weight vector of the network; parameters c_i and σ_i are centres and radii of the basis functions, respectively. The adjustable parameters of the RBF network are M, c_i and σ_i, which can be regulated or optimized by least-square method, gradient descent algorithm, genetic algorithm, etc.

Then for the single-input and single-output case (SISO) in this paper, the output of the controller based on the RBF is

$$w_N(t) = \sum_{i=1}^{n} M_i \exp\left(-\frac{\|s-c_i\|^2}{2\sigma_i^2}\right) = \boldsymbol{M}^{\mathrm{T}}\boldsymbol{\phi}, \quad (11)$$

where $\boldsymbol{M} = [M_1,\cdots,M_n]^{\mathrm{T}}$ is the weight vector; n is the number of hidden neurons; parameters c_i and σ_i are centres and radii of the basis functions, respectively.

3.2.1. Weights updating rule

Based on the Lyapunov theorem, the sliding-surface's reaching condition is $s\dot{s} < 0$. If a control input can be chosen to satisfy this reaching condition, the control system will converge to the origin of the phase plane. Since the RBF neural network is employed to approximate the nonlinear mapping between the sliding input variable and the control law, the weight of the neural network should be regulated based on the reaching condition $s\dot{s} < 0$.[44] An adaptive law is used to adjust the weights for searching the optimal weighting value and obtaining the stable convergence property. The adaptive law is derived from the steep descent rule to minimize the value of $s\dot{s}$ with respect to M_i, and then the updated equation of the weighting parameters is

$$\dot{M}_i = -\gamma\frac{\partial s\dot{s}}{\partial M_i}.$$

The rule of chain differential yields

$$\begin{aligned}\dot{M}_i &= -\gamma\frac{\partial s\dot{s}}{\partial M_i} = -\gamma\frac{\partial s\dot{s}}{\partial w_N}\frac{\partial w_N}{\partial M_i}\\ &= -\gamma s\boldsymbol{C}\boldsymbol{K}\phi_i, \end{aligned} \quad (12)$$

where γ is the learning rate parameter. Then the weighting between hidden and output layer can be adjusted on-line to achieve the learning ability.

3.2.2. Sliding surface design

The equivalent control approach is found from the fact that $\dot{s}(\boldsymbol{e}) = 0$ is a necessary condition for the state trajectory to stay on the switching surface $s(\boldsymbol{e}) = 0$. Hence, when in a sliding mode, the controlled system satisfies the following conditions:

$$s(\boldsymbol{e}) = 0 \quad (13)$$

and

$$\dot{s}(\boldsymbol{e}) = 0. \quad (14)$$

Now, using Eq.(8), Eq.(13) and Eq.(14), one can obtain an equivalent control

$$w_{\mathrm{eq}}(t) = -(\boldsymbol{C}\boldsymbol{K})^{-1}\boldsymbol{C}r\boldsymbol{e}, \quad (15)$$

where the existence of $(\boldsymbol{C}\boldsymbol{K})^{-1}$ is a necessary condition.

3.2.3. Designing RBF sliding mode controller

We assume that the constant-rate reaching law is applied. The reaching law can be chosen such that

$$\dot{s} = -\lambda s. \quad (16)$$

Parameter $\lambda > 0$ is determined such that the sliding condition is satisfied and the sliding mode motion occurs.

From Eq.(8) and Eq.(16) it can follow that

$$\dot{s} = \boldsymbol{C}r\boldsymbol{e} + \boldsymbol{C}\boldsymbol{K}w(t) = -\lambda s. \quad (17)$$

Now from the above Eq.(17), the control input is determined as

$$w_{\mathrm{R}}(t) = -(\boldsymbol{C}\boldsymbol{K})^{-1}\lambda s - (\boldsymbol{C}\boldsymbol{K})^{-1}\boldsymbol{C}r\boldsymbol{e}. \quad (18)$$

4. Robust stability analysis

In this paper, we use an RBF neural network to calculate the real control input. If w_{N} can approach the control input with no error, we can obtain

$$w_{\mathrm{N}} - w_{\mathrm{R}} = 0. \tag{19}$$

However, the output w_{N} of the RBF neural network may have a certain difference from ideal input w_{R}. The following theorem indicates that this difference can be limited to a certain bound, which does not affect the convergence of the closed loop system.

Theorem[46]: For any given constant ξ and a continuous function $f : X \to R^n$, there must exist one weight matrix $M = M^*$, which makes the output of an RBF network with n hidden neurons satisfy the following condition

$$\max_{x \in X} \left\| \hat{f}\left(x, M^*\right) - f\left(x\right) \right\| \leq \xi,$$

where $\hat{f}\left(\cdot\right)$ is the corresponding RBF network.

Therefore, the optimal weighting vector $\bar{\boldsymbol{M}}$ of the RBF NN exists, which makes the control law w_{N} to approximate the perfect control law w_{R} with an error smaller than ξ, i.e.

$$\begin{aligned} &\max \left| w_{\mathrm{R}}(t) - w_{\mathrm{RBF}}(t) \right| \\ =&\max \left| \boldsymbol{M}_{\mathrm{N}}^{\mathrm{T}} \boldsymbol{\phi} - \bar{\boldsymbol{M}}^{\mathrm{T}} \boldsymbol{\phi} \right| \leq \xi, \end{aligned} \tag{20}$$

where $\bar{\boldsymbol{M}}$ is the optimal weighting vector, $w_{\mathrm{N}} = \boldsymbol{M}_{\mathrm{N}}^{\mathrm{T}} \boldsymbol{\phi}$, $w_{\mathrm{R}} = \bar{\boldsymbol{M}}^{\mathrm{T}} \boldsymbol{\phi}$ and $\tilde{\boldsymbol{M}} = \boldsymbol{M}_{\mathrm{N}}^{\mathrm{T}} - \bar{\boldsymbol{M}}^{\mathrm{T}}$.

$$w_{\mathrm{N}} - w_{\mathrm{R}} = \boldsymbol{M}_{\mathrm{N}}^{\mathrm{T}} \boldsymbol{\phi} - \bar{\boldsymbol{M}}^{\mathrm{T}} \boldsymbol{\phi} = \tilde{\boldsymbol{M}} \boldsymbol{\phi} = \tilde{w}. \tag{21}$$

Define the Lyapunov function as

$$V = \frac{1}{2} s^2 + \frac{1}{2\gamma} \tilde{\boldsymbol{M}}^{\mathrm{T}} \tilde{\boldsymbol{M}}, \tag{22}$$

then its first derivation with respect to time will be

$$\begin{aligned} \dot{V} =& s\dot{s} + \frac{1}{\gamma} \tilde{M}^{\mathrm{T}} \dot{\tilde{M}} \\ =& -\lambda s^2 + \frac{1}{\gamma} \tilde{M}^{\mathrm{T}} (-\gamma s \boldsymbol{C}\boldsymbol{K}\boldsymbol{\phi}) \\ =& -\lambda s^2 - s\boldsymbol{C}\boldsymbol{K}(\boldsymbol{M}_{\mathrm{N}} - \bar{\boldsymbol{M}})^{\mathrm{T}} \boldsymbol{\phi}. \end{aligned} \tag{23}$$

Considering $\max \left| \boldsymbol{M}_{\mathrm{N}}^{\mathrm{T}} \boldsymbol{\phi} - \bar{\boldsymbol{M}}^{\mathrm{T}} \boldsymbol{\phi} \right| \leq \xi$ from expression (20), we can obtain

$$\left| \boldsymbol{M}_{\mathrm{N}} - \bar{\boldsymbol{M}} \right|^{\mathrm{T}} \boldsymbol{\phi} \leq \xi. \tag{24}$$

Therefore

$$\begin{aligned} \dot{V} =& -\lambda s^2 + s\boldsymbol{C}\boldsymbol{K}(-\boldsymbol{M}_{\mathrm{N}} + \bar{\boldsymbol{M}})^{\mathrm{T}} \boldsymbol{\phi} \\ \leq& -\lambda s^2 + s\boldsymbol{C}\boldsymbol{K}\xi. \end{aligned} \tag{25}$$

Furthermore, the following expression can be derived:

$$\dot{V} \leq |s| \left(-\lambda |s| + \boldsymbol{C}\boldsymbol{K}\xi\right). \tag{26}$$

Therefore, if $|s| \geq \boldsymbol{C}\boldsymbol{K}\xi/\lambda$, then $\dot{V} \leq 0$ which confirms the presence of sliding mode dynamics. That means that this Lyapunov function will decrease gradually and the sliding surface variable s will converge to a boundary layer $\pm\boldsymbol{C}\boldsymbol{K}\xi/\lambda$ around $s = 0$ in the phase plane. Furthermore, for $\dot{V} \leq 0$ and $V > 0$, it can be concluded that the proposed controller is stable. From the above analysis, we know that sliding surface variable s will converge to a boundary layer in the phase plane, and then the system output errors e_1, e_2 and e_3 at least converge to a small error bound. When the accuracy of the RBFNN non-linear mapping increases, the system output error decreases correspondingly. In this way, the convergence of the closed loop controller is provided.

5. Numerical results and analysis

For illustrating the effectiveness of the proposed method, we have applied this method to synchronizing five pairs of different chaotic systems. In the numerical simulations, the fourth-order Runge–Kutta integration method is used to solve the differential equations with time steps of 0.001s. Parameter λ in sliding surface is set as 0.01. The spread factor of these Gaussian functions is taken to be a constant 0.7, and then each RBF has about one-third overlap.

5.1. System descriptions

The Lorenz system[40] is one of the paradigms of chaos, for it captures lots of features of chaotic systems. It is given by

$$\begin{cases} \dot{x}_1 = a(x_2 - x_1), \\ \dot{x}_2 = cx_1 - x_1x_3 - x_2, \\ \dot{x}_3 = x_1x_2 - bx_3, \end{cases} \tag{27}$$

where $a = 10$, $b = 8/3$ and $c = 28$.

The Chen system is a typical chaos anti-control model, which has a more complicated topological structure than Lorenz attractor.[41] The nonlinear differential equations that describe the Chen system are

$$\begin{cases} \dot{y}_1 = \alpha(y_2 - y_1), \\ \dot{y}_2 = (\gamma - \alpha)y_1 - y_1y_3 + \gamma y_2, \\ \dot{y}_3 = y_1y_2 - \beta y_3, \end{cases} \tag{28}$$

where $\alpha = 35$, $\beta = 3$ and $\gamma = 28$.

The Lü system is a typical transition system, which combines the Lorenz and the Chen attractors and represents the transition from one to another.[42] The Lü system is described by

$$\begin{cases} \dot{z}_1 = \rho(z_2 - z_1), \\ \dot{z}_2 = -z_1 z_3 + \nu z_2, \\ \dot{z}_3 = z_1 z_2 - \mu z_3, \end{cases} \tag{29}$$

where $\rho = 36$, $\mu = 3$ and $\nu = 20$.

The unified chaotic system created by Lü *et al* [43] bridges the gap between the Lorenz system and the Chen system. From its advent, the unified chaotic system has become an important plant of the synchronization control.[14,37–39] It is described by

$$\begin{cases} \dot{x}_4 = (25\alpha + 10)(y_4 - x_4), \\ \dot{y}_4 = (28 - 35\alpha)x_4 - x_4 z_4 + (29\alpha - 1)y_4, \\ \dot{z}_4 = x_4 y_4 - (\alpha + 8)z_4/3, \end{cases} \tag{30}$$

where $\alpha \in [0, 1]$. The unified chaotic system belongs to an extended Lorenz system when $0 \leq \alpha < 0.8$, Lü system when $\alpha = 0.8$, and extended Chen system when $0.8 < \alpha \leq 1$.

5.2. Synchronization between Lorenz and Chen systems

In this case we assume that the Lorenz system drives the Chen system. The initial conditions for the master and the slave system are $[x_1(0), x_2(0), x_3(0)] = [2, 0, 1]$ and$[y_1(0), y_2(0), y_3(0)] = [10, 5, -14]$, respectively. Other control parameters are chosen to be $r = 1.5$, $\boldsymbol{C} = [-2, 6, -3]$ and $\boldsymbol{K} = [1, 2, 2]^{\mathrm{T}}$. Seven radial basis functions are chosen to calculate the control input based on the trial and learning speed parameter γ that is set to be 1.5. Figure 1 illustrates that the states of the slave system can follow the master system successfully.

5.3. Synchronization between the Chen and the Lü systems

It is assumed that the Chen system drives the Lü system. The initial conditions for the master and the slave systems are $[y_1(0), y_2(0), y_3(0)] = [2, 0, 10]$ and $[z_1(0), z_2(0), z_3(0)] = [10, -14, -40]$, respectively. Other control parameters are chosen to be $r = 1.5$, $\boldsymbol{C} = [-2, 6, -3]$ and $\boldsymbol{K} = [1, 2, 2]^{\mathrm{T}}$. Nine radial basis functions are chose to calculate the control input based on the trial and learning speed parameter γ that is set to be 100. Figure 2 shows the simulation results for these two chaotic systems.

5.4. Synchronization between the Lorenz and the Lü systems with parametric disturbances

In this case we assume that the Lorenz system with parametric disturbances drives the Lü system. For the master system the disturbance are taken as $\delta a = 0.1, \delta b = 0.1$ and $\delta c = 0.2$. The initial conditions for the master and the slave systems are $[x_1(0), x_2(0), x_3(0)] = [2, 0, 1]$ and $[z_1(0), z_2(0), z_3(0)] = [10, -20, -1]$, respectively. Other control parameters are chosen to be $r = 1.5$, $\boldsymbol{C} = [-2, 6, -3]$ and $\boldsymbol{K} = [1, 2, 2]^{\mathrm{T}}$. Nine radial basis functions are chosen to calculate the control input based on the trial and learning speed parameter γ that is set to be 15. Figure 3 shows the numerical analysis results for these two chaotic systems.

5.5. Synchronization between the Lorenz and the Chen systems with noise added

To demonstrate the robustness of the proposed method for a noisy master chaotic system, we assume that the Lorenz system drives the Chen system. The initial conditions for the master and the slave systems are $[x_1(0), x_2(0), x_3(0)] = [2, 0, 1]$ and $[y_1(0), y_2(0), y_3(0)] = [10, 5, -14]$, respectively. Other control parameters are chosen to be $r = 1.5$, $\boldsymbol{C} = [-2, 6, -3]$ and $\boldsymbol{K} = [1, 2, 2]^{\mathrm{T}}$. Seven radial basis functions are chosen to calculate the control input based on the trial and learning speed parameter γ that is set to be 1.5. In this simulation, we add zero mean Gaussian noise with a variance of 0.1 to all states of the master system. Figure 4 shows simulation results for this case. Compared with the ideal case (5.2), the control input is not so smooth due to the noise effect. It also shows that the chattering does not appear with the continuous control.

5.6. Synchronization between the unified chaotic systems

The unified chaotic systems with different values of α are studied here, we assume that the extended Chen system ($\alpha = 0.9$) drives the extended Lorenz

system ($\alpha = 0.5$). The initial conditions for the master and the slave systems are $[x_{4m}(0), y_{4m}(0), z_{4m}(0)] = [2, 0, 1]$ and $[x_4(0), y_4(0), z_4(0)] = [10, 5, -14]$, respectively. Other control parameters are chosen to be $r = 1.5$, $\boldsymbol{C} = [-2, 6, -3]$ and $\boldsymbol{K} = [1, 2, 2]^{\mathrm{T}}$. Nine radial basis functions are chosen to calculate the control input based on the trial and learning speed parameter γ that is set to be 20. The simulation results are given in Fig.5 for this case.

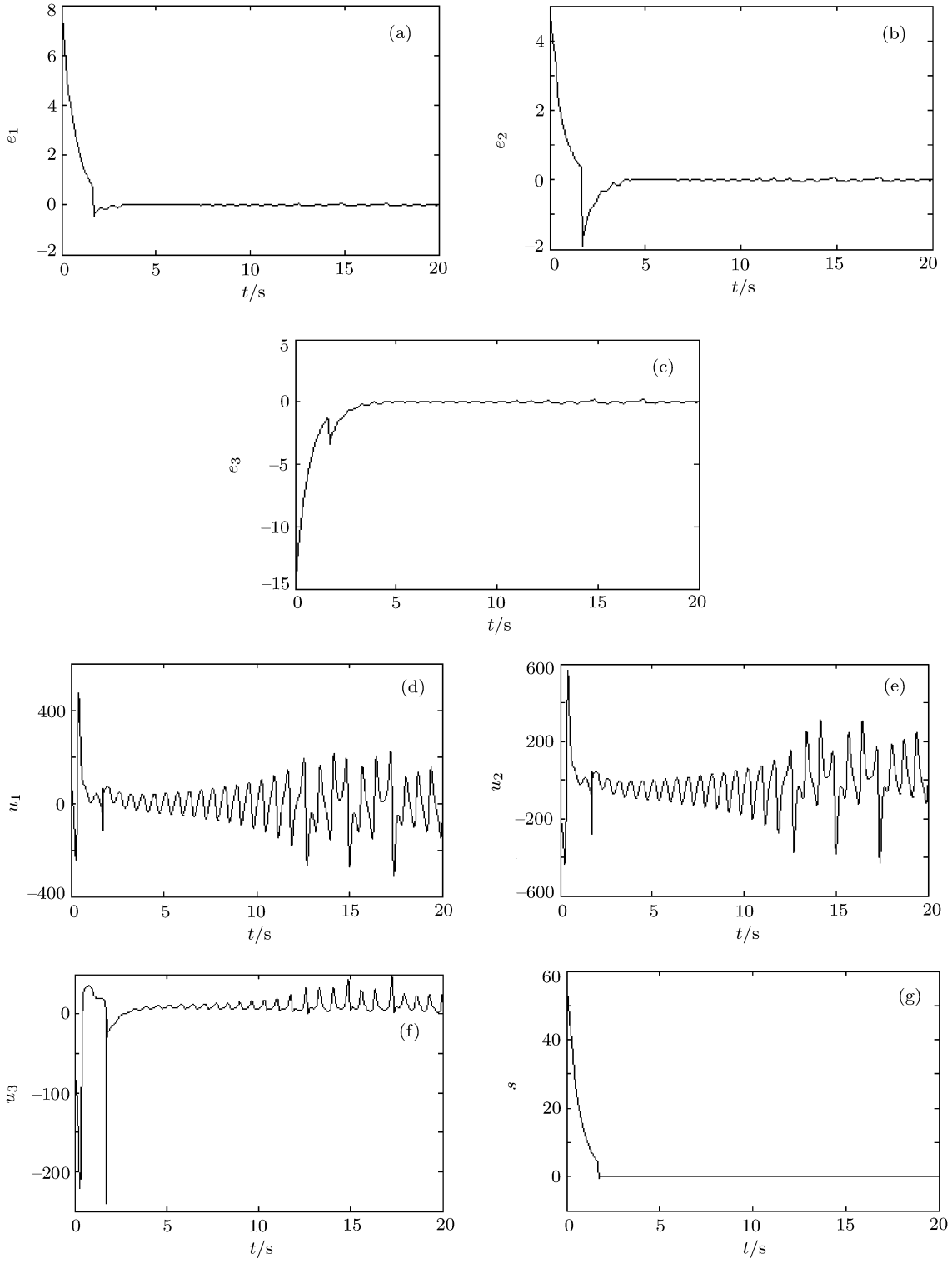


Fig.1. Synchronization between Lorenz and Chen systems, where (a) is for synchronization error e_1, (b) for synchronization error e_2, (c) for synchronization error e_3, (d) for control input u_1, (e) for control input u_2, (f) for control input u_3, and (g) for sliding variable s.

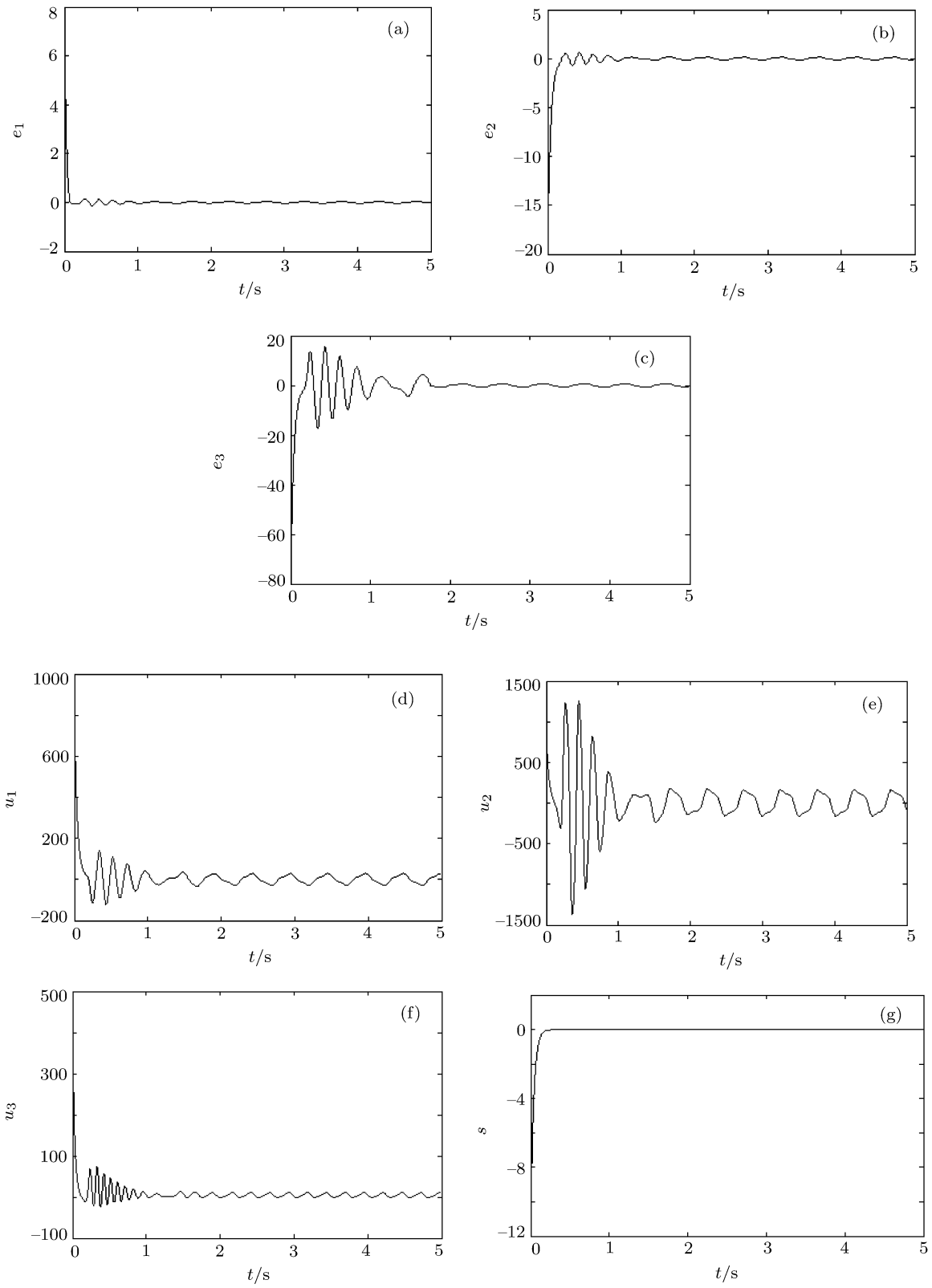


Fig.2. Synchronization between Chen and Lü systems, where (a) is for synchronization error e_1, (b) for synchronization error e_2, (c) for synchronization error e_3, (d) for control input u_1, (e) for control input u_2, (f) for control input u_3, and (g) for sliding variable s.

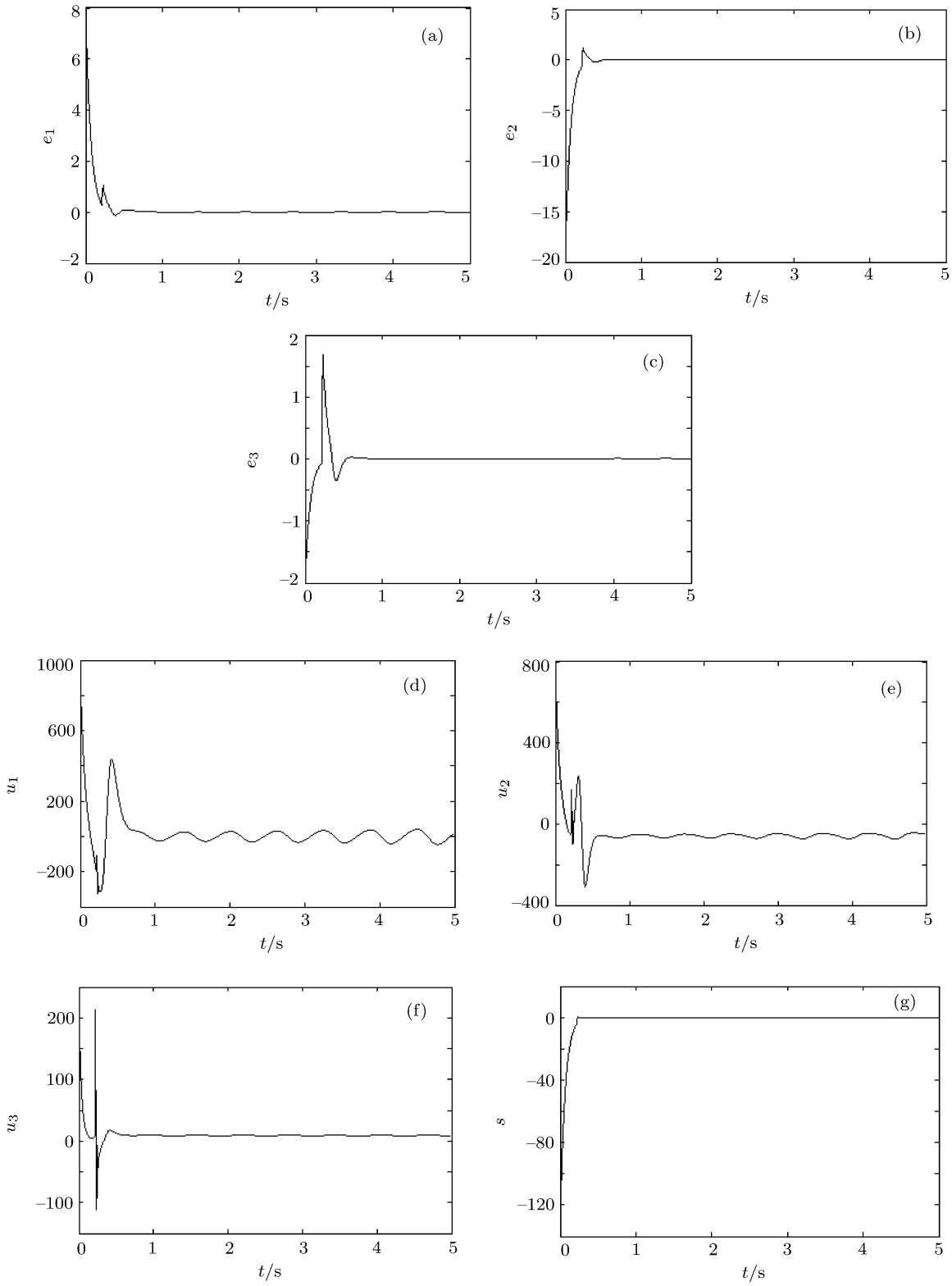


Fig.3. Synchronization between Lorenz and Lü systems, where (a) is for synchronization error e_1, (b) for synchronization error e_2, (c) for synchronization error e_3, (d) for control input u_1, (e) for control input u_2, (f) for control input u_3, and (g) for sliding variable s.

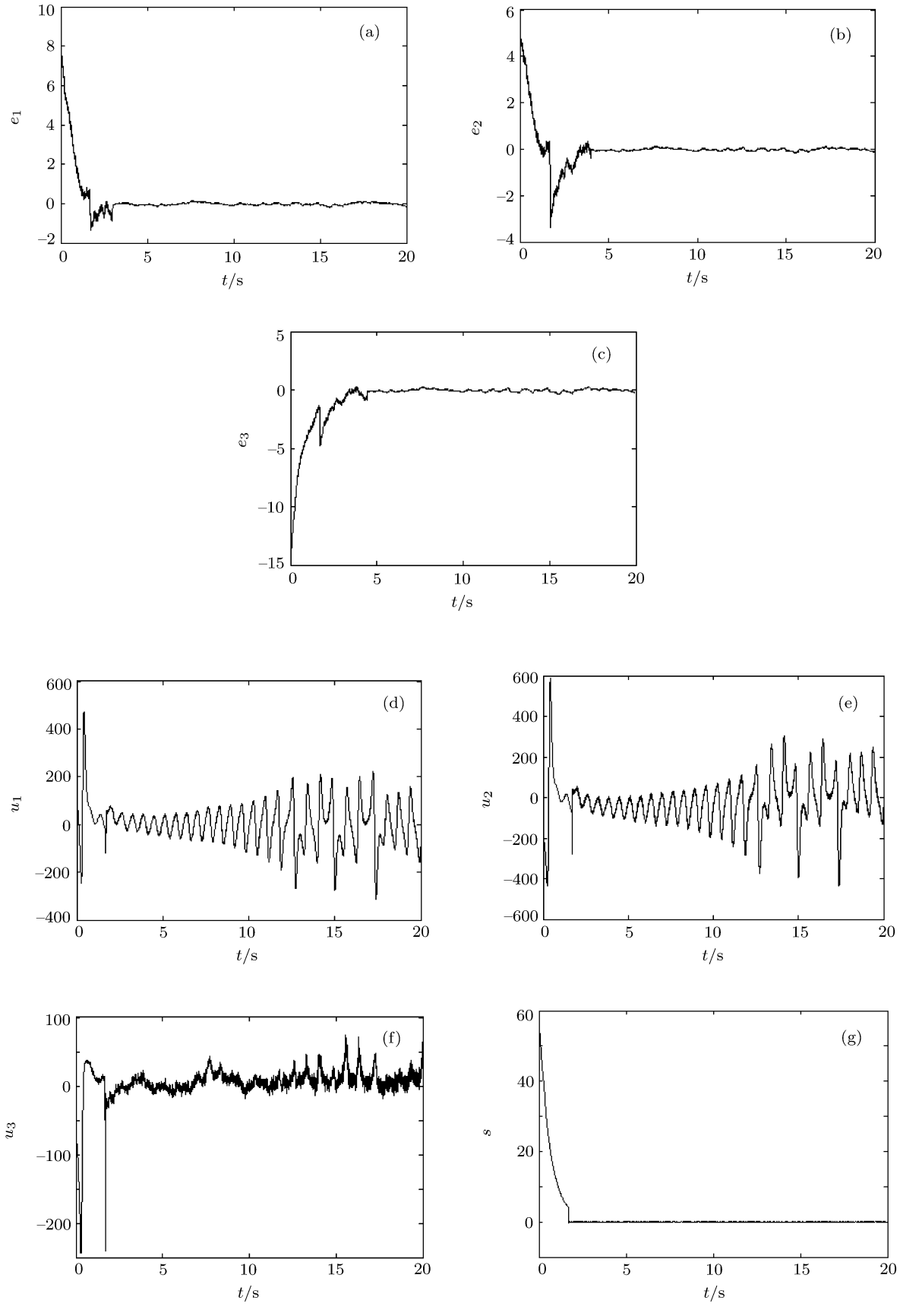


Fig.4. Synchronization between Lorenz and Chen systems with noise, where (a) is for synchronization error e_1, (b) for synchronization error e_2, (c) for synchronization error e_3, (d) for control input u_1, (e) for control input u_2, (f) for control input u_3, and (g) for sliding variable s.

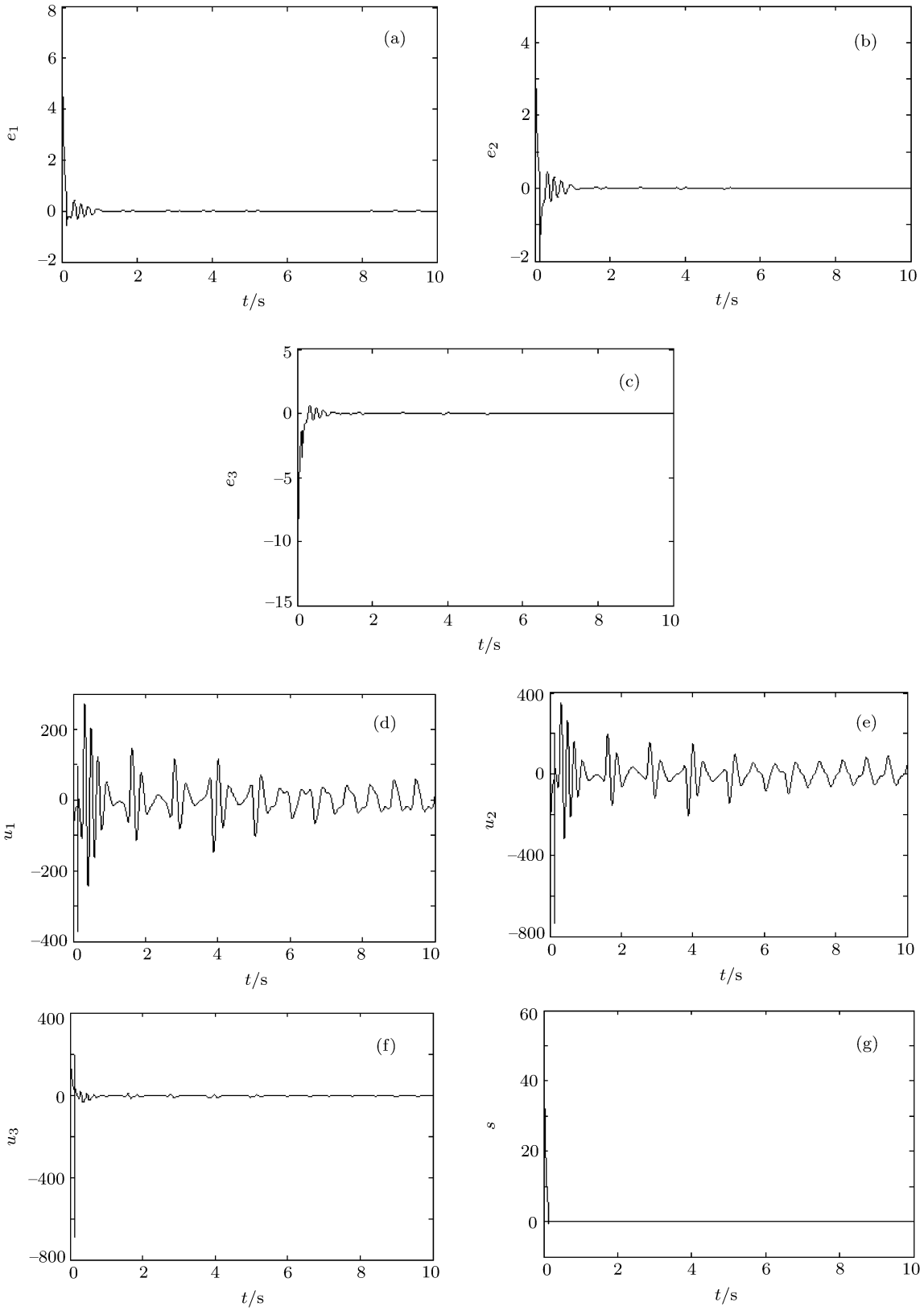


Fig.5. Synchronization between unified chaotic systems, where (a) is for synchronization error e_1, (b) for synchronization error e_2, (c) for synchronization error e_3, (d) for control input u_1, (e) for control input u_2, (f) for control input u_3, and (g) for sliding variable s.

6. Analysis and discussion

From the above simulation results we can conclude that this synchronization controller is valid and feasible, robustness to parametric disturbances and external disturbances of states are also provided. Meanwhile, we must note this fact that the control input is somewhat big just like other methods based on the sliding mode control. According to the theory of variable structure systems, the amplitude of control input is closely related to the sliding surface and sliding mode motion reaching law. So a suitable sliding surface design should be taken such that the control input will not exceed the physical limitations of the system. Considering the on-line computation burden, the number of radial basis functions should not be too large. Therefore, we must take a careful trade-off between the accuracy and the on-line computation burden to reduce the control cost.

7. Conclusions

In this paper, a chaotic synchronization method of different chaotic systems via an active RBF sliding controller is proposed. Starting from the active control, a controller based on the RBF neural networks and sliding mode control theory is designed. This single layer intelligent controller is very simple, and it has less than 10 RBF neurons and no more than 10 weights required to be regulated on-line. The weights of the RBF controller are updated on-line according to the reaching law of the sliding mode motion. In this way the external disturbance can be overcome by adjusting the control parameters. Numerical simulations illustrate the robustness and effectiveness of the proposed control strategy.

However, applying this control approach to synchronizing two different uncertain chaotic systems with parametric disturbances remains to be further studied.

Acknowledgment

The authors would like to thank Professor Liu Ding (Xi'an University of Technology, China) for his help.

References

[1] Pecora L M and Carroll T L 1990 *Phys. Rev. Lett.* **64** 821
[2] Boccaletti S, Kurths J, Osipov G, Valladares D L and Zhou C S 2002 *Phys. Rep.* **366** 1
[3] Gao X, Liu X W and Shao S Q 2007 *Chin. Phys.* **16** 2612
[4] Xiao Y Z and Xu W 2007 *Chin. Phys.* **16** 1597
[5] Chen F X and Zhang W D 2007 *Chin. Phys.* **16** 937
[6] Gao B J and Lu J A 2007 *Chin. Phys.* **16** 666
[7] Hu A H, Li F and Xu Z Y 2006 *Chin. Phys.* **15** 507
[8] Wang X Y and Wu X J 2006 *Acta Phys. Sin.* **55** 6261 (in Chinese)
[9] Zhang H, Ma X K and Liu W Z 2004 *Chaos, Solitons and Fractals* **21** 1249
[10] Park J H 2006 *Chaos, Solitons and Fractals* **27** 1369
[11] Liu C X and Wang F Q 2006 *Chin. Phys.* **15** 1971
[12] Zhu Z Y 2006 *Acta Phys. Sin.* **55** 6248 (in Chinese)
[13] Lin J S, Yan J J and Liao T L 2005 *Chaos, Solitons and Fractals* **24** 371
[14] Yu D C, Wu A G and Wang D Q 2006 *Chin. Phys.* **15** 306
[15] Yan J J, Yang Y S, Chiang T Y and Chen C Y 2007 *Chaos, Solitons and Fractals* **34** 947
[16] Wang Y W, Guan Z H and Wen X J 2004 *Chaos, Solitons and Fractals* **19** 899
[17] Chen S H and Lü J H 2002 *Chaos, Solitons and Fractals* **14** 643
[18] Guo Z, Lü L and Zhang C 2007 *Chin. Phys.* **16** 1603
[19] Jiang C S, Jiang Y M, Lin C S and Liu Y Z 2007 *Chin. Phys.* **16** 660
[20] Li S, Li R H, Li Y P and Xu W 2006 *Acta Phys. Sin.* **55** 5681 (in Chinese)
[21] Park J H 2006 *Chaos, Solitons and Fractals* **27** 549
[22] Guo Z A, Li Y, Lü L and Xia X L 2007 *Acta Phys. Sin.* **56** 95 (in Chinese)
[23] Li G H 2007 *Chin. Phys.* **16** 2608
[24] Gao J F, Xu L and Zhang C F 2007 *Acta Phys. Sin.* **56** 5124 (in Chinese)
[25] Cai G L and Huang J J 2006 *Acta Phys. Sin.* **55** 3997 (in Chinese)
[26] Yassen M T 2005 *Chaos, Solitons and Fractals* **23** 131
[27] Ho M C and Huang Y C 2002 *Phys. Lett.* A **301** 424
[28] Haeri M and Emadezadeh A A 2007 *Chaos, Solitons and Fractals* **31** 119
[29] Lü J, Zhou T and Zhang S 2002 *Chaos, Solitons and Fractals* **14** 529
[30] Yassen M T 2005 *Chaos, Solitons and Fractals* **23** 1245
[31] Zhang H G, Huang W, Wang Z L and Chai T Y 2006 *Phys. Lett.* A **350** 263
[32] Li N, Li J F and Lin H 2006 *Acta Phys. Sin.* **55** 3992 (in Chinese)
[33] Li A P, Meng Z Y, Yang D S and Zhang H G 2007 *Acta Phys. Sin.* **56** 3121 (in Chinese)
[34] Chen M, Jiang C S, Wu Q X and Chen W H 2007 *Chin. Phys. Lett.* **24** 890

[35] Fan Z P, Guan X P, Peng H P and Wang Y Q 2001 *Acta Phys. Sin.* **50** 1670 (in Chinese)

[36] Fei S M and Liu Y Z 2006 *Acta Phys. Sin.* **55** 1035 (in Chinese)

[37] Min F H and Wang Z Q 2005 *Acta Phys. Sin.* **54** 4026 (in Chinese)

[38] Lu J, Wu X, Han X and Lü J 2004 *Phys. Lett.* A **329** 327

[39] Li D M, Lu J A and Wu X Q 2005 *Chaos, Solitons and Fractals* **23** 79

[40] Lorenz E N 1963 *J. Atmos. Sci.* **20** 130

[41] Chen G R and Ueta Y 1999 *Int. J. Bifurcat. Chaos* **9** 1465

[42] Lü J and Chen G R 2002 *Int. J. Bifurcat. Chaos* **12** 659

[43] Lü J, Chen G R, Chen D and Celikovsky S 2002 *Int. J. Bifurcat. Chaos* **12** 2917

[44] Edwards Ch and Spurgeon S K 1998 *Sliding Mode Control—Theory and Applications* (London, Bristol: Taylor and Francis) p121

[45] Powell M J D 1992 *Advances in Numerical Analysis* (Oxford: Oxford University Press) p46

[46] Broomhead D S and Low D 1998 *Complex Systems* **2** 321