PRAMANA
— journal of
physics

© Indian Academy of Sciences

Vol. 79, No. 6
December 2012
pp. 1375–1383

A universal projective synchronization of general autonomous chaotic system

FUZHONG NIAN[1,*], XINGYUAN WANG[2], MING LI[1] and GE GUO[2]

[1]School of Computer & Communication, Lanzhou University of Technology, Lanzhou 730 050, China

[2]Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116 024, China

*Corresponding author. E-mail: gdnfz@lut.cn

MS received 14 May 2011; revised 23 April 2012; accepted 20 June 2012

Abstract. This paper investigates the generalized projective synchronization in general autonomous chaotic system. A universal controller is designed and the effectiveness is verified via theoretical analysis and numerical simulations. The controller design is irrelevant to concrete system structure and initial values. It has strong robustness and broad application perspective.

Keywords. Chaos; projective synchronization; universal.

PACS Nos 05.45.Xt; 05.45.−a; 05.45.Gg

1. Introduction

Chaos synchronization has been intensively studied since 1990. Various schemes of synchrony, such as complete synchronization [1], generalized synchronization [2], phase synchronization [3,4] and lag synchronization [5], have been studied. In recent years, projective synchronization has attracted a great deal of attention and it has been extensively investigated. However, the scaling factor is unpredictable [6], because it depends on both initial condition and the structure of specific chaotic system. This is inconvenient in real application. There are many works which study how to identify the scaling factor, e.g. Chee proposed a control algorithm based on Schur-Chon stability criteria to direct the scaling factor onto a predestined value in lower dimensional discrete-time systems [7]. Hu *et al* [8] realized the desired projective synchronization by introducing improved control scheme. In ref. [9], pinning control techniques were also adopted to direct the scaling factor onto the desired value. Many other methods can be found in refs [10–16]. However, all these works are limited only to partially linear systems, and practically there are many limitations. In this paper, a generalized projective self-synchronization of autonomous chaotic system was investigated, and a corresponding universal controller was designed,

DOI: 10.1007/s12043-012-0351-0; ***e*Publication: 26 September 2012**

and the validity of our method was verified through theoretical analysis and numerical simulations.

2. Generalized projective synchronization controller design

Consider a general autonomous system:

$$\dot{\mathbf{X}} = \mathbf{AX} + \mathbf{B}. \tag{1}$$

Here $\mathbf{X} = [x_1, x_1, \ldots, x_n]^{\mathrm{T}}$ is the state vector of the drive system, $\mathbf{B}$ is a $n \times 1$ constant vector, and $\mathbf{A}$ is a $n \times n$ square matrix:

$$\mathbf{A} = \begin{pmatrix} a_{11}(\mathbf{X}) & \ldots & a_{1n}(\mathbf{X}) \\ \vdots & \ddots & \vdots \\ a_{n1}(\mathbf{X}) & \cdots & a_{nn}(\mathbf{X}) \end{pmatrix}.$$

Here, $a_{ij}(\mathbf{X})$ is a function of $\mathbf{X}$. For simplicity, we write $a_{ij}(\mathbf{X})$ as a_{ij}. So, the matrix $\mathbf{A}$ can be represented as follows:

$$\mathbf{A} = \begin{pmatrix} a_{11} & \ldots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}.$$

Let eq. (1) be a drive system, with the corresponding response system as follows:

$$\dot{\mathbf{X}}' = \mathbf{A}'\mathbf{X}' + \mathbf{U} + \mathbf{B}, \tag{2}$$

where $\mathbf{X}' = [x_1', x_1', \ldots, x_n']^{\mathrm{T}}$ is the state vector of the response system, $\mathbf{A}'$ is a $n \times n$ square matrix:

$$\mathbf{A}' = \begin{pmatrix} a_{11}'(\mathbf{X}') & \ldots & a_{1n}'(\mathbf{X}') \\ \vdots & \ddots & \vdots \\ a_{n1}'(\mathbf{X}') & \cdots & a_{nn}'(\mathbf{X}') \end{pmatrix}.$$

Here, $a_{ij}'(\mathbf{X}')$ is a function of $\mathbf{X}'$. For simplicity, we rewrite $\mathbf{A}'$ as follows:

$$\mathbf{A}' = \begin{pmatrix} a_{11}' & \ldots & a_{1n}' \\ \vdots & \ddots & \vdots \\ a_{n1}' & \cdots & a_{nn}' \end{pmatrix}.$$

Here, $\mathbf{A}'$ is not the transfer of $\mathbf{A}$. $\mathbf{B}$ is a $n \times 1$ constant vector, and $\mathbf{U} = (u_1, u_2, \ldots, u_n)^{\mathrm{T}} \in R^n$ is the projective synchronization controller which we want to design.

However, what form is the controller $\mathbf{U}$? First, let us analyse what conditions should be satisfied when systems (1) and (2) achieve generalized projective synchronization.

Define the error function as follows:

$$\mathbf{e} = \mathbf{X} - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{I}\mathbf{X}', \tag{3}$$

where $\mathbf{I}$ is the $n \times n$ unit matrix, $\mathbf{e} = [e_1, e_2, \ldots, e_n]^{\mathrm{T}}$ is the error vector, $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_n]^{\mathrm{T}}$ is the scaling factor vector, $e_i = x_i - \beta_i x_i'$. From eqs (1) and (2), the corresponding error dynamical system is as follows:

$$\dot{\mathbf{e}} = \dot{\mathbf{X}} - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{I}\dot{\mathbf{X}}' = \mathbf{A}\mathbf{X} + \mathbf{B} - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{I}(\mathbf{A}'\mathbf{X}' + \mathbf{B} + \mathbf{U}). \tag{4}$$

To enable systems (1) and (2) to achieve the generalized projective synchronization, in the course of evolution, eq. (3) must be convergent. Choose Lyapunov function as follows:

$$V = \frac{1}{2}\mathbf{e}^{\mathrm{T}}\mathbf{e} = \frac{1}{2}\sum_{i=1}^{n} e_i^2. \tag{5}$$

According to Lyapunov stability principle, the time derivation along the system evolution trajectory $\dot{V}$ should be less than zero, that is,

$$\dot{V} = \sum_{i=1}^{n} e_i \dot{e}_i < 0. \tag{6}$$

Let $\dot{e}_i = -e_i$, thus

$$\dot{V} = \sum_{i=1}^{n} e_i \dot{e}_i = -\sum_{i=1}^{n} e_i^2 < 0. \tag{7}$$

Obviously, the result is what we want. So, let

$$\dot{e}_i = -e_i \tag{8}$$

and substituting eq. (4) into eq. (8), we obtain

$$\mathbf{A}\mathbf{X} + \mathbf{B} - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{I}(\mathbf{A}'\mathbf{X}' + \mathbf{B} + \mathbf{U}) = -\mathbf{e}.$$

By simplifying it, we can obtain

$$\boldsymbol{\beta}^{\mathrm{T}}\mathbf{I}\mathbf{U} = \mathbf{A}\mathbf{X} + \mathbf{B} - \boldsymbol{\beta}^{\mathrm{T}}\mathbf{I}(\mathbf{A}'\mathbf{X}' + \mathbf{B}) + \mathbf{e}. \tag{9}$$

According to eq. (9),

$$u_i = \frac{1}{\beta_i}\left[\sum_{j=1}^{n}\left(a_{ij}x_j - \beta_i a_{ij}' x_j'\right) + (1 - \beta_i)\, b_i + e_i\right], \quad i = 1, 2, \ldots, n. \tag{10}$$

Equation (10) is the controller we want to design. Reversing the above analysis process can prove that the controller (10) enables the response system (2) and the drive system (1) to asymptotically achieve generalized projective synchronization with scaling factor vector $\boldsymbol{\beta}$.

From above, we can see that the β is "arbitrary". In other words, it is not dependent on the initial values. This point was also verified by the following numerical simulations.

3. Numerical simulation

In order to verify the validity of the above-mentioned method, numerical simulations are given respectively for Lorenz system and the Rössler system.

According to the drive system (1) and the response system (2), for Rössler system, thus,

$$\mathbf{A} = \begin{pmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ x_3 & 0 & -c \end{pmatrix}, \quad \mathbf{A}' = \begin{pmatrix} 0 & -1 & -1 \\ 1 & a & 0 \\ x_3' & 0 & -c \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 0 \\ b \end{pmatrix}.$$

Here, $a = 0.2$, $b = 0.2$, $c = 5.7$. According to eq. (10), we can obtain the following controller as

$$\begin{cases} u_1 = \dfrac{1}{\beta_1}[\beta_1(x_3' + x_2') - (x_1 + x_2) + e_1] \\ u_2 = \dfrac{1}{\beta_2}[-\beta_2(x_1' + ax_2') + (x_1 + ax_2) + e_2] \\ u_3 = \dfrac{1}{\beta_3}[-\beta_3(x_3'x_1' - cx_3' + b) + x_3\,(x_1 - c) + e_3] \end{cases}. \tag{11}$$

Now consider Lorenz system as

$$\mathbf{A} = \begin{pmatrix} -a & a & 0 \\ r & -1 & -x_1 \\ x_2 & 0 & -b \end{pmatrix}, \quad \mathbf{A}' = \begin{pmatrix} -a & a & 0 \\ r & -1 & -x_1' \\ x_2' & 0 & -b \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix},$$

where $a = 10$, $b = 8/3$, $r = 28$. According to eq. (10), we can obtain the following controller as

$$\begin{cases} u_1 = \dfrac{1}{\beta_1}[ax_2 - ax_1 - \beta_1(ax_2' - ax') + e_1] \\ u_2 = \dfrac{1}{\beta_2}[rx_1 - x_2 - x_1x_3 - \beta_2(rx_1' - x_2' - x_1'x_3') + e_2] \\ u_3 = \dfrac{1}{\beta_3}[x_1x_2 - bx_3 - \beta_3(x_2'x_1' - bx_3') + e_3] \end{cases}. \tag{12}$$

For Lorenz system, let

$$\boldsymbol{\beta} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix},$$

and for Rössler system, let

$$\boldsymbol{\beta} = \begin{pmatrix} 2 \\ 0.5 \\ 1.5 \end{pmatrix}.$$

For both Lorenz system and Rössler system, let the initial value of the drive system and the response system respectively be

$$\mathbf{X} = [0.05, 0.05, 0.05]^{\mathrm{T}}, \quad \mathbf{X}' = [0.2, 0.2, 0.2]^{\mathrm{T}}.$$

Solving these two systems using fourth-order Runge–Kutta method, when the drive system and the response system achieve generalized projective self-synchronization, for Lorenz and Rössler systems, the attractor's projections on two-dimensional plane are given respectively as figures 1 and 2 (note: for the sake of clarity, in Lorenz system, the projections of the response system are moved down with 20 units, namely, in figures 1a, 1b and 1c, the projections of the response system are moved 20 units along the negative y-axis, the negative z-axis and the negative z-axis respectively, and in Rössler system, the projections of the response system are moved right with 15 units. That is, in figures 2a, 2b and 2c, the projections of the response system are moved 15 units along the

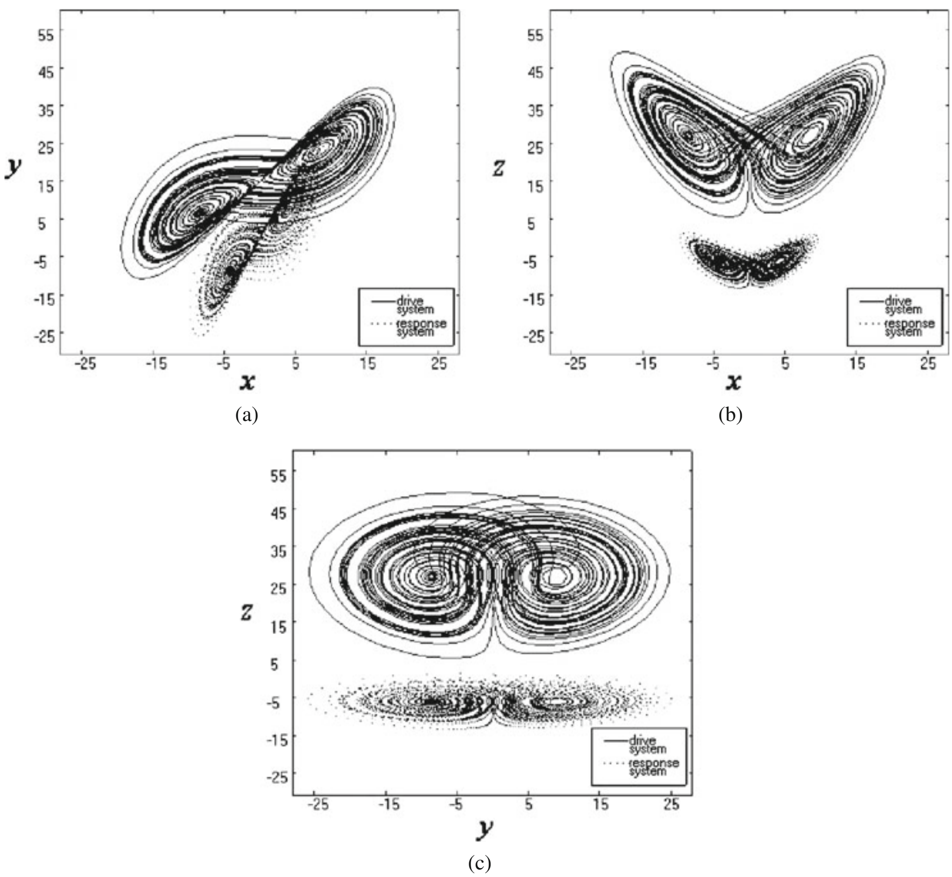

Figure 1. Lorenz system's projection after projective synchronization. **(a)** Projection on the plane of x–y, **(b)** projection on the plane of x–z, **(c)** projection on the plane of y–z.

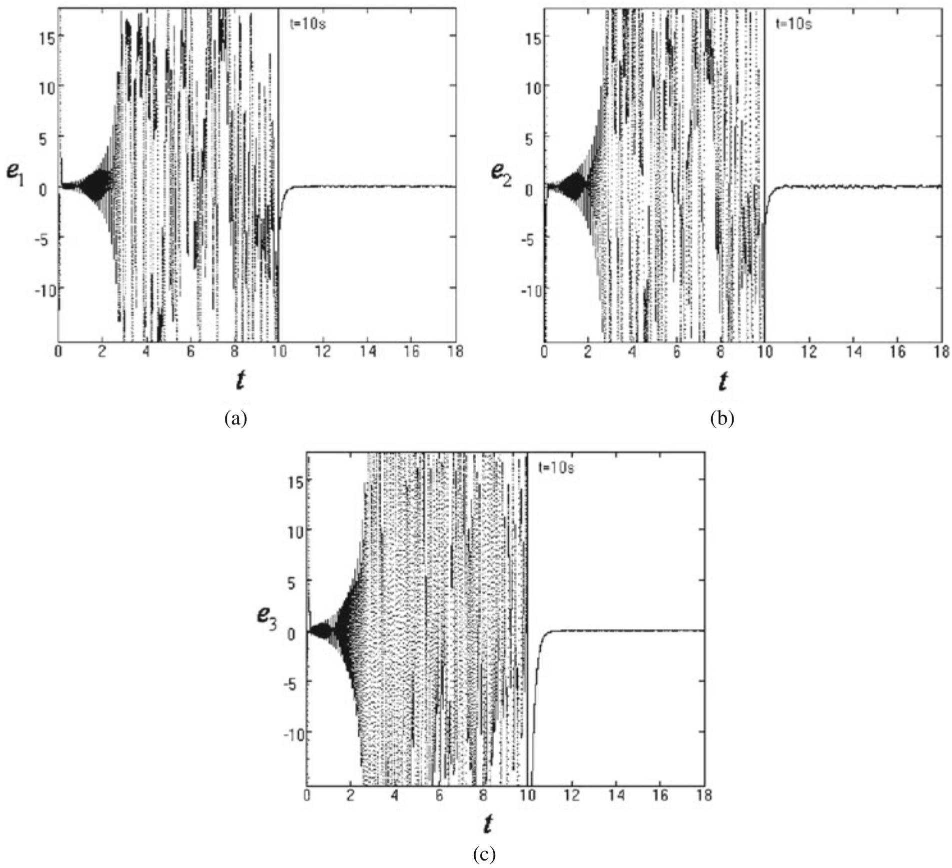

Figure 2. Error evolution of Lorenz system projective synchronization. **(a)** Evolution of e_1, **(b)** evolution of e_2 and **(c)** evolution of e_3.

positive x-axis, the positive x-axis and the positive y-axis respectively). In figure 1, when the Lorenz system achieves generalized projective self-synchronization, the scaling factor along the x-axis, y-axis and z-axis is 1, 2 and 3 respectively, and in figure 3, when the Rössler system achieves generalized projective self-synchronization, the scaling factor along the x-axis, y-axis and z-axis is 2, 0.5 and 1.5 respectively. As a result of using three different scaling factors in different directions, we observe that the attractor projections of the response system are deformed to some extent.

The comparison of error evaluation before and after achieving generalized projective self-synchronization are given in figures 3 and 4, respectively for Lorenz system and Rössler system. The fourth-order Runge–Kutta method is employed to solve the error evaluation of dynamical system (4) with time step $\Delta t = 0.0005$ s. For both Lorenz and Rössler systems, the initial conditions of the drive system and the response system are respectively (0.05, 0.05, 0.05) and (0.2, 0.2, 0.2). The control is activated at $t = 10$ s. We can observe from figures 3 and 4 that before activate control (corresponding to the

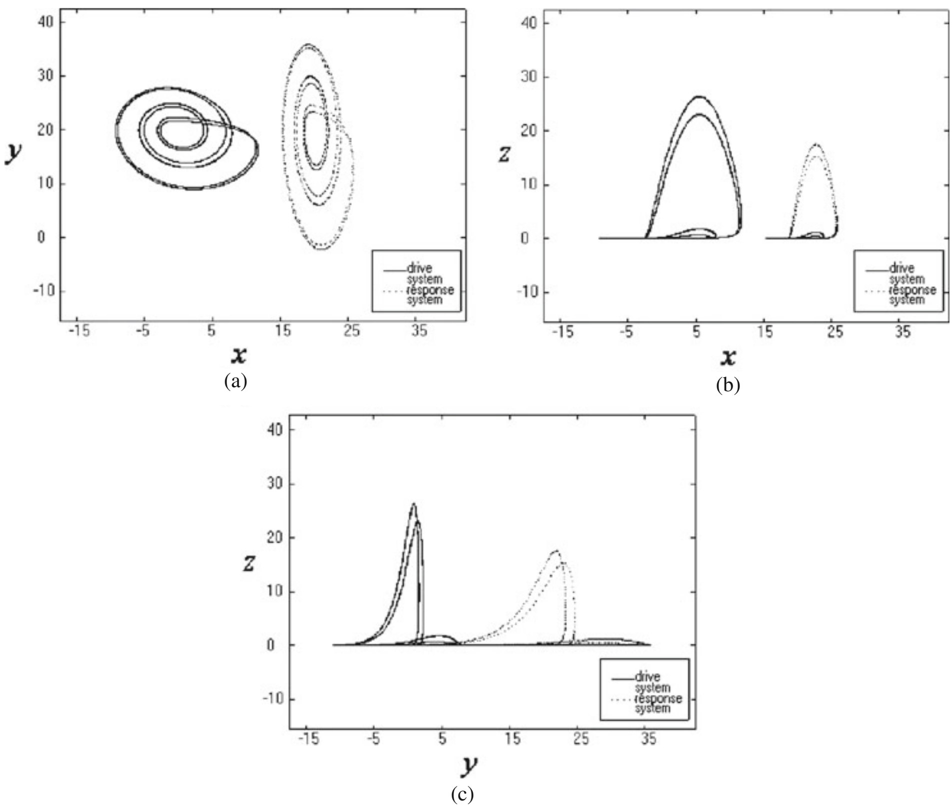

Figure 3. Rössler's projection after projective synchronization. **(a)** Projection on the plane of x–y, **(b)** projection on the plane of x–z and **(c)** projection on the plane of y–z.

time interval of 0–10 s in figures 3 and 4), the error jumps up and down occasionally with the passage of time. This characterizes the phenomenon of 'extension' and 'folding' in chaos, as well as sensitive dependence on initial conditions. After activate control (corresponding to the time interval of 10–18 s in figures 3 and 4), we can observe that the three errors along the x-, y- and z-axes quickly stabilize near zero, and this shows that the Lorenz and Rössler systems reached a generalized projective self-synchronization.

Choose a different scaling factor vector $\boldsymbol{\beta}$. Here, β_1, β_2 and β_3 were randomly selected from (0,10], and by the numerical simulation of the Lorenz and Rössler systems, we obtain results similar to the above results. This indicates that the shortcoming of scaling factor depending on initial condition has been overcome. In this paper, three different scaling factors are chosen for Lorenz and Rössler systems under different initial conditions, and from figures 3 and 4, we observe that both give good results. This shows that our method breaks the limitation that the same scaling factor is needed in projective synchronization. In addition, in this paper, obviously, the limitation of partially linear system is not emphasized, as well as other constraint conditions are not mentioned. Our

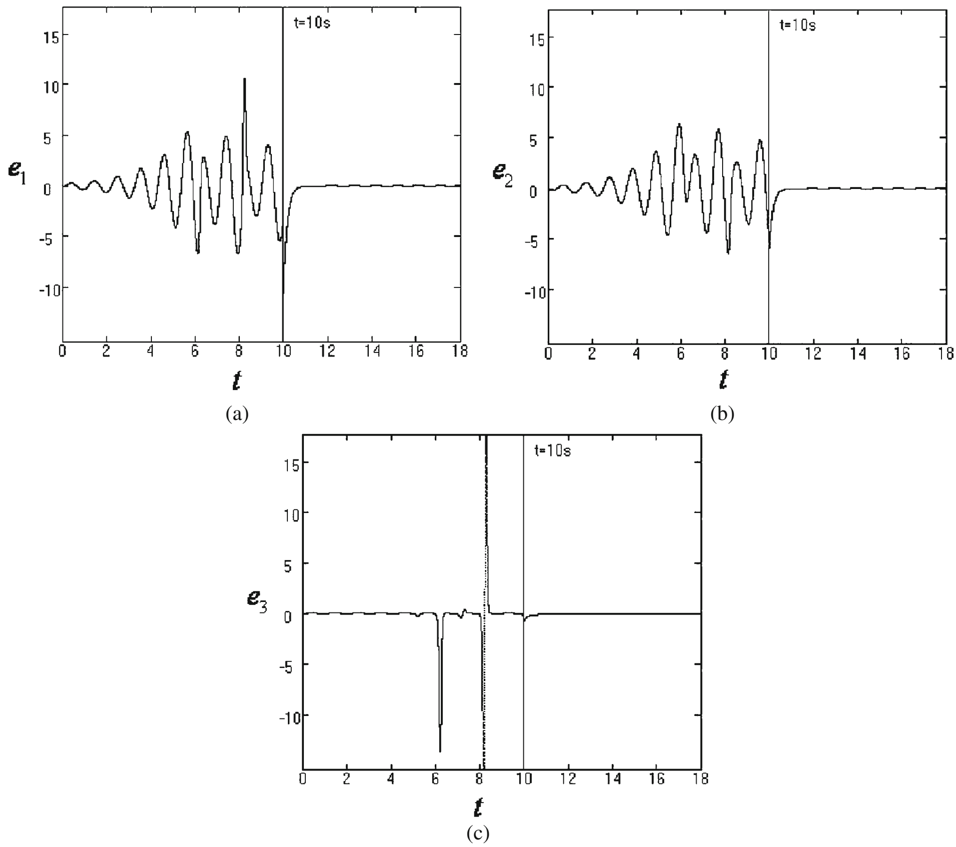

Figure 4. Error evolution of Rössler system projective synchronization. **(a)** Evolution of e_1, **(b)** evolution of e_2 and **(c)** evolution of e_3.

method aims at general autonomous system. Numerical simulation results show that by employing our method to different Lorenz and Rössler systems, both acquire satisfactory generalized projective synchronization. It shows our method is effective and has broad application perspectives.

4. Conclusion

This paper investigates the generalized projective synchronization in the general autonomous chaotic system. A generalized controller was designed, and the effectiveness was verified by theoretical analysis. The numerical simulations in different nonlinear autonomous chaotic system indicate that this method is robust. This controller does not depend either on initial conditions or on the structure of specific coupling system. It has good robustness and broad application perspective.

Acknowledgements

This research is supported by the National Natural Science Foundation of China (No. 61263019), the Fundamental Research Funds for the Gansu Universities (No. 1114ZTC144), the Natural Science Foundation of Gansu Province (No. 1112RJZA029) and the Doctoral Foundation of LUT.

References

[1] L Pecora and T Carroll, *Phys. Rev. Lett.* **64**, 821 (1990)
[2] X Wang and J Meng, *Acta Phys. Sinica* **56**, 6288 (2007)
[3] X Wang and J Meng, *Acta Phys. Sinica* **56**, 5142 (2007)
[4] G R Michael, S P Arkady and K Jurgen, *Phys. Rev. Lett.* **76**, 1804 (1996)
[5] C Li and X Liao, *Phys. Lett.* **A329**, 301 (2004)
[6] D Xu, *Phys. Rev.* **E63**, 027201 (2001)
[7] C Chee and D Xu, *Phys. Lett.* **A318**, 112 (2003)
[8] M Hu, Y Yang and Z Xu, *Phys. Lett.* **A**, DOI:10.1016/j.physleta.2008.01.054 (2008)
[9] M Hu, Y Yang, Z Xu, R Zhang and L Guo, *Physica* **A381**, 457 (2007)
[10] Y Hung, J Yan and T Liao, *Mathematics and Computers in Simulation* **77**, 374 (2008)
[11] F Nian and X Wang, *Chaos* **21**, 043131 (2011)
[12] Q Jia, *Phys. Lett.* **A373**, 1754 (2009)
[13] A Miliou, I Antoniades, S Stavrinides and A Anagnostopoulos, *Nonlinear Analysis: Real World Applications* **8**, 1003 (2007)
[14] F Nian and X Wang, *Int. J. Mod. Phys.* **C21**, 457 (2010)
[15] F Nian and X Wang, *Appl. Math. Comput.* **217**, 2481 (2010)
[16] G Grassi, *Chaos, Solitons and Fractals* **40**, 1029 (2009)