ELSEVIER

6 December 1999

PHYSICS LETTERS A

Physics Letters A 263 (1999) 315–322

www.elsevier.nl/locate/physleta

Finding the chaotic synchronizing state with gradient descent algorithm

J.Y. Chen [a,b], K.W. Wong [a,*], J.W. Shuai [c]

[a] *Department of Electronic Engineering, City University of Hong Kong, Kowloon Tong, Hong Kong*
[b] *Department of Physics, Xiamen University, China*
[c] *Department of Biomedical Engineering, Case Western Reserve University, Cleveland, OH 44106, USA*

Received 5 February 1999; received in revised form 12 July 1999; accepted 5 October 1999
Communicated by A.P. Fordy

Abstract

In this Letter, the gradient descent algorithm is proposed to search for the optimal synchronization between a drive and a response hyperchaotic systems that are coupled with a scalar signal. The energy corresponds to the largest Lyapunov exponent of the response system is defined. Its convergence indicates that the corresponding Lyapunov exponent is always negative.

PACS: 05.45.+b
Keywords: Nonlinear dynamics; Chaotic synchronization; Lyapunov exponents; Gradient descent algorithm

Chaotic synchronization proposed in Refs. [1–3] is widely studied in recent years. It was developed into various types of synchronization such as phase synchronization [4,5], noise-induced synchronization [6,7] and lag synchronization [8]. One of the important methods introduced in Ref. [9] is to synchronize chaos from a high-dimensional hyperchaotic coupled map system using a single scalar feedback signal formed by the linear combination of the original phase space variables. By varying the parameters of the feedback, all the Lyapunov exponents of the response system can be made negative. The amplitude of the feedback perturbation then diminishes as time passes, and synchronization is obtained. Theoretically, this method can be used to synchronize any drive-response systems because there are a large number of adjustable variables in the response system. Some special or extended cases of this method are also discussed [10–12].

The method described in Ref. [9] is general and important for synchronizing hyperchaotic systems. However, too many parameters need to be adjusted and there is not an effective way in finding the optimal parameters. For a m-dimensional system, there are $2m$ parameters to be adjusted. If the simple search method presented in Ref. [9] is used, the computational complexity will be extremely high for large m. Therefore the method is not

* Corresponding author. E-mail: eekww@cityu.edu.hk

0375-9601/99/$ - see front matter
PII: S0375-9601(99)00722-7

practical when it is applied to high-dimensional systems with many positive Lyapunov exponents. To deal with this problem, we propose to use the gradient descent algorithm to find out the optimal parameters.

Gradient descent algorithm is a widely-used approach for multidimensional optimization using first-order derivatives [13]. With multi-layered networks, it has been developed into the back-propagation algorithm and has been shown that can, in principle, solve any task which can be formulated as a continuous mapping from a given input space to a given output space [14,15].

In this Letter, the gradient descent algorithm is used to train the feedback control parameters so as to obtain the optimal values with which the largest Lyapunov exponent of the response system is negative.

The necessary and sufficient stability conditions of the synchronous state have been investigated extensively [16,17]. Given a chaotic system as the drive system,

$$\boldsymbol{x}(t+1) = \boldsymbol{F}(\boldsymbol{x}(t)) \tag{1}$$

where $\boldsymbol{x}(t) \in \boldsymbol{R}^m$ is a m-dimensional vector. In addition, we choose a scalar variable as a signal to be transmitted. It is governed by $u(t) = \boldsymbol{K}^T\boldsymbol{x}(t) = K_1 x_1(t) + K_2 x_2(t) + \cdots + K_m x_m(t)$ where $\boldsymbol{K}$ is a constant column vector and T denotes matrix transpose. The response system is then written as

$$\boldsymbol{y}(t+1) = \boldsymbol{F}(\boldsymbol{y}(t)) - \boldsymbol{B}(v(t) - u(t)) \tag{2}$$

where $\boldsymbol{y}(t)$ is a m-dimensional vector and t is the *iteration time index*. $v(t) = \boldsymbol{K}^T\boldsymbol{y}(t)$ is another scalar signal and $\boldsymbol{B} = (B_1, B_2, \ldots, B_m)^T$ is a m-dimensional constant vector [9]. The linear coupling for continuous-time system is realized by additive linear terms while for discrete mapping systems, a natural linear coupling is described by a multiplicative operator. However, for simplicity, here we only discuss a linear additive coupling in the mapping system (2).

To evaluate the largest Lyapunov exponent of the response system Eq. (2) with respect to the trajectory $\boldsymbol{y}(t) = \boldsymbol{x}(t)$, we define the infinitesimal deviation between $\boldsymbol{x}(t)$ and $\boldsymbol{y}(t)$ by $\boldsymbol{W}(t) = \boldsymbol{y}(t) - \boldsymbol{x}(t)$. As a result, Eqs. (1) and (2) lead to the following linearized equation:

$$\boldsymbol{W}(t+1) = \left[D\boldsymbol{F}|_{\boldsymbol{y}(t) \cong \boldsymbol{x}(t)} - \boldsymbol{B}\boldsymbol{K}^T \right] \boldsymbol{W}(t) \tag{3}$$

If the optimal values of the vectors $\boldsymbol{K}$ and $\boldsymbol{B}$ (a total of $2m$ parameters) are selected, the largest Lyapunov exponent of the response system (denoted as λ) can be made negative and the two chaotic systems are synchronized. However, if m is large, it is difficult to find the $2m$ optimal parameters. The definition of λ is

$$\lambda = \lim_{\xi \to \infty} \frac{1}{\xi} \ln \frac{|\boldsymbol{W}(\xi)|}{|\boldsymbol{W}(0)|} \tag{4}$$

where ξ denotes the total number of iterations. Practically, if ξ is large enough, the approximate value of λ can be obtained. When $\lambda < 0$, for typical $\boldsymbol{y}(0) \neq \boldsymbol{x}(0)$,

$$\lim_{t \to \infty} |\boldsymbol{y}(t) - \boldsymbol{x}(t)| = 0, \tag{5}$$

This implies that synchronization phenomena are observed.

In many practical cases, computing λ using Eq. (4) is not easy. There are two problems: Firstly, the exponential growth property in Eq. (4) may lead to numerical errors and overflows in computation. Secondly, it cannot hold all the time because in many cases the accessible phase space is bounded [18]. In order to solve the above two problems, we rewrite Eq. (3) in the following new form,

$$\boldsymbol{S}(t+1) = \boldsymbol{A}\boldsymbol{V}(t) \tag{6}$$

where matrix $\boldsymbol{A} = D\boldsymbol{F}|_{\boldsymbol{y}(t) \cong \boldsymbol{x}(t)} - \boldsymbol{B}\boldsymbol{K}^T$. $\boldsymbol{S}(t+1)$ is a vector analog to $\boldsymbol{W}(t+1)$ and $\boldsymbol{V}(t)$ a scaled unit vector at time t, defined as

$$\boldsymbol{V}(t) = \frac{\boldsymbol{S}(t)}{d(t)} \tag{7}$$

where

$$d(t) = |\boldsymbol{S}(t)|. \tag{8}$$

After the new vector $\boldsymbol{S}(t+1)$ is computed, the next scaled trajectory can be obtained from the equations $V(t+1) = \boldsymbol{S}(t+1)/d(t+1)$ and $d(t+1) = |\boldsymbol{S}(t+1)|$. If we start at time $t = 0$ with a trajectory at a distance $d(0) = |\boldsymbol{S}(0)|$, the computing process continues and yielding a sequence of distance $d(0), d(1), \ldots, d(t)$ from which the largest Lyapunov exponent can be computed:

$$\lambda = \lim_{\xi \to \infty} \frac{1}{\xi} \sum_{t=0}^{\xi-1} \ln d(t) \tag{9}$$

In practical calculations, λ is calculated in a finite time interval instead of $\xi \to \infty$. This corresponds to the finite-time Lyapunov exponent. The exact Lyapunov exponent can be approximated by this finite time value if the latter remains constant after a finite time interval. In the following analysis, λ is considered as a finite time approximation of the exact Lyapunov exponent.

We define the energy E as:

$$E = \exp(\lambda) \tag{10}$$

From this definition, we find that $E < 1$ if $\lambda < 0$ and $E \to 0$ as $\lambda \to -\infty$. The vectors $\boldsymbol{B}$ and $\boldsymbol{K}$ can be considered as two groups of parameters that can be trained by using the gradient descent algorithm to optimize the energy function. This means that the parameter vectors $\boldsymbol{B}$ and $\boldsymbol{K}$ can be modified automatically so as to make $E < 1$ or even $E \to 0$ in the learning process. The corresponding equations for these parameters are as follows:

$$\boldsymbol{B}(\tau+1) = \boldsymbol{B}(\tau) - \eta \frac{\partial E}{\partial \boldsymbol{B}(\tau)} \tag{11}$$

$$\boldsymbol{K}(\tau+1) = \boldsymbol{K}(\tau) - \eta \frac{\partial E}{\partial \boldsymbol{K}(\tau)} \tag{12}$$

where η is the scalar step size and τ is the *learning cycle*. the latter parameter is the integer part of t/ξ defined by $\tau = \mathrm{int}(t/\xi)$. This means that every ξ iterations correspond to a *learning cycle*. Note that the *learning cycle* τ is different from the *learning time* t. The latter is the original time scale while the former is used in the gradient descent algorithm to modify the vectors $\boldsymbol{B}(\tau)$ and $\boldsymbol{K}(\tau)$. Both of the parameters are modified in every *learning cycle* because the value of $\lambda(\tau)$ is only available in such interval. Here $\lambda(\tau)$ indicates the value of λ in the τth *learning cycle*. By this means, the larger the value of ξ, the higher is the precision of $\lambda(\tau)$ obtained. However, the learning speed will be slower. Besides ξ, η is also an important parameter that will affect the learning speed. A large η always results in a fast learning speed, but with strong perturbations. On the other hand, a small η will make the learning speed very slow and the energy easily traps in local minima. Therefore the parameters ξ and η should be selected properly in different systems so as to achieve the best learning process.

Detailed calculation of $\partial E/\partial \boldsymbol{B}(\tau)$ in a *learning cycle* is shown here. With Eqs. (9) and (10), we get,

$$\frac{\partial E}{\partial \boldsymbol{B}(\tau)} = \exp(\lambda) \lim_{\xi \to \infty} \frac{1}{\xi} \sum_{t=0}^{\xi-1} \left(\frac{1}{d(t)} \frac{\partial d(t)}{\partial \boldsymbol{B}(\tau)} \right) \tag{13}$$

From Eq. (8),

$$\frac{\partial d(t)}{\partial \boldsymbol{B}(\tau)} = \frac{1}{d(t)}\left(\boldsymbol{S}(t)\cdot\frac{\partial \boldsymbol{S}(t)}{\partial \boldsymbol{B}(\tau)}\right) \tag{14}$$

and from Eq. (6) we get

$$\frac{\partial \boldsymbol{S}(t)}{\partial \boldsymbol{B}(\tau)} = \frac{\partial \boldsymbol{A}}{\partial \boldsymbol{B}(\tau)}\boldsymbol{V}(t-1) + \boldsymbol{A}\frac{\partial \boldsymbol{V}(t-1)}{\partial \boldsymbol{B}(\tau)} \tag{15}$$

With Eq. (7), we get

$$\frac{\partial \boldsymbol{V}(t-1)}{\partial \boldsymbol{B}(\tau)} = \frac{1}{d(t-1)}\frac{\partial \boldsymbol{S}(t-1)}{\partial \boldsymbol{B}(\tau)} - \frac{\boldsymbol{S}(t-1)}{d(t-1)^2}\frac{\partial d(t-1)}{\partial \boldsymbol{B}(\tau)} \tag{16}$$

Eqs. (13)–(16) imply that if $\partial d(t)/\partial \boldsymbol{B}(\tau)$ and $\partial \boldsymbol{S}(t)/\partial \boldsymbol{B}(\tau)$ are calculated at every *iterating time*, the value of $\partial E/\partial \boldsymbol{B}(\tau)$ can be attained. In a similar process, the value of $\partial E/\partial \boldsymbol{K}(\tau)$ can also be obtained.

We consider a discrete hyperchaotic coupled map system that can be treated analytically as an example to illustrate the mechanism of the gradient descent method. The drive system is

$$x_i(t+1) = F_i(\boldsymbol{x}(t)) = (1-\varepsilon_{1x}-\varepsilon_2)f(x_i(t)) + \varepsilon_{1x}f(x_{i-1}(t)) + \varepsilon_2 f(x_{i+1}(t)) \tag{17}$$

And the response one is

$$y_i(t+1) = F_i(\boldsymbol{y}(t)) = (1-\varepsilon_{1y}-\varepsilon_2)f(y_i(t)) + \varepsilon_{1y}f(y_{i-1}(t)) + \varepsilon_2 f(y_{i+1}(t)) - B_i(v(t)-u(t)) \tag{18}$$

where i $(i=1,2,\ldots,m)$ labels the system site and $f(x)$ is the local chaotic dynamic at the system site. $\varepsilon_{1x},\varepsilon_{1y}$ and ε_2 are the coupling strengths between the two different system sites. $u(t)=\boldsymbol{K}^T\boldsymbol{x}(t)=K_1x_1(t)+K_2x_2(t)+\cdots+K_mx_m(t)$ is the transmitted scalar signal, and $v(t)=\boldsymbol{K}^T\boldsymbol{y}(t)=K_1y_1(t)+K_2y_2(t)+\cdots K_my_m(t)$. To facilitate the analysis, let $f(x)=2x \bmod 1$ which is the so-called one-dimensional baker map [9,18], $m=5$, $\varepsilon_{1x}=\varepsilon_{1y}=0.8$ and $\varepsilon_2=0.05$. Without the transmitted scalar signal, the five Lyapunov exponents of Eq. (17) are calculated to be 0.69, 0.50, 0.45, 0.33 and 0.30, all being positive.

According to Eq. (6), the components of the vectors $\boldsymbol{S}(t)=[S_1(t),S_2(t),\cdots,S_m(t)]$ are given by:

$$S_i(t) = \sum_{j=1}^{m}\left(\frac{\partial F_i(y_i(t-1))}{\partial y_j(t-1)} - B_i(\tau)K_j\right)V_j(t-1) \tag{19}$$

$$\frac{\partial S_i(t)}{\partial B_p(\tau)} = \begin{cases} \displaystyle\sum_{j=1}^{m}\left[-K_jV_j(t-1)+\left(\frac{\partial F_i(y_i(t-1))}{\partial y_j(t-1)}-B_i(\tau)K_j\right)\left(\frac{1}{d(t-1)}\frac{\partial S_j(t-1)}{\partial B_p(\tau)}-\frac{S_j(t-1)}{d(t-1)^2}\frac{\partial d(t-1)}{\partial B_p(\tau)}\right)\right] & p=i \\ \displaystyle\sum_{j=1}^{m}\left[\left(\frac{\partial F_i(y_i(t-1))}{\partial y_j(t-1)}-B_i(\tau)K_j\right)\left(\frac{1}{d(t-1)}\frac{\partial S_j(t-1)}{\partial B_p(\tau)}-\frac{S_j(t-1)}{d(t-1)^2}\frac{\partial d(t-1)}{\partial B_p(\tau)}\right)\right] & p\neq i \end{cases} \tag{20}$$

$$\frac{\partial d(t)}{\partial B_p(\tau)} = \frac{1}{d(t)}\sum_{j=1}^{m}\left(S_j(t)\frac{\partial S_j(t)}{\partial B_p(\tau)}\right) \tag{21}$$

where $p,i,j=1,2,\cdots,m$ are the system sites. With the initial vectors $\boldsymbol{S}(0)$, $\boldsymbol{K}(0)$ and $\boldsymbol{B}(0)$, one can calculate the values of $d(0),d(1),\cdots,d(t)$ and the vectors $\boldsymbol{S}(0),\boldsymbol{S}(1),\cdots,\boldsymbol{S}(t)$ from Eqs. (6)–(8). According to Eqs. (20) and (21) and let $\partial S(0)/\partial B_p(\tau)=\partial d(0)/\partial B_p(\tau)=0$, one can get

$$\frac{\partial d(0)}{\partial B_p(\tau)} \xrightarrow{Eq.\,(20)} \frac{\partial S_i(1)}{\partial B_p(\tau)} \xrightarrow{Eq.\,(21)} \frac{\partial d(1)}{\partial B_p(\tau)} \xrightarrow{Eq.\,(20)} \frac{\partial S_i(2)}{\partial B_p(\tau)},\cdots.$$

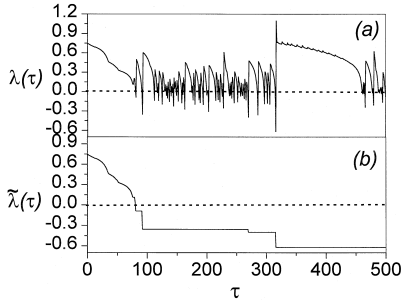

Fig. 1. Synchronization of hyperchaotic systems, (a) the largest Lyapunov exponent of the response system $\lambda(\tau)$, and (b) its transformation $\tilde{\lambda}(\tau)$ vs the learning cycle τ.

From Eqs. (11) and (13), the vector $\boldsymbol{B}(\tau)$ is updated to $\boldsymbol{B}(\tau+1)$ after a *learning cycle*. In a similar way, the expressions $\partial S_i(t)/\partial K_p(\tau)$ and $\partial d(t)/\partial K_p(\tau)$ can be attained and both values are computed at every *iterating time*. As a result, the vector $\boldsymbol{K}(\tau)$ is also updated to $\boldsymbol{K}(\tau+1)$ after a *learning cycle*.

Now suppose that the drive system transmits a scalar signal $u(t)$ to the response system for the purpose of achieving synchronization. In traditional ways, the task of searching the parameter spaces of $\boldsymbol{B}(\tau)$ and $\boldsymbol{K}(\tau)$ is difficult for there are 10 parameters to be optimized in Eq. (18). If all of them are optimized between $[-5,5]$ and the largest Lyapunov exponent of the response system is calculated in the interval of 0.1 among these ten parameters, the maximum number of calculations required is 101^{10}. For this system with all the Lyapunov exponents positive, the parameter space with the largest Lyapunov exponent negative is always restricted to a very small region [9]. However, with our gradient descent method, it is not difficult to get the optimal parameters.

All the ten parameters are initialized randomly between $[-1,1]$ and let $\xi = 1000$, $\eta = 0.05$. In each *learning cycle*, all the parameters are modified once using the gradient descent method. Moreover, the modifications are made at the same time. To facilitate the analysis, we consider $\lambda(\tau) < 0$ as global convergence. The learning process is shown in Fig. 1(a), in which the perturbations are the result of oscillations in the converged state since the corresponding energy is not equal to zero. These perturbations lead to an evident characteristic: The more negative $\lambda(\tau)$ achieves, the higher is the jump of $\lambda(\tau)$ in the next *learning cycle*. When $\lambda(\tau) = -0.62$ which is the largest negative value in all the learning processes, it jumps to the highest value in the next *learning cycle*. This is because the convergent region in the parameter space is so small that a small distance from the convergent point may cause the energy jumps up to a high value. If we select a m-dimensional chaotic system that possesses fewer positive Lyapunov exponents among all the m Lyapunov exponents, the perturbations will be much lower. However, all the Lyapunov exponents are positive in our example.

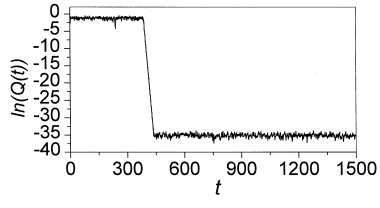

Fig. 2. A plot of time versus the logarithm of the average absolute difference $Q(t)$ between the drive and the response systems when $\lambda(\tau) = -0.62$.

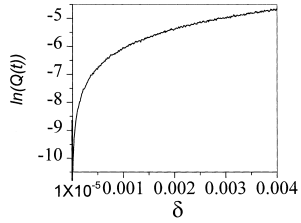

Fig. 3. A plot of mismatch parameter δ between ε_{1x} and ε_{1y} in the drive and the response systems versus the logarithm of the average absolute difference $Q(t)$. The interval of δ used in the simulation is $\Delta\delta = 1 \times 10^{-5}$.

In order to view the decrease in λ clearly, we perform a transformation as follows:

$$\tilde{\lambda}(\tau+1) = \begin{cases} \lambda(\tau+1) & \text{if } \lambda(\tau+1) \le \lambda(\tau) \\ \lambda(\tau) & \text{if } \lambda(\tau+1) > \lambda(\tau) \end{cases} \tag{22}$$

where $\tilde{\lambda}(\tau)$ is the convergent orbit of $\lambda(\tau)$ without agitation. The result is shown in Fig. 1(b). The drive and the response systems can achieve the most stable synchronization with the largest negative $\tilde{\lambda}(\tau)$. After $\tau = 316$ *learning cycles*, the largest negative value is $\tilde{\lambda}(\tau) = -0.62$ and the optimal parameters are $\boldsymbol{B} = (-0.38, 0.26, -0.53, -1.2, -0.39)^T$ and $\boldsymbol{K} = (0.08, 0.16, -0.89, -0.70, -0.92)^T$. This state can be considered as the global convergence with which the drive and the response systems can achieve stable synchronization even if there are small parameter mismatches between these systems [10].

In Fig. 2, we present the result of our synchronization experiment performed between the drive and the response systems by plotting the distance between the two trajectories. The optimal parameters found above are adopted here. The average absolute difference $Q(t)$ is defined as:

$$Q(t) = \frac{1}{m} \sum_{i=1}^{m} |y_i(t) - x_i(t)| \tag{23}$$

Within the resolution of the figure, chaotic synchronization is achieved in about 430 iterations. The figure shows that $Q(t)$ reduces rapidly after a finite number of iterations. After that, it remains at a small value (about e^{-35}) but never vanishes. It is this small value that provides feedback to the response system so as to keep the stable synchronization between the two systems.

We further test the stability of synchronization on a mismatch in the system parameters. Let $\varepsilon_{1y} = 0.8$ and $\varepsilon_{1x} = \varepsilon_{1y} + \delta$ where δ is a small mismatch parameter. The parameter δ ranges from 1×10^{-5} to 0.004. When $\delta = 0.0$, $\ln(Q(t)) \approx -35.1$ as found from Fig. 2. However, when $\delta = 1 \times 10^{-5}$, $\ln(Q(t))$ jumps up to -10.7. Further increase δ from 1×10^{-5} to 0.004 leads to a percentage mismatch between parameters ε_{1x} and ε_{1y} ranges from 0.00125% to 0.5%. Accordingly, $\ln(Q(t))$ increases from -10.7 to -4.7, as shown in Fig. 3. From the results, we find that even when the percentage mismatch is up to 0.5%, the value of $\ln(Q(t))$ is still small enough to maintain the synchronization of the two systems.

Table 1
Learning results of the hyperchaotic systems. Here Max., Min., Ave. and Fai. correspond to Maximum, Minimum, Average values and Number of Failure, respectively

ξ	τ			$\tilde{\lambda}(\tau)$			Fai.
	Max.	Min.	Ave.	Max.	Min.	Ave.	
1,000	500	107	330	−0.75	−0.31	−0.50	0
5,000	497	64	276	−0.84	−0.32	−0.51	0

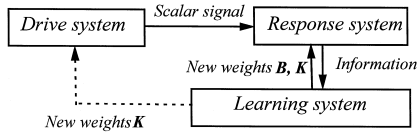

Fig. 4. A schematic diagram showing the relationship between the drive, response and learning systems.

We randomly set initial $\boldsymbol{B}(0)$ and $\boldsymbol{K}(0)$ between $[-1,1]$ to further demonstrate the effectiveness of the gradient descent method. If the number of *learning cycles* exceeds 500 but $\tilde{\lambda}(500)$ is still positive, this *learning process* is considered a failure. This means that 500 *learning cycles* from a *learning process*. In order to find out the effect of τ on the precision of $\lambda(\tau)$ (or $\tilde{\lambda}(\tau)$), simulations with $\xi = 1{,}000$ and $\xi = 5{,}000$ are performed and the results are listed in Table 1. In 100 *learning processes*, *the number of learning cycles* and $\tilde{\lambda}(\tau)$ in every *learning process* are stable and no *learning process* failed. This indicates that our method is effective and stable for chaotic systems with many positive Lyapunov exponents. The different values of ξ only affect the precision of $\tilde{\lambda}(\tau)$ slightly. This implies that $\xi = 1000$ is capable of finding the optimal parameters for stable synchronization in this system.

Our method for seeking the optimal parameters automatically leads to great advantages in the applications of synchronization. The corresponding schematic diagram is shown in Fig. 4. According to the information extracted from the response system, the learning system can find the optimal parameters with which the largest Lyapunov exponent of the response system is always negative. Then these optimized parameters are fed back to both the drive and the response systems. If the optimal parameters of $\boldsymbol{B}$ cannot be fed back to the drive system, we can only train the optimal vector $\boldsymbol{K}$ to achieve stable synchronization on line, as shown by a dotted line in Fig. 4.

In summary, the optimal parameters in forming a scalar signal to achieve stable synchronization between the drive and the response systems can be found by the gradient descent algorithm. We define the energy function as the exponent of the largest Lyapunov exponent of the response system and find the optimal parameters by minimizing this energy function. A chaotic system with five positive Lyapunov exponents is adopted to show the effectiveness of our method. As a result, it is not necessary to examine the largest Lyapunov exponent in getting the optimal parameters because they can be found automatically by the *learning process*. This method is particularly useful for high-dimensional chaotic systems, and so is an effective technique for the synchronization of practical chaotic systems.

Acknowledgements

This work was supported by the Strategic Research Grant provided by the City University of Hong Kong. The authors would like to thank Dr. Kelvin S.Y. Yuen for fruitful discussions and the anonymous reviewers for valuable comments.

References

[1] H. Fujisaka, T. Yamada, Progr. Theor. Phys. 69 (1983) 32.
[2] A.S. Pikosvshy, Z. Phys. B55 (1984) 149.
[3] L.M. Pecora, T.L. Caroll, Phys. Rev. Lett. 64 (1990) 821.
[4] A.S. Pikovsky, M.G. Rosenblum, G.V. Osipov, J. Kurths, Physica D 104 (1997) 219.
[5] F.S. De San Roman, S. Boccaletti, D. Maza, H. Mancini, Phys. Rev. Lett. 81 (1998) 3639.
[6] J.W. Shuai, K.W. Wong, Phys. Rev. E 57 (1998) 7002.
[7] P. Parmananda, Y. jiang, Phys. Lett. A 241 (1998) 173.

[8] M.G. Rosenblum, A.S. Pikovsky, J. Kurths, Phys. Rev. Lett. 78 (1997) 4193.
[9] J.H. Peng, E.J. Ding, M. Ding, W. Yang, Phys. Rev. Lett. 76 (1996) 904.
[10] G.A. Johnson, D.J. Mar, T.L. Carroll, L.M. Pecora, Phys. Rev. Lett. 80 (1998) 3956.
[11] R. Brown, N.F. Rulkov, Phys. Rev. Lett. 78 (1997) 4189.
[12] J.M. Gutierrez, A. Iglesias, Phys. Lett. A 241 (1998) 174.
[13] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, Numerical Recipes, Cambridge University Press, Cambridge, London, 1986.
[14] Y. Chauvin, D.E. Rumelhart, Backpropagation: Theory, Architecture and Application, Lawrence Erlbaum, Hillsdale, NJ, 1995.
[15] P.T. Quinlan, Neural Networks 11 (1998) 577.
[16] N.F. Rulkov, M.m. Sushchik, Int. J. Bifurcation Chaos Appl. Sci. Eng. 7 (1997) 625.
[17] D.J. Gauthier, J.C. Bienfang, Phys. Rev. Lett. 77 (1996) 1751.
[18] H.J. Korsch, H.J. Jodl, Chaos: A Program Collection for the PC, Springer-Verlag, Berlin, Heidelberg, 1994.