Monte Carlo Optimization Search for Synchronization of Coupled Systems

Jiann-Horng Lin

***Abstract*— A scheme for the synchronization of two chaotic oscillators is proposed when a mismatch between the parameter values of the systems to be synchronized is present. We have shown how the proposed Monte Carlo optimization search algorithm can be used to adapt the parameters in two coupled systems such that the two systems are synchronized although their behaviour is chaotic and they have started with different initial conditions and parameter settings. The Markov chain sampling is a powerful tool for the optimization of complicated objective functions. It is introduced in order to more efficiently search the domain of the objective function. In many applications these functions are deterministic and randomness. The maximum statistics converge to the maximum point of probability density which establishing links between the Markov chain sampling and optimization search. This statistical computation algorithm demonstrates convergence property of maximum statistics in large samples and it is global search design to avoid on local optimal solution restrictions. The controlled system synchronizes its dynamics with the control signal in the periodic as well as chaotic regimes. The method can be seen also as another way of controlling the chaotic behaviour of a coupled system. In the case of coupled chaotic systems, under the interaction between them, their chaotic dynamics can be cooperatively self-organized.**

***Index Terms*— nonlinear dynamical system, adaptive synchronization, Monte Carlo, Markov chain**

I. Introduction

Synchronization of chaos is a cooperative behavior of coupled nonlinear systems with chaotic uncoupled behavior. This behavior appears in many physical and biological processes. It would seem to play an important role in the ability of complex nonlinear oscillators, such as neurons, to cooperatively act in the performance of various functions. In recent years, there has been particular interest in the study of chaos synchronization of similar oscillators. Such synchronization strategies have potential applications in several areas, such as secure communication [1] [2] and biological oscillators [3] [4] [5]. In most of the analysis done on two coupled chaotic systems, the two systems are assumed to be identical. In practical implementations this will not be the case. In this paper, we study the synchronization of two coupled nonlinear, in particular chaotic, systems which are not identical. We show how adaptive controllers can be used to adjust the parameters of the systems such that the two systems will synchronize.

Jiann-Horng Lin, Dept. of Information Management, I-Shou University, Taiwan

We first present the problem we are dealing with in a mathematical fashion then we discuss the general principle of the proposed sampling algorithm and explain how to use this algorithm for conformational search. Finally, we present our computational results.

II. Cooperative Self-Synchronization of Coupled Chaotic Systems

One of the properties of oscillations generated by nonlinear dynamical systems is their ability to be synchronized. Synchronization between periodic oscillations of mutually coupled dynamical systems is a well-known phenomenon in physics, engineering and many other scientific disciplines. Chaos synchronization may seem unlikely due to the extreme sensitivity of chaos to initial conditions as well as small random disturbances and parameter variation. However, it has been realized that even chaotic systems may be coupled in a way such that their chaotic oscillations are synchronized. Mutual synchronization can be considered as a form of cooperative self-organization of interacting systems. In contrast to the case of coupled periodic systems, even in the case of coupled chaotic systems, under the interaction between them their chaotic dynamics can be cooperatively self-organized. When this phenomenon occurs there is complete or almost complete coincidence of regular or chaotic realizations generated by identical or almost identical systems with different initial conditions. We consider the case of synchronized chaos where two coupled systems evolve identically in time. Given two chaotic systems, the dynamics of which are described by the following two sets of differential equations:

$$\dot{x} = f(x) \quad (1)$$

$$\dot{y} = f(y) \quad (2)$$

where $x \in R^n$, $y \in R^n$, and $f : R^n \to R^n$ is a nonlinear vector field, systems (1) and (2) are said to be synchronized if

$$e(t) = (y(t) - x(t)) \to 0, \ t \to \infty \quad (3)$$

where e represents the synchronization error. A common system to study such synchronized behavior is the system of two uni-directionally identical coupled oscillators

$$\dot{x} = f(x) \quad (4)$$

$$\dot{y} = \tilde{f}(y, h(x)) \quad (5)$$

where x and y are n-dimensional vectors, $h : R^n \to R^m$, $m \le n$, and $\tilde{f}(x, h(x)) = f(x)$. This condition implies existence of an invariant n-dimensional synchronization manifold $x = y$. If $z(t)$ is a chaotic solution of $\dot{x} = f(x)$, then a synchronous chaotic state is defined by $x = y = z(t)$, and it resides on the manifold. In the following the first and second oscillator (4) and (5) will be called *drive* and *response*, respectively. To emphasize the drive-response nature of equations (4) and (5), one can rewrite equation (5) as

$$\dot{y} = \tilde{f}(y, s(t)) \tag{6}$$

where $s(t)$ is a driving signal, and hence, equation (5) can be considered as a special case of a system driven by chaotic signals. Following [6], synchronization of uni-directionally coupled systems may be defined as follows:

Definition 1. The response system (5) synchronizes with the drive system (4) if the *synchronization manifold*

$$M = \{(x, y) \in R^n \times R^n : x = y\}$$

is an attracting set with a *basin of attraction* $B = B_x \times B_y \supset M$ such that:

$\lim_{t\to\infty} \| x(t) - y(t) \| = 0$, $\forall (x(0), y(0)) \in B$.

We will explicitly examine one type of drive-response coupling for identical systems that have chaotic uncoupled dynamics. Assume the dynamics of the driving system is given by equation (4). In the presence of coupling the dynamics of the response system becomes

$$\dot{y} = f(y) + E(x - y) \tag{7}$$

where E is a vector function of its argument and represents coupling between the systems. We assume $E(0) = 0$, hence synchronization occurs on the invariant manifold given by $x = y$.

III. Error-Feedback Coupling and Observer Design for Synchronization

The problem just described is closely related to the observer problem from control theory. An observer is a typical device designed for estimation of unknown state vectors of a dynamical system. Observers are often used in control systems. In the case of error-feedback synchronization (equation (7)), design of the feedback and analysis of the stability of the error system can be viewed as a special case of the observer design problem, which is well known in the control theory literature. The formalism offered via the observer theory allows us to provide a reasonable comprehensive framework for synchronization issues. A standard approach in solving the observer problem in control theory is to use as receiver a copy of the transmitter (of course with unknown initial state) modified with a term depending on the difference between the received signal and its prediction derived from the observer. The synchronization problem requires one to establish global asymptotic stability for the zero solution of the error dynamics, the dynamics governing the difference between the transmitter state and observer state. An observer is a dynamical system designed to be driven by the output of another dynamical system (plant) and having the property that the state of the observer converges to the state of the plant. More precisely, the following definition [7] [8] is given.

Definition 2. Given dynamical system

$$\dot{x} = f(x) \tag{8}$$

and let $s = h(x) \in R^m$ be an output of the system or an observed signal, the dynamical system

$$\dot{y} = f(y) + g(s - h(y)) \tag{9}$$

is said to be a nonlinear observer of system (8) if y converges to state x as $t\to\infty$, where $g : R^m \to R^m$ is a suitably chosen nonlinear function.

Moreover, system (9) is said to be a global observer of system (8) if $y \to x$ as $t \to \infty$ for any initial condition $y(0)$, $x(0)$. The pair (f, h) is called *observable*, if the full state vector x can be reconstructed from the signal s. Furthermore, system (9) is a (global) observer of system (8) if the error system

$$\begin{aligned}\dot{e} &= f(y) + g(h(x) - h(y)) - f(x)\\ &= f(x + e) + g(h(x) - h(x + e)) - f(x)\\ &= \mathcal{E}(e, t)\end{aligned} \tag{10}$$

has a (globally) asymptotically stable equilibrium point for $e = 0$ [8]. For example, if f and h are linear transformations given by $\dot{x} = A(x)$ and $s = Cx$ where A is an $n \times n$ matrix and C is a $1 \times n$ matrix, then $g = KC(x - y)$ is chosen in such a way that the equation:

$$e^{\cdot} = x^{\cdot} - y^{\cdot} = (A - KC)(x - y)$$

has an asymptotically stable fixed point at the origin, that is, $A - KC$ is a stable matrix. Sufficient and necessary conditions for the existence of such an $n \times 1$ *gain matrix* K are given, for example, in [7]. Consider the nonlinear system given below

$$\dot{x} = A_\mu(x) + g_\mu(x) \ , s = Cx, \tag{11}$$

where $A_\mu \in R^{n\times n}$ is a constant matrix with certain plant parameters denoted by μ, $C_\mu \in R^{m\times n}$ is a constant matrix, $g_\mu : R^n \to R^n$ is a differentiable function. Assume that g_μ satisfies the following Lipschitz condition:

$$\| g_\mu(x_i) - g_\mu(x_j) \| \le \rho \| x_i - x_j \|, \ \forall x_i, x_j \in R^n \tag{12}$$

where $\rho > 0$ is a Lipschitz constant. For the system given by (11), we choose the following observer for synchronization:

$$\dot{x} = A_\mu(x) + g_\mu(x) + K(s - \hat{s}), \ \hat{s} = Cy \tag{13}$$

where $K \in R^{n\times m}$ is a gain matrix to be determined. In this formulation, equation (11) represents the drive system, and equation (13) represents the response system. The output s of the drive system is used in the response system and the problem is to choose the gain K so that the solutions of the equations (11) and (13) asymptotically synchronize, i.e. $\lim_{t\to\infty} \| x(t) - y(t) \| = 0$. The observer given by equation (13) is known as the full order observer (see e.g. [9]). Let us define the synchronization error as $e = x - y$. By using equations (11) and (13) we obtain the following error equation

$$\dot{e} = (A_\mu - KC)e + g_\mu(x) - g_\mu(y) \tag{14}$$

It has been shown that there exists a gain vector K such that the system (13) is an exponential observer for (11). For details and the procedure to determine the gain vector see [10].

Remark 1 For observable pair (C, A_μ), there always exists a matrix K such that $A_c = A_\mu - KC$ is stable. For some pairs (C, A_μ) there may exist a matrix K such that A_c is stable, even if the pair is not observable. Such pairs are called *detectable* (see e.g. [11]).

Remark 2 The condition (12) may seem too restrictive. However for chaotic systems, the trajectories always remain in a bounded region of the state space. Hence whenever $g_\mu(x)$ differentiable, a Lipschitz condition can be found in that bounded region.

Remark 3 The above observer construction scheme is still valid when the system is not in the form equation (11) but can be transformed into this form by a coordinate transformation.

Note that we have assumed that the chaotic systems (equation (11)) consists of a linear part plus a nonlinear part. This is common to a wide class of systems such as the Duffing oscillator, the Chua circuit and the Lorenz equation. For example, consider the Duffing equation [12]:

$$\ddot{x} + p\dot{x} + p_1 x + x^3 = q\cos(\omega t) \tag{15}$$

The state space description of Duffing equation is

$$\dot{x}_1 = x_2 \tag{16}$$

$$\dot{x}_2 = -p_1 x_1 - x_1^3 - p x_2 + q\cos(\omega t) \tag{17}$$

It is well known that the solutions may exhibit chaos for particular parameter combinations. We note that the system is in the form of (11) when $s = x_1$ is chosen as output. The matrix pair (A_μ, C) with

$$A_\mu = \begin{pmatrix} 0 & 1 \\ -p_1 & -p \end{pmatrix}, C = (1\ 0) \tag{18}$$

is observable whenever $p_1 \neq 0$. A full observer may be constructed as

$$\dot{y}_1 = y_2 + k_1(x_1 - y_1) \tag{19}$$

$$\dot{y}_2 = -p_1 y_1 - y_1^3 - p y_2 + q\cos(\omega t) + k_2(x_1 - y_1) \tag{20}$$

By appropriate selection of the gain $K = (k_1 \quad k_2)^T$ the error dynamics $\dot{e} = (A_\mu - KC)e$ can be made asymptotically stable.

IV. Chaos Synchronization Through Parameter Adaptation

The error-feedback coupling described in the previous section can synchronize the trajectory of a nonlinear system to a desired unstable orbit, provided the desired unstable orbit has the same parameter values as the system under control. If there are deviations in the system parameters the synchronization will be degraded. In order to avoid no synchronization in the case of parameter mismatch, we introduce an adaptive algorithm that allow us to adapt the receiver's parameter to the transmitter one and synchronize chaotic signal. Consider two chaotic systems with evolution equations $\dot{x} = f(x, \mu_x)$, $\dot{y} = f(y, \mu_y)$, where μ_x, μ_y are parameters of the systems. Complete synchronization between the drive and response can be realized by matching the parameters of the response to that of the drive through a loop of adaptive control. This is implemented by augmenting the evolution equation for the dynamical system by an additional equation for the evolution of the parameter(s) as described below.

$$\dot{x} = f(x, \mu_x)$$

$$\dot{y} = f(y, \mu_y) + F(x, y)$$

$$\dot{\mu}_y = G(\bullet)$$

Here, $F(x, y)$ denotes coupling between the drive system (x) and the response system (y). The function G acts on the Monte Carlo optimization as presented in the next section. A block diagram for this adaptive synchronization mechanism is shown in Figure 1. The scheme is adaptive since in the above procedure the parameters which determine the nature of the dynamics self-adjust or adapt themselves to yield the desired dynamics. Using such an adaptive control function the drive system and the response system eventually synchronized, although their behavior is chaotic and they have started with different initial conditions and parameter settings.

Our aim is to devise an algorithm to adaptively adjust the parameters in the secondary system, μ_y, until the system variables, y, and the parameters themselves converge to their counterparts in the primary system, i.e., both $y \to x$ and $\mu_y \to \mu_x$. In this way, synchronization between both systems is achieved and the parameters of the primary system are identified. Let $x = [x_1, x_2, \cdots, x_N]^T \in R^N$ be the state vector of the chaotic system, $\dot{x}$ is the derivative of the state vector x. Based on the measurable state vector $x = [x_1, x_2, \cdots, x_N]^T$, for particle i, we define the following fitness function

$$\textit{fitness} = \sum_{t=0}^{k} ((x_1(t) - x_{i1}(t))^2 + \cdots + (x_N(t) - x_{iN}(t))^2)$$

where t = 0, …, k. Therefore, the problem of parameter identification is transformed to that of using the Monte Carlo optimization to search for the suitable value of the parameter μ_y such that the fitness function is globally minimized.

Reconstruction of chaotic systems using chaotic synchronization

The chaos synchronization described in this section can also be used to the parameter identification of chaotic systems. In practical applications we may know a good deal about the structure of the system. In many cases, only a set of parameters in the system are needed to be determined. Thus the problem is reduced to the parameter identification. For the purpose of the parameter identification, a special system has been designed. When this specific system has the same parameters as those of the original unknown system whose parameters are unknown, these two systems are synchronized under one single driving signal from the unknown system. In general, they have different parameters in the beginning and are not synchronized. But, the parameters of the special system can be tuned so that the two systems are synchronized. Therefore, the parameters of the unknown system are obtained.

Let $\dot{x} = f(x, \mu_x)$ be the dynamical system whose parameters $\mu_x \in R^m$ are to be estimated. The only information available is a time series $s(t)$ given by a (scalar) observable $s = h(x)$ and the structure of the model f. Furthermore, let us assume that we are able to construct a dynamical system $\dot{y} = \tilde{f}(s, y, \mu_y)$ that synchronizes ($y \to x$ for $t \to \infty$) if $\mu_x = \mu_y$. If the functional form of the vector field f is known, such a system can be constructed by the method described in this paper. This method observes the previous time series of a single variable. However, when the parameters are identified, the future time series of all variables can be predicted.

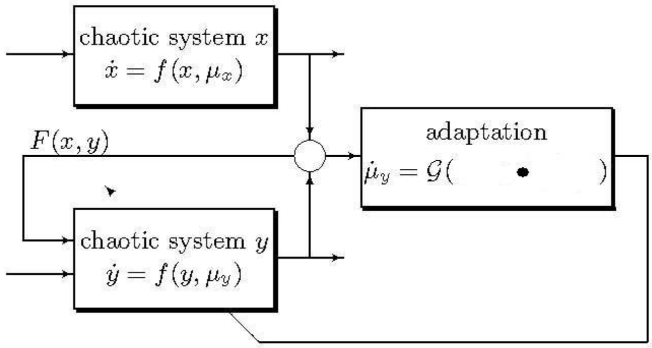

Figure 1. Synchronization of two coupled chaotic systems

V. Markov Chain Monte Carlo

Markov chain Monte Carlo methods are a class of sample-generating techniques by controlling how a random walk behaves. It attempts to directly draw samples from some complex probability distribution based on constructing a Markov chain that has the desired distribution as its equilibrium distribution. The state of the chain after a large number of steps is then used as a sample of the desired distribution. The quality of the sample improves as a function of the number of steps. Usually it is not hard to construct a Markov chain with the desired properties. The more difficult problem is to determine how many steps are needed to converge to the stationary distribution within an acceptable error. The Markov chain Monte Carlo has become a powerful tool for Bayesian statistical analysis, Monte Carlo simulations, and potentially optimization with high nonlinearity. There are many ways to choose the transition probability, and different choices will result in different behaviour of Markov chain. In essence, the characteristics of the transition kernel largely determine how the Markov chain of interest behaves, which also determines the efficiency and convergence of Markov chain Monte Carlo sampling. There are several widely used sampling algorithms, such as Metropolis-Hasting Algorithm [13] and Gibbs Sampler [14].

A. Metropolis-Hastings Sampling Algorithm

The basic idea of Markov chain Monte Carlo methods is to construct a Markov chain with the specified stationary distribution, namely $\pi(\theta)$, then run the chain with full length till the sample chain value close enough to its stationary distribution. Then take stationary chains as the samples of $\pi(\theta)$ and make variety of statistical inference based on these samples. The most popular Markov chain Monte Carlo sampling method is Metropolis-Hastings algorithm, which means sampling starts from another easily known reversible Markov chain Q, and obtain the new Markov chain by comparing. It generates a random walk using a proposal density and a method for rejecting proposed moves.

To draw samples from the target distribution, we let $\pi(\theta) = \beta \cdot p(\theta)$, where β is a normalizing constant which is either difficult to estimate or not known. We will see later that the normalizing factor β disappear in the expression of acceptance probability. The Metropolis-Hastings algorithm essentially expresses an arbitrary transition probability from state θ to ϕ as the product of an arbitrary transition kernel $q(\theta,\phi)$ and a probability $\alpha(\theta,\phi)$. That is,

$$P(\theta,\phi) \equiv P(\theta \to \phi) = q(\theta,\phi)\alpha(\theta,\phi)$$

Here q is the proposal distribution function, while $\alpha(\theta,\phi)$ can be considered as the acceptance rate from state θ to ϕ, and can be determined by

$$\alpha(\theta,\phi) = \min\left\{\frac{\pi(\phi)q(\phi,\theta)}{\pi(\theta)q(\theta,\phi)}, 1\right\} = \min\left\{\frac{p(\phi)q(\phi,\theta)}{p(\theta)q(\theta,\varphi)}, 1\right\}$$

The essence of Metropolis-Hastings algorithm is to first propose a candidate θ^*, then accept it with probability α. That is, $\theta_{t+1} \leftarrow \theta^*$ if $\alpha \geq u$ where u is a random value drawn from an uniform distribution in [0, 1], otherwise $\theta_{t+1} \leftarrow \theta_t$. It is straightforward to verify that the reversibility condition is satisfied by the Metropolis-Hastings kernel

$$q(\theta,\phi)\alpha(\theta,\phi)\pi(\theta) = q(\phi,\theta)\alpha(\phi,\theta)\pi(\phi),$$

for all θ, ϕ. Consequently, the Markov chain will converge to a stationary distribution which is the target distribution $\pi(\theta)$.

In a special case, when the transition kernel is symmetric is its arguments, or $q(\theta,\phi) = q(\phi,\theta)$, for all θ, ϕ, then the acceptance rate $\alpha(\theta,\phi)$ become

$$\alpha(\theta,\phi) = \min\left\{\frac{p(\phi)}{p(\theta)}, 1\right\},$$

And the Metropolis-Hastings algorithm reduces to the classic Metropolis algorithm. In this case, the associated Markov chain is called as symmetric chain. In a special case when $\alpha = 1$ is used, that is the acceptance probability is always 1, then the Metropolis-Hastings degenerates into the classic widely used Gibbs sampling algorithm. However, Gibbs sampler becomes very inefficient for the distributions that are non-normally distributed or highly nonlinear.

B. Random Walk and Levy Flight

A random walk is a random process which consists of taking a series of consecutive random steps. The sum of each consecutive step which is a random step drawn from a random distribution forms a random walk. It means the next state will only depend on the current existing state and the transition from the existing state to the next state. This is typically the main property of a Markov chain. If the step size obeys the Gaussian distribution, the random walk becomes the Brownian motion. In theory, as the number of steps increases, the central limit theorem implies that the random walk should approaches a Gaussian distribution. If the step size obeys other distribution, we have to deal with more generalized random walk. A special case is when the step size obeys the Levy distribution, such a random walk is called a Levy flight or Levy walk. Levy flight is a random walk whose step length is drawn from the heavy-tailed Levy distribution often in terms of a simple power law formula. It is worth to point out that a power law distribution is often link to some scale free characteristics, and Levy flights can thus show self-similarity and fractal behaviour in the fight patterns.

VI. Markov Chain Sampling for Optimization Search

A simple random walk can be considered as a Markov chain. In a probability distribution, the largest density area is mostly tending to be sampled. So the sampling density function should converge to the maximum point of maximum probability if the sample is sufficiently large. Thus establishing links between the function maximum value and sampling extreme statistics. We can use Markov chain Monte Carlo to simulate a sample of this distribution. And the optimal will appear most frequently in the sample. That is, the optimal state will have the greatest probability.

Suppose that we are interested in exploring solutions x that minimize an objective function $f(x) \in \mathrm{R}$, where $x = (x_1, ..., x_n) \in R^n$. That is, if we want to find the minimum of an objective function $f(x) \in \mathrm{R}$ at $x = x^*$ so that $f^* = f(x^*) \le f(x)$. We can convert it to a target distribution for a Markov chain

$$\pi(x) \propto e^{-\beta f(x)}$$

where $\beta > 0$ is a parameter which act as a normalized factor. β should be chosen so that the probability is close to 1 when $x \to x^*$. At $x = x^*$, $\pi^* = \pi(x^*) \ge \pi(x)$. This often requires that the formulation of $f(x)$ should be non-negative, which means that some objective functions can be shifted by a large constant $\mathrm{C} > 0$ if necessary. Then, a Markov chain is constructed to sample $\pi(x)$. Typically, the solutions in the vicinity of the global minimum of $f(x)$ are most likely to be drawn in the sampling process. Therefore, Markov chain Monte Carlo can also be used for optimization purposes. To design a Markov chain with stationary distribution $\pi(x)$, the maximum point in finite sampling from distribution $\pi(x)$ will be sufficiently close to the maximum point of $f(x)$ in the feasible region. When the transition kernel is symmetric is its arguments, or $q(y, x_t) = q(x_t, y)$, then the acceptance rate $\alpha(x_t, y)$ become

$$\alpha(x_t, y) = \min\left\{1, \frac{\pi(y)q(y, x_t)}{\pi(x_t)q(x_t, y)}\right\} = \min\left\{1, \frac{\pi(y)}{\pi(x_t)}\right\}$$

The proposed Markov chain sampling for optimization search algorithm is:

(1) Start with x_0, at $t = 0$, $x_0^* = x_0$

(2) Propose a new solution y

(3) Drawn u from the uniform distribution $U(0,1)$

(4) Compute $\alpha(x_t, y) = \min\left\{1, \frac{\pi(y)}{\pi(x_t)}\right\}$

(5) Take $x_{t+1} = \begin{cases} y & u \le \alpha \quad \text{(with probability } \alpha) \\ x_t & u > \alpha \quad \text{(with probability } 1-\alpha) \end{cases}$

(6) Take $x_{t+1}^* = \begin{cases} x_t^* & f(x_t^*) > f(x_{t+1}) \\ x_{t+1} & f(x_t^*) \le f(x_{t+1}) \end{cases}$ . . . . . .

Repeat (2) to (6). If the iteration times are large enough, then x_t^* will convergence to the maximum point of the objective function $f(x)$ in distribution. We can see from the problem analysis above that the key points of Markov chain sampling method are designing of general probability density function $\pi(x)$ and uniform sampling from conditional constraint region.

In order to solve an optimization problem, we can search the solution by performing a random walk starting from a good initial but random guess solution. However, to be computationally efficient and effective in searching for new solutions, we can keep the best solutions found so far, and to increase the mobility of the random walk so as to explore the search space more effectively. We can find a way to control the walk in such a way that it can move towards the optimal solutions more quickly, rather than wander away from the potential best solutions. These are the challenges for the most meta-heuristic algorithms. The same issues are also important for Monte Carlo simulations and Markov chain sampling techniques. An important link between Markov chain and optimization is that some heuristic or meta-heuristic search algorithms such as simulated annealing use a trajectory-based approach. They start with some initial random solution, and propose a new solution randomly. Then the move is accepted or not, depending on some probability. It is similar to a Markov chain. In fact, the standard simulated annealing is a random walk. Simulated annealing is a probabilistic method for finding global minimum of some cost function introduced by Kirkpatrick et al. [15]. It searches local minimum, and finally stays at the global minimum given enough time. This sampling method was originally extended from Metropolis Algorithm [16] by implanting a temperature function *T*. *T* is used to control the difficulty for the stochastic sampler to escape from a local minimum and reach the global optimal for a non-optimal state. Algorithms such as simulated annealing which use a single Markov chain may not be very efficient. In practice, it is usually advantageous to use multiple Markov chains in parallel to increase the overall efficiency. In fact, the algorithms such as particle swarm optimization can be viewed as multiple interacting Markov chains, though such theoretical analysis remains almost intractable. The theory of interacting Markov chains is complicated and yet still under development. However, any progress in such areas will play a central role in the understanding how population- and trajectory-based meta-heuristic algorithms perform under various conditions.

In addition, a Markov chain is said to be ergodic or irreducible if it is possible to go from every state to every state. Furthermore, the use of a uniform distribution is not the only way to achieve randomization. In fact, random walks such as Levy flights on a global scale are more efficient. On the other hand, the track of chaotic variable can travel ergodically over the whole search space. In general, the chaotic variable has special characters, i.e., ergodicity, pseudo-randomness and irregularity. To enrich the searching behavior and to avoid being trapped into local optimum, chaotic sequence and a chaotic Levy flight can be incorporated in the meta-heuristic search for efficiently generating new solutions. In the paper

[17], we presented synergistic strategies for meta-heuristic optimization learning, with an emphasis on the balance between intensification and diversification. We showed some promising efficiency for global optimization. Interestingly, it can be viewed to link with optimization search and Markov chain sampling under appropriate conditions.

VII. SIMULATION RESULTS

For the numerical experiment, we consider a nonlinear oscillator, whose dynamics are described by the Duffing equation [12]:

$$\dot{x} + px + p_1 x + x^3 = q\cos(\omega t) \qquad (21)$$

where $p = 0.4$, $p_1 = -1.1$, $\omega = 1.8$.

This equation can be used to model various physical phenomena. By varying the forcing amplitude q, the system will undergo several types of bifurcations, ranging from a regular periodic evolution to chaos. The evolutions of two chaotic Duffing oscillators, when starting from different initial conditions $(x(0),\dot{x}(0)) = (x_1(0), x_2(0)) = (0,-2)$, $(y(0),\dot{y}(0)) = (y_1(0), y_2(0)) = (1,0)$ and different parameter values q for time period $t_0 = 0$ to $t_f = 50$, are shown in Figure 2. As shown, with different parameters $q = q_x = 2.10$ and $q = q_y = 2.00$, the solutions of (21) displays chaotic responses. In order to synchronize the two chaotic systems, we apply our adaptive controller for synchronization of the drive (x) and response (y) systems. The system that we are investigating is given by the following system of equations:

$$\begin{aligned} &\dot{x} + px + p_1 x + x^3 = q\cos(\omega t) \\ &\dot{y} + py + p_1 y + y^3 = q_y\cos(\omega t) + k(x-y) \qquad (22) \\ &\dot{q}_y = G(\bullet) \end{aligned}$$

The function G acts on the Monte Carlo optimization as presented in this paper for adaptive synchronization. Our simulation results show that the adaptive controller produces the desired synchronization after a relatively short transient period. Figure 2 shows the error trajectories and the parameter q. In Figure 3 (f) we show the convergence of the perturbed parameter q_y to its desired value q_x from which the desired trajectory was constructed, when the Duffing oscillator is controlled to drive its trajectory to a desired chaotic orbit. The convergence of the system variables $(y,\dot{y}) = (y_1, y_2)$ to the desired chaotic orbit $(x,\dot{x}) = (x_1, x_2)$ are also shown in Figure 3 (a) and (b), respectively. The dynamics of the two component of the error system are shown in Figure 3 (d) and (e).

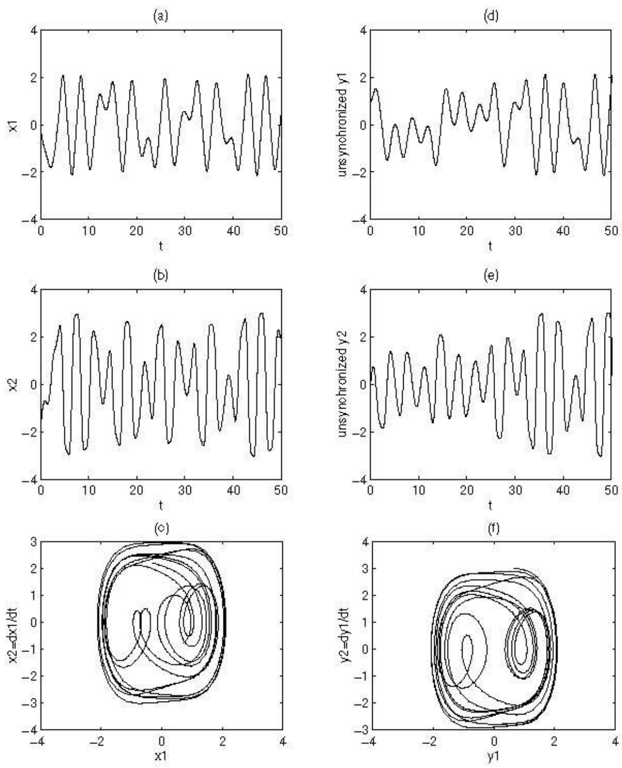

Figure 2. Chaotic orbits of the Duffing equation: (a)(b)(c) q = 2.10, (d)(e)(f) q = 2.00

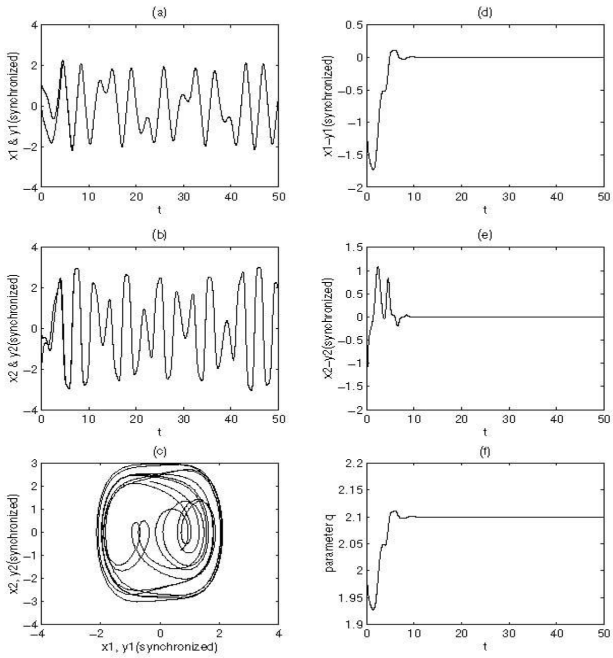

Figure 3. Synchronization between two chaotic systems described by Duffing equations

VIII. Conclusion

Markov chain Monte Carlo is a family of simulation methods, which generate samples of Markov Chain processes. In this paper, we set up a framework of Markov chain sampling for optimization search. We have shown how the proposed Monte Carlo optimization search algorithm can be used to adapt the parameters in two coupled systems such that the two systems are synchronized. Simulation results have been presented to show the effectiveness of the approach. We have also studied the possibility to reconstruct a chaotic system based on the concept of chaos synchronization.

References

[1] Tao Yang and Leon O. Chua. Secure communication via chaotic parameter modulation. *IEEE Trans. Circuits and Systems - I: Fundamental Theory and Applications*, 43(9):817–819, 1996.

[2] K. M. Cuomo and A. V. Oppenheim. Chaotic signals and systems for communications. In *1993 IEEE ICASSP III*, pages 137–140, 1993.

[3] R. C. Elson, A. I. Selverston, and R. Huerta. Synchronous behavior of two coupled biological neurons. *Physical Review Letters*, 81(25):5692–5695, 1998.

[4] J. M. Kowalski, G. L. Albert, B. K. Rhoades, and G. W. Gross. Neuronal networks with spontaneous, correlated bursting activity: theory and simulations. *Neural Networks*, 5:805–822, 1992.

[5] D. Hansel and H. Sompolinsky. Synchronization and computation in a chaotic neural network. *Physical Review Letters*, 68(5):718–721, 1992.

[6] I. I. Blekhman, A. L. Fradkov, H. Nijmeijer, and A. Yu. Pogromsky. On synchronization and controlled synchronization. *Systems & Control Letters*, 31:299–305, 1997.

[7] M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice-Hall, Englewood Cliffs, 1993.

[8] B. Friedland. *Advanced Control System Design*. Prentice-Hall, Englewood Cliffs, 1996.

[9] T. Kailath. *Linear Systems*. Prentice-Hall, Englewood Cliffs, 1980.

[10] G. Ciccarella, M. Dalla Mora, and A. Germani. A Luenberger-like observer for nonlinear systems. *International Journal of Control*, 57:537–556, 1993.

[11] W. M. Wonham. *Linear Multivariable Control: A Geometric Approach*. Springer Verlag, New York, 1985.

[12] T. Kapitaniak. *Chaotic Oscillations in Mechanical Systems*. Manchester University Press, 1991.

[13] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," Biometrika, vol.57, 1970, pp. 97-109.

[14] S. Geman, and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 6, no. 6, 1984, pp. 721-741.

[15] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. science, 220(4598):671, 1983.

[16] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, et al. Equation of state calculations by fast computing machines. The journal of chemical physics, 21(6):1087, 1953.

[17] Jiann-Horng Lin, "Synergistic Strategies for Meta-heuristic Optimization Learning Algorithms", International Journal of Engineering & Technical Research , vol. 4, no. 4, pp. 89-97, 2016.04.

Jiann-Horng Lin received his B. S. and M. S. both in Computer Science and Information Engineering from Feng-Chia University, Taiwan in 1987 and 1989, respectively. He then received his Ph.D. in Electrical Engineering and Computer Science from Syracuse University, New York in 1999. He is currently an assistant professor at the Department of Information Management at I-Shou University, Taiwan. He is also the department chair from 2004 to 2007. His research interests include artificial intelligence, data mining, chaos and information theory. He is also interested in the area of evolutionary computation and bioinformatics. Dr. Lin is a member of the IEEE, the Association for Computing Machinery and the Institute of Information and Computing Machinery.