



## CONTROLLING CHAOS BY AN IMPROVED ESTIMATION OF DISTRIBUTION ALGORITHM

Xingli Huang<sup>1</sup>, Peifa Jia<sup>1</sup>, Bo Liu<sup>2</sup>

<sup>1</sup> Department of Computer Science & Technology  
Tsinghua University, 100084, Beijing, P. R. China

<sup>2</sup> Center for Forecasting Science, Academy of Mathematics and Systems Science  
Chinese Academy of Sciences, 100190, Beijing, P. R. China  
liub01@mails.tsinghua.edu.cn

**Abstract-** Control and synchronization of chaotic systems are important issues in nonlinear sciences. This paper proposes an effective estimation of distribution algorithm (EDA)-based memetic algorithm (MA) to direct the orbits of discrete chaotic dynamical systems as well as to synchronize chaotic systems, which could be formulated as complex multi-modal numerical optimization problems. In EDA-based MA (EDAMA), both EDA-based searching operators and simulated annealing (SA)-based local searching operators are designed to balance the exploration and exploitation abilities. On the other hand, global information provided by EDA is combined with local information from SA to create better solutions. In particular, to enrich the searching behaviors and to avoid premature convergence, SA-based local search is designed and incorporated into EDAMA. To balance the exploration and exploitation abilities, after the standard EDA-based searching operation, SA-based local search is probabilistically applied to some good solutions selected by using a roulette wheel mechanism with a specified probability. Numerical simulations based on Hénon Map demonstrate the effectiveness and efficiency of EDAMA, and the effects of some parameters are investigated as well.

**Keywords-** Estimation of Distribution Algorithm, Chaotic dynamics

### 1. INTRODUCTION

Since the pioneering work of Huber's on chaos control in 1989 [1] and work of Pecora and Carroll's on chaos synchronization [2] in 1990, control and synchronization of chaotic systems have been important issues in nonlinear science and attracting increased interests from various fields [3]. Nowadays, a wide variety of approaches have been proposed for control and synchronization of chaotic systems [4-8].

Recently, a new population-based evolutionary technique, estimation of distribution algorithm (EDA), has been proposed [9] as an alternative to genetic algorithm (GA) [10] and particle swarm optimization (PSO) [11] for continuous or discrete optimization problems. In EDA, a population of solutions is initialized randomly, which is evolved to find optimal solutions through selection, modeling, sampling, and replacement operation procedures. Compared with GA and PSO, EDA has some attractive characteristics. It samples new solutions from a probability model which approximates the distribution of promising solutions, and avoids premature

convergence and selection pressure exhibited by GA's crossover and mutation operation. Additionally, the priori information about the problem structure can be captured by the probability model estimated during the search, which facilitating a more efficient global search. This paper proposes an effective estimation of distribution algorithm (EDA) – based memetic algorithm (MA) to direct the orbits of chaotic dynamical systems as well as to synchronize chaotic systems, which could be formulated as multimodal numerical optimization problems. Simulations based on Hénon Map demonstrate the effectiveness and efficiency of EDAMA, and the effects of some parameters are investigated as well.

## 2. PROBLEM FORMULATION

Consider the following discrete chaotic dynamical system:

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k)), \quad k = 1, 2, \dots, N \tag{1}$$

where state  $\mathbf{x}(k) \in R^n$ , and  $\mathbf{f} : R^n \rightarrow R^n$  is continuously differentiable.

Let  $\mathbf{x}_0 \in R^n$  be an initial state of the system. If small perturbation  $\mathbf{u}(k) \in R^n$  is added to the chaotic system, then

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k)) + \mathbf{u}(k), \quad k = 0, 1, \dots, N-1 \tag{2}$$

where  $\|\mathbf{u}(k)\| \leq \mu$ , and  $\mu$  is a positive real constant.

The goal is to determine suitable  $\mathbf{u}(k)$  so as to make  $\mathbf{x}(N)$  in the  $\varepsilon$ -neighborhood of the target  $\mathbf{x}_t$ , i.e.,  $\|\mathbf{x}(N) - \mathbf{x}_t\| < \varepsilon$ , where a local controller is effective for chaos control. Without loss of generality, assume that  $\mathbf{u}(k)$  acts only on the first component of  $\mathbf{f}$ , then the problem can be re-formulated as follows.

(P1):

$$\min \|\mathbf{x}(N) - \mathbf{x}_t\| \text{ by choosing suitable } u(k), \quad k = 0, 1, \dots, N-1 \tag{3a}$$

$$s.t. \begin{cases} x_1(k+1) = f_1(\mathbf{x}(k)) + u(k) \\ x_i(k+1) = f_i(\mathbf{x}(k)), i = 2, \dots, n \end{cases}, \quad |u(k)| \leq \mu, \quad \mathbf{x}(0) = \mathbf{x}_0 \tag{3b}$$

Let  $\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k))$  and  $\mathbf{y}(k+1) = \mathbf{f}(\mathbf{y}(k))$  be two given chaotic dynamical systems with the same structure but different initial state. Chaotic synchronization by feedback is to select the feedback matrix  $\mathbf{K}(k) \in R^{n \times n}$  such that  $\|\mathbf{x}(N) - \mathbf{y}(N)\| \rightarrow 0$ , where  $\mathbf{x}(0) = \mathbf{x}_0 \neq \mathbf{y}(0) = \mathbf{y}_0$  and

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k)) \\ \mathbf{y}(k+1) = \mathbf{f}(\mathbf{y}(k)) + \mathbf{K}(k) \cdot (\mathbf{y}(k) - \mathbf{x}(k)) \end{cases} \tag{4}$$

It was regarded that chaos synchronization also can be formulated as the above problem [4]. Similarly, assume that feedback only acts on the first component, i.e.,  $K_{11}(k) \neq 0$  and all other components of  $\mathbf{K}(k)$  are zeros. For convenience, denote  $K_{11}(k) = K(k)$ . Then the problem can be formulated as follows.

(P2):

$$\min \|\mathbf{x}(N) - \mathbf{y}(N)\| \text{ by choosing suitable } K(k), \quad k = 0, 1, \dots, N-1 \tag{5a}$$

$$s.t. \begin{cases} \mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k)) \\ y_1(k+1) = f_1(\mathbf{y}(k)) + K(k) \cdot (y_1(k) - x_1(k)), \quad |K(k)| \leq \kappa, \quad \mathbf{x}(0) = \mathbf{x}_0 \neq \mathbf{y}(0) = \mathbf{y}_0 \\ y_i(k+1) = f_i(\mathbf{y}(k)), i = 2, \dots, n \end{cases} \tag{5b}$$

However, (P2) does not totally equivalent to the origin problem, because

$\mathbf{x}(N) \neq \mathbf{y}(N)$  will cause the two systems apart from each other eventually due to the sensitivity to initial state of chaotic systems, even though  $\|\mathbf{x}(N) - \mathbf{y}(N)\|$  is arbitrarily small. So we should deal with (P2) on-line at some step  $k'$  with  $\mathbf{x}_0 = \mathbf{x}(k')$ ,  $\mathbf{y}_0 = \mathbf{y}(k')$ , when  $\|\mathbf{x}(k') - \mathbf{y}(k')\|$  is larger than a threshold  $\delta$ . For the above problems, it needs to determine suitable  $u(k)$  or  $K(k)$  to minimize objective function value. Thus, from the viewpoint of optimization, (P1) and (P2) are both multi-dimensional constrained numerical optimization problems.

### 3. EDAMA

#### 3.1. Estimation of distribution algorithm (EDA)

In EDA, the  $i$ -th individual in the  $d$ -dimensional search space at generation  $t$  can be represented as  $X_i(t) = [x_{i,1}^t, x_{i,2}^t, \dots, x_{i,d}^t]$ , ( $i = 1, 2, \dots, NP$ , where  $NP$  denotes the size of the population). The procedure of EDA is summarized as follows.

Step 1: Randomly initialize the population of individual for EDA, where each individual contains  $d$  variables (i.e.,  $d = N$ ). Repair individuals if required.

Step 2: Evaluate the objective values of all individuals, and recode the best individual  $X_{best}$  together with its objective value.

Step 3: Select  $M$  individuals from population based on the selection strategy.

*Remark:* In this case, truncation selection strategy is employed by which individuals are sorted according to their objective values and the best  $M$  ( $M = NP/2$ ) individuals are selected.

Step 4: Build a probabilistic model based on the statistical information of the selected individuals in Step 3 using a learning method.

*Remark:* Gaussian model with diagonal covariance matrix (GM/DCM) [9] is used, in which the  $d$ -dimensional joint probability distribution is factorized as a product of  $d$  univariate and independent normal distributions:

$$p_t(x) = \prod_{j=1}^d N(x_j; \mu_j^t, \sigma_j^t), \quad N(x_j; \mu_j^t, \sigma_j^t) = \frac{1}{\sqrt{2\pi\sigma_j^t}} e^{-\frac{1}{2}\left(\frac{x_j - \mu_j^t}{\sigma_j^t}\right)^2} \quad (6)$$

In Eq. (6), the mean  $\mu_j^t$  and standard deviation  $\sigma_j^t$  of the  $j$ -dimension variable for the  $t$ -th generation can be estimated respectively as follows:

$$\hat{\mu}_j^t = \frac{1}{M} \sum_{i=1}^M x_{i,j}^t, \quad \hat{\sigma}_j^t = \sqrt{\frac{1}{M} \sum_{i=1}^M (x_{i,j}^t - \hat{\mu}_j^t)^2} \quad (7)$$

Step 5: Sample  $NP$  new offspring individuals from the model built in Step 4. Repair individuals if required.

Step 6: Evaluate the objective values of the offspring individuals.

Step 7: Update population using replacement strategy.

*Remark:* As for the replacement strategy, steady-state replace is used, by which the pool of the parents and offspring are ranked based on their objective values, and top individuals are selected for next generation.

Step 8: Determine the best individual of the current new population with the best objective value. If the objective value is better than the objective value of  $X_{best}$ , then update  $X_{best}$  and its objective value with the value and objective value of the current

best individual.

Step 9: If a stopping criterion is met, then output  $X_{best}$  and its objective value; otherwise go back to Step (3).

As for the problems to direct chaotic orbit and to synchronize chaotic systems, the searching solutions are  $(u(0), u(1), \dots, u(N-1))^T$  and  $(K(0), K(1), \dots, K(N-1))^T$  respectively. Obviously, multiple variables need to be determined if  $N$  is large. Besides, in Steps 1 and 5, if newly generated  $u$  or  $K$  is exceeded its bound (i.e.  $\pm\mu$  or  $\pm\kappa$ ), it will be set as its nearest bound value based on the repairing strategy.

### 3.2. EDAMA

Starting from an initial state, simulated annealing (SA) randomly generates a new state in the neighborhood of the original one, which causes a change of  $\Delta E$  in the objective function value. For minimization problems, the new state is accepted with probability  $\min\{1, \exp(-\Delta E/T)\}$ , where  $T$  is a control parameter. SA provides a mechanism to probabilistically escape from local optima and the search process can be controlled by the cooling schedule [12].

In EDAMA, SA-based local search is inserted after Step 7 in EDA to enrich the local searching behaviors and to avoid premature convergence. Based on its performances, each solution is assigned a probability to be selected by the rank-based fitness assignment technique [10]. Then, the roulette wheel mechanism [10] is used to decide which solutions will be selected. Subsequently, the selected solutions will perform the SA-based local search with a predefined probability  $p_{ls}$ . Due to the mechanism of roulette wheel rule, good solution will gain more chance for exploitation. Besides, it is easy to control such an exploitation process by adjusting the value of  $p_{ls}$ . Moreover, for the parameter settings of SA, proper initial temperature should be high enough so that all states of the system have an equal probability of being visited and at the same time it should not be rather high so that a lot of unnecessary searches in high temperature will be avoided. In EDAMA, an initial temperature is set as  $t_0 = -(c_w - c_b) / \ln p_r$ , where  $c_w$  and  $c_b$  denote the worst and best objective values in the initial population, respectively;  $p_r$  denotes the acceptance probability of the worst solution compared with the best one. Exponential cooling schedule,  $t_k = \lambda t_{k-1}$ ,  $0 < \lambda < 1$ , is applied, which is believed to be an excellent cooling recipe, since it provides a rather good compromise between a computationally fast schedule and the ability to reach low-energy state.

## 4. SIMULATION

### 4.1. Simulation on directing chaotic orbits

As a typical discrete chaotic system, Hénon Map is described as follows.

$$\begin{cases} x_1(k+1) = -px_1^2(k) + x_2(k) + 1 \\ x_2(k+1) = qx_1(k) \end{cases} \quad (8)$$

where  $p=1.4$ , and  $q=0.3$ .

Consider (P1) first, the target  $\mathbf{x}_r$  is set to be a fixed point  $(0.63135, 0.18941)^T$ .

Let  $\mathbf{x}_0 = (0, 0)^T$ , and  $u(k)$  is only added to  $x_1$  with the bound  $\mu = 0.01$ . To test the performance of EDAMA, EDAMA is compared with standard EDA and SA. In EDAMA, the population size is 100, the maximum generation is 200,  $p_{ls} = 0.1$ ,  $p_r = 0.05$ , and cooling rate  $\lambda = 0.9$ . In EDA, the population size is 100, and the maximum generation is 200. In SA,  $p_r = 0.05$ ,  $\lambda = 0.9$ , the step of Metropolis sampling process under each temperature is set to 1000, and the maximum generation is set to 20. Under different values of  $N$ , Table 1 lists the mean objective value and the best objective value of 100 independent runs for each of the above three algorithms. From Table 1, it can be seen that EDAMA is superior to EDA and SA in term of searching quality and derivation of the results. It could be concluded that EDAMA is more effective and more robust on initial conditions.

**Table 1 Statistics performance of EDAMA, EDA and SA under different  $N$**

$N$		EDAMA	EDA	SA
$N=6$	Mean	0.01913	0.09784	0.03922
	Best	0	0.09405	0
	Standard Deviation	0.04033	0.0028	0.0537
$N=7$	Mean	0.00281	0.02091	0.00585
	Best	0	0.01385	0
	Standard Deviation	0.00593	0.00766	0.008
$N=8$	Mean	0.00082	0.02838	0.00066
	Best	0	0.00124	0
	Standard Deviation	0.00173	0.03057	0.00091
$N=9$	Mean	0.00007	0.05913	0.00002
	Best	0	0.00078	0
	Standard Deviation	0.00015	0.05706	0.00002
$N=10$	Mean	0	0.06831	0.00005
	Best	0	0.00138	0
	Standard Deviation	0	0.03425	0.00006

#### 4.2. Simulation on synchronization of chaotic systems

Next, we consider chaos synchronization problem based on EDAMA. The driven system is Eq. (8) with  $\mathbf{x}(0) = (0.2, 0.3)^T$ , and the response system is as follows.

$$\begin{cases} y_1(k+1) = -py_1^2(k) + y_2 + 1 + K(k) \cdot (y_1(k) - x_1(k)) \\ y_2(k+1) = qy_1(k) \end{cases} \quad (10)$$

where  $\mathbf{y}(0) = (0.8, 0.5)^T$ .

Chaos synchronization can be formulated as (P2). We set  $N = 5$ ,  $\kappa = 1$  and  $\delta = 0.03$ . One of the typical results is showed in Fig. 1. Fig.1 (a) is the error of the first component between two systems by dealing with (P2) online; (b) is the corresponding value of  $K$ ; (c) illustrates the bias of  $x_1 - y_1$  without dealing with (P2) online. It can be seen from Fig. 1 (a) that the response system synchronizes the driven system very well, but without dealing with (P2) online the two systems get apart from each other only

after a few steps as shown in Fig. 1 (c). In a word, it is concluded that EDAMA can effectively and efficiently solve the problem synchronizing chaotic dynamical systems.

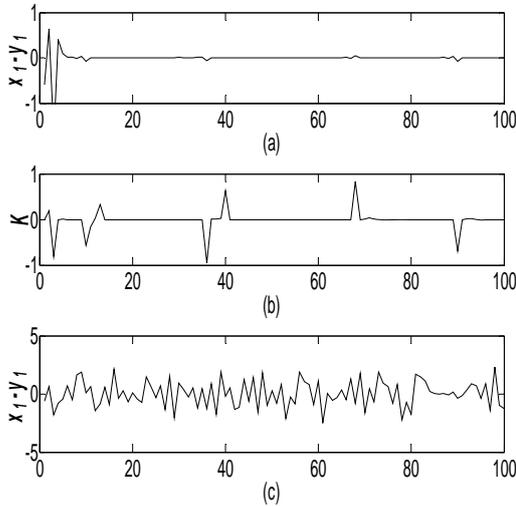


Fig.1 Chaos synchronization

## 5. CONCLUSION

To our knowledge, this is the first report of hybridizing estimation of distribution algorithm and simulated annealing for chaos control and chaos synchronization. The proposed approach not only performs exploration by using the population-based evolutionary searching ability of EDA, but also performs exploitation by using the SA-based local searching behavior. Simulation results based on Hénon Map demonstrated the effectiveness and efficiency of EDAMA.

## 6. REFERENCES

1. A. W. Hubble, Adaptive control of chaotic system, *Helv Phys Acta* **62**, 343-346, 1989.
2. L. Pecora and T. Carroll, Synchronization in chaotic systems, *Phys Rev Lett* **64**, 821-824, 1990.
3. G. Chen and X. Dong, *From Chaos to Order: Methodologies, Perspectives, and Applications*, World Scientific, Singapore, 1998.
4. J. P. Su and C. C. Wang, Synchronizing cascade-connected chaotic systems with uncertainties and breaking chaotic cryptosystems via a novel adaptive control scheme, *Int J Bifurcat Chaos* **15**, 2457-2468, 2005.
5. E. Ott, C. Grebogi and J. A. Yorke, Controlling chaos, *Phys Rev Lett* **64**, 1196-1199, 1990.
6. C. C. Fuh and P. C. Tung, Controlling chaos using differential geometric method, *Phys Rev Lett* **75**, 2952-2955, 1995.
7. X. Zhong, S. Shao and J. Fang, Directing orbits of chaotic dynamical systems using genetic algorithms, *Control Decis* **13**, 165-168, 1998.
8. B. Liu, L. Wang, Y. H. Jin, et al., Directing orbits of chaotic systems by particle swarm optimization, *Chaos, Solitons & Fractals* **29**, 454-461, 2006.
9. P. Larrañaga and J. A. Lozano, *Estimation of Distribution Algorithms: A New Tool for Evolutionary Computation*, Kluwer Academic Publishers, 2002.
10. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, 1989.
11. L. Wang and B. Liu, *Particle Swarm Optimization and Scheduling Algorithms*, Tsinghua University Press, Beijing, 2008.
12. S. Kirkpatrick, C. D. Gelat and M. P. Vecchi, Optimization by simulated annealing, *Science* **220**, 671-680, 1983.