Available online at www.sciencedirect.com
ScienceDirect
Chaos, Solitons and Fractals 34 (2007) 1366–1375
CHAOS
SOLITONS & FRACTALS
www.elsevier.com/locate/chaos

On the efficiency of chaos optimization algorithms for global optimization

Dixiong Yang *, Gang Li, Gengdong Cheng

Department of Engineering Mechanics, State Key Laboratory of Structural Analysis of Industrial Equipment, Dalian University of Technology, Dalian 116023, China

Accepted 27 April 2006

Accepted by Professor G. Iovane

Abstract

Chaos optimization algorithms as a novel method of global optimization have attracted much attention, which were all based on Logistic map. However, we have noticed that the probability density function of the chaotic sequences derived from Logistic map is a Chebyshev-type one, which may affect the global searching capacity and computational efficiency of chaos optimization algorithms considerably. Considering the statistical property of the chaotic sequences of Logistic map and Kent map, the improved hybrid chaos-BFGS optimization algorithm and the Kent map based hybrid chaos-BFGS algorithm are proposed. Five typical nonlinear functions with multimodal characteristic are tested to compare the performance of five hybrid optimization algorithms, which are the conventional Logistic map based chaos-BFGS algorithm, improved Logistic map based chaos-BFGS algorithm, Kent map based chaos-BFGS algorithm, Monte Carlo-BFGS algorithm, mesh-BFGS algorithm. The computational performance of the five algorithms is compared, and the numerical results make us question the high efficiency of the chaos optimization algorithms claimed in some references. It is concluded that the efficiency of the hybrid optimization algorithms is influenced by the statistical property of chaotic/stochastic sequences generated from chaotic/stochastic algorithms, and the location of the global optimum of nonlinear functions. In addition, it is inappropriate to advocate the high efficiency of the global optimization algorithms only depending on several numerical examples of low-dimensional functions.

1. Introduction

Many problems in various kinds of fields can be formulated as a typical optimization problem, and it is important to search for a global optimum for most of them. However, a global optimum is not characterized by any mathematical conditions, unlike a local optimum that is characterized by the local behavior of the problem functions, such as the gradients and Hessians of functions. Therefore, the global optimization problem remains a challenge from mathemat-

* Corresponding author. Tel.: +86 411 84748226; fax: +86 411 84708400.
E-mail address: yangdx@dlut.edu.cn (D. Yang).

0960-0779/$ - see front matter
doi:10.1016/j.chaos.2006.04.057

ical as well as computational viewpoint [1]. The methods for global optimization can be divided into two broad categories: deterministic and stochastic. The deterministic method includes covering methods, methods of generalized descent, zooming method, tunneling method, etc.; and the stochastic method includes basic stochastic global optimization methods such as pure random search i.e. Monte Carlo method and multi-start method, clustering methods, controlled random search, acceptance–rejection based optimization methods, tabu search, simulated annealing, genetic algorithm and so on. A number of papers identified and discussed the main characteristics of some methods from the above two categories [1,2].

In recent years, the theories and applications of nonlinear dynamics, especially of chaos, have drawn more and more attention in many fields. One is chaos controlling [3] and synchronization [4]. Another field is the potential applications of chaos in various disciplines including optimization. At present, there are three groups working on the research of chaotic optimization. The first group aimed at global optimization by chaotic neural network proposed by Aihara et al. [5], which mainly used the transiently chaotic neural network with chaotic annealing developed to find globally optimal solutions for combinatorial problems, such as traveling salesman problem and maintenance scheduling problem [6–10]. The second group searched the global optimum by the chaos optimization algorithm, which utilized the nature of chaos sequences such as pseudo-randomness, ergodicity and irregularity [11,12]. The third group utilized computational instabilities to solve non-linear equations and optimization problems that resulted in the development of a new method, chaotic descent, which is based on descending to the global minima via regions that are the source of computational chaos [13,14]. In this paper we focus on the clarification of computational efficiency and performance of chaos optimization algorithms for global optimization of nonlinear functions, as belonged to the second group.

The basic process of chaos optimization algorithm [11] generally includes two major steps. Firstly, define a chaotic sequences generator based on the Logistic map. Generate a sequence of chaotic points and map it to a sequence of design points in the original design space. Then, calculate the objective functions with respect to the generated design points, and choose the point with the minimum objective function as the current optimum. Secondly, the current optimum is assumed to be close to the global optimum after certain iterations, and it is viewed as the center with a little chaotic perturbation, and the global optimum is obtained through fine search. Repeat the above two steps until some specified convergence criterion is satisfied, then the global optimum is obtained. However, further numerical simulation [15] showed that the method is effective only in small design space. Therefore, mutative scale chaos optimization algorithm was presented in which the fine search strategy with mutative scale is introduced [15]. Due to the pseudo-randomness of chaotic motion, the motion step of chaotic variables between two successive iterations is always big, which resulted in the big jump of the design variables in design space. Thus, even if the above chaos optimization algorithms have reached the neighborhood of the optimum, it needs to spend much computational effort to approach the global optimum eventually by searching numerous points. Hence, the hybrid methods attracted the attention of some researchers, in which the chaos optimization algorithm was used for global search and the conventional optimization algorithms were employed for local search near the global optimum. The hybrid methods can save much CPU time and enhance the computational efficiency of algorithms. In Refs. [16–19], the chaos optimization algorithm is combined with the regularization algorithm, sequential linear programming (SLP) and genetic algorithm to solve various global optimizations for heat conductivity of geotechnical engineering, power flow of electrical engineering, nonlinear multi-modal function, and operating conditions of aromatic hydrocarbon isomerization process of chemical engineering, respectively. Recently, Sun et al. [20] adopted chaos search algorithm to implement the node repositioning for tetrahedral mesh in the computer-aided design. Lu et al. [21–23] utilized chaos optimization algorithm to search and construct optimal sliding mode for tracking control of the Chen's chaotic system. Ji and Tang [24] and Liu et al. [25] suggested a hybrid method of simulated annealing and particle swarm optimization combined with chaos search, and examined its efficiency with several nonlinear functions, respectively.

These chaos optimization algorithms in the referenced papers were all based on the Logistic map. We noticed that the probability density function of the chaotic sequences for Logistic map is a Chebyshev-type one with very high density near the two ends of the chaotic variable interval $(0,1)$ and low density at middle part, not uniformly distributed at all. This type of density distribution may affect the global searching capacity and computational efficiency of chaos algorithms remarkably. Considering this fact, the improved hybrid chaos-BFGS (BFGS is a quasi-Newton method for function optimization developed by Broyden et al. in 1970 [28]) optimization algorithm was proposed in our study by cutting the parts near the two ends of chaos variable interval $(0,1)$ during the chaos searching. Since the probability density function of chaotic sequences of Kent map is the uniform function in $(0,1)$, the Kent map based hybrid chaos optimization-BFGS algorithm was also studied herein. In order to get the deep understanding of the efficiency of chaos optimization algorithms presented in some references, five nonlinear functions (three 2- or 3-dimensional and two 5-dimensional functions) were evaluated to compare the numerical performance of five hybrid optimization algorithms, which are the improved Logistic map based chaos-BFGS algorithm, Kent map based chaos-BFGS algorithm, conventional Logistic map based chaos-BFGS algorithm, Monte Carlo-BFGS algorithm, and mesh-BFGS algorithm. The remainder of this paper is organized as follows. In Section 2, we introduce the hybrid chaos-BFGS algorithms based on Logistic map and improved Logistic map as well as Kent map. In Section 3, the hybrid Monte Carlo-BFGS and mesh-BFGS algorithms are described. Then, five nonlinear functions are tested to examine the computational efficiency and performance of the five algorithms in Section 4. Some factors affecting their performance are analyzed and discussed in detail. Finally, Section 5 is devoted to some conclusions and our opinions about the chaos optimization algorithms.

2. Hybrid chaos-BFGS optimization algorithm

Chaos often exists in nonlinear systems. It is a kind of highly unstable motion of deterministic systems in finite phase space. Chaos theory is typically described as the so-called 'butterfly effect', detailed by Lorenz [26]. Attempting to simulate a global weather system numerically, Lorenz discovered that minute changes in initial conditions steered subsequent simulations towards radically different final results. This sensitive dependence on initial conditions is generally exhibited by the systems containing multiple elements with nonlinear interactions, particularly when the system is forced and dissipative. Sensitive dependence on initial conditions is not only observed in complex systems, but even in the simplest Logistic map discussed by May [27]. The well-known Logistic map is written as

$$z^{k+1} = f(\mu, z^k) = \mu z^k(1 - z^k) \tag{1}$$

in which μ is a control parameter, $k = 0, 1, 2, \ldots$, and z is a variable. Suppose $0 < z^0 < 1$, $0 \leqslant \mu \leqslant 4$. It is easy to find that Eq. (1) is a deterministic dynamic system without any stochastic disturbance. It seems that its long-time behavior can be predicted. But that is not true. The behavior of system Eq. (1) is greatly changed with the variation of μ. The value of the control parameter μ determines whether z stabilizes at a constant value, or oscillates between a limited sequence of values, or behaves chaotically in an unpredictable pattern. In the latter case, i.e. $\mu = 4$, the equation is changed to

$$z^{k+1} = 4z^k(1 - z^k) \tag{2}$$

And, very small change in the initial value of z will cause large difference in its long-term behavior, which is just the typical characteristic of chaos. The variable z is referred to chaotic variable. Thus, a minute change of the initial value of the chaotic variable would give rise to a considerable difference of the values of chaotic variable later. The track of chaotic variable can move over the whole space of interest ergodically. The variation of the chaotic variable has a delicately inherent rule in spite of the fact that it looks like in disorder. In general, there are three main characteristics of the variation of the chaotic variable, i.e. pseudo-randomness, ergodicity and irregularity. By using these properties, the chaos optimization algorithm is proposed and developed to solve various kinds of global optimization problem [16–25].

Consider the optimization problem for nonlinear multimodal function with boundary constraints:

$$\text{Minimize} \quad f(\mathbf{x}) = f(x_1, x_2, \ldots, x_n) \tag{3a}$$

$$\text{Subject to} \quad XL_i \leqslant x_i \leqslant XU_i, \quad i = 1, 2, \ldots, n \tag{3b}$$

where f denotes objective function with respect to n design variables x_i ($i = 1, 2, \ldots, n$) with the assumption of continuously twice differentiability. XL_i and XU_i are lower and upper bounds for variable x_i, and the n-dimensional variable space S is defined in the region $\{(XL_i, XU_i), i = 1, 2, \ldots, n\}$.

The chaotic variables of the chaos optimization algorithms used in the referenced papers were generated by the Logistic map, which is expressed as Eq. (1). As mentioned above, for $\mu = 4$, Logistic map is in chaotic state, and the interval $(0, 1)$ is chaotic invariable set of the map. In order to generate the design variable by the Logistic map, a linear mapping between the design variable x_i and chaos variable z_i is defined, namely,

$$z_i = (x_i - XL_i)/(XU_i - XL_i) \tag{4}$$

Then, the n chaotic variables z_i ($i = 1, 2, \ldots, n$) with different initial value derived from Logistic map can produce a design vector $\boldsymbol{x}$. The basic procedure of the chaos optimization algorithm is as follows.

(1) *Initialization*. Initialize the number M of chaotic search and different initial value of n chaos variables z_i^0 (not the fixed points of Logistic map i.e. 0.25, 0.50, 0.75). Set the initial design variables $x_i^0 = XL_i + z_i^0(XU_i - XL_i)$, objective function $f^* = f(x^0)$ and iteration counter as $k = 1$.
(2) *Variable mapping*. Calculate the new chaotic variables $z_i^{k+1} = 4z_i^k(1 - z_i^k)$, and the new design variables by linear mapping: $x_i^{k+1} = XL_i + z_i^{k+1}(XU_i - XL_i)$.

(3) *Searching for optimal solution.*

If $k \leqslant M$, then

If $f(x^{k+1}) \leqslant f^*$, then $x^* = x^{k+1}$, $f^* = f(x^{k+1})$

Set $k = k + 1, z^k = z^{k+1}$, and go to step 2.

Else if $k > M$ is satisfied, then stop.

The above chaos optimization algorithm sometimes can obtain global optimum, but the effectiveness and efficiency is not good due to the limited local optimization ability of chaotic search. The conventional gradient descent algorithms are good at local optimization. By integrating these two algorithms we can develop a more efficient algorithm for global optimization. Because BFGS algorithm is one of robust and efficient quasi-Newton algorithms of local optimization [28], we select it to combine with chaos optimization algorithm to form the hybrid global optimization algorithm.

2.1. Improved hybrid Logistic map based chaos-BFGS algorithm

Following the above algorithm, a chaotic sequences with each component randomly distributed in $(0, 1)$ is generated. The probability density function (PDF) of the chaotic sequences with the nature of pseudo-randomness can be derived as follows [27,29]:

$$\rho(z) = \frac{1}{\pi\sqrt{z(1-z)}} \tag{5}$$

which is called as Chebyshev distribution shown in Fig. 1. From Fig. 1, it can be observed that the values of density function are nearly uniform around the middle of the interval, but much larger near two ends of $(0, 1)$. Therefore, during the searching iterations of chaos optimization, numerous searches are close to the two ends. Obviously, it is not beneficial for finding the global optimum if the global optimum does not locate near the two ends, which indicates that the intervals near the two ends should be properly thrown away. Based on the above observation, the hybrid improved chaos-BFGS algorithm is proposed. In addition, it is noted that the PDF of Kent map is uniform function, and then the hybrid Kent based chaos-BFGS algorithm is also suggested.

In the improved hybrid chaos-BFGS algorithm, the chaotic variable interval is partitioned into two subsets $S1 = (0, r) \cup (1 - r, 1)$ and $S2 = [r, 1 - r]$, in which $r \in (0, 0.5)$ is a adjustable parameter. Each time a value of chaotic variable z_i is generated from the Logistic map, and its validity is evaluated. If the chaotic variable z_i belongs to the subset $S1$ it is skipped. Otherwise, if it belongs to the subset $S2$, then it is reserved and used to calculate the objective function and check its optimum. Furthermore, the linear mapping from chaotic variable to design variable is changed as follows:

$$x_i^{k+1} = XL'_i + z_i^{k+1}(XU'_i - XL'_i) \tag{6}$$

where XL'_i and XU'_i are

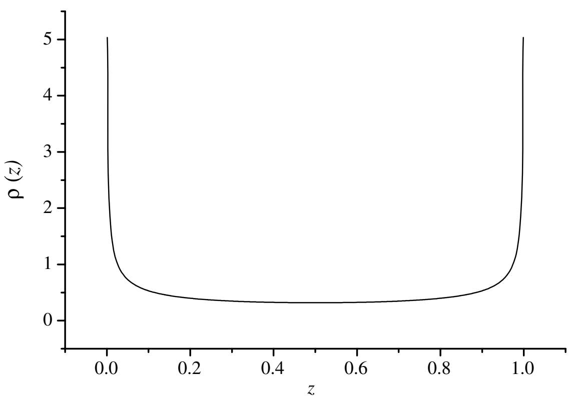

Fig. 1. The PDF of chaotic orbit points of Logistic map.

$$XL'_i = XL_i - \frac{r}{1-2r}(XU_i - XL_i) \tag{7a}$$

$$XU'_i = XU_i + \frac{r}{1-2r}(XU_i - XL_i) \tag{7b}$$

Consequently, it can be easily observed that for the interval of chaotic variable $S2 = [r, 1-r]$, the corresponding interval of design variable (XL_i, XU_i) is retained. The procedure of improved hybrid Logistic map based chaos-BFGS algorithm is shown as follows.

(1) Initialize the number M of chaotic search and starting point z^0, which consists of n different values for z_i ($i = 1, 2, \ldots, n$). Set iteration counter iter = 0, and objective function value f^{iter} as certain large value.
(2) Call the subroutine of chaos optimization algorithm, and get the current optimum x^{*C} and f^{*C}.
(3) Take x^{*C} as starting design point and run BFGS algorithm, and obtain the current optimum x^* and f^*. And set iter = iter + 1.
(4) If $|f^* - f^{\text{iter}}| > \varepsilon$ is satisfied for a specified small tolerance ε, then $f^{\text{iter}} = f^*$, and take x^* as the starting point of next chaos optimization, and go to step 2. Otherwise, obtain the final optimum x^* and f^* for hybrid algorithm, and stop the iteration.

Since we have cut the 'bad' part of chaotic variable interval $(0, 1)$ during the generation of chaotic sequences, which is distinct from the conventional chaos optimization algorithms in the papers [15–25], the presented hybrid optimization algorithm enhances the computational efficiency significantly. This will be confirmed by the numerical examples later.

2.2. Kent map based chaos-BFGS algorithm

The Kent map and Logistic map can be transformed each other, and there is a relationship of topological conjugacy between them [29]. The Kent map is written as [30,31]

$$f(z) = \begin{cases} z/\beta & 0 < z \leqslant \beta \\ (1-z)/(1-\beta) & \beta < z \leqslant 1 \end{cases} \tag{8}$$

where the parameter β is set the value within the interval $(0, 1)$. The Lyapunov exponent of Kent map is greater than 0, so Kent map is in chaotic state within $(0, 1)$. Because the chaotic sequences of Kent map are influenced by the limited byte length and the limited precision of computer, they are easy to trend to be periodic or converge to the stable value when $\beta = 0.5$ [31]. In this paper, in order to generate useful chaotic sequences from Kent map by computer, the parameter β should not be equal to 0.5 (here, $\beta = 0.4$ is assumed). Replacing the Logistic map by Kent map in the algorithm in Section 2.1, we can write the procedure of Kent map based chaos-BFGS algorithm similarly.

According to the hybrid optimization algorithms based on the improved Logistic map, Kent map and conventional Logistic map, some examples of global optimization of nonlinear functions are implemented and calculated. Furthermore, the comparative results will be shown in the latter section. However, the numerical results make us question the high efficiency of the chaos optimization algorithms claimed in some papers. Accordingly, we investigate the Monte Carlo-BFGS algorithm and mesh-BFGS algorithm further.

3. Monte Carlo-BFGS algorithm and mesh-BFGS algorithm

Monte Carlo method is a direct search method of global optimization [1], which converges slowly and obtains the global optimum with low probability. We combine Monte Carlo method with BFGS, and establish the Monte Carlo-BFGS algorithm for comparison with chaos-BFGS algorithms. The procedure of hybrid Monte Carlo-BFGS algorithm is similar to that of chaos-BFGS algorithm as follows.

Firstly, select the seed and generate the random sequences (numbers) with long period by the linear congruent generator (LCG) in computer. Pick n random numbers consecutively and map them to be the n components of the design vector. Then calculate and judge objective function value until the number of random search M is achieved. Secondly, take the current optimum obtained by Monte Carlo search as a starting point and minimize the objective function with BFGS method. Thirdly, implement the next round Monte Carlo-BFGS search and optimization until the given convergence criterion is satisfied.

Unlike the chaos optimization method and Monte Carlo method, the mesh method is a simply deterministic enumerative method. It evaluates the objective function at all mesh points with equal distance or predefined unequal one. This procedure, however, is quite inefficient due to the huge number of function evaluations when the number of design variables of optimization problem is large, namely, the problem dimension is high. Here, the mesh method is also coupled with BFGS method, and the hybrid algorithm has the same procedure as the chaos-BFGS algorithm.

4. Numerical examples and discussion

The efficiency and performance of the above five optimization algorithms with the following five nonlinear functions [11,12,15] is evaluated:

$$f_1(x,y) = \left(4 - 2.1x^2 + \frac{x^4}{3}\right)x^2 + xy + (-4 + 4y^2)y^2 \quad -10 < x,y < 10 \tag{9}$$

$$f_2(x,y) = 0.5 - \frac{\sin^2\sqrt{x^2+y^2} - 0.5}{(1 + 0.001(x^2+y^2))^2} \quad -4 < x,y < 4 \tag{10}$$

$$f_3(X) = \sum_{i=1}^{3}(x_i^2 - 10\cos(2\pi x_i) + 10) \quad -4 < x_i < 4, \ i = 1,2,3 \tag{11}$$

$$f_4(X) = \frac{1}{4000}\sum_{i=1}^{5}x_i^2 - \prod_{i=1}^{5}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad -5 < x_i < 5, \ i = 1,2,\ldots,5 \tag{12}$$

$$f_5(X) = \frac{1}{5}\sum_{i=1}^{5}(x_i^4 - 16x_i^2 + 5x_i) \quad -10 < x_i < 10, \ i = 1,2,\ldots,5 \tag{13}$$

Function f_1 is the Camel function, which has six local minima and two global minima $x^* = (-0.0898, 0.7126)$, $(0.0898, -0.7126)$, and optimal objective function value $f^* = -1.031628$. Function f_2 is the Schaffer's function, which has infinite local maxima and one global maximum $x^* = (0,0)$, and $f^* = -1.0$. Function f_3 is the Rastrigin's function, which has many local minima and one global minimum $x^* = (0,0,0)$, and $f^* = 0$. Function f_4 is the Griewank's function, which has several thousand local minima and one global minimum $x^* = (0,0,\ldots,0)$, and $f^* = 0$. Function f_5 has five variables, which has 32 local minima and one global minimum $x^* = (2.9051, 2.9051, \ldots, 2.9051)$, and $f^* = -78.3323$. These five nonlinear multimodal functions are often used to test the convergence, efficiency and accuracy of optimization algorithms [11,15]. Among them the former three functions have two or three design variables, which is 2- or 3-dimensional problem, and the latter two have five variables which is 5-dimensional problem.

The five hybrid optimization algorithms are represented as follows respectively. ILC-BFGS refers to the improved Logistic map based chaos-BFGS algorithm; KC-BFGS, the Kent map based chaos-BFGS algorithm; LC-BFGS, the conventional Logistic map based chaos-BFGS algorithm; MC-BFGS, the Monte Carlo-BFGS algorithm; and the fifth is the mesh-BFGS algorithm. Among them the first three algorithms are hybrid chaos optimization algorithms via chaotic search, and the fourth algorithm belongs to stochastic method based on the random search, and the fifth is a deterministic method.

Before presenting the results of the chaotic/stochastic algorithms, the performance of mesh-BFGS algorithm is examined. There are $K = (N+1)^m$ uniform mesh points by making use of mesh-BFGS algorithm, with each variable axes divided into N equidistant parts, and m is the number of variable axes, or design variables of the optimization problem. The computational results are listed in Table 1 and explained as follows. (1) For f_1, when $N = 9$ and $K = 100$, the mesh-BFGS algorithm does not obtain global optimum within the required accuracy of design variables,

Table 1
Computational results of global optimization by mesh-BFGS algorithm

	f_1		f_2		f_3		f_4	f_5	
N	9	19	49	59	45	47	9	1	2
K	100	400	2500	3600	97,336	110,592	100,000	32	243
Global optimum	No	Yes	No	Yes	No	Yes	No	Yes	Yes

Table 2
Probability of the global optimum for 2- or 3-dimensional functions (%)

	Function f_1			Function f_2			Function f_3		
M	100	500	1000	1000	5000	10,000	10,000	40,000	100,000
ILC-BFGS	82	98	100	69	100	100	25	47	65
KC-BFGS	75	98	100	61	100	100	17	40	61
LC-BFGS	67	89	95	36	81	95	9	21	34
MC-BFGS	70	96	100	61	100	100	18	48	64

Table 3
Probability of the global optimum for 5-dimensional functions (%)

	Function f_4			Function f_5		
M	10,000	40,000	100,000	10,000	40,000	100,000
ILC-BFGS	54	81	90	39	56	68
KC-BFGS	23	28	32	31	40	46
LC-BFGS	6	7	4	12	28	32
MC-BFGS	18	22	24	21	42	50

while when $N = 19$ and $K = 400$, it does. (2) For f_2, when $N = 49$ and $K = 2500$, the algorithm does not obtain global optimum, while when $N = 59$ and $K = 3600$, it does. (3) For f_3, when $N = 45$ and $K = 97{,}336$, the algorithm does not get global optimum, while when $N = 47$ and $K = 110{,}592$, it does. (4) For f_4, when $N = 9$ and $K = 100{,}000$, the algorithm does not obtain global optimum. (5) For f_5 when $N = 1$, $K = 32$, and $N = 2$, $K = 243$, the algorithm achieves global optimum, which is a special example. It should be pointed out that it is straightforward to obtain global optimum by setting $N =$ even number, because the global optimum of f_2, f_3, f_4 lies in the center of the design space. To avoid such a case, only the case of $N =$ odd number is considered.

For the chaotic/stochastic algorithms and all test functions, 100 stochastic starting points from the random numbers uniformly distributed in $(0, 1)$ by LCG of computer are employed to perform the optimization process. Several different values are set for the number M of chaotic/stochastic search. Table 2 shows the results of the first three 2- or 3-dimensional test functions, and Table 3 indicates the results of the last two 5-dimensional test functions. The probability in Table 2 and 3 means the number of obtaining the global optimum successfully from the 100 stochastic starting points via the algorithms, and is expressed as P (%). These results in Tables 2 and 3 are also shown in the Fig. 2. Several dozens of objective function evaluations is needed for BFGS and thousands of objective function evaluations for chaotic/stochastic search. In other words, the most computational effort of hybrid optimization algorithms is attributed to the chaotic/stochastic search of global optimization.

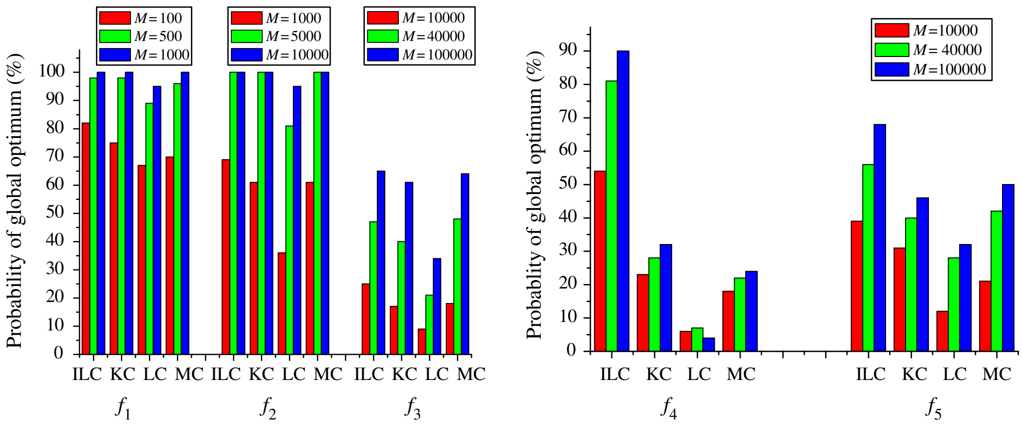

Fig. 2. Probability of the global optimum for five functions (%).

From the numerical results of five hybrid algorithms of global optimization for five nonlinear functions, the following observations can be made.

(1) The mesh-BFGS algorithm performs pretty well for 2- or 3-dimensional functions, as shown in Table 1. In fact it is better than any chaotic/stochastic algorithms. For 5-dimensional function, the mesh-BFGS algorithm faces the difficulty of combinatorial explosion due to the increased design variables of global optimization, and can be hardly applied. To compare the performance of different algorithms for global optimization, we should not just use 2- or 3-dimensional problems, because the difficulty of global optimization is closely related to the combinatorial explosion, which only appears in higher-dimensional problems.

(2) Generally, among the improved Logistic map based chaos-BFGS algorithm, Kent map based chaos-BFGS algorithm and Monte Carlo-BFGS algorithm, there is minor difference in computational efficiency as presented in Tables 2 and 3 and Fig. 2. And the efficiency between Kent map based chaos-BFGS algorithm and Monte Carlo-BFGS algorithm is very close.

(3) Efficiency of chaos optimization approaches depends on the location of the global optimum of problems.
The probability density function (PDF) of chaotic sequences (numbers) from Logistic map follows Chebyshev-type distribution, and that from Kent map follows uniform distribution. From intuitive viewpoint, the more sampled points lie in the vicinity of global optimum, the higher probability to find the global optimum. This behavior can be observed from the performance of Logistic map based chaos-BFGS algorithm. For example, as for f_4 and f_5, the global optimum of f_5 lies in the left part of design space, and the global optimum of f_4 locates in the center of design space. Because more points at two ends of design space are generated from the Logistic map, the results of f_5 are better than that of f_4. But the points from the improved Logistic map are uniform relatively, when the improved Logistic map chaos-BFGS algorithm is employed, the results of f_4 are superior to that of f_5.
If it is known where the global optimum of optimization problem is located, many sampled points from the Logistic map or other maps with non-uniform probability distribution (e.g. normal distribution) can be put in the neighborhood of global optimum to enhance the effectiveness and efficiency of hybrid algorithm. This idea is similar to the Monte Carlo importance sampling method. Unfortunately, it is quite difficult to predict the location of global optimum of actual problem functions. Consequently, it is appropriate to adopt the hybrid Kent map based chaos algorithm or the hybrid Monte Carlo algorithm with the uniform PDF. Furthermore, we think that Monte Carlo and Kent map based algorithms have good applicability without influence of the location of global optimum; and the optimization results by means of improved or conventional Logistic map based algorithm are related to the practical problem functions to some extent.

(4) In most cases, the conventional Logistic map based chaos-BFGS algorithm is computationally much less efficient than the other three chaotic/stochastic algorithms. Moreover, it becomes very bad for the more difficult problems such as the optimization of f_3, f_4, f_5 (Fig. 2). The statistical property of the sequences can explain the observed comparison. And the following more detailed discussions are presented.

The probability density function of chaotic or stochastic sequences (variables) can demonstrate the distribution of the sequences. And given the equal mean value, the standard deviation of sequences reflects the density distribution of sequences and the degree of scatter to mean value quantitatively.

As is known, the PDF of random sequences uniformly distributed in the interval $(0,1)$ is written as $\rho(z)=1$, and its mean value and standard deviation is as follows, respectively:

$$E\xi = 0.5, \quad \sqrt{D\xi} = \sqrt{\frac{1}{12}} = 0.28867 \tag{14}$$

With the same PDF as random sequences, the numeral characteristic of chaotic sequences from Kent map is also equal to that of random sequences. The PDF of chaotic sequences from Logistic map is expressed as Eq. (5), namely, the Chebyshev function, its mean value and standard deviation is written as

$$E\xi = 0.5, \quad \sqrt{D\xi} = \sqrt{\frac{1}{8}} = 0.35355 \tag{15}$$

While the PDF of chaotic sequences in the interval $(0.1, 0.9)$ generated from the improved Logistic map is derived as

$$\rho(z) = \frac{\lambda}{\pi\sqrt{z(1-z)}}, \quad \lambda = \frac{\pi}{2\arcsin(0.8)} = 1.69395 \tag{16}$$

Table 4
Probability of the global optimum for improved chaos optimization algorithm with different interval of chaotic variable (%)

	f_1		f_2		f_3		f_4		f_5	
M	100	500	1000	5000	10,000	40,000	10,000	40,000	10,000	40,000
(0.1, 0.9)	82	98	69	100	25	47	54	81	39	56
(0.2, 0.8)	88	100	88	100	43	71	100	100	64	87
(0.3, 0.7)	99	100	100	100	66	98	100	100	100	100

And the numeral characteristic of the new chaotic sequences in $(0.1, 0.9)$ is

$$E\xi = 0.5, \quad \sqrt{D\xi} = 0.24555 \tag{17}$$

From Eqs. (14), (15) and (17), it can be seen that the standard deviation of conventional Logistic sequences is larger than those of other three sequences, which implies that there are fewer sample points adjacent to the mean value of sequences relatively. Since the global optimum for the five test functions locate near the center of variable domain corresponding to the mean value of sequences, the performance (including computational efficiency and optimization capability) of conventional Logistic map based chaos optimization algorithm is much worse than those of the other hybrid optimization algorithms. The standard deviation of sequences from Kent map based hybrid algorithm equals to that from Monte Carlo-BFGS hybrid algorithm, therefore, the difference of performance between these two algorithms is very small.

Moreover, as for the improved chaos optimization algorithm, when the value r of chaotic variable interval of Logistic map is set 0.1, 0.2, 0.3, the standard deviation of chaotic sequences is 0.24555, 0.23752, 0.23358, respectively, in the standard interval $[r, 1-r]$. The calculated results of improved chaos optimization algorithm with different interval of chaotic variable for five test functions are listed in the Table 4. It is found that from the numerical results in Table 4, the bigger the value r up to 0.3 is, the smaller the standard deviation of chaotic sequences, and the better the performance of hybrid chaos optimization algorithm. In a word, to account for the computational performance of improved Logistic map based hybrid algorithm comprehensively, the following two factors should be considered, the statistical property of new chaotic sequences in the interval $[r, 1-r]$ and the mathematic property of all local optima of problem functions. Additionally, it is worth noting that the new chaotic sequences of Logistic map $z^{k+1} = \mu z^k(1-z^k)$ in the interval $[r, 1-r]$ add an adjustable variable r besides the control parameter μ and initial value z^0 for chaotic number generator [31]. Therefore, it becomes more difficult to be decoded for the chaotic sequences and this is beneficial for the safety of chaotic cryptogram communication.

5. Conclusion

In the present paper, five nonlinear multimodal functions are used to examine the performance of five hybrid algorithms including three chaos optimization ones, and the computational efficiency and the optimization capability of hybrid algorithms are compared. Moreover, the influential factors on the performance of hybrid optimization algorithms are discussed and clarified. It is concluded that the computational efficiency of hybrid optimization algorithms is affected at least by the following factors: the statistical property of chaotic/stochastic sequences generated from optimization algorithms, and the location of global optimum of nonlinear functions. In addition, through adding an adjustable parameter r to the Logistic map and retaining the chaotic points in the interval $[r, 1-r]$, the new chaotic sequences is generated, and it is more difficult to be decoded for the chaotic sequences and beneficial for the safety of chaotic cryptogram communication.

Finally, we would like to point out that the characteristic of chaos optimization is chaotic search by utilizing chaotic sequences with infinitely length period theoretically generated from simply nonlinear chaotic map. Unlike the other chaos optimization methods based on chaotic neural networks, chaotic annealing [8–10] and chaotic descent method [13,14] that use the dynamical bifurcation process and fractal set concept respectively, the chaos optimization algorithms discussed herein [15–25] only utilize the same pseudo-randomness and ergodicity of chaotic sequences as stochastic sequences rather than the unique properties of chaos. Additionally, in some references the low-dimensional functions were solved to test the performance and efficiency of chaos optimization algorithm. From our observation, it is inappropriate to conclude the high efficiency of global optimization algorithms only depending on several numerical examples of low-dimensional functions. To date, the global optimization is still a great challenge to us.

Acknowledgments

The support of the National Natural Science Foundation of China (Grant Nos. 10421002, 50578028), the National High Technology Research and Development Program of China (Grant No. 2001AA602015) and the National Basic Research Program of China (Grant No. 2006CB601205) is much appreciated.

References

[1] Arora JS, Elwakeil OA. Global optimization methods for engineering application: a review. Struct Optim 1995;9:137–59.
[2] Pardalos PM, Romeijn HE, Tuy H. Recent developments and trends in global optimization. J Comput Appl Math 2000;124:209–28.
[3] Ott E, Grebogi C, Yorke JA. Controlling chaos. Phys Rev Lett 1990;64(11):1196–9.
[4] Pecora L, Carroll TL. Synchronization in chaotic system. Phys Rev Lett 1990;64(8):821–4.
[5] Aihara K, Takabe T, Toyoda M. Chaotic neural networks. Phys Lett A 1990;144(6/7):333–40.
[6] Chen L, Aihara K. Chaotic simulated annealing by a neural network model with transient chaos. Neural Networks 1995;8(6):915–30.
[7] Chen L, Aihara K. Global searching ability of chaotic neural networks. IEEE Trans Circuits Syst I-Fundam Theory Applicat 1999;46(8):974–93.
[8] Zhou CS, Chen TL. Chaotic annealing for optimization. Phys Rev E 1997;55(3):2580–7.
[9] Zhou CS, Chen TL. Chaotic neural networks and chaotic annealing. Neurocomputing 2000;30:293–300.
[10] Kowk T, Smith K. A unified framework for chaotic neural network approaches to combinatorial optimization. IEEE Trans Neural Networks 1999;10(4):978–81.
[11] Li B, Jiang WS. Optimizing complex function by chaos search. Cybern Syst 1998;29(4):409–19.
[12] Zhang P, He C, Zhang X, Qiao Y. The survey and future development on global optimization algorithms in engineering. In: Extended abstracts of the fourth world congress of structural and multidisciplinary optimization (WCSMO-4), Dalian, China, June 4–6 2001. p. 328–9.
[13] Jovanovic V. Chaotic descent method and fractal conjecture. Int J Numer Methods Eng 2000;48(1):137–52.
[14] Jovanovic V, Kazerounian K. Optimal design using chaotic descent method. ASME J Mech Des 2000;122(3):265–70.
[15] Zhang T, Wang HW, Wang ZC. Mutative scale chaos optimization algorithm and its application. Control Decis 1999;14(3):285–8.
[16] Wang DG, Liu YX, Li SJ. Chaos-regularization hybrid algorithm for nonlinear two dimensional inverse heat conduction problem. Appl Math Mech 2002;23(8):973–80.
[17] Liu SS, Wang M, Hou ZJ. Hybrid algorithm of chaos optimization and SLP for optimization power flow problems with multimodal characteristic. IEE Proc Generat Transm Distribut 2003;150(5):543–7.
[18] Wang Y, Sun HW, Sun YK. A hybrid genetic algorithm based on mutative scale chaos optimization strategy. J Univ Sci Technol Beijing 2002;9(6):470–3.
[19] Yan XF, Chen DL, Hu LS. Chaos-genetic algorithms for optimizing the operating conditions based on RBF-PLS model. Comput Chem Eng 2003;27(10):1393–404.
[20] Sun SL, Liu JF. An efficient optimization procedure for tetrahedral mesh by chaos search algorithm. J Comput Sci Technol 2003;18(6):796–803.
[21] Lu Z, Shieh LS, Chen GR. On robust control of uncertain chaotic systems: a sliding-mode synthesis via chaotic optimization. Chaos, Solitons & Fractals 2003;18:819–27.
[22] Lu Z, Shieh LS, Chandra J. Tracking control of nonlinear systems: a sliding mode design via chaotic optimization. Int J Bifurcat Chaos 2004;14(4):1343–55.
[23] Lu Z, Shieh LS, Chen GR, Coleman NP. Simplex sliding mode control for nonlinear uncertain systems via chaos optimization. Chaos, Solitons & Fractals 2005;23(3):747–55.
[24] Ji MJ, Tang HW. Application of chaos in simulated annealing. Chaos, Solitons & Fractals 2004;21:933–41.
[25] Liu B, Wang L, Jin YH, Tang F, Huang DX. Improved particle swarm optimization combined with chaos. Chaos, Solitons & Fractals 2005;25:1261–71.
[26] Lorenz EN. Deterministic nonperiodic flow. J Atmos Sci 1963;20:130–41.
[27] May R. Simple mathematical models with very complicated dynamics. Nature 1976;261:459–67.
[28] VR&D. DOT (design optimization tools) users manual version 4.20. Colorado: Vanderplaats Research & Development Inc; 1995.
[29] Hao BL. Introduction to chaotic dynamics: starting from parabola. Shanghai: Shanghai Science Technology and Education Press; 1993.
[30] Chua LO, Yao Y. Generating randomness from chaos and constructing chaos with desired randomness. Int J Circuit Theory Applicat 1990;18:215–40.
[31] Zhao CM, You XH, Cheng SX. Digital generator of chaos sequence and its application in communication. J Southeast Univ China 1995;25(4):137–42.