

非线性函数全局最优化的一种混沌优化混合算法

杨迪雄, 李 刚

(大连理工大学工程力学系, 工业装备结构分析国家重点实验室, 辽宁 大连 116023)

摘 要: 混沌优化方法是近年出现的利用混沌的遍历性、随机性作为全局优化机制的一种优化技术。已有的混沌优化方法都是利用 Logistic 映射作为混沌序列发生器, 而由 Logistic 映射产生的混沌序列的概率密度函数服从两头多、中间少的切比雪夫型分布, 这种分布特性会严重影响混沌优化全局搜索能力和效率。利用 Logistic 映射的特点, 在混沌搜索时预先筛选掉劣质点, 建立改进的混沌—BFGS 混合优化算法。复杂非线性测试函数计算结果表明, 与文献中不加改进的混沌混合算法相比, 本算法以同样的混沌搜索次数找到全局最优解的概率提高了 10-30%, 而以概率 1 获得全局最优解的最大混沌搜索次数减少了 8-10 倍。另外, 还将细搜索策略引入到改进的混沌—BFGS 混合算法中, 对具有较大边界约束范围的非线性函数进行了优化计算。

关键词: 全局最优化; 混沌—BFGS 混合优化算法; 非线性函数; Logistic 映射; 概率密度函数

中图分类号: O39 **文献标识码:** A

A HYBRID CHAOS OPTIMAZATION ALGORITHM FOR GLOBAL OPTIMIZATION OF NONLINEAR FUNCTIONS

YANG Di-xiong, LI Gang

(Department of Engineering Mechanics, Dalian University of Technology,

State Key Laboratory of Structural Analysis of Industrial Equipment, Dalian 116023, China)

Abstract: The chaos optimization technique proposed in recent years searches the global optimum, using the properties of ergodicity and randomness of the chaos sequence. The chaos optimization algorithms in the literature are all based on Logistic map. However, it is noticed that the probability density function of chaotic sequence of Logistic map is Chebyshev-type function, which may affect the global searching capacity and computational efficiency of chaos optimization algorithm severely. In this paper, considering the property of probability density function of chaotic sequence, an improved hybrid chaos-BFGS optimization algorithm is established by eliminating the bad design points during the chaos searching process. Numerical results of the improved chaos-BFGS algorithm for the nonlinear test functions show that the probability of getting the global optimum increases by 10-30 % for the same number of chaos search, and the number of chaos search reduces 8-10 times with the probability of 1.0 for obtaining the global optimum, compared to the results in the reference. In addition, the fine search strategy is introduced into the hybrid algorithm so that it can optimize the nonlinear functions with large boundary constraints.

Key words: global optimization; hybrid chaos optimization-BFGS algorithm; nonlinear functions; Logistic map; probability density function

收稿日期: 2002-10-20; 修改日期: 2003-05-28

基金项目: 国家自然科学基金资助项目(59895410, 50008003, 10032030)和 863 计划资助项目(2001AA602015)

作者简介: 杨迪雄(1970), 男, 湖南人, 博士生, 从事计算力学和结构工程研究(E-mail: yangdixiong@hotmail.com);

李 刚(1966), 男, 山东人, 副教授, 博士, 从事计算力学和结构工程研究

1 引言

在许多应用中找到全局最优解是十分重要的。然而,全局最优解没有任何数学条件可以表征,这与问题函数的局部最优解可以用其局部性质(如函数的梯度和海森矩阵)来表征是不同的。因此,无论从数学的还是计算的观点来看,求解全局最优化问题都仍然是巨大的挑战。混沌是非线性动力学系统中的一种普遍现象,它指由确定性系统产生的对于初始条件极为敏感的貌似随机的非周期运动,其长程行为不能预测。混沌具有其独特的性质,即:随机性、遍历性、规律性、普适性、具有分形的性质等。混沌的遍历性和随机性特点可以作为搜索过程中避免陷入局部极小的一种优化机制,这与模拟退火的概率性劣向转移和禁忌搜索的禁忌表检验存在一定的区别^[1]。混沌优化作为一种新的优化技术,受到了一定的关注,并在函数优化、组合优化以及化工、机械、电力系统等领域的优化问题中得到了应用。

Chen 等^[2]提出了混沌模拟退火算法,即先利用瞬时混沌动态进行全局遍历性搜索,然后转入梯度下降搜索,成功解决了 TSP、MSP 等组合优化问题。此后,多位学者利用混沌神经网络、混沌遗传算法解决组合优化问题^[1]。在函数优化方面,李兵、蒋慰孙^[3,4]利用 Logistic 混沌映射和二次载波的概念,提出了一种应用于非线性复杂函数的混沌优化方法,其基本思想是把混沌变量线性映射到优化设计变量的取值区间,然后利用混沌变量进行搜索。但是,进一步的仿真计算表明,该方法对于搜索区间小时效果显著,反之不能令人满意。于是,张彤等^[5]提出变尺度混沌优化法,该方法不断缩小优化变量的搜索区间并不断提高搜索精度,从而有较高的搜索效率。由于混沌运动的随机性,并且混沌变量迭代前后两次的运动幅度往往较大,导致优化设计变量在设计空间中产生较大的跳跃,所以即使上述的混沌优化方法已经搜索到了最优解附近,算法仍然需要计算大量远离最优解的劣质点才能向全局最优解逐渐靠近。这样,为了达到一定的计算精度,这些混沌优化算法的计算时间将会较长。钱富才等^[6]把共扼梯度法与混沌优化法结合,提出了一种非线性函数的混合全局优化算法。文献[7~9]分别将混沌优化法与 BFGS、Powell、罚函数法结合求解具有简单边界约束和复杂约束的非线性函数优化问题。

这些混沌优化方法都是利用 Logistic 映射作为混沌序列发生器,我们注意到由 Logistic 映射产生的混沌序列的概率密度函数服从两头多、中间少的切比雪夫型分布^[10],这种分布特性可能会严重影响到混沌优化的全局搜索能力和效率。因此,本文利用 Logistic 映射混沌轨道点概率密度函数的特点,划分混沌变量区间,搜索时预先筛选掉劣质点,建立了改进的混沌—BFGS 混合优化算法。

2 混沌优化方法

具有简单边界约束的非线性函数最优化问题可以表示为:

$$\text{Min } f(x) = f(x_1, x_2, \dots, x_n) \quad (1a)$$

$$\text{s.t. } XL_i \leq x_i \leq XU_i \quad (i=1, 2, \dots, n) \quad (1b)$$

混沌优化算法的混沌变量由 Logistic 映射^[10]—— $z^{k+1} = \mu z^k (1 - z^k)$ ——产生。描述生态学上虫口模型的 Logistic 映射自 May 于 1976 年开始研究以来,受到了非线性科学家的高度关注,Feigenbaum 就是从 Logistic 映射出发于 1978 年发现了 Feigenbaum 普适常数。Logistic 映射是混沌理论发展史上不可多得的典范性的混沌模型,它几乎具有迄今为止所有混沌模型所拥有的特征和性质。难能可贵的是,Logistic 映射只是一个简单的一维映射,其非线性行为却是那么丰富多彩,使初学者通过这一简单映射便可以初窥混沌的奥妙。Logistic 映射表达式中, z^k 表示混沌变量 z 在 k 次迭代时的值。已经证明, $\mu=4$ 时,映射处于混沌状态,区间(0, 1)是 Logistic 映射的混沌不变集^[10],见(2)式。正是(0, 1)混沌不变集,使得混沌优化方法总是在函数的设计空间上进行全局搜索。

$$\begin{aligned} z_i^{k+1} &= 4z_i^k (1 - z_i^k) & 0 < z_i^k < 1 \\ i &= 1, 2, \dots, n, & k = 0, 1, 2, 3, \dots \end{aligned} \quad (2)$$

利用混沌搜索求解问题(1)时,先建立混沌变量 z 与设计变量 x 的对应关系

$$x_i = XL_i + z_i(XU_i - XL_i) \quad (3)$$

然后,由 Logistic 映射产生 n 个轨迹不同的混沌变量 $z_i (i=1, 2, \dots, n)$ 进行优化搜索。混沌优化方法的基本步骤如下:

(1) 初始化:给定最大混沌运动次数 M ; 给式(2)中的 z 赋 n 个相异的初值 z_i^0 (不能取 Logistic 映射的不动点: 0.25, 0.50, 0.75), 根据式(3)计算 x_i^0 , $f^* = f(x^0)$; 置 $k=0$, $i=1, 2, \dots, n$;

(2) 利用(2)和(3)式,计算产生混沌变量序列

z_i^{k+1} 及设计变量序列 x_i^{k+1} ;

(3) 若 $k \leq M$,

若 $f(x^{k+1}) \leq f^*$, 令 $x^* = x^{k+1}$,
 $f^* = f(x^{k+1})$;

若 $f(x^{k+1}) > f^*$, x^* 和 f^* 保持不变 ;

然后, 令 $k = k + 1$, $z^k = z^{k+1}$, 转向(2)。

若 $k > M$, 则混沌搜索结束。

文献[4,5]的混沌优化算法在此基本步骤基础上增加了细搜索、变尺度搜索功能。

3 改进的混沌—BFGS 混合优化算法

混沌优化方法具有全局搜索能力, 而其局部寻优能力不足, 计算效率不高。常规的梯度下降类算法局部寻优能力很强, 计算速度快。将二者结合在一起, 发挥两种方法各自的优势, 可以有效地求取非线性复杂函数的全局最优解。文献[6,7]的数值算例说明了这一点。如何将混沌优化方法与常规优化算法良好地结合, 使混合算法的全局搜索能力强、计算效率较高, 是混合算法程序设计中需要注意的问题。由于 BFGS 方法是求解无约束最优化问题的拟牛顿类方法最有代表性的算法之一, 本文也选择它与混沌优化法结合, 组成非线性复杂函数全局最优化的混合算法。

3.1 改进的混沌—BFGS 混合优化算法

公式(2)表示的 Logistic 映射经过一定次数的迭代后, 在(0, 1)线段上分布了大量的轨道点, 经过简单运算其概率密度函数 $r(z)$ 的解析表达式为^[10] :

$$r(z) = \frac{1}{p \cdot z(1-z)} \quad (4)$$

(4)式表示为切比雪夫分布, 如图 1 所示。

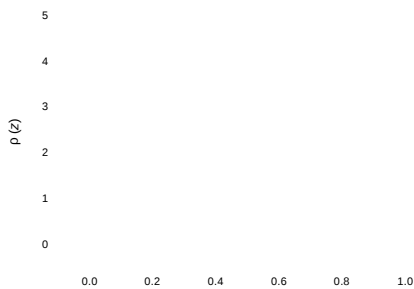


图 1 Logistic 映射混沌轨道点的概率密度函数

Fig.1 Probability density function of chaotic sequence of Logistic map

从图 1 可知, Logistic 映射混沌轨道点的分布特点是中间分布比较均匀, 两头分布特别多, 在端

点达到无穷多。所以, 混沌优化时, 许多次搜索都在设计变量空间的两端进行, 这对找到最优点是很不利的, 位于区间(0, 1)两端的点便是混沌搜索的劣质点。

本文把混沌变量空间分成两部分, $S1 = (0, r) \cup (1-r, 1)$ 和 $S2 = [r, 1-r]$, 用于区间划分的 $r \in (0, 0.5)$, r 一般取 0.10。用 Logistic 映射产生的点序列, 如果位于 $S1$ 内, 则放弃该点, 映射产生新的点, 重新进行搜索、判断; 如果位于 $S2$ 内, 则按前述算法进行搜索、判断。相应的, 混沌变量到设计变量的映射式(3)修改为

$$x_i^{k+1} = XL_i' + z_i^{k+1}(XU_i' - XL_i') \quad (5)$$

其中

$$\begin{aligned} XL_i' &= XL_i - \frac{r}{1-2r}(XU_i - XL_i), \\ XU_i' &= XU_i + \frac{r}{1-2r}(XU_i - XL_i) \end{aligned} \quad (6)$$

这相当于混沌变量区间虽然仍为(0, 1), 对应的设计变量区间变为

$$\begin{aligned} &(XL_i - \frac{r}{1-2r}(XU_i - XL_i), \\ &XU_i + \frac{r}{1-2r}(XU_i - XL_i)) \end{aligned} \quad (7)$$

上述做法的实质是把混沌映射在 $S1$ 内产生的点看作劣质点, 每次混沌搜索时预先筛选掉这些劣质点, 而混沌变量区间的有效长度为 $1-2r$, 与其对应的设计变量区域仍为原设计变量空间, 即混沌变量区间($r, 1-r$)经过线性映射后对应的设计变量区间为(XL_i, XU_i)。然后混沌优化法与 BFGS 法交替在可行域内搜索最优解。BFGS 法的可行域仍限定在原设计变量区间。改进的混合算法具体步骤如下:

(1) 初始化: 给定最大混沌运动次数 M 和优化收敛判据 e , 给混沌变量赋 n 个随机初值, $iter=0$, 给目标函数赋大值 f^{iter} ;

(2) 调用混沌优化程序, 得混沌搜索后的当前最优点 x^{*C} 和不大于 f^{iter} 的最优解 f^{*C} ;

(3) 以 x^{*C} 作为初始点, 调用 BFGS 程序, 得 BFGS 寻优后的当前最优解 x^* 和 f^* , $iter=iter+1$;

(4) 若 $f^* - f^{iter} > e$, 则 $f^{iter} = f^*$, 并且 x^* 作为下次混沌优化的初始解, 转向(2);

否则, 得到混合算法的最优解 x^* 和 f^* , 计算结束。

混合算法直接调用优化程序 DOT(Design Optimization Tools)中的 BFGS 方法^[11], 其数值稳定

性好,对复杂问题求解具有良好的适用性和健壮性。

算例 1: 本文采用文献[7]中的两个复杂测试函数作为算例,并与文献[7]的结果做了比较。

$$f_1(x, y) = (4 - 2.1x^2 + \frac{x^4}{3})x^2 + xy + (-4 + 4y^2)y^2 \quad (8)$$

$-10 < x, y < 10$

$$f_2(x, y) = 0.5 - \frac{\sin^2 x^2 + y^2 - 0.5}{(1 + 0.001(x^2 + y^2))^2} \quad (9)$$

$-4 < x, y < 4$

函数 f_1 称为 Camel 函数,该函数有六个局部极小点,其中有两个全局最优点 $P_1=(-0.0898, 0.7126)$ 和 $P_2=(0.0898, -0.7126)$,最小值为-1.031628。函数 f_2 称为 Schaffer 函数,该函数有无数个极大点,其中只有一个全局最优点(0,0),最大值为 1.0。此函数的最大峰值周围有一圈脊,它们的取值均为 0.990283,因此函数优化时很容易停留在这些局部极大点处。文献[7]采用其混沌—BFGS 混合算法,由计算机内部随机函数自动产生 100 个不同的初始点进行 100 次函数优化。每次函数优化时,最大混沌搜索次数 M 取不同数值时对函数 f_1, f_2 的计算结果如表 1、2 所示^[7],一般每次混合算法迭代均为 2~4 次。

为了便于和上述计算结果比较,本文的 100 个不同的初始点也由随机函数自动产生,对这两个测试函数进行优化计算。 $r=0.10$, M 取不同数值时对函数 f_1, f_2 的计算结果也列于表 1、2 中。计算结果中未搜索到全局最优点时,算法都获得了局部最优点,每次混合算法迭代均为 2~4 次。BFGS 算法一般计算目标函数几十次,而混沌搜索的目标函数计算次数为: $M \times \text{迭代次数}(\text{iter})$,一般需要计算几百上千次,所以混合算法的计算时间大部分花在混沌搜索的目标函数计算上。

表 1 M 取不同数值时函数 f_1 的计算结果比较

Table 1 Comparison of numerical results for function f_1 with different value of M

	最大混沌 搜索次数 M	搜索到全局最优点的次数		搜索到 最优的概率
		点 P_1	点 P_2	
文献[7] 算法	1000	44	39	83%
	3000	53	45	98%
	5000	53	47	100%
本文算法	500	47	53	100%

表 2 M 取不同数值时函数 f_2 的计算结果比较

Table 2 Comparison of numerical results for function f_2 with different value of M

最大混沌 搜索次数 M	文献[7]算法		本文算法	
	搜索到全局最 优点的次数	搜索到最 优的概率	搜索到全局最 优点的次数	搜索到最 优的概率
1500	40	40%	68	68%
5000	73	73%	99	99%
6000	—	—	100	100%
10000	88	88%	—	—
50000	100	100%	—	—

从表 1 计算结果的比较可知,对 f_1 即 Camel 函数,随机选择 100 个不同的初始点进行优化计算,本文改进的混沌算法最大混沌搜索 500 次,100 次优化计算都找到了全局最优解。文献[7]不加改进的混沌算法最大混沌搜索 1000 次,同样 100 次随机选择初始点的优化计算只有 83 次找到了全局最优解。而文献[7]算法要找到全部全局最优解,则需要增大混沌搜索次数到 5000 次,是本文算法的 10 倍。同时,从表 2 计算结果的比较可知,对 f_2 即 Schaffer 函数,最大混沌搜索 1500 次,100 次随机选择初始点进行优化计算,本文算法有 68 次找到了全局最优解,而文献[7]算法只有 40 次得到了全局最优解;最大混沌搜索 5000 次时,本文算法找到全局最优解的概率提高了 26%。再考察 100 次随机选择初始点进行优化计算,两种算法全部获取全局最优解的最大混沌搜索次数,本文改进算法是 6000 次,文献[7]算法是 50000 次,是改进算法的 8.3 倍。总之,与文献[7]混沌算法相比较,本文算法以同样的混沌搜索次数找到全局最优解的概率提高了 10-30 个百分点,而以概率 1 获得全局最优解的最大混沌运动次数 M 减少了 8-10 倍,表明混沌搜索的筛选机制使得改进的混合算法的计算效率和全局搜索能力得到大幅度的提高。当最大混沌搜索次数 M 足够大时,混合算法将以概率 1 获得全局最优解,文献[6]给出了证明,本文算例再次验证了这一点。需要说明的是,由于混沌映射对初值高度敏感,初值的细微差别导致搜索结果可能不一致,因此要求 BFGS 程序具有良好的数值稳定性,以使混合算法较快地找到全局最优点。

3.2 改进算法的细搜索策略

上面建立的混合算法,当非线性函数的边界约

束范围放大时，换句话说函数的设计空间扩大时，计算效率降低不少。这里在混沌优化过程中引入与张彤等^[5]提出的变尺度搜索类似的细搜索策略，使得混合算法即使函数具有较大的设计空间也能保持较高的计算效率。具体办法是在改进的混合算法的第 2 步混沌优化阶段，直接对混沌变量搜索采取以下措施。从第 2 次开始每一次进行混沌优化时不断缩小混沌变量区间为：

$$(x_i^{*C} - t / iter, x_i^{*C} + t / iter) \tag{10}$$

其中 $t \in (0,0.5)$, $iter$ 是混沌优化的迭代次数 , x^{*C} 每一次都在变化 , 当混沌搜索次数 M 较大时 x^{*C} 接近全局最优点 , 前后变化不大 , 因此要求 M 取值较大 , 以达到细搜索到全局最优解的目的。这样 , 当混沌变量位于缩小的变量区间时 , 进行目标函数的计算和判断 , 否则放弃该点。

表 3 有、无细搜索策略的函数 f_1 计算结果比较
Table 3 Comparison of numerical results for function f_1
with and without fine search strategy

最大混沌 搜索次数 M	无细搜索策略		有细搜索策略	
	搜索到全	搜索到	搜索到全	搜索到
	局最优点	最优的	局最优点	最优的
	的次数	概率	的次数	概率
500	66	66%	99	99%
1000	82	82%	100	100%

算例 2：将前面 f_1 、 f_2 函数的边界约束改为 (-100,100)，重新对函数进行优化计算。 $r=0.10$, $t=0.025$, M 取不同值时，有、无细搜索策略的 f_1 、 f_2 函数的计算结果见表 3、4。结果表明，无细搜

索策略的混合算法对大空间的全局优化能力和效率都不高；引入细搜索策略的混合算法，即使优化问题的边界约束范围较大时，全局优化效率仍然较高。

表 4 有、无细搜索策略的函数 f_2 计算结果比较
Table 4 Comparison of numerical results for function f_2
with and without fine search strategy

最大混沌	无细搜索策略		有细搜索策略	
	搜索到全	搜索到	搜索到全	搜索到
	局最优点	最优的	局最优点	最优的
	的次数	概率	的次数	概率
5000	1	1%	62	62%
20000	—	—	96	96%
40000	—	—	100	100%
50000	14	14%	—	—

4 结 语

混沌优化方法是利用混沌的遍历性、随机性特点作为全局优化机制的一种直接搜索算法，它与非线性规划算法结合，可以更好地求解优化问题。本文利用 Logistic 映射混沌轨道点概率密度函数的特点，建立的改进混沌—BFGS 混合优化算法，与文献中算法相比，计算效率和全局搜索能力得到显著提高。当然，混沌优化方法是近几年才发展起来的一种优化技术，还有许多问题需进一步研究^[1]，如算法停止准则的建立、优化效率和性能的进一步提高、在复杂和大规模实际问题中的应用潜力探讨等。

功的。由此表明：本文提出的方法是可行的。

(2) 对端承型 PHC 桩,本文方法需要进行一定程度的修正,但本文的逐次线性概似法的基本原理却普适于非只 PHC 桩在内的一般桩型的承载力预测。

参考文献：

- [1] JGJ94-94, 建筑桩基技术规范[S]. 北京: 中国建筑工业出版社, 1995.
JGJ94-94, Technical Code for Building Pile foundations [S]. Beijing: China Construction Industry Press, 1995. (in Chinese)
- [2] 李庆扬, 王能超, 易大义. 数值分析[M]. 武汉: 华中理工大学出版社, 1986.
Li Qingyang, Wang Nengchao, Yi Dayi. Numerical analysis [M]. Wuhan: Press of Huazhong University of

Technology, 1986. (in Chinese)

- [3] 樊映川. 高等数学讲义[M]. 北京: 人民教育出版社, 1964.
Fan Yingchuan. Higher mathematics teaching materials [M]. Beijing: People Education Press, 1964. (in Chinese)
- [4] 宰金珉, 宰金璋. 高层建筑基础分析与设计[M]. 北京: 中国建筑工业出版社, 1994.
Zai Jinmin, Zai Jinzhang. Analysis and design on high-rise building foundations[M]. Beijing: China Construction Industry Press, 1994. (in Chinese)
- [5] 李巨文, 周书敬, 黄光春. 水泥深层搅拌桩极限承载力的灰色预测[J]. 建筑结构, 1999, 5: 37-39.
Li Juwen, Zhou Shujing, Huang Guangchun. Gray forecast on the cement deep mixing pile bearing capacity[J]. Building Structures, 1999, 5: 37-39. (in Chinese)

(上接 110 页)

参考文献：

- [1] 王凌, 郑大钟, 李清生. 混沌优化方法的研究进展[J]. 计算技术与自动化, 2001,20(1):1-4.
Wang Ling, Zheng Dazhong, Li Qingsheng. Survey on chaotic optimization methods [J]. Computing Technology and Automation, 2001, 20(1): 1-4. (in Chinese)
- [2] Chen L, Aihara K. Chaotic simulated annealing by a neural network model with transient chaos [J]. Neural Networks, 1995, 8(6): 915-930.
- [3] Li Bing, Jiang Wei Sun. Optimizing complex function by chaos search [J]. Cybernetics and Systems, 1998,29(4): 409-419.
- [4] 李兵, 蒋慰孙. 混沌优化方法及其应用[J]. 控制理论与应用, 1997,14(4):613-615.
Li Bing, Jiang Weisun. Chaos optimization method and its application [J]. Control Theory and Application, 1997, 14(4): 613-615. (in Chinese)
- [5] 张彤, 王宏伟, 王子才. 变尺度混沌优化方法及其应用[J]. 控制与决策, 1999,14(3):285-288.
Zhang Tong, Wang Hongwei, Wang Zicai. Mutative scale chaos optimization algorithm and its application [J]. Control and Decision, 1999, 14(3): 285-288. (in Chinese)
- [6] 钱富才, 费楚红, 万百五. 利用混沌搜索全局最优的一种混合算法[J]. 信息与控制, 1998, 27(3): 232-235.
Qian Fucai, Fei Chuhong, Wan Baiwu. A hybrid algorithm for finding global minimum [J]. Information and Control, 1998, 27(3): 232-235. (in Chinese)
- [7] 王登刚, 刘迎曦, 李守巨. 非线性最优化问题的一种混合解法[J]. 工程力学, 2001, 18(3): 61-66.
Wang Denggang, Liu Yingxi, Li Shouju. A hybrid approach for nonlinear optimization [J]. Engineering Mechanics, 2001, 18(3): 61-66. (in Chinese)
- [8] 黄文培, 王金诺, 于兰峰. 混沌—Powell 混合算法在机械优化设计中的应用研究[J]. 四川大学学报, 2001, 33(5): 31-34.
Huang Wenpei, Wang Jinruo, Yu Lanfeng. A hybrid algorithm combining Powell method with chaos optimization method and its application to mechanical optimum design [J]. Journal of Sichuan University (Engineering Science Edition), 2001, 33(5): 31-34. (in Chinese)
- [9] 张春慨, 徐立云, 邵惠鹤. 改进混沌优化及其在非线性约束优化问题中的应用[J]. 上海交通大学学报, 2000, 34(5): 593-595,599.
Zhang Chunkai, Xu Liyun, Shao Huihe. Improved chaos optimization algorithm and its application in nonlinear constraints optimization problems [J]. Journal of Shanghai Jiaotong University, 2000, 34(5): 593-595,599. (in Chinese)
- [10] MaCauley J L. Chaos, Dynamics and Fractals [M]. Cambridge University Press, 1993.
- [11] VR&D. DOT (Design Optimization Tools) Users Manual Version 4.20 [M]. Vanderplaats Research & Development Inc., Colorado, USA, 1995.