文章编号: 1001-0920(2004)12-1337-04

混沌梯度组合优化算法

胡志坤[1,2], 桂卫华[1], 彭小奇[2]
(1. 中南大学 信息科学与工程学院, 湖南 长沙 410083; 2. 中南大学 物理科学与技术学院, 湖南 长沙 410083)

摘 要: 提出一种混沌梯度组合全局优化算法, 并对该算法进行了收敛性分析. 算法首先采用改进的变步长梯度法得到某个优化值, 然后利用变尺度混沌搜索跳出局部极小, 经过反复组合迭代, 直至到达最优解. 仿真结果表明, 该算法能充分发挥梯度法寻优的快速性和混沌法寻优的全局搜索能力.

关键词: 混沌优化; 梯度搜索; 组合算法
中图分类号: TP18 **文献标识码**: A

Chaotic gradient combination optimization algorithm

HU Zhi-kun[1,2], *GUI Wei-hua*[1], *PENG Xiao-qi*[2]
(1. School of Information Science and Engineering, Central South University, Changsha 410083, China; 2. School of Physics Science and Technology, Central South University, Changsha 410083, China. Correspondent: HU Zhi-kun, E-mail: hzk@mail.csu.edu.cn)

Abstract: A coupled optimization algorithm combined gradient search with chaotic search is proposed and its convergence is discussed. By the algorithm, a local minimum is obtained by an improved mutative step gradient descending search. A more optimal minimum is obtained to replace the local minimum by mutative scale chaotic search. The global optimal value will be attained by iterating repeatedly. The simulation results show that the coupled algorithm is able to make full use of the quickness of gradient search and the ability of global optimization of chaotic search.

Key words: chaotic optimization; gradient search; coupled algorithm

1 引 言

梯度法具有寻优速度快的特点, 但可能出现局部极小, 特别是对于多极值目标函数, 如果初值选择不当, 搜索过程极易陷入局部极小. 模拟退火优化方法和遗传算法从本质上都是按某种概率接受较差解来跳出局部极小以实现全局优化, 但搜索速度较慢[1,2]. 混沌优化方法利用混沌运动具有的遍历性和随机性, 采用混沌变量在目标函数的定义域内进行搜索, 可实现全局优化[3]. 张彤等[4]采用逐步缩小尺度的变尺度混沌运动进行寻优, 得到最优值. 这两种方法都有全局寻优的能力, 但寻优速度较慢. 张春慨等[5]首先利用混沌运动搜索到一个优化值, 然后在该点的邻域进行线性搜索, 继续迭代, 直至得到最优值. 但如果搜索得到的是一个局部极小值, 那么最后得到的优化值并不能确保为全局最优.

为了快速得到全局优化解, 本文提出一种综合混沌遍历寻优的全局性和梯度下降寻优的快速性混沌梯度组合优化算法. 该算法首先采用一种改进的变步长梯度搜索算法得到一个优化值, 然后利用不断扩大尺度的变尺度混沌运动来跳出因梯度下降搜

收稿日期: 2003-10-22; **修回日期**: 2003-12-30.
基金项目: 国家 973 计划项目(2002cb312200); 国家自然科学基金资助项目(50374079); 教育部科技研究重点项目(02146); 湖南省自然科学基金资助项目(01JJY2110).
作者简介: 胡志坤(1976—), 男, 湖北鄂州人, 讲师, 博士生, 从事智能决策与优化算法的研究; 桂卫华(1950—), 男, 湖北襄樊人, 教授, 博士生导师, 从事复杂工业过程建模与优化控制等研究.

索导致的局部极小, 以减少混沌搜索的范围, 并得到一个新的优化值. 该过程反复进行, 直至按某种标准搜索到全局优化值. 因此, 本文算法既拥有梯度优化法的快速性, 又拥有混沌优化法的全局收敛性.

2 混沌梯度组合优化算法

考虑如下一类优化问题:

$$\min f(x), x = \{x_1, x_2, \cdots, x_n\}, \quad (1)$$

$$\text{s.t.} \quad a_i \le x_i \le b_i. \quad (2)$$

其中$[a_i, b_i]$为x_i的定义域.

2.1 改进的变步长梯度下降法

为保证快速进行优化搜索, 可采用如下自适应变步长梯度下降算法, 且梯度采用前向差分方法计算:

$$x^{(k+1)} = x^{(k)} - \lambda^{(k)} \frac{f(x^{(k)}) - f(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}. \quad (3)$$

其中: k 为迭代次数, 第 k 次迭代时的梯度向量采用差分法自适应求得, 故适合任意的函数形式; $\lambda^{(k)}$ 为迭代步长. 可以首先随机初始化一个步长, 仿照前馈神经网络基于梯度下降的权值学习BP 算法[6], 在搜索过程中令 $\lambda^{(k)} = \delta * \lambda^{(k)}$, 如果目标函数值在降低, 则 $\delta = 0.9$; 否则 $\delta = 1.1$. 这样可保证梯度下降能平稳进行, 不会发生太大的振荡.

设 k 是当前的步数, $e[k-m]$ 是前第 m 步的目标函数的差值, 令 $e[k-m] = f(x^{(k-m+1)}) - f(x^{(k-m)})$. 设 $\bar{e}$ 是前 C (C 为给定的正整数) 步的目标函数差值, 则梯度下降算法的收敛条件为

$$\bar{e} = \frac{1}{C}\sum_{m=0}^{C-1}(e[m-k+1] - e[m-k]) < \epsilon_1, \quad (4)$$

其中 ϵ_1 为事先给定的一个小正数.

为减少梯度下降法的搜索次数, 避免在优化点的邻域附近长期搜索, 首先利用如下规则来判断是否搜索到优化点:

规则 1: 如果满足式(4), 则表示该次已搜索到一个局部优化点 x^* 和对应的函数值 $f(x^*)$, 停止这次梯度下降搜索;

规则 2: 如果式(4) 不满足, 但搜索次数已大于规定的最大迭代步数, 则终止这次梯度下降搜索, 并记下当前的变量 x^* 和对应的函数值 $f(x^*)$.

然后, 采用基于变尺度混沌遍历跳出由规则 1 或规则 2 得到的局部极小值. 若无法获得比规则 1 获得的局部优化值更优的值, 则认为找到全局最优.

2.2 改进的变尺度混沌优化算法

采用梯度法搜索成功后, 无法判定该优化点是否为全局优化点, 因此必须采用一种全局优化搜索法来获得全局优化点. 本文利用混沌运动的遍历性和不重复搜索性, 以跳出局部极小.

混沌优化搜索主要是为了跳出因梯度下降搜索导致的局部极小, 因此为了避免在整个定义域空间内的大范围搜索而影响寻优速度, 本文提出以梯度下降搜索法所得优化点为起点, 先用小尺度搜索, 然后逐渐增大尺度, 即逐步扩大混沌搜索的空间, 直至在多峰函数中找到另一个更优解后, 即刻退出混沌搜索; 再以混沌搜索得到的新的优化点为起点, 用变步长梯度下降法继续搜索, 从而加快搜索速度, 以便更快地找到全局优化点. 若混沌搜索阶段未获得更优解, 则认为当前优化点即为全局优化点.

考虑Logistic 映射

$$Z_{k+1} = \mu Z_k(1 - Z_k), 0 \le Z_k \le 1. \quad (5)$$

其中: Z_k 为混沌变量, k ($k = 1, 2, \cdots$) 为迭代次数, μ 为控制变量. 当 $\mu = 4$ 时, Logistic 映射为[0, 1] 区间的满映射, 并且系统处于完全的混沌状态[4].

一般多峰(谷) 的函数都有若干个局部极小点. 为保证在进入局部极小点后能迅速跳出, 可在以局部极小为原点的同心圆上进行混沌遍历, 以寻求用最小的搜索代价跳出局部极小. 令 $k = 1, r = 0$, 综合考虑搜索的可行性和算法的收敛速度, 令尺度因子 $\eta = 0.05$, 保证算法最多会进行 10 次混沌遍历, r 为变尺度标志. 设自变量 x_i 的取值范围为$[a_i, b_i]$, 变尺度混沌优化算法如下:

1) 设变尺度梯度下降搜索获得的最优解为 x_i^*, 设置优化变量的取值范围为

$$a_{i+1}[r+1] = x_i^* - \eta(b_i[r] - a_i[r]), \quad (6)$$

$$b_{i+1}[r+1] = x_i^* + \eta(b_i[r] - a_i[r]). \quad (7)$$

当 $a_i[r] \le a_i$, 令 $a_i[r] = a_i$; 当 $b_i[r] \ge b_i$, 令 $b_i[r] = b_i$. 这意味着搜索范围从原范围的 10% 开始搜索.

2) 将 x_i^* 根据由1) 确定的变量范围线性映射到[0, 1] 之间, 按式(5) 得到 l 个混沌序列$(Z_1^{(k)}, Z_2^{(k)}, \cdots, Z_n^{(k)})$, $k = 1, 2, \cdots, l$.

3) 将 l 个混沌变量分别变换到原变量, 使其成为混沌变量

$$x_i^{(k)} = a_i[r] + (b_i[r] - a_i[r]) Z_i^{(k)}. \quad (8)$$

4) 设由梯度法获取的优化点为 x^*, 相应的目标函数值为 $f(x^*)$, 用变换后所得到的混沌变量 $x^{(k)}$ 进行搜索.

Step 1: 如果函数值 $f(x^{(k)}) < f(x^*)$, 则 $f(x^*) = f(x^{(k)})$, $x^* = x^{(k)}$, 终止本次搜索, 退出.

Step 2: 否则放弃 $x^{(k)}$, 并令 $k = k + 1$. 如果 $k <$

l, 则转向 3) 重新搜索; 否则转向 5).

5) 若本次搜索未能找到新的优化点, 用 $\eta+0.05$ 代替 η. 然后按式(6) 和式(7) 修改各变量的搜索范围, 获取新的混沌变量后, 按式(5) 获得新的混沌变量, 再从 2) 开始再次进行混沌搜索. 若经过 10 次变尺度搜索, 即 $\eta=0.5$ 或者 $f(x^*)$ 均保持不变, 优化过程结束.

于是, 混沌梯度组合优化算法如下:

1) 将目标函数的自变量随机初始化, 用改进的变步长梯度法寻找满足规则 1 或规则 2 的优化点 x^* 和 f^*.

2) 用变尺度混沌优化算法搜索 $f<f^*$ 的点. 若结束混沌遍历, 则执行 3); 否则若 η 为 1, 则搜索成功, 停止搜索, x^* 和 f^* 即为所求的优化变量和优化目标值.

3) 以新的优化点为起点, 用改进的变步长梯度法寻找新的优化点 x^* 和 f^*, 然后返回 2).

3　混沌梯度组合优化算法的收敛性分析

组合优化算法通过由小到大变尺度因子 η 来实现梯度下降优化和混沌遍历的组合. 当 $\eta=0$ 时, 组合算法就蜕变为一个变步长梯度下降优化算法; 当 $\eta=1$, 而又没有搜索到比梯度下降得到的局部优化值更优, 则表明该值是全局优化值. 这就是本算法具备的全局优化性.

设 $x^*=\arg\min f(x)$, $\{x_k'\}$ 是由混沌搜索保留下来的解序列, $\{x_k''\}$ 是组合优化算法产生的解序列. 显然, 对于 $i\le j$, 有 $f(x_i')\ge f(x_j')$ 和 $f(x_i'')\ge f(x_j'')$. 根据算法结构可知, $f(x_k')\ge f(x_k'')$. 文献[4] 已证明 $\{f(x_k')\}$ 是收敛序列, 且

$$\lim_{k\to\infty} P\{f(x_k')=f(x^*)\}=1. \tag{9}$$

根据收敛的夹逼定理可知, 因 $f(x_k')\ge f(x_k'')\ge f(x^*)$, $\{f(x_k')\}$ 是收敛序列, 故 $\{f(x_k'')\}$ 是收敛序列, 且

$$\lim_{k\to\infty} P\{f(x_k'')=f(x^*)\}=1. \tag{10}$$

故本文所提出的混沌梯度优化算法以概率 1 收敛到全局最优解.

4　仿真研究

不失一般性, 考虑如下多峰函数最小值问题:

$$f(x)=4+4.5x_1-4x_2+x_1^2+2x_2^2-2x_1x_2+x^4-2x_1^2x_2,$$
$$-8\le x_1, x_2\le 8. \tag{11}$$

该函数为一典型的多峰函数, 有一个全局最优解 $\{-1.053, 1.028\}$, 其目标函数值为 -0.5134. 该函数的一个局部极小解为 $\{1.941, 3.854\}$, 其函数值为 0.985 5. 对不同初值, 利用梯度下降法、混沌优化法以及变尺度混沌梯度下降优化法对该函数寻优进行仿真, 结果如表 1 所示.

从表 1 可见, 当初值接近最优解时, 梯度下降法具有明显的优势, 与组合优化算法的计算时间相当. 但当初值离最优解相对远时, 梯度下降法就易进入局部极小或出现振荡等现象; 当初值离局部极小值很近时, 进入局部极小就在所难免了. 而混沌优化法利用混沌自身的遍历性, 可以跳出局部极小, 使得组合算法由于拥有梯度下降法的快速性和混沌优化算法的全局优化能力而独具优势.

为进一步验证方法的有效性, 考虑如下函数的全局最小优化问题:

$$f(x)=\frac{1}{100}\sum_{i=1}^{100}(x_i^4-16x_i^2+5x_i),$$

表 1　几种优化算法的仿真结果比较

序号	初值	目标值		找到全局优化值的时间	
		变步长梯度下降法	组合优化算法	变步长梯度下降法	组合优化算法
1	(0, 1)	(-1.052 7, 1.027 6)	(-1.052 7, 1.027 6)	2 s	2 s
2	(3, 3)	(-1.052 7, 1.027 6)	(-1.052 7, 1.027 6)	4 s	2 s
3	(2, 4)	(1.940 9, 3.854 1)	(-1.052 7, 1.027 6)	> 2 min	10 s
4	(-0.2, 2.5)	(-1.052 7, 1.027 6)	(-1.052 7, 1.027 6)	3 s	2 s
5	(2.5, 3)	(1.940 9, 3.854 1)	(-1.052 7, 1.027 6)	> 2 min	10 s

表 2　4 种优化方法求解的计算结果

算法类型	外循环次数最小值	外循环次数平均值	外循环次数最大值
变步长梯度下降优化算法	18 760	29 542	37 553
文献[3] 方法	1 197	5 983	8 892
组合优化算法	536	2 246	5 387

$$-10 \le x_i \le 100,\ i = 1, 2, \cdots, 100.$$

函数 $f(x)$ 在可行域内有 2^{100} 个局部最优解, 全局最优解的目标值为 -78.332 3. 对初值进行随机赋值, 进行 10 次仿真, 用优化搜索循环中的外循环次数来衡量优化搜索的速度, 则仿真结果如表 2 所示.

5 结 论

1) 变步长梯度下降法结合优化点判断规则, 能有效地避免寻优过程在优化点附近出现振荡现象并加快寻优速度.

2) 变尺度混沌优化算法能有效地逃离局部极小并实现全局优化搜索.

3) 仿真结果表明, 混沌梯度组合优化算法既具有快速性, 又具有全局收敛性, 是一种行之有效的优化算法.

参考文献(References):

[1] Ingber L. Simulated annealing: Practice versus theory [J]. *Mathematical and Computer Modeling*, 1993, 18(1): 29-57.

[2] Rudolph G. Convergence properties of control parameters for genetic algorithms[J]. *IEEE Trans on Neutral Networks*, 1994, 5(1): 65-67.

[3] 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): 163-165.
(Li B, Jiang W S. Chaos optimization method and its application [J]. *Control Theory and Application*, 1997, 14(4): 163-165.)

[4] 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 T, Wang H W, Wang Z C. Mutative scale chaos optimization algorithm and its application [J]. *Control and Decision*, 1999, 14(3): 285-288.)

[5] 张春慨, 李霄峰, 邵惠鹤. 基于线性搜索的混沌优化及其在非线性约束优化问题中的应用[J]. 控制与决策, 2001, 16(1): 123-128.
(Zhang C K, Li X F, Shao H H. Chaos optimization algorithm based on linear search and its application to nonlinear constraint optimization problems[J]. *Control and Decision*, 2001, 16(1): 123-128.)

[6] 史天运, 贾利民. 基于进化策略的动态递归神经网络建模与辨识[J]. 控制与决策, 2000, 15(4): 439-442.
(Shi T Y, Jia L M. Evolutionary strategy based on dynamic recursive neural network modeling and identification[J]. *Control and Decision*, 2000, 15(4): 439-442.)

(上接第 1326 页)

[48] Bullnheimer B, Hartl R F, Strauss C. Applying the ant system to the vehicle routing problem [A]. *Meta-Heuris-tics: Advances and Trends in Local Search Paradigms for Optimization* [C]. Kluwer, Boston, 1998. 109-120.

[49] 马良, 姚俭, 范炳全. 蚂蚁算法在交通配流中的应用[J]. 科技通报, 2003, 19(5): 377-380.
(Ma L, Yao J, Fan B Q. The application of ant algorithm in traffic assignment[J]. *Bulletin of Science and Technology*, 2003, 19(5): 377-380.)

[50] Ismail E, Otman A B, Paul C. An experimental study of a simple ant colony system for the vehicle routing problem with time windows [A]. *Proc of 3rd Int Workshop ANTS* [C]. Brussels, 2002. 100-110.

[51] Tsai C F, Wu H C, Tsai C W. A new data clustering approach for data mining in large databases[A]. *Proc of the Int Symposium on Parallel Architectures, Algorithms and Networks*[C]. Makati, 2002. 315-321.

[52] Abbaspour K C, Schulin R, Genuchten M T V. Estimating unsaturated soil hydraulic parameters using ant colony optimization[J]. *Advances in Water Resources*, 2001, 24(8): 827-841.

[53] Salima Q, Mohamed B, Catherine G. Ant colony system for imagine segmentation using markov random field[A]. *Proc of 3rd Int Workshop ANTS* [C]. Brussels, 2002. 294-295.

[54] Costa D, Hertz A. Ants can colour graphs[J]. *J of the Operational Research Society*, 1997, 48(3): 295-305.

[55] Ding Y P, Wu Q S, Su Q D. Ant colony algorithm and optimization of test conditions in analytical chemistry [J]. *Chinese J of Chemistry*, 2003, 21(6): 607-609.

[56] 王成华, 夏绪勇, 李广信. 基于应力场的土坡临界滑动面的蚂蚁算法搜索技术[J]. 岩石力学与工程学报, 2003, 22(5): 813-819.
(Wang C H, Xia X Y, Li G X. Ant algorithm in search of the critical slip surface in soil slops based stress fields[J]. *Chinese J of Rock Mechanics and Engineering*, 2003, 22(5): 813-819.)

[57] Shin Ando, Hitoshi Iba. Ant algorithm for construction of evolutionary tree[A]. *Proc of the Genetic and Evolutionary Computation Conference*[C]. New York, 2002. 1552-1557.