

求解约束优化问题的混合粒子群算法

裴胜玉, 周永权, 罗淇方

(广西民族大学 数学与计算机科学学院, 广西 南宁 530006)

摘 要: 针对约束优化问题提出一种混合粒子群求解算法, 该算法根据可行性规则, 引入自适应惩罚函数, 结合模拟退火算法, 不断地寻找更优可行解, 逐渐达到搜索全局最优解. 通过对一些标准函数测试, 计算机仿真结果表明, 该方法是有效和可行的, 且具有较高的计算精度, 相比传统算法, 最优解精度达到 10^{-15} .

关键词: 粒子群; 约束优化; 可行性规则; 模拟退火; 自适应惩罚函数

中图分类号: TP18

文献标识码: A

文章编号: 1000-7180(2010)04-0005-04

Hybrid Particle Swarm Algorithm for Solving Constrained Optimization

PEI Sheng-yu, ZHOU Yong-quan, LUO Qi-fang

(College of Mathematics and Computer Science, Guangxi University for Nationalities, Nanning 530006, China)

Abstract: This paper presents a hybrid particle swarm optimization for solving constrained optimization problems, which is based on constraint-handling mechanism, SA(Simulated Annealing) and dynamically changing penalty function. And can continuously find better feasible solutions, gradually leading the search near the true optimum solution. This constraint handling approach have been tested on some problems commonly used in the literature. In all cases, our results show that the proposed approach is an efficient and can reach a higher precision, the precision is more than 10^{-15} .

Key words: particle swarm optimization; constrained optimization; feasibility rules; simulated annealing; dynamically changing penalty function

1 引言

粒子群优化(PSO)算法具有计算简单、收敛性良好等优点,已在各类连续空间优化问题、神经网络训练和模糊系统控制等应用领域中均取得了良好的效果.但基于 PSO 的约束优化问题还研究得较少,文中结合模拟退火算法的特点,提出了一种基于 PSO 的混合优化算法求解约束优化问题的粒子群算法,该算法利用了基于可行性的更新规则,加入自适应惩罚函数,不断地寻找更优可行解,逐渐达到搜索全局最优解.通过对一些标准函数测试,计算机仿真实验结果表明该方法有效可行,且具有较高的计算精度^[1].

2 约束优化问题

不失一般性,一个非线性规划问题可描述为

$$\text{minimize } f(\mathbf{x}), \mathbf{x} = (x_1, x_2, \dots, x_n) \in R^n$$

(1)

subject to $g_j(\mathbf{x}) \leq 0, j = 1, 2, \dots, l.$

$h_j(\mathbf{x}) = 0, j = L + 1, \dots, p.$

式中, $\mathbf{x} \in \Omega \subseteq S$ 为决策向量, Ω 为可行域, S 为决策空间.一般地, S 为 R^n 中的 n 维长方: $l(i) \leq x_i \leq u(i), l(i), u(i)$ 为常数, $i = 1, 2, \dots, n$. $f(\mathbf{x}), g_j(\mathbf{x}), h_j(\mathbf{x})$ 均为 R^n 上的 n 元函数, $f(\mathbf{x})$ 为目标函数, $g_j(\mathbf{x}) \leq 0$ 为第 j 个不等式约束条件, $h_j(\mathbf{x}) = 0, j = l + 1, \dots, p$ 为第 j 个等式约束条件. l 表示不

收稿日期: 2009-03-13; 修回日期: 2009-05-13

基金项目: 国家自然科学基金项目(60461001); 广西自然科学基金项目(0832082; 0991086); 国家民委科研基金项目(08GX01); 广西民族大学科研项目启动基金资助项目

等式约束条件的个数, $p-l$ 表示等式约束条件的个数. 这里, 通常将等式约束条件改写为两个不等式约束条件.

3 基于可行性的约束处理方法

文中采用区别于单纯罚函数的做法, 运用可行性规则将目标和约束分开处理. 其中可行性规则规定: (1) 可行解总是优于不可行解; (2) 若两个解均可行, 则目标值好的解为优; (3) 若两个解均不可行, 则约束违反量小的解为优. 其主要用于指导解的比较, 规则(1)和(3)指导算法向约束违反量小的方向搜索, 规则(2)指导算法在可行域内向目标值好的方向搜索. 下面为一个适应度函数^[2-4]:

$$F(\mathbf{x}) = \begin{cases} f(\mathbf{x}), & \text{if } g_j(\mathbf{x}) > 0, \forall j = 1, 2, \dots, m \\ f_{\max} + \sum_{j=1}^m \max[g_j(\mathbf{x}), 0] & \text{otherwise} \end{cases} \quad (2)$$

式中, $f_{\max}$ 为当前代对应可行解的最大目标函数值. 因此, 不可行解的适应度值不仅受约束违反量影响, 还由当前代种群决定.

3.1 粒子群优化算法

粒子群算法模拟鸟集群飞行觅食的行为, 通过鸟之间的集体协作使群体达到目的. 相对于其他演化方法, 它具有原理简单, 易于编程实现等特点. PSO 数学描述如下:

初始化一群随机粒子(随机解). 然后通过迭代找到最优解. 在每一次的迭代中, 粒子通过跟踪两个“极值”(pbest, gbest) 来更新自己. 在找到这两个最优值后, 粒子通过下面的公式来更新自己的速度和位置^[5].

$$\begin{aligned} V_{i,j}(t+1) &= wV_{i,j}(t) + c_1 \times \text{rand}() \times (\text{pbest}_{i,j} - x_{i,j}(t)) + c_2 \times \text{rand}() \times (\text{gbest}_{i,j} - x_{i,j}(t)) \\ x_{i,j}(t+1) &= x_{i,j}(t) + V_{i,j}(t+1), j = 1, 2, \dots, d \end{aligned} \quad (3)$$

$$(4)$$

在式(3)、式(4)中, $i = 1, 2, \dots, M$. M 是该群体中粒子的总数, V_i 是粒子的速度; pbest 和 gbest 分别定义为粒子最佳位置和种群最佳位置. rand() 是介于 (0, 1) 之间的随机数; x_i 是粒子的当前位置. c_1 和 c_2 是学习因子, 通常取 $c_1 = c_2 = 2$, w 为惯性权因子.

3.2 模拟退火算法

模拟退火算法主要用于概率突跳, 有效避免搜索过程陷入局部极小. SA 算法在搜索过程中不但绝对接受好的解, 同时也允许以一定概率接受差的解.

同时, 这种突跳概率受到温度参数的控制, 即概率大小随着温度的下降而减小, 当温度趋于零时概率值也将趋于零. 理论上已证明 SA 算法在一定条件下以概率 1 收敛到全局最优解.

3.3 基于混合 PSO 的约束优化算法

采取可行性规则处理约束问题, 对于不可行解, 适当应用罚函数确定其特殊适应度值. 其中适应度函数定义如下:

$$\text{penalty}(x) = \sum_{i=1}^{m_1} K_i \cdot |h_i(x)| + \sum_{j=1}^{m_2} M_j \cdot \max\{0, g_j(x)\} \quad (5)$$

式中, $h_i(x)$, $g_j(x)$, 分别为等式约束和不等式约束条件.

在 PSO 算法中, 合理控制与平衡算法的全局搜索和局部改良能力, 对于有效找到问题的最优解至关重要. 在微粒群算法的可调整参数中, 惯性权重 w 是最重要的参数, 较大的 w 有利于提高算法的全局搜索能力, 而较小的 w 会增强算法的局部搜索能力. 算法采用了文献[6]中公式: $w = w_{\max} - \text{iter} \times \frac{w_{\max} - w_{\min}}{\text{iter}_{\max}}$. 其中 iter 为当前迭代次数, $\text{iter}_{\max}$ 为

最大的迭代次数. 文中采用结合模拟退火算法的局部搜索技术来避免算法早熟收敛及合粒子群算法的特点, 提出了一种求解约束优化问题的混合粒子群算法, 混合粒子群优化算法的流程描述如下:

Step 1: 按搜索范围初始化种群粒子, 令 $k = 0$, $t(k) = t_0$;

Step 2: 判断粒子可行性, 通过自适应罚函数计算适应度值;

Step 3: 确定各粒子最优位置和种群最佳位置;

Step 4: 满足条件, 输出种群最佳位置, 否则运行 Step 5;

Step 5: 计算: $w = w_{\max} - \text{iter} \times \frac{w_{\max} - w_{\min}}{\text{iter}_{\max}}$, 更新种群粒子位置和速度;

Step 6: 判断粒子可行性, 通过自适应罚函数计算适应度值;

Step 7: 确定更新各粒子最优位置和种群最佳位置;

Step 8: 对种群最佳位置应用基于可行性规则和 SA 的局部搜索技术.

(1) 令 $m = 1$, $p_g = p'_g$.

(2) 采用式(6)产生一个新解

$$x' = p'_g + \eta * (x_{\max} - x_{\min}) * N(0, 1) \quad (6)$$

式中, η 为搜索步长, $N(0,1)$ 为服从标准高斯分布随机数, $x_{\max}$ 和 $x_{\min}$ 为变量的上下界.

(3) 若 x' 可行, p'_g 不可行, 则 $p_a = 1$;

若 x' 不可行, p'_g 可行, 则 $p_a = 0$;

若 x', p'_g 均可行, 则

$$p_a = \min\{1, \exp[f(p'_g) - f(x')]/T(k)\};$$

若 x', p'_g 均不可行, 则

$$p_a = \min\{1, \exp[(\text{viol}(p'_g) - \text{viol}(x'))/T(k)]\}$$

(4) 随机产生服从 $(0,1)$ 均匀分布随机数, 若 $p_a \geq U(0,1)$, 则 $p'_g = x'$;

(5) 令 $m = m + 1$ 若条件满足, 则输出 p'_g 作为种群新的最佳位置, 否则返回(2).

4 仿真结果和性能分析

4.1 算法参数的设置

在此选用标准函数和工程设计问题测试算法的性能, 算法在 Genuine1.86GHz(内存1GB) 兼容机上采用统一的参数对每一问题均独立运行 30 次. 种群规模 $N_{\text{pop}} = 250$, 最大迭代次数 $\text{iter}_{\max} = 300$, 惯性权重 w 从 0.9 线性减小至 0.4, 局部搜索步数 $L = 20$, 局部搜索步长 $\lambda = 0.94$, 退温速度 $\eta = 0.001$.

4.2 标准函数测试

为检验算法的有效性, 对以下几个标准函数进行测试.

(1) 测试函数

$$\min f(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$$

$$\text{s.t. } g_1(x) = 4.84 - x_1^2 - (x_2 - 2.5)^2 \leq 0$$

$$g_2(x) = (x_1 - 0.05)^2 + (x_2 - 2.5)^2 - 4.84 \leq 0$$

$$0 \leq x_1 \leq 6, 0 \leq x_2 \leq 6$$

算法求解结果:

$$x = (2.243720637267753,$$

$$2.344712837533228),$$

$$\min f(x) = 13.661348597844016.$$

约束条件 g_2 是活跃的, 可行区域约占搜索的 0.7%.

(2) 测试函数

$$\min f(x) = \frac{\sin^3(2\pi x_1)\sin(2\pi x_2)}{x_1^3(x_1 + x_2)}$$

$$\text{s.t. } g_1(x) = x_1^2 - x_2 + 1 \leq 0$$

$$g_2(x) = 1 - x_1 + (x_2 - 4)^2 \leq 0$$

$$0 \leq x_1 \leq 10, 0 \leq x_2 \leq 10$$

算法求得最优解 $x = (1.224305673597622, 4.217124005454183)$, 最优函数值 $f(x) = 0.094239287843330$. 优于文献 [7], $x^* = (1.2279713, 4.2453733)$, 最优值为 $f(x^*) = 0.095825$.

(3) 测试函数

$$f(x) = (x_1 - 2)^2 + (x_2 - 1)^2$$

$$\text{s.t. } x_1 - 2x_2 + 1 = 0$$

$$-\frac{x_1^2}{4} - x_2^2 + 1 \geq 0$$

$$0 \leq x_1 \leq 10, 0 \leq x_2 \leq 10$$

算法求得最优解: $x = (0.822889944855232, 0.911439729869010)$, 最优函数值 $f(x) = 1.393431003368593$. 优于文献 [8], $f(x^*) = 1.393608$. 表 1 为解的比较.

表 1 测试函数(3) 与传统方法结果的比较

	文中算法	最好解			
		自适应 乘子法	Gen	Horniafar	GRG
x_1	0.822889944855232	0.8228	0.8080	0.8102	0.8229
x_2	0.911439729869010	0.9112	0.88544	0.9026	0.9115
$f(x)$	1.393431003368593	1.3937	1.4339	1.4251	1.3934

(4) 测试函数

$$f(x) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.29329x_1 - 40792.141$$

$$g_1(x) = 85.334407 + 0.0056858x_2x_5 + 0.00026x_1x_4 - 0.0022053x_3x_5$$

$$g_2(x) = 80.51429 + 0.0071317x_2x_3 + 0.0029955x_1x_2 + 0.00218x_3^2$$

$$g_3(x) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 - 0.0019085x_3x_4$$

$$78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45, 27 \leq x_{3,4,5} \leq 45$$

$$0 \leq g_1(x) \leq 92, 90 \leq g_2(x) \leq 110, 20 \leq g_3(x) \leq 25$$

算法求得最优解结果优于文献 [8] 结果 - 31025.1412. 表 2 为解的比较.

从表中可以看出该算法在处理约束问题时, 使用可行性规则, 保持粒子的多样性, 收敛速度较快, 算法采用双精度计算, 与传统算法比较发现, 最优解精度达到 10^{-15} , 比传统算法最优解精度 10^{-7} 高出许多.

表 2 测试函数(4) 与传统方法结果的比较

	文中算法	最好解			
		自适应 乘法子法	Gen	Homaifar	GRG
x_1	78.08228	78.008	81.490	78.000	78.620
x_2	33.031439814512858	33.011	34.090	33.000	34.440
x_3	27.071944261449740	27.109	31.240	29.995	31.070
x_4	44.893914606454146	45.000	42.200	45.000	44.180
x_5	44.879107075616076	44.822	44.822	34.370	35.220
$f(x)$	- 3.103115687213954e + 004	- 31025.1412	- 30183.586	- 30665.609	- 30373.949

5 结束语

文中根据可行性规则求解约束优化的方法出发,利用自适应惯性权因子,适当应用罚函数确定其特殊适应度值,结合模拟退火的局部搜索算法,提出一种改进的混合粒子群算法来解决约束优化问题.从实验结果可以看出,文中算法有效可行,计算精度较高且解的稳定性较好,为解决复杂约束函数优化问题提供了一种行之有效的方法.但对算法的收敛性理论证明有待于进一步研究.

参考文献:

[1] Eberhart R C, Kennedy J. A new optimizer using particles swarm theory[C]// Proc. Sixth International Symposium on Micro Machine and Human Science. Japan, Nagoya, 1995:30-43.

[2] 王凌,刘波.微粒群优化与调度算法[M].北京:清华大学出版社,2008:59-93.

[3] 胡宏梅,董恩清.基于粒子群算法的码书设计研究[J].

微电子学与计算机,2009,26(1):97-100.

[4] 周鲜成,申群态,王俊年.基于微粒群的颜色量化算法[J].微电子学与计算机,2008,25(3):51-34.

[5] Deb K. An efficient constraint handing method for genetic algorithms[J]. Computer Methods in Applied Mechanics and Engineering, 2000(186):311-338.

[6] Runarsson T P, Yao X. Stochastic ranking for constrained evolutionary optimization[J]. IEEE Trans. on Evolutionary Computation, 2000(4):284-294.

[7] 谢柏桥,戴光明,石红玉.一种改进的求解约束函数优化问题的演化算法[J].计算机应用与软件,2008,25(7):48-50.

[8] 李炳宇,萧蕴诗,汪镭.PSO算法在工程优化问题中的应用[J].计算机工程与应用,2004,40(18):74-76.

作者简介:

裴胜玉 男,(1986-),硕士.研究方向为计算智能及其应用.

周永权 男,(1962-),博士,教授.研究方向为计算智能、神经网络及应用.

罗淇方 (1976-),工程师.研究方向为并行计算.

(上接第 4 页)

[6] Le Changyong, Yao Xin. Evolutionary programming using mutations based on the levy probability distribution[J]. IEEE Transactions on evolutionary computation, 2004, 8 (1):1-13.

作者简介:

罗 梅 女,(1974-),硕士,讲师.研究方向为软件工程、计算机算法及应用.

贾志娟 女,(1973-),硕士,副教授.研究方向为智能算法及信息挖掘.