



SAGACIA 全局优化方法及应用¹⁾

李 兵 蒋慰孙

(华东理工大学自动化研究所 上海 2000237)

关键词 模拟退火法, 遗传算法, 趋化性方法, 优化

1 引言

模拟退火法(Simulated Annealing Algorithm, 简称 SAA), 遗传算法(Genetic Algorithm, 简称 GA)和趋化性算法(Chemotaxis Algorithm, 简称 CA)都是模拟自然界中的某些客观规律得出的优化算法, 能较好地解决一些传统优化技术难以解决的优化问题, 近年来得到了许多的关注和发展.

SAA 在搜索过程中, 随机地接受某些劣化解, 因而具有较强的全局搜索能力, 能够跳出局部极小点, 但算法参数的选择比较困难, 所得解的质量与搜索时间之间存在着矛盾. GA 是一种群体优化算法, 通过选择、交叉、变异等操作使解群性能不断得到进化. GA 比 SAA 具有更强的全局搜索能力, 但应用过程中往往需要编码和解码, 还会遇到进化缓慢、过早收敛等问题. CA 结构简单, 搜索过程中只向使解性能变好的方向搜索, 全局搜索能力较差, 但收敛速度较快.

SAA, GA, CA 各有所长, 目前已出现了一些 SAA 和 GA 的混合算法(文献[1, 2]等), 为进一步提高算法的效率, 本文提出了一种新的 SAA、GA 和 CA 集成的优化方法, 简称 SAGACIA (SA——SAA, GA——GA, C——CA, I——Integrated, A——Algorithm).

2 SAGACIA 的结构及特点

从本质上说, 随机优化方法在搜索过程中存在两种搜索方式: “确定性”搜索和“盲目性”搜索. 如在搜索过程中主要沿着使性能变好的方向进行搜索, 这种搜索就具有某种程度的确定性, 而搜索过程中又以一定的方式随机的接受一些使解的性能变差的方向, 这种搜索又具有一定的盲目性. “确定性”搜索可以使算法收敛于性能优良的解, 而“盲目性”搜索则使算法能够跳出局部极小. 如果这两种搜索方式配合得当, 就能够使算法具有良好的

1) 国家自然科学基金重点项目.

性能.

2.1 SAGACIA 的主要步骤

设优化问题为

$$\begin{aligned} \min C(\mathbf{x}) \\ \text{s. t. } \mathbf{x} \in S, \end{aligned} \quad (1)$$

其中 S 为所有可能解构成的解集, $C(\mathbf{x})$ 为目标函数, 对任意 $\mathbf{x} \in S$ 均有一个确定的目标值 $C(\mathbf{x})$. \mathbf{x}^* 表示当前最优解. SAGACIA 的主要步骤如下:

步骤1. 随机产生 n 个初始解 $\mathbf{x}_n \in S$ 作为当前解群. 给定各参数初值;

步骤2. “粗搜索”: 在每个当前解的邻域内随机产生一个新解 $\mathbf{x}'_n \in S$, 若 $c(\mathbf{x}'_n) < c(\mathbf{x}_n)$, 则接受 \mathbf{x}'_n 为当前解; 否则以一定的概率 P_c 接受 \mathbf{x}'_n 为当前解;

步骤3. “细搜索”: 将当前解群中性能最好的解保存于 \mathbf{x}^* 中, 并在 \mathbf{x}^* 的邻域内随机产生 m 个新解 $\mathbf{x}_m \in S$, 若 $c(\mathbf{x}_m) < c(\mathbf{x}^*)$, 则 $\mathbf{x}^* = \mathbf{x}_m$.

步骤4. 从当前解群中按扰动选择率 P , 随机选出某些解, 按一定的突变扰动率 P_m 对其进行突变扰动后, 代替原来的解;

步骤5. $k = k + 1$, 如果 \mathbf{x}^* 满足要求则终止搜索, 否则转步骤2.

算法中邻域的定义如下:

连续变量 \mathbf{x}^* ($\mathbf{x}^* \in S$) 的 δ 邻域为

$$N_\delta(\mathbf{x}^*) = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}^*\| < \delta, \delta > 0\}, \quad (2)$$

离散变量 $\mathbf{x}^* = d_m d_{m-1} \cdots d_1$ 的 δ 邻域为

$$N_\delta(\mathbf{x}^*) = \{\mathbf{x} \mid \left| \sum_{i=1}^m (d_i - d_i^*) \right| < \delta, \delta > 0\}, \quad (3)$$

其中

$$d_i - d_i^* = \begin{cases} 1 & d_i \neq d_i^*, \\ 0 & d_i = d_i^*. \end{cases} \quad (4)$$

2.2 SAGACIA 的特点

1) SAGACIA 是一种群体优化算法, 算法具有并行性;

2) 步骤2和步骤4都有可能向使解的性能变坏的方向搜索, 因此, 算法有很强的全局搜索能力, 搜索过程能够比较容易地跳出局部极小点. 特别是突变扰动的合理运用, 不仅能使算法搜索到全局最优点, 而且还可以大大提高搜索速度;

3) 步骤3加强了性能优良的解周围的局域搜索, 提高了算法的局部搜索能力, 将步骤2作为“粗”搜索, 步骤3作为“细”搜索, 可以大大提高搜索效率;

4) SAGACIA 既适用于离散对象的优化问题, 也适用于连续对象的优化问题, 使用方便.

2.3 主要参数的选择

接受概率 P_c 一般与 $c(\mathbf{x}'_n) - c(\mathbf{x}_n)$ 及迭代次数有关. 搜索开始时 P_c 较大, 随着迭代的进行逐渐下降为零.

扰动选择率 P , 一定时, 突变扰动率 P_m 的大小对算法效果影响很大, P_m 不能太小也不能太大, 太小了会使搜索陷入局部最优值, 太大了又会使算法蜕化为纯随机搜索. 一般 P_m 取 0.1—0.3 左右即可, 也可根据具体情况稍作调整.

3 应用举例

生产调度问题

已知 UIS(10,6) 的 Flowshop 调度问题^[2], 以 Makespan 最小为目标对其进行优化调度, 结果如表1所示.

表 1 优化调度结果的比较

优化方法	搜索到全局最优解的百分率(%)	搜索到全局最优解时的平均迭代次数	搜索到全局最优解时的解空间搜索率(%)	未找到全局最优解时的平均相对误差(%)
SAGACIA	100	275	0.08	0
SAA	83	1 088	0.32	0.7
IGA	58	183	0.35	1.12

单凭迭代次数并不能准确地反映算法的搜索速度, 还要看其解空间搜索率(或搜索过的可行解的个数). 用 GA 对本问题进行优化时, 出现了严重的过早收敛现象. 因此使用了改进遗传算法(IGA)^[3], 从优化结果可以看出, SAGACIA 的优化速度及精度都高于 SAA 和 GA, 也优于文献[2]中的混合算法.

4 结论

综上所述, SAGACIA 吸收了 SAA、GA 和 CA 的优点, 既有很强的全局搜索能力, 又有较强的局部搜索能力, 搜索速度快、精度高、实施方便, 具有明显的优越性. 此外, SAGACIA 还被应用于解决神经网络训练、模糊控制、数据校正等问题, 均取得了令人满意的效果.

参 考 文 献

- 1 Lin F T *et al.* Applying the genetic approach to simulated annealing in solving some NP-hard problems. *IEEE Transactions Systems, Man, and Cybernetics*, 1993, **23**(6):1752-1767
- 2 田澎. 离散最优化的 Darwin & Boltzman 混合策略[博士学位论文]. 沈阳: 东北大学, 理论及应用, 1993
- 3 李兵, 蒋慰孙. 改进的遗传算法及在一类组合优化问题中的应用. 中国控制与决策学术年会论文集. 沈阳: 东北大学出版社, 1995, 360-362
- 4 Das H, Cumming P T *et al.* Scheduling multiproduct batch process via simulated annealing. *Computers Chem. Engng.*, 1990, **14**(12):1352-1362

SAGACIA OPTIMIZATION ALGORITHM AND ITS APPLICATION

LI BING JIANG WEISUN

(Res. Ins. of Auto. Contr., East China Univ. of Sci. & Tech., Shanghai 200237)

Key words Simulated annealing algorithm, genetic algorithm, chamotaxis algorithm, optimization