

基于模拟退火的混合遗传算法研究*

周 丽¹, 黄素珍²

(1. 温州师范学院 数学与信息科学学院, 浙江 温州 325035; 2. 温州师范学院 计算机科学系, 浙江 温州 325027)

摘要: 针对常规遗传算法会出现早熟现象、局部寻优能力较差等不足, 在遗传算法运行中融入模拟退火算法算子, 实现了模拟退火的良好局部搜索能力与遗传算法的全局搜索能力的结合。经验证, 该混合算法可以显著提高遗传算法的运行效率和优化性能。

关键词: 遗传算法; 模拟退火; 混合算法; 非线性约束

中图分类号: TP302.7 **文献标识码:** A **文章编号:** 1001-3695(2005)09-0072-02

Study of Hybrid Genetic Algorithm Based on Simulated Annealing

ZHOU Li¹, HUANG Su-zhen²

(1. School of Mathematics & Information Science, Wenzhou Normal College, Wenzhou Zhejiang 325035, China; 2. Dept. of Computer Science, Wenzhou Normal College, Wenzhou Zhejiang 325027, China)

Abstract: Taking a modified Simulated Annealing algorithm as a genetic operator realized the combination of the local searching ability of SA and global searching ability of GA. A new hybrid algorithm of Genetic Simulated Annealing had been designed with dynamic probability of crossover and mutation, and tested by a nonlinear function optimization. The results indicated the hybrid algorithm can improve significantly the efficiency of GA for solving nonlinear optimization.

Key words: Genetic Algorithm(GA); Simulated Annealing(SA); Hybrid Algorithm; Nonlinear Optimization

遗传算法(Genetic Algorithm, GA)^[1]因其高度的并行处理能力、强鲁棒性和全局搜索能力而被广泛地应用于诸多领域。理论上遗传算法依“概率 1”收敛于问题的最优解, 然而实践应用中, 遗传算法会表现出早熟现象、局部寻优能力较差等不足。所以一些常规遗传算法并不一定是针对某一问题的最佳求解方法^[2]。

模拟退火(Simulated Annealing, SA)具有较强的局部寻优能力, 并能使搜索过程避免陷入局部最优解, 但把握整个搜索过程的能力不够, 不便于使搜索过程进入最有希望的搜索区域, 从而使得模拟退火的运算效率不高^[1]。为了提高 GA 的优化性能和运行效率, 本文利用模拟退火算法与遗传算法的优势互补, 提出基于模拟退火的混合遗传算法, 并用一个非线性优化实例进行验证。

1 模拟退火混合遗传算法(SHGA)

1.1 SHGA 的实现思想

模拟退火是一种基于热力学的退火机理而建立的随机搜索算法。SA 使用基于概率的双方向随机搜索技术, 当基于邻域的一次操作使当前解的质量提高时, SA 接收这个被改进的解, 作为新的当前解; 在相反的情况下, SA 以一定的概率接收相对当前解来说质量较差的解, 作为新的当前解。SA 和 GA 都是概率搜索算法, 两者的优缺点正好互补。如果将两者相结合, 互相取长补短, 可开发出性能良好的新的全局搜索算法。

GA 与 SA 混合算法的实现有两种思路: 一种是在 GA 的运行中融入 SA 的思想, 称为混合遗传算法; 另一种是在 SA 中融入 GA 的思想, 称为混合模拟退火算法。

本文提出的模拟退火混合遗传算法是将常规的模拟退火算法改良后, 作为遗传算法的一个独立的算子, 置于遗传算法进化过程中。其过程是随机产生一组初始群体, 先通过选择、交叉、变异等遗传操作来产生一组新的个体, 然后再独立对每个新个体进行模拟退火操作, 其结果作为下一代群体, 如此反复迭代进行, 直到满足某个终止条件为止。

1.2 模拟退火算子

为利于混合算法的实现, 本文将 GA 的当前进化代数作为 SA 的退火时间。根据文献[3]中提出的模拟退火温度更新函数的启发式准则, 从而确定了 SA 的温度更新函数和随机向量的产生方式。模拟退火操作具体步骤如下:

(1) 设模拟退火初始温度为 T_0 , 遗传算法当前群体中的个体 i 记为 $X^i = (X_1^i, \dots, X_j^i, \dots, X_l^i)$, l 为个体编码串的长度。当前进化代数 t 对应的温度更新函数为 T_e :

$$T_e = T_0 / t^\theta \quad (1)$$

式(1)中, $\theta \geq 1$ 为给定常数。

(2) 通过下式产生随机向量:

$$Z^i = (Z_1^i, \dots, Z_j^i, \dots, Z_l^i) \\ Z_j^i = \text{sig } n(r_1) \cdot T_e \cdot (|r_1|^{-\theta} - 1), j=1, 2, \dots, l \quad (2)$$

式(2)中, r_1 为 $(-1, 1)$ 上均匀分布的随机数, $\text{sig } n(r_1)$ 为符号函数。

(3) 在 X^i 的基础上产生一个新的试探解 $Y^i = X^i + Z^i$, 并计算其对应的适应度 $F(Y^i)$ 和新试探解被接受的概率:

$$P_a = \min \left\{ 1, \exp \left(\frac{f(X') - f(Y)}{\psi \cdot T_c} \right) \right\} \tag{3}$$

式(3)中,ψ是对目标函数进行适当比例变换的常数。它适合于目标极小化。

(4)产生一个(0,1)上的均匀分布的随机数r₂,若r₂≤P_a则接受新试探解,即置X'=Y'。对种群中的个体X'(i=1,2,...,M)均进行式(3)、式(4)的操作。

1.3 模拟退火混合遗传算法设计

(1)染色体的编码。常用的编码方法是二进制编码和实数编码。二进制编码影响算法运行效率,且不利于混合算法的实现。所以本文对优化问题中的每个变量采用实数编码,即将其真实值作为变量的编码,将l个变量的编码按一定顺序连接在一起,形成个体编码串。解码时只需将染色体各个基因座上的基因值按序赋给相应的变量即可。若某一个优化问题有三个决策变量x_i(i=1,2,3),则个体X:

6.4	3.7	1.0
-----	-----	-----

对应决策变量取值为x₁=6.4,x₂=3.7,x₃=1.0。

(2)适应度评价函数。适应度是遗传算法中用来度量个体能达到或接近于最优解的优良程度。个体适应度函数的构造需与交叉、变异等算子及约束条件统筹兼顾。对于约束条件的处理,可采用罚函数法。由目标函数f(X)确定个体适应度评价函数F(X)。其公式如下:

$$F(X) = f(X) \mp \sigma g(X) \tag{4}$$

式(4)中,σ为罚因子,g(X)为约束条件表达式。目标极小化问题取“+”,否则取“-”。值得一提的是,目标极小化时,个体X适应度越小则越接近最优解,这与常规的适应度意义不同。

(3)选择操作。它采用正规几何排序选择法。只涉及个体适应度的大小次序关系,并不需要具体数据,所以适应度并不一定要求非负,且对目标极小化和极大化都适用。其基本思想是先对群体中所有个体按照其适应度大小进行排序(目标极大化时按升序排序,目标极小化时按降序排序)。群体中各个体被选中的概率为

$$P_i = \frac{q(1-q)^{r-1}}{1-(1-q)^M} \tag{5}$$

式(5)中,q为选择最优个体的可能性,0<q<0.1,r为第i个个体的排序序号,M为群体规模。

(4)交叉操作。交叉运算使用非均匀算术交叉。假设对X_Aⁱ,X_Bⁱ两个个体进行交叉,则交叉运算后所产生的两个个体是:

$$\begin{cases} X_A^{i+1} = \alpha X_B^i + (1-\alpha) X_A^i \\ X_B^{i+1} = \alpha X_A^i + (1-\alpha) X_B^i \end{cases} \tag{6}$$

式(6)中,α=exp(-α₀T/t),α₀为非均匀算术交叉系数,T为遗传算法进化最大代数,t为当前代数。

(5)变异操作。变异运算使用均匀变异。在进行x_k向x'_k的均匀变异操作时,x'_k为

$$x'_k = \begin{cases} x_k + (U_{\max}^k - x_k) c_2, c_1 \geq 0.5 \\ x_k - (x_k - U_{\min}^k) c_2, c_1 < 0.5 \end{cases} \tag{7}$$

式(7)中,c₁,c₂分别为[0,1]上均匀分布的随机数。

(6)模拟退火操作。在交叉、变异运算之后进行模拟退火运算,过程如本文1.2节。

(7)动态交叉率和变异率。遗传算法在群体进化的不同时期,交叉率或变异率始终取相同值不利于遗传算法的进化。

尤其在进化后期,个体相似性较大,交叉率应逐渐减小,变异率应逐渐增大,以增强变异作用而减轻交叉作用。笔者根据群体中较优个体应具有相对较小的交叉率和变异率的思想,并考虑到交叉和变异操作在不同时期的作用,提出了在进化过程中动态调整个体交叉率和变异率的方法,称为动态交叉率和变异率。计算公式如下:

$$p_c = \begin{cases} \eta_1 \cdot \frac{F' - F_{\min}}{F - F_{\min}} - \mu_1 \frac{t}{T}, & F' \leq \bar{F} \\ \eta_1 - \mu_1 \frac{t}{T}, & F' > \bar{F} \end{cases} \tag{8}$$

$$p_m = \begin{cases} \eta_2 \cdot \frac{F' - F_{\min}}{F - F_{\min}} + \mu_2 \frac{t}{T}, & F' \leq \bar{F} \\ \eta_2 + \mu_2 \frac{t}{T}, & F' > \bar{F} \end{cases} \tag{9}$$

式(9)中,F_{min}为群体中最小个体适应值,̄F为群体平均适应值,F'为参与交叉操作的两个个体适应值中较小者,η₁,η₂,μ₁,μ₂为调整系数。式(8)、式(9)适合于目标极小化问题。

(8)程序终止条件。若当前最优个体目标函数值达到最优值或当前代数达到最大遗传代数,则输出最优解,终止运行。

2 算例

以文献[4]中的非线性约束优化问题为例,对比验证本文的模拟退火混合遗传算法的有效性和实用性。

【算例】min f(x)=(x₁-2)²+(x₂-1)² (10)

s.t. $\begin{cases} c_1(x) = 0 \\ c_2(x) \geq 0 \end{cases} \tag{11}$

式中,c₁(x)=x₁-2x₂+1,c₂(x)=x₁²/4-x₂²+1,x₁∈[-1.82,0.84],x₂∈[-0.41,0.92]。

根据模拟退火混合遗传算法运行过程,编制了计算机程序。其程序框图如图1所示。混合遗传算法的运行参数:群体规模M=20,终止代数T=60,罚因子σ=100,模拟退火参数T₀=15 000,ψ=1.5,θ=3.0,动态交叉率和变异率参数μ₁=μ₂=0.1,η₁=1.0,η₂=0.5。

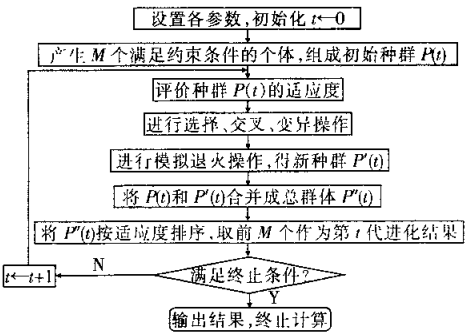


图1 模拟退火混合遗传算法程序框图

为了全面评价 SHGA 算法,对比说明模拟退火在 SHGA 中的作用,本文设计了没有模拟退火算子的遗传算法(OGA),即程序中不再进行模拟退火操作,其他算子和参数均不变。由运行过程中平均适应度的变化(图2)和最佳适应度的变化(图3)可知,进化过程中 SHGA 的种群整体性能优于 OGA,且 SHGA 能很快向最优解方向定位,而 OGA 向最优解搜索较缓慢;而且最后 SHGA 收敛到最优解,而 OGA 只收敛到次优解。这说明模拟退火对提高遗传算法的运行效率有显著作用。

(下转第76页)