

基于 DNA 计算的混合遗传算法研究*

唐天兵¹, 申文杰¹, 韦凌云², 谢祥宏¹

(1. 广西大学 计算机与电子信息学院, 南宁 530004; 2. 北京邮电大学 自动化学院, 北京 100876)

摘要: 提出一种混合遗传算法。该算法从微观策略和宏观遗传策略两方面考虑, 在微观方面将 DNA 计算引入到交叉算子和变异算子中; 在宏观方面提出一种以遗传算法流程为基础, 引入改进的小生境模拟退火操作, 构造合理的混合框架。典型算例的实验结果表明, 该算法有效, 收敛性好。

关键词: 遗传算法; DNA 计算; 小生境; 模拟退火

中图分类号: TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2010)01-0089-03

doi:10.3969/j.issn.1001-3695.2010.01.026

Research on hybrid genetic algorithm based on DNA computing

TANG Tian-bing¹, SHEN Wen-jie¹, WEI Ling-yun², XIE Xiang-hong¹

(1. School of Computer & Electronics Information, Guangxi University, Nanning 530004, China; 2. School of Automation, Beijing University of Posts & Telecommunications, Beijing 100876, China)

Abstract: This paper proposed a hybrid genetic algorithm. By considering the micro and macro strategies both, introduced DNA computing into the crossover and mutation in the micro aspects, and introduced improved niche simulated annealing operation into genetic algorithm to construct reasonable structure of the hybrid framework in the macro aspects. Experimental results of typical numerical example show that the algorithm is efficient and presents good convergence.

Key words: genetic algorithm(GA); DNA computing; niche; simulated annealing

0 引言

遗传算法(GA)是一种以 Darwin 自然进化论与 Mendel 遗传变异理论为基础的求解复杂全局优化问题的仿生型算法。它的显著不足是过早收敛和局部搜索能力差^[1], 前者常使算法收敛于局部最优解, 后者导致进化后期搜索效率低。其解决办法之一是把 DNA 计算引入到遗传算法中。DNA 计算是 Adleman 博士^[2]在 1994 年提出, 他成功解决了著名的 Hamilton 路径问题。DNA 携带丰富的遗传信息, 是重要的遗传物质。DNA 计算是利用 DNA 分子的各种生化反应来完成计算, 而遗传算法是一种以分子水平模拟生物进化过程来求解复杂问题的有效算法, 两者天生就具有某种必然的联系。文献[3]提出了一种基于遗传算法的 DNA 计算来解决最大团问题。文献[4]提出了一种具有 DNA 编码特征的改进遗传算法, 并引入了新的操作算子。解决办法之二是在遗传算法中引入较强的局部搜索算法。文献[5, 6]提出了一种遗传算法与模拟退火操作相结合的混合遗传算法, 但两者的结合没有采用小生境技术。

本文构造了一个基于 DNA 计算的混合遗传算法(DNA-hybrid genetic algorithm, DNA-HGA)。该算法从微观遗传策略和宏观遗传策略两方面考虑, 从而改善遗传算法过早收敛和局部搜索能力差的问题。在微观方面, 采用基于 DNA 特征的四进制编码, 把 DNA 计算引入到交叉算子和变异算子中; 在宏观

方面, 提出一种以遗传算法流程为基础, 通过对小生境淘汰操作后的个体进行改进的模拟退火操作。本文通过七个典型算例的计算结果表明, 该算法有效, 收敛性好。

1 算法设计

1.1 编码

编码是遗传算法首要解决的问题。传统的编码方法很多, 二进制编码为得到较高的精度, 需要增加二进制串的长度, 而造成计算量的迅速增加, 引起精度和效率的冲突。DNA 计算采用 DNA 链结构进行问题的编码, 可以有效克服遗传算法的 Hamming 悬崖距离和早熟现象, 增加种群的多样性。DNA 链是双链结构, 而 RNA 是以 DNA 为模板, 根据互补方式合成的单链结构。文中为了简化设计, 采用 RNA 的四种碱基 C(胞嘧啶)、U(尿嘧啶)、A(腺嘌呤)和 G(鸟嘌呤)分别对应 0(00), 1(01), 2(10), 3(11)这四个数字, 在括号中的二进制数, 高位为结构编码位, 低位为功能编码位, 这就是用 0、1、2 和 3 进行的四进制编码^[7]。

1.2 适应度计算

适应度函数直接影响遗传算法的性能, 即适应度较高的个体遗传到下一代的概率较大, 而适应度较低的个体遗传到下一代的概率较小。对于只有上下限约束的优化问题, 将目标函数直接作为适应度函数; 对于有不等式约束的优化问题, 采用罚函

收稿日期: 2009-05-17; 修回日期: 2009-06-27 基金项目: 国家自然科学基金资助项目(50605010)

作者简介: 唐天兵(1972-), 男, 四川成都人, 副教授, 硕导, 硕士, 主要研究方向为并行分布式计算和优化算法(tbtang@gxu.edu.cn); 申文杰(1981-), 男, 山西晋城人, 硕士研究生, 主要研究方向为并行分布式计算和优化算法; 韦凌云(1974-), 男, 广西柳州人, 副教授, 硕导, 博士后, 主要研究方向为并行分布式计算和优化算法; 谢祥宏(1984-), 男, 广西梧州人, 硕士研究生, 主要研究方向为并行分布式计算和优化算法。

数方法处理约束,把有约束问题改造为无约束问题,假设目标函数为极小化形式,本文构造的适应度函数表达式如下:

$$F = \frac{U}{obj + M \sum_{i=1}^n cons_i + N} \quad (1)$$

其中:obj 为目标函数;cons_{*i*} 为第 *i* 个约束违反量的绝对值;*M* 为惩罚因子,设为很大的数(如 *M* = 10 000 000);*N* 为一个很小的数(如 *N* = 0.000 001),以防止被 0 除;*U* 为与 obj 值同一量级的数。

1.3 遗传算子

1) 选择算子

选择操作的目的是为了避免基因缺失,使优良的个体以更大的概率遗传到下一代,从而提高全局收敛性和计算效率。本文中采用随机联赛选择,联赛规模为 2。

2) 交叉算子和变异算子

文献[8]总结了关于 RNA 序列的常用操作算子。交叉操作是遗传算法的主要操作,只有不断地交叉才能产生新的个体,从而得到优良的个体。本文在交叉过程中引入置换、换位和转位操作。置换操作是将 $T = T_5 T_4 T_3 T_2 T_1$ 中的一个子序列 T_4 被另一个子序列 T_4' 替换,形成新的 RNA 序列 $T = T_5 T_4' T_3 T_2 T_1$ 。换位操作是将 $T = T_5 T_4 T_3 T_2 T_1$, 交换子序列 T_4 与 T_2 , 形成新的 RNA 序列 $T' = T_5 T_2 T_3 T_4 T_1$ 。转位操作是将一个 RNA 序列 $T = T_5 T_4 T_3 T_2 T_1$ 中的一个子序列转换到新的位置,转换成 $T' = T_5 T_2 T_4 T_3 T_1$ 。交叉过程如图 1 所示。

变异操作是为了保持种群的多样性。本文引入三种变异操作:结构位交换是指结构编码位交换而功能编码位不变,即将 C(00)与 A(10)交换,U(01)与 G(11)交换;功能位交换是指结构编码位不变而功能编码位交换,即 C(00)与 U(01)交换,A(10)与 G(11)交换;结构编码位和功能编码位全交换是指结构编码位和功能编码位都改变,即 A(10)与 U(01)交换,C(00)与 G(11)交换。变异过程如图 2 所示。

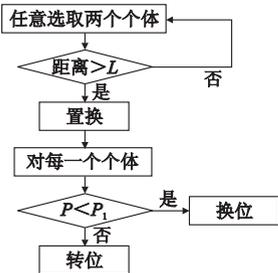


图1 交叉过程

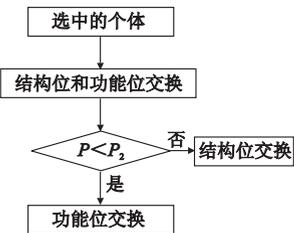


图2 变异过程

文中采用自适应交叉算子和自适应变异算子,交叉概率 *p_c* 和变异概率 *p_m* 能够随着个体的适应度自动变化。其思想是当种群适应度比较集中或局部最优时,使 *p_c* 和 *p_m* 增大;当种群适应度比较分散时,使 *p_c* 和 *p_m* 减少。因此 *p_c* 和 *p_m* 按如下公式自适应调整:

$$p_c = \begin{cases} k_1 \frac{f_{\max} - f'}{f_{\max} - f_{\text{avg}}} & f' \geq f_{\text{ave}} \\ k_2 & f' < f_{\text{ave}} \end{cases} \quad (2)$$

$$p_m = \begin{cases} k_3 \frac{f_{\max} - f'}{f_{\max} - f_{\text{avg}}} & f \geq f_{\text{ave}} \\ k_4 & f < f_{\text{ave}} \end{cases} \quad (3)$$

其中:*f_{max}* 是种群的最大适应度;*f_{ave}* 是种群的平均适应度;*f'* 是交叉的两个个体中较大的适应度;*f* 是变异个体的适应度,

k₁ = 0.85, *k₂* = 1, *k₃* = 0.5, *k₄* = 1。*L* 是两个个体之间的欧式距离;*P* 是一个随机数;*P₁* 是运行换位或转位的概率;*P₂* 是运行结构位交换或功能位交换的概率;*L* = 0.01, *P₁* = 0.5, *P₂* = 0.5。

1.4 小生境模拟退火操作

遗传算法采用小生境技术,可以在一次搜索过程中得到多峰值函数的多个极值点^[1]。本文根据个体间距离判定个体相似性,对适应度低的个体全部死亡,剩下一定数量的适应度高的个体占有小生境中的全部资源,执行模拟退火操作,从而使个体分散在整个搜索空间达到维持种群多样性的目的。

模拟退火(simulated annealing, SA)是一种随机搜索技术,它来源于金属退火原理,先将金属加热到某一高温状态,金属内部粒子随温度升高变为无序状态,内能增大,而慢慢冷却时粒子渐渐变得有序,在每个温度都达到平衡态,最后在常温时达到基态,内能减为最小。本文将模拟退火操作应用到遗传算法的局部搜索过程中,主要改进的地方有:a)在降温终止准则的基础上增加即时终止准则,当有更优解产生时则结束当前模拟退火操作;b)在扰动过程中,当在一个方向找不到更好解时,则向另一个方向搜索;c)对邻域大小的设置采取一种随进化代数增加而减小的自适应变化机制。算法参数设置为初始温度 *t₀* = 30,终止温度 *t_f* = 0.05,模拟退火的迭代次数 step = 10,温度衰减控制参数 *L* = 0.5,算法如下:

```

SA()
{
    当前温度 t = t0;
    sn = s0; // s0 为初始解
    从经过小生境操作后的群体中选择适应度高的个体;
    while (t > tf) {
        迭代因子 i;
        for (i = 0; i < step; i++) {
            根据一定扰动策略产生当前解 si;
            if (eval(si) > eval(sn)) // eval() 为评价函数
                sn = si;
            else {
                根据 Metropolis 准则计算接受概率 P;
                if (P > rand())
                    sn = si;
            }
        }
        t = t * L; // 进行温度衰减
    }
}
    
```

2 算法实现

根据上述思想,基于 DNA 计算和改进的小生境模拟退火操作相结合的混合遗传算法实现如下:

- a) 设置进化代数计数器 *t* = 1, 最大进化代数 *T*。
- b) 初始化种群。采用 RNA 编码,随机生成由 *M* 个个体组成的初始种群 *P(t)*, 计算出各个个体的适应度,将 *P(t)* 作为父代种群。根据个体的适应度进行降序排序,记忆前 *N* 个个体 (*N* < *M*)。
- c) 选择运算。采用随机联赛选择方式,对种群 *P(t)* 进行选择运算,得到种群 *P(t)'*。
- d) 自适应交叉操作。采用自适应的 *P_c* 对种群 *P(t)'* 进行

交叉操作,得到种群 $P(t)''$ 。

e) 自适应变异操作。采用自适应的 P_m 对种群 $P(t)''$ 进行变异操作,得到种群 $P(t)'''$ 。

f) 小生境生成操作。将步骤 e) 得到的 M 个个体和所记忆的 N 个个体合并成 $M+N$ 个个体的新群体,对这 $M+N$ 个个体,按照式(4)计算每两个个体 X_i 与 X_j 之间的归一化欧式距离,小生境半径为 D ,对每一个个体 X_i ,当其适应度 $F(X_i) > 0$ 时,若 $\|X_i - X_j\| < D$ 时,则惩罚个体 X_i ,令 $F(X_i) = 0$ 。得到种群 $P(t)''''$ 。

$$\|X_i - X_j\| = \sqrt{(1/n) \sum_{k=1}^n (x_{ik} - x_{jk})^2} \begin{cases} i=1, 2, \dots, M+N-1 \\ j=i+1, \dots, M+N \end{cases} \quad (4)$$

g) 模拟退火操作。对种群 $P(t)''''$ 进行改进的模拟退火操作,降序排序之后取前 M 个个体形成种群 $P(t)$ 作为下一代进化种群。

h) 终止条件判断。若 $t \leq T$,更新进化迭代计数器 $t = t + 1$,转到 c);若 $t > T$,则输出结果,计算结束。

3 算例

为了验证混合遗传算法的有效性和收敛性,本文采用七个典型算例对算法进行测试,并将计算结果与其他算法所得到的结果进行比较。实验程序采用 MATLAB 语言编写,硬件环境使用 AMD Dual Core Processor 2.30 GHz CPU,1 GB 内存。从表 1~3 的计算结果可知,该混合算法有效,收敛性好,并且在算法精度和稳定性方面都体现出了一定的优势。

$$P_1^{[4]}: \begin{cases} \max & x \sin(10\pi x) + 1 \\ \text{s. t.} & -1 \leq x \leq 2 \end{cases}$$

$$P_2^{[9]}: \begin{cases} \max & 21.5 + x_1 \sin(4\pi x_1) + x_2 \sin(20\pi x_2) \\ \text{s. t.} & -3.0 \leq x_1 \leq 12.1 \\ & 4.1 \leq x_2 \leq 5.8 \end{cases}$$

$$P_3^{[9]}: \begin{cases} \max & -(1-x_1)^2 + 10(x_2-x_1^2)^2 + x_1^2 - 2x_1x_2 + \exp(-x_1-x_2) \\ \text{s. t.} & x_1^2 + x_2^2 - 16 \geq 0 \\ & x_1^2 - x_2 + 1 \geq 0 \\ & -x_1 - x_2 + 20 \geq 0 \\ & 0 \leq x_1 \leq 100 \\ & 0 \leq x_2 \leq 100 \end{cases}$$

$$P_4^{[9]}: \begin{cases} \min & \sum_{i=1}^5 i \cos[(i+1)x_1 + i] + \sum_{i=1}^5 i \cos[(i+1)x_2 + i] \\ \text{s. t.} & -10.0 \leq x_1 \leq 10.0 \\ & -10.0 \leq x_2 \leq 10.0 \end{cases}$$

$$P_5^{[9]}: \begin{cases} \min & 0.01x_1^2 + x_2^2 \\ \text{s. t.} & x_1x_2 - 25 \geq 0 \\ & x_1^2 + x_2^2 - 25 \geq 0 \\ & 2 \leq x_1 \leq 50 \\ & 0 \leq x_2 \leq 50 \end{cases}$$

$$P_6^{[9]}: \begin{cases} \min & (x_1-2)^2 + (x_2-1)^2 \\ \text{s. t.} & -x_1^2 + x_2 \geq 0 \\ & 2 - x_1 - x_2 \geq 0 \end{cases}$$

$$P_7^{[9]}: \begin{cases} \min & 100(x_2-x_1^2)^2 + (1-x_1)^2 \\ \text{s. t.} & x_1 + x_2^2 \geq 0 \\ & x_1^2 + x_2 \geq 0 \\ & -0.5 \leq x_1 \leq 0.5 \\ & x_2 \leq 1 \end{cases}$$

表 1 P_1 函数的优化结果

测试函数	算法类型	种群规模	迭代次数	x_{\max}	y_{\max}
P_1	DNA-HGA	20	100	1.850547	2.850274
	文献[4]	50	145	1.850773	2.850227
	SGA ^[4]	50	145	1.850773	2.850227

表 2 P_2, P_3 和 P_4 函数的优化结果

测试函数	最优结果		平均结果		最差结果	
P_2	最优解	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]
	38.8503	38.8503 38.8503	38.8497 38.8185	38.8441 38.5873		
P_3	最优解	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]
	-	1600039.0 1600039.0	1599832.7 1599828.8	1598477.1 1599264.9		
P_4	最优解	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]	DNA-HGA 文献[9]
	-186.73	-186.73 -186.73	-186.73 -186.56	-186.72 -180.21		

表 3 P_5, P_6 和 P_7 函数的优化结果

测试函数	最优解	DNA-HGA	文献[9]	HHE-MUM-APP ^[9]	TPEP ^[9]	Hybrid EP ^[9]	EP alone ^[9]
P_5	f_{\min}	5.00000	5.00000	5.00000	5.00000	5.00000	4.99800
	误差	0	0.0	3.0×10^{-12}	0.0	0.0	发散
P_6	f_{\min}	1.00000	1.00000	1.00000	1.00000	1.00000	0.99559
	误差	0	0.0	0.0	0.0	0.0	发散
P_7	f_{\min}	0.25000	0.25000	0.25000	0.25000	0.25000	0.83636
	误差	0	0.0	0.0	0.0	0.0	发散

4 结束语

为改善遗传算法过早收敛和局部搜索能力差的问题,本文从微观遗传策略和宏观遗传策略两方面进行考虑,在微观方面,采用基于 DNA 特征的四进制编码,把 DNA 计算引入到交叉算子和变异算子中;在宏观方面,把遗传算法与改进的小生境模拟退火操作有机结合,构造合理的混合框架。本文采用七个典型算例对算法进行测试,计算结果表明,该混合算法有效,收敛性好。

参考文献:

- [1] 周明,孙树栋. 遗传算法原理和应用[M]. 北京:国防工业出版社,2005.
- [2] ADLEMAN L M. Molecular computation of solutions to combinatorial problems[J]. Science,1994,266(5187):1021-1024.
- [3] LI Yuan, FANG Chen, OUYANG Qi. Genetic algorithm in DNA computing: a solution to the maximal clique problem[J]. Chinese Science Bulletin, 2004,49(9):967-971.
- [4] 爱新觉罗·川, 闵乐泉. 一种基于 DNA 计算的改进遗传算法[J]. 哈尔滨理工大学学报, 2008,13(3): 57-61.
- [5] 孙强,马光胜,刘晓晓. 求解 SAT 问题的退火遗传算法[J]. 小型微型计算机系统, 2008,29(7):1268-1271.
- [6] CHEN P H, SHAHANDASHTI S M. Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints[J]. Automation in Construction, 2009,18(4): 434-443.
- [7] LI Shu-chao, XU Jin. Digital coding for RNA based on DNA computing[J]. Computer Engineering and Application, 2003,39(5): 46-47.
- [8] LI Shu-chao, XU Jin, PAN Lin-qiang. Operational rules for digital coding of RNA sequences based on DNA computing in high dimensional space[J]. Bulletin of Science and Technology, 2003,18(6):388-398.
- [9] SARIMVEIS H, NIKOLAKOPOULOS A. A line up evolutionary algorithm for solving nonlinear constrained optimization problems[J]. Computers & Operations Research, 2005,32(6): 1499-1514.