

基于改进的选择算子和交叉算子的遗传算法

曹道友¹,程家兴²

(1. 安徽大学 计算机科学与技术学院,安徽 合肥 230039;

2. 安徽大学 计算智能与信号处理教育部重点实验室,安徽 合肥 230039)

摘要:为了解决遗传算法中收敛速度与局部最优解的矛盾,文中提出了一种具有改进的选择算子和改进的交叉算子的遗传算法。使用文中改进的选择算子,能够增加算法收敛于全局最优解的概率,从而不容易陷入局部最优,也就增加了找到最优解的概率,使用文中改进的交叉算子可以加快算法的收敛速度,从而缩短寻找最优解的时间。实验证明,这两种改进算子的结合能以较快速度收敛于全局最优解,因此能很好地解决遗传算法中收敛速度与局部最优解之间的矛盾。

关键词:遗传算法;选择算子;交叉算子;适应度;相似度

中图分类号:TP301.6

文献标识码:A

文章编号:1673-629X(2010)02-0044-04

A Genetic Algorithm Based on Modified Selection Operator and Crossover Operator

CAO Dao-you¹, CHENG Jia-xing²

(1. School of Computer Science and Technology in Anhui University, Hefei 230039, China;

2. Ministry of Edu. Key Lab. of Intelligent Computing & Signal Processing, Anhui Univ., Hefei 230039, China)

Abstract: In order to solve the conflict between algorithm convergence and the best local answer effectively, puts forward an improved genetic algorithm with a modified selection operator and a modified crossover operator. It can increase the probability of the best answer and well avoid approaching the best local solution by using the modified selection operator, it also increased the probability of finding the best answer, and using the modified crossover operator can speed up the convergence rate, thus shortening the time to find the best answer. The experimental result indicates that the two modified operators' combination can converge to the best answer at higher speed, so it can well solve the contradiction between the convergence rate of genetic algorithm and the best local solution.

Key words: genetic algorithm; selection operator; crossover operator; fitness; similarity degree

0 引言

遗传算法(Genetic Algorithm, GA)^[1,2]是模仿自然界生物遗传进化过程中“物竞天择、适者生存”的原理,而开发出的一种全局优化随机搜索算法,是由美国J. Holland教授于1975年在其论文“自然系统和人工系统的适配”中提出的,它的特点是应用广泛、使用简单、鲁棒性强等。GA的基本思想来自Darwin的进化论和Mendel的遗传学。它借用了生物遗传学的观点,通过自然选择、交叉、变异等遗传操作,实现各个体适应性的提高。算法拥有一群个体组成的种群。每个体在种群演化过程中都被评价优劣并得到其适应值,个体在

选择、交叉以及变异算子的作用下向更高的适应度进化以达到寻求问题最优解的目标^[3]。

目前,传统的遗传算法已经在工业工程和设计、经济管理、交通运输等领域得到了广泛的应用。不过,它仍然存在过早收敛和局部最小等问题。为了解决这些问题,人们从遗传算法的三种基本遗传算子(选择算子,交叉算子和变异算子)入手,这些参数的选择将直接影响算法的性能和搜索速度,所以选择合适的遗传算子是算法能高效地收敛到全局最优解的关键所在。文中使用改进的选择算子,从而不容易陷入局部最优,使用改进的交叉算子加快算法收敛速度,两种改进算子的结合在一定程度上解决了上述问题。

收稿日期:2009-05-21;修回日期:2009-08-21

基金项目:国家自然科学基金(60273043)

作者简介:曹道友(1984-),男,安徽滁州人,硕士研究生,研究方向为智能计算;程家兴,教授,博士生导师,研究方向为智能计算、算法分析及最优化方法。

1 遗传算法的基本概念及步骤

1.1 遗传算法的基本概念

现在简单介绍遗传算法的几个基本概念:

(1)基因(Gene):它是组成染色体的单元,可以表示为一个二进制位,一个整数或者一个字符等,它是遗传算法中的最小单位。

(2)染色体(Chromosome):在用遗传算法解决问题的时候,需要把问题的解编成一个合适的码子。一般采用二进制编码,二进制编码类似于生物染色体的组成,编码简单易于用生物遗传理论来解释并且易于算法进行各种遗传操作,这种具有固定长度的0,1二进制字符串就是染色体。二进制字符串中的每一位二进制代码代表一个基因,二进制字符串代码的总位数称为染色体的长度。一个染色体就代表问题的一个可能的解,每个染色体也被称作一个个体,它是遗传算法操作的基本对象。

(3)群体(Population):在遗传算法中每代所产生的染色体的总数被称为群体。一个群体包含了该问题在这一代的一些解的集合,它表示遗传算法的遗传搜索空间。

(4)适应度(Fitness):对群体中每个染色体进行编码后,每个个体对应一个具体问题的解,并且每个解都对应于一个函数值。而该函数就是适应度函数,函数值即适应度,它代表一个个体所对应解的优劣,也即衡量染色体对环境适应能力的指标。适应度是群体中个体生存机能的唯一确定性指标,它的选取直接影响到遗传算法的收敛速度以及能否找到最优解,适应度函数是反映实际问题的目标函数,适应度基本上依据该目标函数来确定。

(5)选择(Select):从当前群体中选择适应度高的个体以便进行繁殖,选择的原理是:适应度高的个体为下一代贡献一个或多个后代的概率就大,所以选择之前必须计算每个个体的适应度,然后按一定的概率从上一代群体中选择 M 对个体作为双亲,直接复制到下一代,染色体不发生变化。

(6)交叉(Crossover):对于选中进行繁殖的两个染色体 X, Y ,以它们为双亲进行交叉操作,组合它们两者的特性产生两个新的后代 X', Y' 。

(7)变异(Mutation):在群体中随机选择一个个体(染色体),对于选中的个体以一定的概率随机地改变符号串结构数据中某一位的值。变异使算法具有一定的局部的随机搜索能力,也可防止发生因遗漏最优值而无法再找到最优解的情况,保持了算法的有效性和种群的多样性,是算法的一个重要操作过程。染色体是否变异由变异率来控制。

1.2 遗传算法的步骤

1.2.1 遗传算法的主要步骤

Holland 教授提出的遗传算法常被称为“简单遗传

算法(SGA)”,其主要步骤如下:

- (1)根据待解问题的参数集进行编码;
- (2)初始化群体;
- (3)计算群体中每个个体的适应度值;
- (4)按照由个体的适应度值所决定的某个规则选择将进入下一代的个体;
- (5)按交叉概率 P_c 进行交叉操作;
- (6)按变异概率 P_m 进行变异操作;
- (7)如果没有满足某种终止条件,则转到第(3)步,否则进入第(8)步。
- (8)输出种群中适应度值最优的染色体作为问题的满意解或最优解。

1.2.2 参数控制

遗传算法中有四个参数需要提前设定,且在实际应用中需要多次测试之后才能确定这些参数的合理取值。

M :称为种群大小,算法的效率明显受到它的影响,种群规模太小会降低种群的多样性,太大会降低算法效率。对于不同问题,种群规模也不同,一般建议的取值范围是20~100。

T :称为终止进化代数,一般建议的取值范围是100~500。

P_c :称为交叉概率,并不是所有被选择的个体(染色体)都要进行交叉或变异操作,而是以一定的概率进行的。在程序设计中交叉概率要比变异概率选取得大几个数量级,一般建议的取值范围是0.4~0.99^[4]。

P_m :称为变异概率,一般建议的取值范围是0.0001~0.5。

终止条件:遗传算法的终止条件通常可以从两方面进行控制,一是预先设定最大的进化代数来终止算法,二是当算法在规定的代数内还没有找到最优解则终止算法。

2 改进的遗传算法

2.1 选择算子的改进

简单遗传算法采用轮盘赌选择方法^[5],它的过程是这样的:将每个个体的适应度值与该种群的总适应度值(也即种群所有个体的适应度值之和)相比,得到该个体的相对适应度,所有个体的相对适应度之和应为1。把每个个体的相对适应度当作其在选择操作中被选中的概率,每一轮选择会产生一个[0,1]均匀随机数,并将该随机数作为选择指针来确定被选个体,适应度大的个体被选中的概率就大。这种轮盘赌选择方法虽得到广泛运用,但依然存在两个问题:一是在进化初期,有可能适应度很高的个体,被选择的概率很大,

从而复制出很多后代,个体单一而无法继续进化使搜索陷入局部最优;二是在进化后期,当各个个体的适应度差距不大时,该方法已经不再具有选择能力,体现不出个体的优劣。

所以下面对选择算子进行改进:将群体中的所有个体按照适应度从大到小进行排列,排在最前面 1/4 的个体复制两份,中间的 2/4 复制一份,最后面的 1/4 抛弃,这样一来种群数目还是没有改变。用这种方法来进行选择,可以将适应度比较低的个体直接淘汰掉,并且增加了适应度较好的个体的数目,从而在一定程度上解决了上述问题。

2.2 交叉算子的改进

交叉算子的好坏直接影响到遗传算法的收敛速度的快慢。在传统的遗传算法中,不用管交叉的两个个体的相似性,而采用固定的交叉率对两个个体进行交叉操作,导致父个体中的好的模式不能被遗传到下一代,因此使得算法的收敛速度下降。为了保证父个体中的好的模式保留到下一代,文中采用了改进的交叉算子,它根据个体间的相似度大小来决定是否进行交叉操作^[6]。假设有两个串 A 和 B,对 A 和 B 的相似度定义如下:

定义 1 相似度:

$$s(i, j) = l/n \quad (0 < i, j \leq m \wedge i \neq j)$$

其中 l 是 A 和 B 的最长公共子串的长度, n 为染色体串长度, m 为种群规模。例如:串 A = 10111100, 串 B = 01101100, 最长的公共子串是 1100, 长度为 4, 染色体长度 n 为 8, 所以 A 和 B 相似度为 4/8。

下面给出阈值 p : 0.5, 只有当两个个体的相似度 s 小于 p 时,这两个个体才可以进行交叉,文中采用单点交叉。

下面是求两个串的最长公共子序列长度的代码:

```
void crossover(char * p1, char * p2, float p)
```

```
{
    int m = 0, n = 0, i = 0;
    float s;
    char * s1 = p1, * s2 = p2;
    while(* s1 != '\0')
    {
        if(* s1 == * s2)
        {
            s1++;
            s2++;
        }
        else
        {
            s1 = &(p1[i++]);
            s2 = p2;
        }
    }
}
```

```
{
    n = strlen(s1);
    i = 1;
    s1 = p1;
    s2 = p2;
    while(* s1 != '\0')
    {
        if(* s1 == * s2)
        {
            s1++;
            s2++;
        }
        else
        {
            s1 = p1;
            s1[strlen(s1) - i] = '\0';
            s2 = p2;
            i++;
        }
    }

    m = strlen(s1);
    if(m > n)
    {
        n = m;
        s = n/strlen(p1);
        if(s < p)
        {
            进行交叉操作;
        }
    }
}
```

3 性能测试

3.1 测试函数

3.1.1 Camel 函数

$$f(x, y) = (4 - 2.1x^2 + x^4/3)x^2 + xy + (-4 + 4y^2)y^2 \quad (1)$$

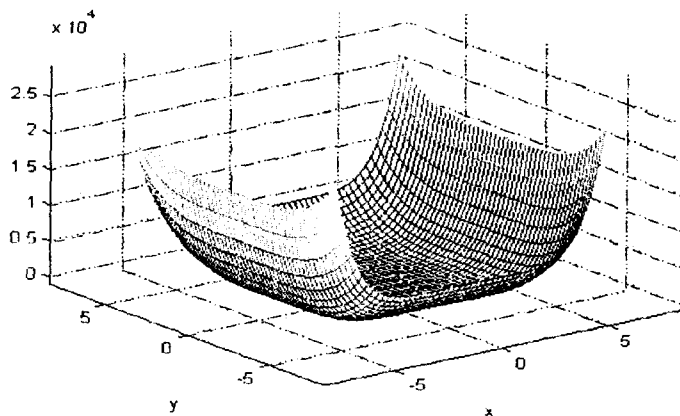


图 1 Camel 函数极值分布图

图 1 为 Camel 函数^[7] 极值分布图,从图中可以看到该函数有多个极小值,其中最小值为 -1.031628 , $-100 < x, y < 100$ 。

3.1.2 Schaffer 函数和 Shubert 函数

$$f_1(x,y) = 0.5 - (\sin^2 \sqrt{x^2 + y^2} - 0.5)/(1 + 0.001(x^2, y^2))^2 \tag{2}$$

其中 f_1 为 Schaffer 函数^[7], $-100 < x, y < 100$, 理论最大值为 1, 收敛值为 0.9999。图 2 为 Schaffer 函数极值分布图。

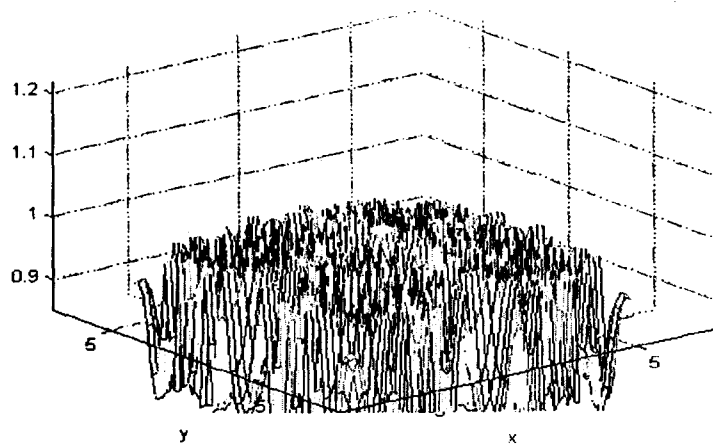


图 2 Schaffer 函数极值分布图

$$f_2(x,y) = \left\{ \sum_{i=1}^5 i \cos[(i+1)x + i] \right\} \cdot \left\{ \sum_{i=1}^5 i \cos[(i+1)y + i] \right\} \tag{3}$$

其中 f_2 为 Shubert 函数^[8], $-10 \leq x, y \leq 10$, 理论最小值为 -186.7309 , 收敛值为 -186.7290 。图 3 为 Shubert 函数极值分布图。

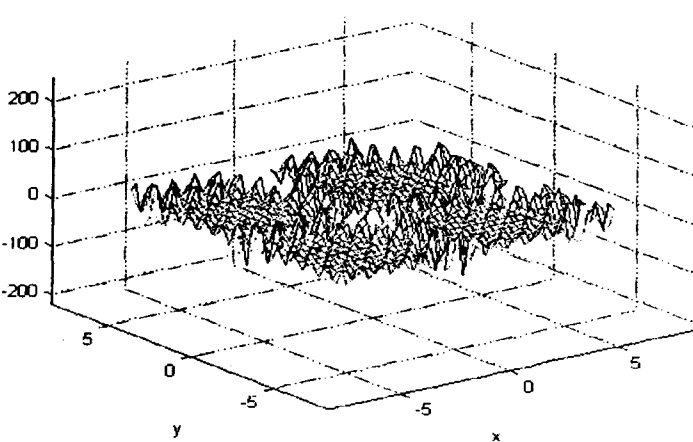


图 3 Shubert 函数极值分布图

3.2 测试结果

分别采用传统遗传算法和文中的改进算法先对 Camel 函数进行测试。实验在 VC++ 6.0 环境下各自运行 200 次,种群规模为 100,进化代数为 200 代。使用两种算法分别对 Camel 测试函数运行 200 次

后,就它们的最终解平均值和最优解作了比较,结果如表 1 所示。

表 1 两种算法的测试对照表

函数	传统遗传算法		文中算法	
	最终解平均值	最优解	最终解平均值	最优解
f	-0.960864	-1.029202	-1.031282	-1.031624

从表 1 中可以看出,文中算法的最优解更加接近测试函数的最小值,从而证明了该算法的有效性。

下面分别采用传统遗传算法和文中的改进算法对上述 3.1.2 中的一组测试函数进行测试。

文中提出的算法和传统遗传算法对这组测试函数分别运行了 500 次。从表 2 和表 3 的对比中可以看出,文中的改进算法具有较好的收敛性和较快的收敛速度。

表 2 传统遗传算法的收敛性能

函数	种群规模	收敛次数	平均收敛代数	平均收敛值
f_1	50	500	25	0.9999
f_2	300	482	31	-186.7290

表 3 文中算法的收敛性能

函数	种群规模	收敛次数	平均收敛代数	平均收敛值
f_1	50	500	25	0.9999
f_2	300	496	24	-186.7302

4 结束语

传统的遗传算法采用固定的交叉率,致使收敛速度慢,容易陷入局部最优解。文中使用改进的选择算子,增加了算法收敛于全局最优解的概率,使用改进的交叉算子加快了算法的收敛速度。实验表明改进算法具有良好的稳定性和较高的运行效率,从而很好地解决了遗传算法收敛速度与局部最优解的矛盾。今后研究的方向是如何找到更好的变异算子使算法以更快的速度收敛于全局最优解。

参考文献:

[1] Holland J H. Adaptation in Natural Artificial Systems[M]. [s.l.]:MIT Press,1975:1-17.
[2] 黄少荣. 遗传算法及其应用[J]. 电脑知识与技术,2008,4(7):1874-1875.
[3] Srinivas M, Parnaik L. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithm[J]. IEEE Trans. on Systems, Man, and Cybernetics, 1994,24(4):656-666.
[4] 刘铁男,刘 斌,梁福贵. 一种带局部搜索策略的遗传算法及其应用[J]. 大庆石油学院学报,2005,29(2):76-78.
[5] Goldberg D E. Genetic Algorithms in Search, Optimization



图 4 原始输入彩色转化成的灰度图像

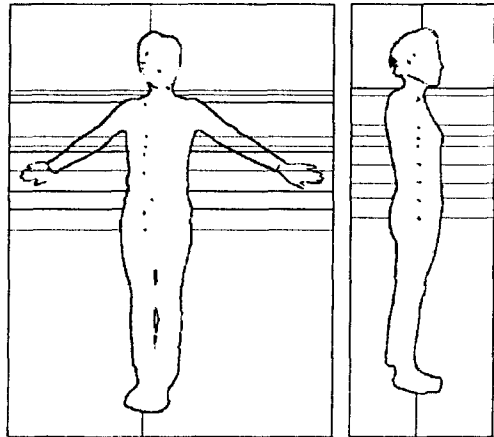


图 5 尺寸顶点标记图像

表 1 人体躯干十个截面尺寸顶点坐标

水平截面	前视图坐标(cm)	侧视图坐标(cm)
后领围线	(-4.0,5.9,-136)	(-8.1,4.6,-136)
前领围线	(-5.9,7.8,-134)	(-8.8,4.3,-134)
肩峰线	(-14.3,17.4,-131)	(-12.2,2.2,-131)
上胸围线	(-13.9,14.7,-118)	(-13.9,8.4,-118)
胸围线	(-13.6,13.6,-113)	(-12.9,10.8,-113)
下胸围线	(-12.8,13.2,-110)	(-12.2,10.1,-110)
腰围线	(-12.4,12.4,-102)	(-10.1,10.1,-102)
腹突线	(-12.4,12.4,-94)	(-8.8,10.5,-94)
臀围线	(-15.1,13.9,-86)	(-10.5,11.2,-86)
基线	(-15.9,13.9,-78)	(-12.2,10.1,-78)

与国外的系统例如 Loughborough 人体影子扫描

仪^[6,7]、TC2 分层轮廓测量方法^[8]以及 Cyberware 公司的三维全身扫描仪^[9,10]等比较起来,该文提高了原型的健壮性。由于用阈值分割提取轮廓,不能保证轮廓一定闭合,可能出现断点,此时程序在测量恰好出现断点的部位时,会出现异常,使程序中途退出。因此需要程序有避开断点的能力,比断点更坏的情况是出现了一断线段,但健壮的程序在即使这样的情况下,也不能让程序中途退出,否则,既不能了解哪出了断点,也不能继续测量本没有断点的部位,因此需要捕捉异常。为了避免断点,测量可以在根据位置比例求得的像素位置上下浮动 2~3 个像素,然后在这几个位置找间距最小的那个。如果遇到断线,则此截面无法测准,但要记录断线位置,并跳过该截面,继续后面的测量。

参考文献:

[1] 周晓晶,赵正旭.基于照片的三维虚拟人重建技术[J].东南大学学报:自然科学版,2008,38(4):632-636.

[2] 魏高峰,王成焘.虚拟人体的研究现状与进展[J].北京生物医学工程,2008,27(4):431-435.

[3] 朱 洲.三维人体信息获取及虚拟服装试穿技术研究[D].武汉:华中科技大学,2004.

[4] 冈萨雷斯.数字图像处理.[M].阮秋琦,译.北京:电子工业出版社,2003.

[5] 夏良正.数字图像处理修订版[M].南京:东南大学出版社,1999.

[6] Jones P R M, West G, Harris D. The Loughborough Anthropometric Shadow Scanner (LASS)[J]. Endeavour, 1989, 13(4):164-168.

[7] Aufmann K. Invasion of the Body Scanners[J]. IEEE Circuits and Devices Magazine, 1997, 13(3):12-17.

[8] Hurley J D, Demers M H, Wulpern R C. Body measurement system using white light projected patterns for made-to-measure apparel[J]. SPIE, 1997(3131):212-223.

[9] Simmons K. Body Measurement Techniques: A Comparison of Three-Dimensional Body Scanning and Physical Anthropometric Methods[C]//ITAA/KSCT 2001 Conference. Seoul, KOREA:[s.n.], 2001.

[10] Daanen H, Taylor S E, Brunsman M A. Absolute accuracy of the Cyberware WB4 whole body scanner[J]. SPIE, 1997(3023):6-12.

(上接第 47 页)

and Machine Learning[M]. MA: Addison-Wesley, 1989:1-83.

[6] 孟佳娜,王立宏.具有自识别能力的遗传算法求解旅行商问题[J].计算机工程与应用,2006,42(13):51-52.

[7] 韩 力,王亚峰,李 伟,等.改进模拟退火算法在三相异

步电动机优化设计中的应用[J].中小型电机,2004,31(6):2-3.

[8] 张 文,李 样.基于优化组合的遗传算子的研究与应用[J].数值计算与计算机应用,2005,26(3):208-214.