

函数优化的遗传算法

刘焕祥^{1,2}

(1.武汉大学 计算机科学学院,湖北 武汉 430072;2.武汉大学 后勤集团,湖北 武汉 430072)

摘要:遗传算法是一种有效的模拟进化算法,针对不同问题,编码方式多种多样。针对函数优化问题,阐述了不同的编码方式及遗传操作,在 Matlab 环境中用遗传算法实现求解函数优化问题。

关键词:遗传算法;函数优化;Matlab

中图分类号:TP312

文献标识码:A

文章编号:1672-7800(2009)02-0044-03

0 引言

近年来,随着计算技术的发展,一些新的智能算法(如遗传算法、模拟退火算法、禁忌搜索算法)得到了迅速发展和广泛应用。特别是模拟进化算法(GA、GP、Es),无论是理论研究还是应用研究都空前活跃。同时,一些新的模拟进化算法也逐渐出现并日益完善,为这类复杂优化问题提供了一定的解决方案。遗传算法是目前研究最多、应用最广的模拟进化算法,在众多领域得到了广泛应用。

1 遗传算法

遗传算法是基于自然选择和自然遗传机制的一种随机搜索算法。遗传算法与传统的搜索算法不同,具有如下特点:①遗传算法不是对变量直接操作,而是对其编码进行操作;②它是从一组点出发进行搜索,而不是从单个点开始;③它不需要导数等信息;④它是一种随机搜索算法。

遗传算法的主要操作有:复制、交叉、变异。复制体现了“适者生存,不适者淘汰”的自然选择机制。交叉操作使得后代能够继承父代的优良特性。变异操作在增加种群多样性方面具有重要作用。

遗传算法的应用过程主要包括:编码、构造初始种群、设计适应度函数、确定遗传算法结构、选择遗传算子、确定遗传算法的控制参数。

2 函数优化问题中的遗传算法

2.1 编码问题

遗传算法编码方式有多种形式,如二进制编码、浮点数编码、整数编码、符号编码、矩阵编码等。对于实值函数编码,最常用的编码方式是二进制编码,二进制编码实际就是对搜索空间

均匀划分,所求解的精度受到编码长度的限制。如果编码长度过短,则求解精度下降;反之,如果编码长度过长,则搜索时间相应增加,不利于实际应用。因而对函数优化问题,实数编码已逐渐引起重视。

2.2 选择方法

常用的选择方式有:①适应度比例选择方法。这是最基本最常用的选择方法,又称赌轮选择。个体被选中的概率与其适应度成比例。②最佳个体保留方法。它把群体中性能最好的个体直接复制到下一代。③期望值方法。它首先计算每个个体在下一代中的期望数目。若某一个体被选中并参与交叉配对,则它在下一代中的期望生存数目减去 0.5。否则,该个体的期望生存数目减去 1。当某一个体的期望生存数目小于 0 时,该个体不参与选择。④联赛选择方法。从群体中任意选择一定数目的个体,将其中性能最好的个体保留到下一代,重复这一过程,直到生成整个种群。⑤排挤方法。在该方法中,新生成的子代将替代或排挤相似的旧父个体。

2.3 遗传算子

以优化问题

$\min f(x)$

$$X=(x_1, x_2, \dots, x_n) \subset R^n, a_i \leq x_i \leq b_i \quad (1)$$

为例进行说明。

(1)变异算子

均匀变异:随机选择某一位 $j: 1 \leq j \leq n$, 将该位改变为区间 $[a_i, b_i]$ 上的随机变量

$$x_i' = \begin{cases} U(a_i, b_i), & i=j \\ x_i, & \text{otherwise} \end{cases} \quad (2)$$

边界变异:随机选定某一位 j , 将该位变为相应元素的上(下)界

$$x_i' = \begin{cases} a_i, & i=j, r < 0.5 \\ b_i, & i=j, r \geq 0.5 \\ x_i, & \text{otherwise} \end{cases} \quad (3)$$

非均匀变异:随机选定某一位 j ,把它变为 $[a_i, b_i]$ 内的非均匀随机数

$$x_i' = \begin{cases} x_i + (b_i - x_i)f(G), & i=j, r_i < 0.5 \\ x_i - (x_i - a_i)f(G), & i=j, r_i \geq 0.5 \\ x_i, & \text{otherwise} \end{cases} \quad (4)$$

其中 $f(G) = (r_2(1 - \frac{G}{G_{\max}}))^b$; $r_1, r_2 \in U(0, 1)$; G, G_{\max} 分别表示当前遗传代数和最大遗传代数; b 是一控制参数。

多点非均匀变异:每次选定多个点进行上面所述的变异。

(2)交叉算子

实值简单交叉:与二进制编码中的简单交叉方式相同。

算术交叉:对个体 X, Y ,交叉后代为($r \in U(0, 1)$)

$$\begin{cases} X' = rX + (1-r)Y \\ Y' = (1-r)X + rY \end{cases} \quad (5)$$

启发式交叉:

$$\begin{cases} X' = X \\ Y' = Y + r(X - Y) \end{cases} \quad (6)$$

同时执行可行性检查,即检查各元素是否超出其上(下)界(其中 $r \in U(0, 1)$)。

3 遗传算法求解函数优化的 Matlab 程序实现

为了体现优化问题的普遍性,这里的优化对象选择一个多峰函数

$$f(x) = x + 10 \sin(5x) + 7 \cos(4x), x \in [0, 9] \quad (7)$$

采用二进制编码,种群中的个体数目为 10,染色体长度为 20,交叉和变异概率分别为 0.95 和 0.08.其 Matlab 的主程序源代码如下:

```

fplot('x+10*sin(5*x)+7*cos(4*x)', [0 9])
% create a random starting population of size 10.
initPop=initialzega(10,[0 9], 'galevall');
hold on
plot(initPop(:,1),initPop(:,2), 'b+')
pause % Strike any key to continue to run the ga for one
generation.
[x endPop]=ga([0 9], 'galevall', [], initPop, [1e-6 1 1], '
maxGenTerm', 1, ...
'normGeomSelect', [0.08], ['arithXover'], [2 0], '
nonUnifMutation', [2 1 3]);
x % The best found
% plot the resulting the resulting population
plot(endPop(:,1),endPop(:,2), 'ro')
pause % Strike any key to continue to run the ga for 25
generations
[x endPop]=ga([0 9], 'galevall', [], initPop, [1e-6 1 1], '
maxGenTerm', 25, ...
'normGeoraSelect', [0.08], ['arithXover'], [2], '
nonUnlfMutation', [2 25 3]);
x % The best found, and plot the resulting the resulting

```

population

```

pause % Strike any key to continue
figure(2)
fplot('x+10*sin(5*x)+7*cos(4*x)', [0 9])
hold on
plot(endPop(:,1),endPop(:,2), 'r*')
pause % Strike any key to continue
figure(3)
plot(trace(:,1),trace(:,3), 'r-')
hold on
plot(trace(:,1),trace(:,2), 'r-')
xlabel('Generation');ylabel('Fitness');
legend('solution', 'average');
% End

```

图 1 为目标函数的图形和初始化随机种群的个体分布图。经过一次遗传迭代后,寻优结果如图 2 所示,图中“o”表示经过一次迭代后的个体分布,此时最优解=2.8979,16.2130。

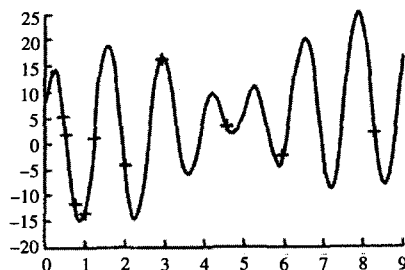


图 1 初始种群分布图

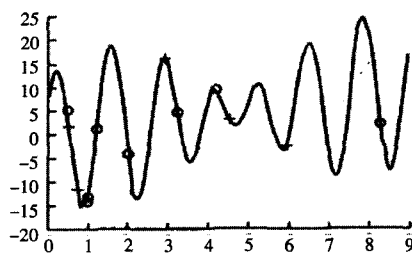


图 2 一次迭代后的寻优结果

经过 25 次遗传迭代后,寻优结果如图 3 所示,图中“*”表示经过迭代后的最优结果,此时最优解 $x=7.8567, 24.8554$ 。迭代过程中迭代次数与函数值的变化如表 1 所示。图 4 则通过进化过程中种群平均值和解的变化绘出了遗传算法的寻优性能。

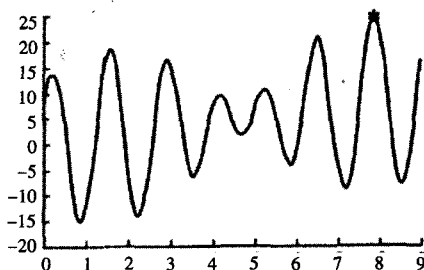


图 3 25 次迭代后的最优结果

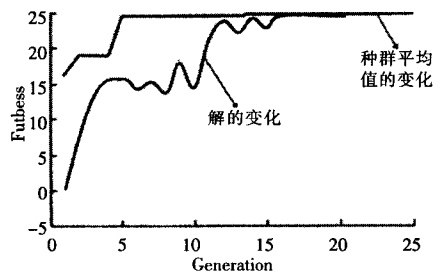


图4 遗传算法的寻优性能

表1 迭代次数与函数值的变化过程

迭代次数	函数值
1	16.212973
2	18.851287
5	24.480041
14	24.839363
17	24.853845
19	24.855305
22	24.855356
23	24.855363
25	24.855363

了综述。并利用 Matlab 中已有的遗传算法工具箱给出了一个多峰函数的实际算法。

参考文献:

- [1] 玄光男,程润伟.遗传算法与工程优化[M].北京:清华大学出版社,2004.
- [2] 王小平,曹立明.遗传算法—理论、应用与软件实现[M].西安:西安交通大学出版社,2002.
- [3] 潘正君,康立山,陈毓屏.演化计算[M].北京:清华大学出版社,1998.
- [4] 孔敦卫,潘凤萍.自适应遗传算法及应用[M].北京:中国矿业大学出版社,2003.
- [5] 殷铭,张兴华.基于 Matlab 的遗传算法实现[J].电子技术应用,2000(1).
- [6] 王玉良.基于 MATLAB 的遗传算法可视化仿真系统研究[J].计算机系统应用,2004(10).
- [7] E.B.马格雷伯.MATLAB 原理与工程应用[M].北京:电子工业出版社,2002.
- [8] 雷英杰,张善文等.Matlab 遗传算法工具箱及应用[M].西安:西安电子科技大学出版社,2005.

(责任编辑:周晓辉)

4 结束语

本文对遗传算法用于函数优化问题的主要遗传算子进行

Genetic Algorithm Based on Function Optimization

Abstract: The inheritable calculation is a kind of effective immitantly evolving calculation, which focuses on different problems and has various coding. The thesis illustrats different coding methods and inheritable operation about the problem on well-organized function. It is a concrete example of realization of function optimization with genetic algorithm in Matlab.

Key Words: Genetic Algorithm; Function Optimization; Matlab