

遗传编程局部最优化问题分析

高波^{1,2} 张忠能¹ 查志琴²

¹(上海交通大学计算机科学与工程系 上海 200030)

²(常州工学院计算机工程系 常州 213002)

摘要 本文介绍了遗传编程的一般步骤,指出程序在遗传的过程中存在的局部最优问题,分析了局部最优状态产生的机理,并提出了局部最优状态的发现方法以及解决方法。

关键词 遗传编程 局部最优 全局最优 异常基因 变异

THE ANALYZE OF LOCAL OPTIMIZATION IN GENETIC PROGRAMMING

Gao Bo^{1,2} Zhang Zhongneng¹ Zha Zhiqin²

¹(Department of Computer Science and Engineering, Shanghai Jiaotong University Shanghai 200030)

²(Department of Computer Engineering, Changzhou Institute of Technology, Changzhou 213002)

Abstract This paper introduces the steps of genetic programming, and point out that there are local optimizations in genetic programming. In this paper, we analyse the mechanism of local optimizations, then give a method to solve the problem.

Keywords Genetic programming Local optimization Global optimization Abnormal gene Variation

1 遗传编程的基本工作原理

遗传编程(Genetic Programming, GP)是一种自动随机产生搜索程序的方法,该算法是一种全局优化搜索算法,对非线性复杂问题显示出很强的求解能力,已被成功地应用于许多不同的领域。理论上凡是根据多个输入值而得到一个值的函数,如:对于 $f(x_1, x_2, \dots, x_n)$ 这样的函数都可以使用GP来生成。

比如在战场上,需要一种能够画出作战区域地图的探测机器人,这个机器人从起始位置开始,无指导地自动沿着障碍物走,并自动记录相应的位置,最后自动画出所有的障碍物的位置及形状,从而得到相应的地图。可以为这个机器人设置8个传感器,分别为n(north)、e(east)、s(south)、w(west)、ne(north_east)、nw(north_west)、se(south_east)、sw(south_west),它们用以探测障碍物的存在(靠障碍物时对应的传感器输入为1,否则为0)。这个机器人有5种动作分别为:上(north)、下(south)、左(west)、右(east)、停(stop)。

由于战场的千变万化,需要考虑的因素很多,要产生它的控制程序往往需要通过相当复杂的逻辑分析。这样的程序就可以通过GP来产生。这类程序一般都可以用一颗树来表示。其实验环境如图1所示。

GP的基本步骤如下:



图 1

1. 预处理

预处理主要包括:建立程序的数据模型、建立评估方法及函数、确定直接复制的数目、交叉生产数目、参与锦标赛选拔的程序数目以及变异的百分比。

2. 随机产生初始群体0代(generation 0)

产生的0代程序是通过随机算法生成的树型程序,初始程序的数量X可以预先确定,比如5000个。

3. 评估程序的合适度

对于不同的应用,需要选择不同的评估函数,用于评估所生成的程序是否是可用的,对于上面的机器人程序,评估函数可以让程序模拟执行100次,每次运行1000步,然后把沿障碍物次数的多少作为返回值,这个返回值称为合适度,合适度越大说明这个程序是越合适的。

4. 若生成了最合适程序,则结束;否则继续

找出当前代中合适度最大的个体,若该程序中已符合要求,则整个算法可以结束,否则还要继续进行进化,产生下一代程序。评估函数的取法,对整个遗传编程的影响是非常大的,有时可能会导致程序不收敛而得不到正确的结果。

5. 产生下一代程序

一般通过以下几种方法来产生:

(1) 直接复制产生下一代。把第i代中的n个程序复制到第i+1代中而不作任何修改。这种方法产生的程序数量比较

收稿日期:2002-12-27。高波,讲师,主研领域:数据挖掘,网络与数据库技术,粗糙集理论。

少,一般控制在 10%左右(比如 500 个)。

(2) 交叉产生下一代(如图 2 所示)。交叉操作是这样进行的:每一个程序均从第 i 代中选择一个“父亲”和“母亲”进行交叉操作而产生,交叉的方法如图 2 所示。而“父亲”和“母亲”程序都是通过从第 i 代中用锦标赛的方法产生的。大部分下一代程序是用这种方法产生的,一般要占到 90%左右(比如 4500 个)。

(3) 变异构造产生下一代。在第 i 代中用锦标赛的方法选出一个单亲,删除此单亲程序的一个子树,这个子树是随机选定的,取而代之以另一个新的子树,这个新的子树也是随机产生的,产生新子树的方法与产生第 0 代程序的方法相同。用这种方法产生的程序的数目很少,一般控制在 1%左右(比如 50 个左右)。

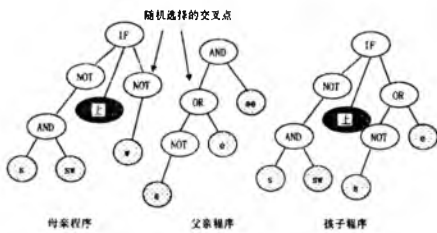


图 2

2 GP 的局部最优问题

遗传编程的求解目标是一个最优的程序,从本质上来讲遗传编程其实是一个最优化方法的问题。局部最优问题是最优化方法的一个重要问题。我们建立了一个模拟的环境,假设整个战场的障碍物都在边上,而且整个战场环绕一周共需走 32 步。图 3a 是一个实际进化的合适度函数曲线。可以看出,当程序进化到第 5 代时,合适度不再继续提高,而是保留在 16 的位置。

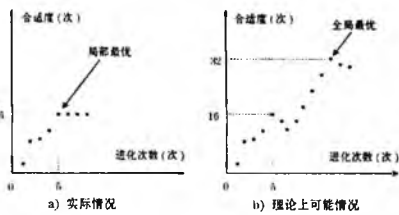


图 3

这就是一个典型的遗传编程中的局部最优问题。从理论上讲,合适度变化的总的趋势应该不断地趋向于最大值 32,从而达到全局最优。但最优状态是一个稳定状态,不管是局部的还是全局的,所以当遗传程序进入到一个局部最优状态时就会保持这个状态而无法继续提高。

经过分析,我们发现局部最优现象主要是由于锦标赛选择个体的方法以及评估函数导致的。当进入局部最优状态时,再进行下一轮进化其合适度可能会降低,此时用锦标赛方法选择时就会选择已进入该状态的程序个体,而更优越的个体基因就无法进入下一代。从而导致了不断地近亲繁殖,也就进入了相对稳定的局部最优状态。

在实验中,我们还发现局部最优总是出现在全局最优的 50%处,在本例中全局最优是 32,而局部最优是 16。经过分析,

这是由于地图的形状所决定的,当地图中存在 90 度的墙角时就有可能出现 50%的局部最优状态。

3 局部最优状态的识别

当出现局部最优时,必须及时发现并立即处理,否则随着不断进化整个遗传程序个体基因将会出现严重的同化,这是由于不断的近亲繁殖所引起的。可采用两种方法来发现局部最优状态。

第一种方法是通过人工发现局部最优状态。每一次进化选出最优个体后,我们都把这个个体进行连续的模拟执行,所以可以直观地从图 4 上看出机器人是否已经进入局部最优状态。其具体表现是机器人在某个区域范围内不断地重复行走,而不会移动到指定区域的其它地方去。最可能发生的情况是机器人在某个 90 度的墙角死循环,如图 4 中所示的三种可能出现的机器人处于局部状态时的运行轨迹。

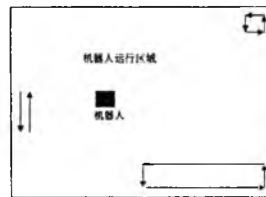


图 4 机器人的三种典型局部最优轨迹

另一种方法是通过程序自动发现机器人程序是否已经进入局部最优状态。这里又可能分为两种方法:一种是总结出可能出现局部最优的情况,在评估函数中对比被测评的程序是否符合局部最优状态;第二种方法是从整体上来进行判别,用一个数组记录下每次进化后最优个体的合适度,当发现连续多次进化后合适度不再发生变化,且还没有达到全局最优时,则可以认为遗传程序已进入局部最优状态。

4 局部最优问题的解决

从遗传程序本身的角度来看,导致局部最优的原因是在遗传程序中存在异常的基因,为了使得机器人能够走出局部最优状态,必须找出含有异常基因的个体并破坏它。异常基因可能存在于一般个体中,也可能存在于最优个体中。我们只要考虑最优个体即可,因为只有最优个体中的异常基因才会反映出来并导致局部最优现象。而在一般个体中的异常基因是隐性遗传基因不会立即反映出来,我们可以暂时不用考虑。

对于具有导致局部最优的异常基因,一种有效的方法是对它进行变异处理。找出具有异常基因的个体,在这个个体上随机选定一棵子树,删除后取而代之以另一棵新的子树,这棵新的子树也是临时随机产生的,产生新子树的方法与产生第 0 代程序的方法相同。

上述的变异操作可以使得遗传程序有效地脱离局部最优状态,如图 5 所示。

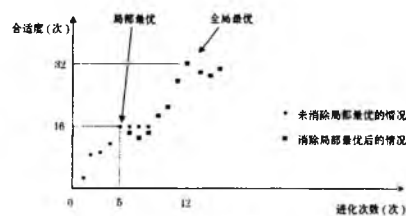


图 5

当判断出局部最优状态发生并采取变异后,其中的最优个体的合适度在一般情况下会降低,但这是暂时的,随着进化的进行,很快就会回升到局部最优状态时的合适度并不断地趋向全局最优,在上升过程中可能还会出现局部最优状态,则需再次调用变异操作,直到达到全局最优为止。但通过实验发现,局部最优一般出现在全局最优的 50%处,再次发生的局部最优一般出现在前一次局部最优处理完毕,合适度下降并回升再次达到 50%时发生。

5 结 论

遗传编程过程中局部最优状态的发现是一个非常重要的问题,若不及时解决会使整个群体的基因发生同化,从而出现严重的退化现象。实验过程中通过人工直观的观察可以发现局部最优状态,并可以从现象中分析出局部最优产生的机理。

实验证明本文提出的局部最优的发现及处理方法可以有效地使遗传程序脱离局部最优,加快了整个程序的全局收敛速度。

参 考 文 献

- [1] Nordin P., Banzhaf W., Brameier M., Evolution of micro-robot Inertial model based on genetic Programming. *Robotics and Autonomous Systems*, 1998, 25(105 ~ 116).
 - [2] 刘大有、卢奕甫、王飞、梁艳春,“遗传程序设计方法综述”,《计算机研究与发展》,第 38 卷第 2 期,2001 年 2 月。
 - [3] 王江、杨家本,“一种搜索全局最优解的工程方法”,《系统工程理论与实践》,第 8 期,1998 年 8 月。
 - [4] Erich, Gamma 设计模式 - 可复用面向对象软件的基础,机械工业出版社。
 - [5] Nilson, 人工智能,机械工业出版社。
-