

一种基于遗传算法的求代数方程组数值解的新方法

胡小兵, 吴树范, 江 驹

(南京航空航天大学 自动控制系, 南京 210016)

摘要: 解方程组是工程研究中的基本问题. 当前的各种求解方法都只针对某一特定类型的方程组, 没有通用性. 根据遗传算法与具体问题无关的特点, 提出了一种通用的代数方程组数值解求解方法. 首先, 基于遗传算法建立了代数方程组求解问题的通用模型; 然后阐述了具体的求解过程; 最后分析了该法的特点和性能. 实验表明该法是有有效可行的, 进一步提高解精度的关键在于对遗传算法本身的研究.

关键词: 遗传算法; 方程组; 适应度函数; 聚类

中图分类号: TP301.6

文献标识码: A

New method based on genetic algorithm for resolving algebraic equation groups

HU Xiao-bing, WU Shu-fan, JIANG Ju

(Department of Automatic Control, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

Abstract: Resolving equation group is a principal problem in engineering study. Any existing method can only deal with a specified kind of equation group, so has no broad applicability. A new method with broad applicability is presented on base of genetic algorithm, because genetic algorithm does not at all care what the problem is. First, a common model is established for all kinds of algebraic equation groups; then a software process is brought up; finally the performance is analyzed. The results of experiment show that the new method is effective and feasible, and the key to improve this method is to study genetic algorithm further.

Key words: genetic algorithm; equation group; fitness function; clustering

1 引言(Introduction)

求解方程组(本文中指的是:求代数方程组的数值解)是实际工程应用研究中的常见问题. 各种方程组的解法长期以来一直是数学界和工程界的一个重要研究内容. 因为方程组的复杂多样性, 目前的研究大都是仅针对某一类特定方程组的解法; 对许多方程形式复杂的方程组, 甚至还没有有效的解法. 遗传算法(genetic algorithm, 简称 GA)是一种解决复杂问题的有效方法, 具有很强的通用性^[1,2]. 本文根据求解方程组问题的共同特点, 基于 GA 建立了一个适用于各类方程组的通用模型. 在此模型基础上, 开发了一种新的求代数方程组数值解的通用算法. 实验证明了此算法的有效可行性; 求解精度的提高与 GA 的品质有关.

2 基于 GA 的方程组求解通用模型(GA based common model of equation groups)

设一个代数方程组, 由 n 个方程组成, 涉及 m

个变量:

$$\begin{cases} f_1(X) = 0, \\ f_2(X) = 0, \\ \vdots \\ f_i(X) = 0, \\ \vdots \\ f_n(X) = 0. \end{cases} \quad X \in \Phi.$$

其中, $f_i()$ 可为任意形式的函数表达式(分段函数除外), $X = [x_1, x_2, \dots, x_j, \dots, x_m]$, $\Phi = \{X \mid x_j \in (a_j, b_j)\}$. 求解此方程组等价于下面一个求极值问题: 求一 X , 以使式(1)取值最小. 当其最小值为 0 时, 所对应的 X , 即为方程组之解; 当其最小值不为 0 时, 则此方程组无解.

应用 GA 的关键在于将问题的求解转化为一种数串染色体, 然后找出染色体在问题环境中的适应度函数:

$$F(X) = \sum_{i=1}^n |f_i(X)|, X \in \Phi. \quad (1)$$

对方程组问题,其求解的每个变量都可用一个二进制数串来表示(如图 1 所示).二进制数串对应的变量值由式(2)计算.通过改变二进制数串的长度 q_j 和定义小数点位置 k ,就可实现各种取值范围和精度.把表示各个变量的二进制数串依次连接起来,就构成了一条染色体.记某代染色体由式(1)算出的最大值为 $F_{\max}$,则该代中某个染色体的适应度由式(3)计算.

$$x_j = (1 - 2 \times b_{j,1}) \sum_{s=2}^{q_j} b_{j,s} \times 2^{k-s}, \quad (2)$$

$$\text{Fit} = F_{\max} - \sum_{i=1}^n |f_i(X)|, X \in \Phi. \quad (3)$$

这个方程组 GA 模型形式简单,通用性强.

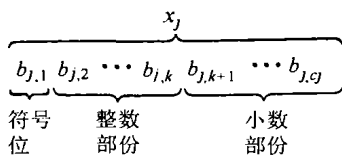


图 1 变量 x_3 的二进制数串
Fig. 1 Binary cluster for x_3

对有 $f_i(\cdot)$ 为分段函数的情况,可将原方程组按分段情况转化为若干个子方程组,然后为每个子方程组建立一个 GA 模型即可.

3 算法的实现过程(The processes of this new method)

3.1 适应度函数的生成(Establishment of fitness function)

要使算法能适用于任意方程组,首先必须根据用户输入的待求解方程组生成适应度函数.由于待求解方程组的多样性,所以算法中的染色体适应度计算函数应是动态可变的.可用如下方法实现:以字符串形式记录用户输入的待求解方程组;当要计算染色体的适应度时,再根据方程组字符串以软件编译的方法进行单步解释执行.这一过程类似于 Basic 语言这种单步解释执行的程序编译系统,只不过要简单许多.

有时由于交叉、变异操作的随机性,会造成病态染色体,即:使某个方程等式无意义,如实数域内负数开平方.则此时直接将病态染色体的 Fit 置为 0.当所有染色体由式(1)算出的适应度都相等时,则 Fit 都为 0.这时一般可确认为所有染色体都集中到了某一个局部极值点上.此时要加大变异以启动新的搜索.

3.2 染色体的生成(Production of chromosomes)

和其它搜索算法一样,当解空间非常大时,GA

的解品质也会下降.解空间的大小与变量的数目、变量的取值范围 and 要求的解精度有关,最后表现为染色体的长度.当某个变量的取值范围很大,如 $(-10^6, 10^6)$,要求的解精度又很高,如 10^{-10} ,则此变量的二进制码串长度将达到 55 位.仅对此一个变量寻优的计算量就很大了.因此本算法规定:一个变量的最长码长为 25 位.根据实际情况,变量码长可以比此短.一旦 25 位仍不够用,则先将变量的取值范围划分为若干个子区间,在满足精度要求的前提下,使变量在各个子区间内的码长不超过 25 位;然后分别在各子区间内求解.这时式(2)需要修正为式(4),其中 $x0_j$ 为与子区间相对应的基准量.采用子区间的求解效果比采用超长染色体的明显要好得多.

$$x_j = x0_j + (1 - 2 \times b_{j,1}) \sum_{s=2}^{q_j} b_{j,s} \times 2^{k-s}. \quad (4)$$

3.3 对 GA 的改进(Improvement of GA)

标准 GA 存在着求解精度不高、容易陷入局部极值等缺点.因此,本算法作了如下改进:

1) 交叉、变异操作、种群规模 n 和进化代数 N .

本文定义了两种交叉操作:一种是将两个染色体中的某一变量片段整个进行交换,称为 I 类交叉;另一种是将变量片段内的子片段进行交换,称为 II 类交叉.文中的变异操作为对染色体的某一位取反.交叉、变异操作是染色体进化的根本方法,是算法寻优能力的基础.由于病态染色体的 Fit 为 0,进化中将被淘汰,所以不用严格检查交叉、变异后解的可行性.

种群规模 n 和进化代数 N 太小,难以找到最优解;反之,计算时间太长.这里取 n 为染色体长度的 2 倍, N 为染色体长度的 4 倍.这样就可与具体的方程组问题相适应.

2) 根据染色体的适应度动态地改变交叉概率 P_c 和变异概率 P_m .

P_c 和 P_m 对 GA 的性能有重要的影响.本算法中, P_c 和 P_m 根据式(5)、式(6)来动态确定,从而增加算法的进化能力和收敛速度^[3].其中, $\text{Fit}_{\max}$ 为某代染色体中的最大适应度; Fit_{avg} 为该代染色体的平均适应度; Fit 为将要进行交叉或变异的染色体的适应度.

$$P_c = \begin{cases} \frac{\text{Fit}_{\max} - \text{Fit}_{\text{avg}}}{\text{Fit}}, & \text{Fit}_{\max} - \text{Fit}_{\text{avg}} < \text{Fit}, \\ 1.0, & \text{Fit}_{\max} - \text{Fit}_{\text{avg}} \geq \text{Fit}, \end{cases} \quad (5)$$

$$P_m = \begin{cases} \frac{0.1(\text{Fit}_{\max} - \text{Fit})}{\text{Fit}_{\max} - \text{Fit}_{\text{avg}}}, & \text{Fit}_{\max} - \text{Fit} < \text{Fit}_{\max} - \text{Fit}_{\text{avg}}, \\ 0.1, & \text{Fit}_{\max} - \text{Fit} \geq \text{Fit}_{\max} - \text{Fit}_{\text{avg}}. \end{cases} \quad (6)$$

3) 根据小生境原理指导染色体进行进化,采用聚类方法加强 GA 的局部寻优能力.

标准 GA 的全局和局部寻优能力在求解方程组问题中都显不足,算法在进化的后期容易陷入局部极值点,且解精度不高.本文按文献[4,5]的方法,采用 Kohonen 网对染色体进行聚类,根据小生境原理指导

复制、交叉和变异操作.这样,通过不断优化小生境中的类群,并根据类群最佳个体进行自适应优化搜索,从而增强算法的全局、局部寻优能力和收敛速度.总结上面的方法,算法的总流程图见图 2.

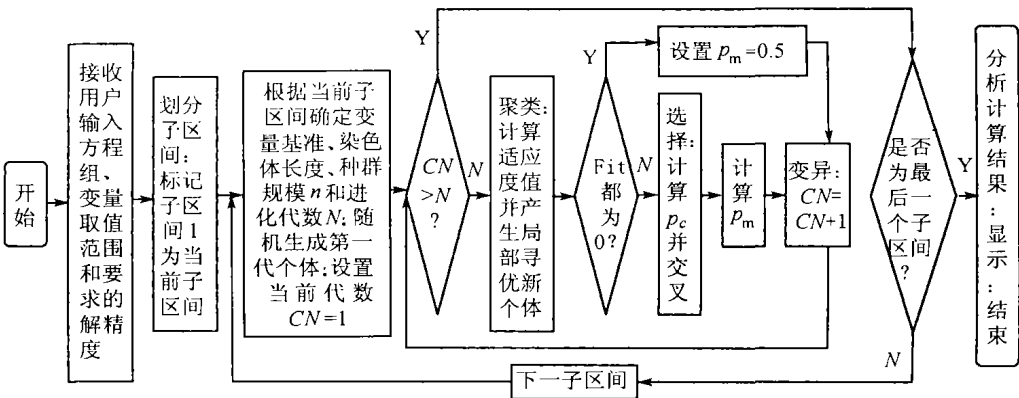


图 2 算法的总流程图
Fig. 2 The flow chart of the method

4 算法的性能分析和实验结果 (Performance analyse and experiment results)

本算法的数学模型概括了代数方程组求解问题的本质特点,从而决定了算法的通用性.此模型还可根据式(7)进一步改进.其中 $\max(|f_i()|)$ 可用 GA 来求得.改进模型均衡了各方程的指标的影响,使指标函数的地形简化,有利于后面的求解;但时间增加.其实对式(1),只要方程组求解算法中的 GA 性能足够好,即使指标函数地形复杂,也不影响求解质量.所以关键在于提高 GA 性能.本算法实际采用式

(1)来建立模型.

$$F = \sum_{i=1}^n |f_i(X)| / \max(|f_i(X)|), X \in \Phi. \tag{7}$$

为提高 GA 性能,本文采用了多种改进措施,目前,GA 研究是一个热点.随着 GA 研究不断深入,必将为本软件算法提供更多更有力的支持.同时,划分子区间的措施保证了算法在各种解空间中的求解精度.虽然牺牲一些时间,但可从 GA 的高效率中得到补偿.

表 1 实验结果
Table 1 The results of experiment

序号	方程组	理论解	算法解	求解时间(s)
1	$ \sin(30x) \times (1 - x /2) - 0.9739626 = 0,$ $x \in (-100, 100)$	$x = \pm 0.05179$	$x = 0.0518$	25.41
2	$\begin{cases} (x + 99.7091)^2 + y^2 - 10000 = 0, \\ \sin(5x) + \cos(5y) - 1.9932 = 0, \end{cases}$ $x \in [-2, 2], y \in [-2, 2]$	$x = 0.2909$ $y = 0$	$x = 0.2909$ $y = 0.0000$	36.90
3	$\begin{cases} x^2 + y^2 + z^2 - 3 = 0, \\ x^2 + y^2 + xy + x + y - 5 = 0, \\ x + y + z - 3 = 0, \end{cases}$ $x \in [-1.732, 1.732], y \in [-1.732, 1.732]$ $z \in [-1.732, 1.732]$	$x_1 = 1$ $y_1 = 1$ $z = 1$	$x = 1.0000$ $y = 0.9998$ $z = 1.0001$	51.57
4	$\begin{cases} 2x + 6y - 6zs - 5 = 0, \\ (x + y)^2 - 4z - 10s = 0, \\ 2.5x + 2^y - 2z - 3s - 4 = 0, \\ xy + z + s + 0.5 = 0, \end{cases}$ $x \in [-10, 10], y \in [-10, 10]$ $z \in [-10, 10], s \in [-10, 10]$	$x = 1$ $y = 0$ $z = -1$ $s = 0.5$	$x = 1.0011$ $y = 0.0009$ $z = -0.9984$ $s = 0.5001$	124.5

本算法具有很强的适应性和鲁棒性. 不管方程是否可微、连续或形式复杂, 方程组是否完备、良态或有解, 都不影响其求解. 对于有通解的方程组, 可用此法多次求解后再拟合求通解. 对于有低效高精度算法的方程组, 可用此法进行预解以提高速度.

表 1 给出了 4 个方程组的求解情况. 其中方程组 1、方程组 2 具有很复杂的多峰地形. 要求解精度为 0.0001; 所用电脑 P III 450. 求解结果表明本软件算法的有效可行性.

5 结论(Conclusion)

本文提出了一种基于 GA 的求代数方程组数值解的通用算法. 首先, 基于 GA 建立了方程组求解问题的通用模型; 然后研究了具体的实现过程; 最后分析了此法的特点和性能. 实验结果表明: 该算法是有效可行的; 进一步提高解精度的关键在于对 GA 本身的研究. 本算法可作为工程应用研究的一个通用基础工具.

参考文献(References)

[1] Wang Yaonan. Intelligent Control System [M]. Kaifeng: Henan

University Press, 1996 (in Chinese)

- [2] Wang Wencheng. The Applications of Neural Network to Motor Industry [M]. Beijing: Beijing University of Technology Press, 1998 (in Chinese)
- [3] Wu Zhiyuan and Shao Huihe. A new adaptive genetic algorithm & its application in multimodel function optimization [J]. Control Theory and Applications, 1999, 16(1): 127 - 129 (in Chinese)
- [4] Xu Jinwu and Liu Jiwen. Niche based genetic algorithm [J]. Pattern Recognition & Artificial Intelligence, 1999, 12(1): 104 - 107 (in Chinese)
- [5] Cao Xianbin, Zheng Zhen and Fan Lei. An improved genetic algorithm mixed with neural network [J]. Pattern Recognition & Artificial Intelligence, 1999, 12(4): 486 - 492 (in Chinese)

本文作者简介

胡小兵 1975 年生. 现为南京航空航天大学自动控制系 98 级硕士研究生. 研究方向: 智能控制与模式识别. Email: h0615@sina.com

吴树范 1963 年生. 现为南京航空航天大学自动控制系教授. 研究方向: 航天体控制.

江 驹 1964 年生. 现为南京航空航天大学自动控制系讲师. 研究方向: 机器人控制.