1000-9825/2001/12(07)0981-06 ©2001 Journal of Software 软件学报 Vol.12, No.7

利用目标函数梯度的遗传算法*

何新贵[1]， 梁久祯[2]

[1](北京系统工程研究所,北京 100101)；

[2](大庆石油学院 计算机科学系,黑龙江 大庆 151400)

E-mail: hexingui@ns.cetin.net.cn; liang_jz@263.net

摘要：多数遗传算法在搜索解时没有充分利用其问题域的知识.提出了一类新的改进的适应度函数的遗传算法.它考虑了函数在搜索点的函数值及其变化率,并将该信息加入适应度函数,使得按概率选择的染色体不但具有较小的函数值(对极小化问题而言),而且具有较大的函数值变化率.实验结果表明,这类方法的收敛速度明显高于标准遗传算法.

关键词：SGA(标准遗传算法)；GMGA(梯度改进的遗传算法)；DGMGA(离散的梯度改进遗传算法)；适应度函数；优化

中图法分类号：TP18 **文献标识码**：A

遗传算法自20世纪60年代发明以来得到了广泛的重视,尤其是模式理论的建立给遗传算法奠定了理论基础.由于遗传算法并行度高,对初值的依赖性小,在求解具有局部极值的问题中表现出了其特有的鲁棒性.近年来,遗传算法作为智能计算(神经网络、模糊处理和进化计算)的重要组成部分,一直是研究的一个热点,在应用领域也取得了相当好的结果,如多极值函数的优化问题、组合优化问题、调度问题等.尽管如此,在使用遗传算法求解具体问题时,染色体群体规模、选择概率、交叉概率以及变异概率等参数的设置仍然较难控制,特别是适应度函数的选择对算法的收敛性以及收敛速度的影响较大,故针对不同的问题需根据经验来确定相应的参数.这说明只是依赖于编码域搜索的遗传算法并没有充分利用问题域中的知识.如何将问题域中的知识作为一个指导信息加入到算法的搜索过程中去,这在遗传算法研究中已引起人们的注意[1,2],并取得了初步的成果.如文献[3]将被优化函数的梯度或一阶差分引入到十进制实数编码的遗传算法变异算子中,实现了定向变异.

本文研究一类新的改进的适应度函数的遗传算法.首先分析标准遗传算法的基本思想以及在利用问题域进行近似解搜索方面的不足;由于标准遗传算法对于求函数极值问题只是考虑在搜索点的函数值信息,而没有充分利用函数的变化趋势,因此按概率选择的染色体不一定是一个“好的”染色体.本文重点考虑在搜索点的变化趋势,并将该信息加入到适应度函数中,使得按概率选择的染色体不但具有较小的函数值(对极小化问题而言),而且具有较大的变化趋势.实验结果表明,这类方法的收敛速度明显高于标准遗传算法,在相同迭代次数控制的搜索过程中,收敛率也有较大的提高.

1 标准遗传算法及其存在的一些问题

标准遗传算法按编码方式可分为二进制编码遗传算法和非二进制编码遗传算法.

* 收稿日期：2001-01-05；修改日期：2001-03-05

作者简介：何新贵(1938－),男,浙江浦江人,研究员,博士生导师,主要研究领域为神经元网络,模糊技术,进化算法；梁久祯(1969－),男,山东东平人,博士,讲师,主要研究领域为神经元网络,进化算法.

二进制遗传算法的特点是:基因表达细腻,问题编码长度较大,有利于求解组合优化问题,而不利于高精度的函数优化问题;染色体的选择需要从编码域到问题域的转换;染色体交换基因的方式丰富;染色体变异可选择的范围较大;可利用模式理论进行算法收敛性分析.

非二进制遗传算法的特点是:编码域与问题域相同,有利于求解高精度的函数优化问题;染色体的选择无须译码;染色体交换不如基因方式丰富;染色体变异较复杂;到目前为止还没有成型的理论用于算法的收敛性分析.

两类算法存在一个共同的问题:适应度函数的选取只与目标函数有关,而没有考虑或没有充分考虑目标函数的变化趋势,致使遗传过程中优选的点(染色体)在一个时期内只具有较优的目标函数值,但可能都是在较平坦的区域内徘徊,经多代遗传后,目标函数值仍然得不到较明显的优化,这种情况有可能导致"早熟"现象.

2 改进适应度函数的遗传算法

大多数遗传算法在进化搜索中仅使用目标函数作为适应度函数.适应度函数评估是选择操作的依据,适应度函数的设计直接影响到遗传算法的性能.在具体应用中,适应度函数的设计要结合求解问题本身的要求而定.本节简要介绍常用的适应度函数及其处理方法,重点考虑基于目标函数变化趋势的适应度函数的选取准则及其对遗传算法的影响.

2.1 常用的适应度函数

对于不同的问题,适应度函数的选取方式有多种表现形式.按照对目标函数的处理方式,可将适应度函数分为以下两大类:目标函数映射成适应度函数和适应度函数标定.

由于在遗传算法中,适应度函数用于计算适应值总和、选择概率和累积选择概率,故适应度函数的值要非负.因此,一般需将目标函数映射成求极大值函数且函数值为非负的适应度函数.对于求解极大值问题,目标函数可采取如下方式转换为适应度函数:

$$fit(x)=\begin{cases}f(x)+c_{\min} & \text{当 } f(x)+c_{\min}>0\\ 0 & \text{其他}\end{cases}. \tag{1}$$

其中 $c_{\min}$ 可以是一个合适的输入值,或者是当前代或前 n 代中 $fit(x)$ 的最小值.对于求解极小值问题,目标函数可用如下方式转换为适应度函数:

$$fit(x)=\begin{cases}c_{\max}-f(x) & \text{当 } f(x)<c_{\max}\\ 0 & \text{其他}\end{cases}. \tag{2}$$

其中 $c_{\max}$ 可以是一个合适的输入值,或者是到目前为止,进化过程中 $f(x)$ 的最大值,也可以是当前代中 $f(x)$ 的最大值.

在遗传算法中,为了避免少数个体霸占整个群体而导致的"早熟"现象以及因平均适应度接近最佳适应度而使优化过程趋于随机漫游的现象,人们引入了适应度调整的方法[2],即适应度标定.常见的标定方式有线性标定、σ 截断、幂率标定、对数标定、窗口技术、正规化、Boltzmann 选择、排序等.

下面,我们将给出两种利用目标函数梯度(即变化率)的改进遗传算法.

2.2 梯度改进的遗传算法(GMGA)

考虑如下连续函数的无约束极小化问题:

$$\min f(x_1,x_2,\ldots,x_n),\ x_i\in[a_i,b_i],\ i=1,2,\ldots,n. \tag{3}$$

原始 $f(x)$ 的适应度函数可取式(2)的形式,如果需要,还要选取相应的适应值标定方式.引进适应度函数是为了衡量遗传种群个体的优劣程度,对连续函数求极小问题而言,自然应该要求各个个体的适应度函数值越小越好.但是,在一些函数求极值的问题中,为了求函数的极小(即下山),往往不但要选择函数值小的搜索点,而且还必须保证该搜索点的函数值变化率足够大,即只有合适地选择那些函数值小而且函数值变化率大的点作为下山过程的搜索点,才能使下山过程迈大步.否则,就可能使下山过程在一块"平坦的高原"上来回漫游,而很难达到欲到达的"山谷".本文把这种思想借用到遗传算法中来.在选取适应度函数时,充分考虑目标函数的变化率,对可微函数而言就是利用其梯度,提出选用如下新的适应度函数:

$$fit'(x)=\lambda\cdot\frac{fit(x)-fit_{\min}}{fit_{\max}-fit_{\min}}+(1-\lambda)\cdot\frac{\|\nabla f(x)-\nabla f_{\min}\|}{\|\nabla f_{\max}-\nabla f_{\min}\|}. \tag{4}$$

其中:权值 $\lambda\in[0,1]$,称为控制因子,用以体现适应度函数值和函数值变化率对欲解问题的重要程度;$fit(x)$ 为由式(2)定义的原始适应度函数,$fit_{\min}$ 和 $fit_{\max}$ 分别表示当前代群体中个体适应度的最小值和最大值(群体中个体数>1);$\nabla f(x)$ 为目标函数 $f(x)$ 在点 x 的梯度,$\nabla f_{\min}$ 和 $\nabla f_{\max}$ 分别定义为

$$\nabla f_{\min}=\left(\min\left\{\frac{\partial f(v_1)}{\partial(v_1)_1},\ldots,\frac{\partial f(v_{ps})}{\partial(v_{ps})_1}\right\},\ldots,\min\left\{\frac{\partial f(v_1)}{\partial(v_1)_n},\ldots,\frac{\partial f(v_{ps})}{\partial(v_{ps})_n}\right\}\right), \tag{5}$$

$$\nabla f_{\max}=\left(\max\left\{\frac{\partial f(v_1)}{\partial(v_1)_1},\ldots,\frac{\partial f(v_{ps})}{\partial(v_{ps})_1}\right\},\ldots,\max\left\{\frac{\partial f(v_1)}{\partial(v_1)_n},\ldots,\frac{\partial f(v_{ps})}{\partial(v_{ps})_n}\right\}\right), \tag{6}$$

其中 $(v_k)_i$ 表示向量 v_k 的第 i 个分量,$\|\cdot\|$ 表示某种范数,$fit'(x)$ 表示新的适应度函数,ps 表示种群规模.

其实不难看出,上面提出的是"一类"算法,读者可以根据从下山优化算法中借用来的上述思想,构造各种各样的既考虑函数值又充分考虑函数值变化率的适应度函数,并且适当调整权值 λ,从而形成各种各样的"利用目标函数变化率的遗传算法".

为了实现上述遗传算法,只需在一般的十进制实数编码的遗传算法中,在计算适应度函数值时增加如下 3 个步骤:

步骤 1. 对所有染色体 $v_k,k=1,2,\ldots,ps$,计算函数 $f(v_k)$ 的梯度 $\nabla f(v_k)$;

步骤 2. 计算群体中 $fit_{\min},fit_{\max},\nabla f_{\min}$ 和 $\nabla f_{\max}$;

步骤 3. 对所有染色体 $v_k,k=1,2,\ldots,ps$,若 $|fit_{\max}-fit_{\min}|<\varepsilon$ 或 $\|\nabla f_{\max}-\nabla f_{\min}\|<\varepsilon$,则 $fit'(v_k)=fit(v_k)$,否则按照式(4)计算 $fit'(v_k)$,其中 ε 为一个小正数.

2.3 离散的梯度改进遗传算法(DGMGA)

考虑如下离散问题的无约束极小化:

$$\min f(x_1,x_2,\ldots,x_n),\ x_i\in\{0,1,\ldots,K\},\ i=1,2,\ldots,n. \tag{7}$$

原始 $f(x)$ 的适应度函数仍取式(2)的形式,如果需要,再选取相应的适应值标定方式.由于 $f(x)$ 不存在梯度,故不能像连续情形那样将梯度信息加入适应度函数,但可以利用相邻两代,即父代和子代的一阶差分代替梯度.适应度函数以如下方式选取:

$$fit'(x)=\lambda\cdot\frac{fit(x)-fit_{\min}}{fit_{\max}-fit_{\min}}+(1-\lambda)\cdot\frac{\widetilde{\nabla} f(x)-\widetilde{\nabla} f_{\min}}{\widetilde{\nabla} f_{\max}-\widetilde{\nabla} f_{\min}}. \tag{8}$$

其中:$\lambda,fit(x),fit_{\min},fit_{\max},fit'(x)$ 的意义同上;$\widetilde{\nabla} f(x)$,$\widetilde{\nabla} f_{\min}$ 和 $\widetilde{\nabla} f_{\max}$ 分别定义为

$$\widetilde{\nabla} f(x)=f(x^p)-f(x^c), \tag{9}$$

$$\widetilde{\nabla} f_{\min}=\min\{f(v_1^p)-f(v_1^c),\ldots,f(v_{ps}^p)-f(v_{ps}^c)\}, \tag{10}$$

$$\widetilde{\nabla} f_{\min}=\max\{f(v_1^p)-f(v_1^c),\ldots,f(v_{ps}^p)-f(v_{ps}^c)\}, \tag{11}$$

其中 x^p,x^c 分别表示父代和子代染色体.

若采用二进制编码的简单遗传算法,只需在选择适应度函数时增加如下 3 个步骤:

步骤 1. 对所有染色体 v_k,$k=1,2,\ldots,ps$,按照式(9)计算函数 $f(v_k)$的差分 $\widetilde{\nabla} f(v_k)$;

步骤 2. 计算群体中 $fit_{\min}$,$fit_{\max}$,$\widetilde{\nabla} f_{\rm m}\mathrm{in}$ 和 $\widetilde{\nabla} f_{\max}$;

步骤 3. 对所有染色体 v_k,$k=1,2,\ldots,ps$,若 $|fit_{\max}-fit_{\min}|<\varepsilon$ 或 $|\widetilde{\nabla} f_{\max}-\widetilde{\nabla} f_{\min}|<\varepsilon$,则 $fit'(v_k)=fit(v_k)$,否则 $fit'(v_k)$按照式(8)来计算,其中 ε 为一个适当选取的小正数.

3 算例结果及分析

3.1 连续问题算例

考虑以下无约束优化问题:

$$\max f(x_1,x_2)=21.5+x_1\sin(4\pi x_1)+x_2\sin(29\pi x_2),\quad -3.0\leqslant x_1\leqslant 12.1,\ 4.1\leqslant x_2\leqslant 5.8. \tag{12}$$

目标函数的三维图形如图 1 所示.

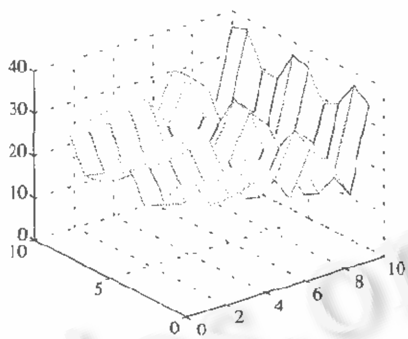

Fig. 1 Object function

图 1 目标函数

分别利用标准十进制编码的遗传算法(SGA)和目标函数梯度的改进遗传算法(GMGA)搜索目标函数的极值点.染色体群体规模取 10,交叉率取 0.3,变异率取 0.1.这 3 次计算的结果见表 1.

Table 1 Computational results by algorithm SGA and algorithm GMGA

表 1 SGA 算法与 GMGA 算法的计算结果

Algorithm SGA①		Algorithm GMGA②	
Iteration number③	Maximum④	Iteration number	Maximum
155	38.86	108	38.86
111	38.83	21	38.84
128	38.80	39	38.85

①SGA 算法,②GMGA 算法,③代数,④极大值.

3.2 离散问题算例

文献[4]中的配词问题,即试图用遗传算法将随机产生的字母序列变为短语“to be or not to be". 用小写字母的 ASCII 整数来编码,标准遗传算法适应值取为匹配的字母数,变异定义为以一定的概率取一个小写字母. 染色体群体规模取 10,交叉率取 0.8,变异率取 0.7. GA 算法与DGMGA 算法取每代中最好的字符串,学习结果见表 2.

Table 2 Comparison between SGA and DGMGA in the discrete case

表 2 离散情形的 SGA 算法与 DGMGA 算法的比较

Iteration number①	Algorithm SGA②		Algorithm DGMGA③	
	String④	Fitness⑤	String	Fitness
1	wxbxsidlcoowy	2	vafcdidanqovc	2
11	tabepfnvztbhh	5	tqbelqnktufte	5
21	tobepvnvvtahu	6	tobepfnottowe	10
31	tobeprnvvtahe	8	tobepxnottowe	12
41	tobeounvvtohe	9	tobeornottobe	13
51	tobeornvltohe	10		
61	tobeornvttobe	12		
71	tobeornwttobe	12		
81	tobeornattobe	12		
88	tobeornottobe	13		

①代数,②SGA 算法,③DGMGA 算法,④字串,⑤适应值.

需要指出的是,上述算法的适应度函数值的计算比一般遗传算法要复杂一些,所以在实际计算时间上需要付出一定的代价.

4 结 论

多数遗传算法在利用问题域进行近似解搜索的时候,例如求函数极值问题时,往往只考虑在搜索点的函数值信息,而没有充分考虑函数的变化趋势或变化率. 简单地按概率随机选择的染色体不一定是一个"好的"染色体. 本文考虑了目标函数在搜索点的变化趋势,并将该信息加入适应度函数,使得按概率选择的染色体不但具有较小的函数值(对极小化问题而言),而且具有较大的变化率,同时保持了遗传算法以小概率发生染色体交叉和单体变异的思想. 从算法的计算复杂度而言,需要计算目标函数的梯度或差分,但总的计算复杂度与标准遗传算法同阶. 实验结果表明,该方法的收敛速度有较大的提高.

References:

[1] Chen, Guo-liang, Wang, Xu-fa, Zhuang, Zhen-quan, *et al*. Genetic Algorithms and Their Applications. Beijing: People's Post and Telecommunications Publishing House, 1999 (in Chinese).

[2] Xuan, Guang nan, Cheng, Run-wei. Genetic Algorithms and Engineering Design. Beijing: Science Press, 2000 (in Chinese).

[3] Michhalewicz, Z. Genetic Algorithm Data Structure Evolution Programs. 2nd ed., New York: Springer-Verlag, 1994.

[4] Freeman, J. Simulating Neural Networks with Mathematics. New York: Addison-Wesley Publishing Company, 1994.

附中文参考文献:

[1] 陈国良,王煦法,庄镇泉,等. 遗传算法及其应用. 北京:人民邮电出版社,1999.

[2] 玄光男,程润伟. 遗传算法与工程设计. 北京:科学出版社,2000.

Genetic Algorithms Using Gradients of Object Functions*

HE Xin-gui[1], LIANG Jiu-zhen[2]

[1](*Beijing Institute of System Engineering*, *Beijing* 100101, *China*);

[2](*Department of Computer Science*, *Daqing Petroleum Institute*, *Daqing* 151400, *China*)

E-mail: hexingui@ns.cetin.net.cn; liang_jz@263.net

Abstract: Most genetic algorithms do not use the knowledge in the related problem fields completely when searching the approximate solutions. A new kind of genetic algorithm with modified fitness functions the presented in this paper. In this algorithms, both the function value at the searching point and the function change rate at the point are combined into fitness functions. It makes the chromosome code chosen by probability be able to have both smaller function value (for minimum problem) and higher function change rate. The experimental results show that the new algorithm is convergent much faster than the standard genetic algorithm is.

Key words: SGA (standard genetic algorithm); GMGA (gradient modified genetic algorithm); DGMGA (discrete gradient modified genetic algorithm); fitness function; optimization

---

* Received January 5, 2001; accepted March 5, 2001