遗传算法机理的研究*

张铃[1,3] 张钹[2,3]

[1](安徽大学人工智能研究所 合肥 230039)

[2](清华大学计算机科学与技术系 北京 100084)

[3](清华大学智能技术与系统国家重点实验室 北京 100084)

E-mail: zling@ahu.edu.cn

摘要 众所周知,"模式定理"和"隐性并行性"是遗传算法(genetic algorithms,简称GA算法)的两大理论基础.该文对这两个原理进行分析,指出这两个原理存在有不严格和不足之处,即作为GA算法的基础,这两个原理尚欠完善.为加深对GA的理解,文章提出遗传算法的一个新的改进模型——理想浓度模型.通过对此模型的分析,得出遗传算法本质上是一个具有定向制导的随机搜索技术.其定向制导原则是,导向以适应度高的模式为祖先的染色体"家族"方向.最后给出两个典型的函数求最大值的模拟例子.从模拟结果看,改进后的GA算法大大提高了算法的速度,解的精度也有所提高.这说明新算法具有应用的潜力.

关键词 遗传算法,模式定理,隐性并行性,遗传算法的理想浓度模型.

中图法分类号 TP18

遗传算法由于其简单易行,在各个领域都得到了广泛应用,但是,GA(genetic algorithms)算法的运行机理并不十分清楚.对于GA算法的运行机理,也有一些研究者进行过研究.Radolph在文献[1]中证明一般的GA算法不收敛,只有每次保存最优个体时才收敛.Perey在文献[2]中就一类特殊的GA算法,给出收敛的充要条件.Qi和Palmieri在文献[3]中证明,当群体的容量无限大时,算法是收敛的.Fogal[4]和Eiben[5]证明了一类抽象GA算法的收敛性.另一方面,人们普遍认为,GA算法之所以高效且具有强鲁棒性,是因为遗传算法中有"模式定理"和"隐性并行性"性质成立之故[6].于是到目前为止,这两个性质被看作是遗传算法理论的两大基石.本文将对这两个性质进行分析研究,指出其不足之处,并在此基础上提出遗传算法的一个新的改进模型——理想浓度模型.通过对此模型的分析,得出结论:遗传算法本质上是一个具有定向制导的随机搜索技术,其定向制导原则是,导向以适应度高的模式为祖先的染色体"家族"方向.最后,给出两个典型的函数求最大值的模拟例子,从模拟结果看,改进后的GA算法大大提高了算法的速度,解的精度也有所提高.新算法具有应用的潜力.

1 模式定理的分析

在文献[7]中,我们曾对模式定理进行过分析,指出模式定理的推论"结构块(即高适应度、低阶、短确定长的模式)在遗传过程中其个数将随代数按指数律增加"在一般情况下是不成立的,这里不再重复,详见文献[7].

关于结构块是否存在的问题也是很值得商讨的,因为从统计上来看,低阶一般就不会是高适应度.

下面给出有关模式之间关系的简单的性质.

定义1. 设$H1$,$H2$是两个给定的模式,若$H1$的确定分量的集合是$H2$确定分量集合的子集,则记为$H1>H2$,称$H2$是$H1$的子模式.

* 本文研究得到国家自然科学基金(No.69675011)、国家863高科技项目基金(No.863-306-05-08 3)和国家973高科技项目基金(No.G1998030509)资助.作者张铃,1937年生,教授,博士生导师,主要研究领域为人工智能理论,人工神经网络理论.张钹,1935年生,教授,博士生导师,中国科学院院士,主要研究领域为人工智能理论及应用,计算机应用.

本文通讯联系人:张铃,合肥 230039,安徽大学人工智能研究所

本文1998-12-29收到原稿,1999-06-17收到修改稿

性质 1. 若 $O(H1)=m$,令 $\boldsymbol{H}$ 是池中 $H1$ 的所有子模式集合,则必存在一 $H2\in\boldsymbol{H}$,满足 $f(H1)\leqslant f(H2)$.

证明:由 $\boldsymbol{H}$ 是 $H1$ 的子模式集合,得到

$$f(H_1)=\sum_{H\in\boldsymbol{H}}f(H)/n,$$

其中 n 是 $\boldsymbol{H}$ 的规模.故得到必存在一 $H2\in\boldsymbol{H}$,使得 $f(H1)\leqslant f(H2)$. □

注:除特殊情况外,上式一般取不等号.这说明低阶者其适应度不会很高.这个结论给结构块的存在蒙上了阴影.

2　遗传算法中"隐性并行性质"的分析

让我们再来分析作为现有遗传算法理论的另一基石:遗传算法的隐性并行性质.

"隐性并行性质"的大意是:遗传算法有效处理的模式总数正比于群体规模 n 的立方,即其阶为 $O(n^3)$[6,8,9].

在进行分析之前,先作几个约定:

染色体 x,记为 $x=(x_1,x_2,\ldots,x_m)$,其中 $x_i\in\{0,1\}$.

模式 H,记为 $H=(h_1,h_2,\ldots,h_m)$,其中 $h_i\in\{0,1,*\}$.

称染色体 x 属于模式 H,若对一切 $h_i\neq *$,有 $x_i=h_i$ 成立.这样,模式 H 就可理解为所有属于模式 H 的染色体的集合.而对一给定的染色体池,可理解为池中所有属于 H 的染色体的集合.

为正确分析"隐性并行性定理",我们摘录文献[8]中一段有关的证明(此段证明基本上是文献[6]相关段落的译文)如下:

"给定一染色体,求其中有多少个定义距$\leqslant l_s$ 的模式.固定第 l_s 位,计算确定分量在前 l_s 位的模式的个数:...显然,这样的模式数为 2^{l_s-1}.为计算整个串中的这类模式,我们将上面的底线向右移动一位.一般地,对于长度为 l 的串,计算定义距小于等于 l_s 的模式数,需将上述底线移动 $l-l_s+1$ 次,由此得出一个长度为 l 的串,定义距小于等于 l_s 的模式数为 $2^{l_s-1}(l-l_s+1)$.对于群体数为 n,则此类模式总数为 $n2^{l_s-1}(l-l_s+1)$.显然,这个结论在群体规模较大的情况下存在着重复计数的问题.为了更准确些,取群体数 $n=2^{l_s}/2$,由此希望阶数大于等于 $l_s/2$ 的模式最多重复计数一次.另一方面.考虑到模式数的分布呈二项式分布,则阶数高于 $l_s/2$ 的模式与低于 $l_s/2$的模式的数目大致相等,各占一半.如果我们只考虑高阶的部分,则有关模式数的下界为

$$n_s\geqslant n(l-l_s+1)2^{l_s-2}.$$

考虑到 $n=2^{l_s/2}$,则有

$$n_s\geqslant(l-l_s+1)n^3/4,$$

即有 $n_s\geqslant cn^3=O(n^3)$,其中 c 是常数.由此得出结论:遗传算法有效处理的模式总数正比于群体数 n 的立方,即 $O(n^3)$".

首先指出,上面的证明在两处有误.一个是在推导一个染色体有多少个定义距$\leqslant l_s$ 的不同的模式数时,所得出的结论是 $2^{l_s-1}(l-l_s+1)$,正确的应为 $2^{l_s-1}(l-l_s+2)$,因为它少算了确定分量落在前 l_s-1 位的模式.这只要看一下下面简单的例子就可明白.如取染色体 $x=(1011)$,$l=4$,$l_s=3$,则按文献[6]中的公式得有 $4\times(4-3+1)=8$,直接计算可得 x 阶$\leqslant 3$ 的模式共有 14 个(1 阶的 4 个,2 阶的 6 个,3 阶的 4 个),而不是 8 个.另一个是,在推导公式 $n_s\geqslant n(l-l_s+1)2^{l_s-2}$时也有差错.它是在推导过程中简单地将 $(l-l_s+1)2^{l_s-1}$除以 2 而得到.这是不对的,因为文献[6]中只证明:池中阶大于 $l_s/2$ 的模式个数与阶小于 $l_s/2$ 模式的个数一样多,而阶小于 $l_s/2$ 的模式在池中不只重复计数一次,而是很多次,故池中不同模式的个数就不能用简单地除以 2 而得到.下面举个简单的例子加以说明.设池中阶大于和小于 $l_s/2$ 的模式各有 a 个,前者的模式平均重复计数 0.6 次,后者的模式平均重复计数 3 次,那么池中(包括重复计数的)模式个数应为 $a(1.6+4)=5.6a$,于是求 $2a$,必须将总数除以 2.8,而不是 2.

现在来分析 Goldberg 的证明.Goldberg 在文献[6]中证明隐性并行性质时,主要利用下面的结论:"池中阶$\geqslant l_s/2$ 的模式平均至多重复计数一次,即在池中至多有两个不同的染色体属于同一模式."

当模式被看成是染色体的集合时,其适应度值等于池中属于该模式的染色体的适应度的均值."均值"只有在统计的个数达到一定程度时才有意义,而如上面所述,这里处理的模式至多只有两个不同的染色体,故由两个

染色体计算得到的均值,其代表性是很值得怀疑的.这样的均值,不能代表对应模式的性质(也就是说,其对应的模式并没有得到有效的处理),它只能代表个体的性质.故隐性并行性质中的"有效处理模式..."的性质就不存在.于是,在 n 个染色体的池中能有效处理的仍旧只是 n 个染色体个体本身,并没有如隐性并行性质所说,能有效地处理对应的模式.

另外,文献[6]中取 $n=2^{l_s/2}$ 是为了使池中有一半的模式的重复计数不多于1次,而恰恰是这一点使池中很多模式因规模太小在计算适应值时失去统计学的意义,也就是说不能被有效处理.于是,文献[6]中取 $n=2^{l_s-1}$ 的根据就很不足了.

总之,从以上分析来看,作为现有遗传算法理论的另一基石"隐性并行性质"也是有些问题的.至少,目前这种解释遗传的有效性的理论是有问题的.

下面,让我们沿文献[6]作者的思路来计算在含有 n 个染色体的池中,定义距$\leqslant l_s$ 模式的平均个数.

先计算在含有 n 个染色体的池中,定义距$\leqslant l_s$ 模式的平均规模 d:

固定一染色体 h,设 h 属于模式 H,不妨设 H 的确定分量位于中间连续 l_s 位.

设 $\boldsymbol{H}$ 为 H 的所有子模式的集合,由文献[8]中隐性并行性质的证明可知,$\boldsymbol{H}$ 共有 2^{l_s-1} 个模式.取 $\boldsymbol{H}$ 中阶数为 i 的模式 $H1$,那么随机取另一染色体 h_1,h_1 属于 $H1$ 的概率为 2^{-i}.另一方面,在 $\boldsymbol{H}$ 中共有 $C_{l_s}^{i}$ 个 i 阶子模式,故折算后,共有 $(n-1)2^{-i}C_{l_s}^{i}$ 次重复的次数.

于是,总的平均重复数 e 为

$$e=2^{-l_s}\sum_{i=1}^{}(n-1)2^{-i}C_{l_s}^{i}=2^{-l_s}(n-1)\left[\sum_{i=0}^{}2^{-i}C_{l_s}^{i}-1\right]=2^{-l_s}(n-1)[(1+1/2)^{l_s}]\leqslant n2^{(1.58-2)l_s}=n2^{-0.42l_s}.$$

注:其中用到 $3\approx 2^{1.58}$ 和 $(1+1/2)^{l_s}$ 的二项式展开公式.

另一方面,池中所有可能的模式个数(其中有重复计算)为

$$n(l-l_s+1)2^{l_s-1}.$$

于是,池中模式的平均规模 $d=e+1$(因为平均重复计数 e 次,即等于总共平均计数 $e+1$ 次).

将上式除以 d,得到池中不同模式的平均数 n_s:

$$n_s\approx O(2^{1.42l\times r-1})=O(2^{1.42l\times r}).$$

例:若按文献[8]中的假设取 $n=2^{l_s/2}$,得 $d=2^{0.08\times l_s}$,取 $l_s=10$,得 $d<2$.这样的 d 对计算模式的适应值就太小了.故除非 l_s 很大(但我们关心的是结构块,而结构块是低阶的,故 l_s 不宜过大),否则,在证明隐性并行性质时,取 $n=2^{l_s/2}$ 就缺乏依据.

为使计算模式的适应度值在统计学上有意义,必须要求 d 充分大,这就要增加 n 的规模,如取 $n=2^{\frac{3}{4}l_s}$.

由上式得 $n_s\approx O(n^2)$,则 n 与 n_s 的关系就不是立方的关系,而是平方的关系.

还有一个问题是,"隐性并行性质"说:"遗传算法有效地处理模式...",何谓"有效处理"?这种"有效处理"对求最优值有何影响?均不清楚.

3 遗传算法机理分析

从上面的分析可以看出,到目前为止,人们对GA算法的机理尚不清楚.为加深对遗传算法的机理的理解,我们对GA算法进行探讨,提出一些看法.这些看法当然也不尽完善,只是起一个"抛砖引玉"的作用.

3.1 几个概念

定义 3.1(浓度). 模式 H 的分量 H_i 于时刻 t 时在池中的浓度 $c(H_i,t)$,定义为池中各染色体的第 i 分量,等于 H_i 的个数除以 N(N 是池的规模).

模式 H 各分量的浓度的几何平均值 $c(H,t)$,称为 H 在时刻 t 在池中的平均浓度.

设在染色体池中,开始时的浓度是均匀的,现按遗传算法进行操作,开始时适应度高的染色体的浓度将逐步增加,池中的平均适应度值也逐步增加.下面我们来分析一下各模式的浓度变化情况.

定义 3.2. 设染色体 $e=(e_1,\ldots,e_n)$,$d=(d_1,\ldots,d_n)$,定义 $e\times d=g$,$g_i=\begin{cases}e_i, & e_i=d_i\\ *, & \text{其他}\end{cases}$,则称 g 所代表的模

式为以染色体 e 与 d 为祖先的"家族".

浓度计算. 对池进行复制时,其浓度将发生变化,其变化规律为

$$c(H_i,t+1)=c(H_i,t)\frac{f(H_i(C),t)}{\overline{f(t)}},$$

其中 $f(\cdot)$ 为适应度,$\overline{f(t)}$ 是 t 时刻的平均适应度.

上式说明,复制操作的目的在于,使模式的浓度与其适应度成正比.

下面来分析杂交操作对浓度的影响,即要计算在进行杂交时,何时会使原有的模式消失,何时会使模式个数增加,各增加多少.这种计算相当复杂,为便于分析,我们引入一个假设和模型,并以此进行上述计算.

理想浓度假设. 设在任一时刻,各分量的浓度是相互独立的.

遗传算法的理想浓度模型. 凡满足理想浓度假设的遗传算法模型.

3.2 杂交操作对模式浓度的影响

有了这个模型,我们就可以来计算在杂交操作下模式的生灭率.

设第 i 分量=0 的浓度为 $c_i(t)$,假设染色体 $A=(a_1,a_2,\ldots,a_n)$,按理想浓度假设,则 A 在模型中出现的概率为 $\prod c_i(a_i,t)$(这里,$c_i(a_i,t)$ 表示当 $a_i=0$,$c_i(0,t)=c_i(t)$,当 $a_i=1$ 时,$c_i(1,t)=1-c_i(t)$).

注:理想浓度假设是一个很强的假设,在实际情况下一般不易满足,这里是为了易于进行分析而引进的.

池中模式个数的估计. 设模式 H,其定义距=$\delta(H)$,其阶=l,在时刻 t 时,其适应度为 $f(H,t)$,此时池中的平均适应度为 $\overline{f(t)}$,设现取杂交概率为 p_c.下面推导在杂交操作(杂交操作在复制操作之后进行)下,池中 H 个数变化的情况.情况无非有两种:一种是原来属于 H 的模式经杂交后,消失了;另一种是原来不属于 H 的模式,经杂交后产生出新的属于 H 的模式.现具体加以分析.

设杂交时所选取的两染色体为 h_1,h_2,设其染色体长度为 L.

(1) 若两者都属于 H,则杂交后的子染色体 h'_1,h'_2 必属于 H.即杂交前后属于 H 的染色体个数不变.

(2) 若 h_1,h_2 中有一个属于 H,则要分几种情况进行讨论.

(2.1) 设 h_1 属于 H,h_2 不属于 H(或反之),若 h_2 只有一个分量不属于 H,则 h'_1,h'_2 中恰有一个属于 H.即杂交前后属于 H 的染色体个数不变.

(2.2) 一般地,若 h_1 属于 H,h_2 恰有 $i(i\geqslant 2)$ 个分量不属于 H,当划分点落在此 i 个分量之间时,则 h'_1,h'_2 均不属于 H.此时经杂交后,属于 H 的染色体少了一个.

经推导可得到这种情况的概率为

$$c(H,t)^1\left\{\sum_{i=2}^{l}C_{l-1}^{i-1}(H,t)^{i-1}(1-c(H,t))^{i}\left[(i-1+\delta(H))/(L-1)\right]\right\}. \tag{1}$$

化简得到

$$a_2=c(H,t)^1\left\{\frac{\delta(H)-1}{(L-1)}-\frac{l(1-C(H,t))}{(L-1)}+\frac{(\delta(H)-1)c(H,t)^1}{(L-1)}-\frac{l(\delta(H)-1)c(H,t)^{l-1}(1-c(H,t))}{(L-1)}\right\}. \tag{2}$$

(3) 当 h_1,h_2 均不属于 H 时,令 $h_1=(h_{11},h_{12})$,$h_2=(h_{21},h_{22})$,$h'_1=(h_{11},h_{22})$,$h'_2=(h_{21},h_{12})$.要 h'_1(或 h'_2)属于 H,则 h_{11}(或 h_{21})必须(比如(h_{11},h_{22})属于 H)恰好是 H 的前半部,h_{22}(或 h_{12})正好是 H 的后半部.这种情况的概率为

$$a_3=2\delta(H)c(H,t)^1(1-c(H,t))^2/(L-1). \tag{3}$$

因篇幅所限,证明略.

综合上述分析可得,经杂交后,最后池中模式 H 的生灭率为

$$(2c(H,t)^1+(a_2-a_3))/2c(H,t)^1=1+(a_2-a_3)/2c(H,t)^1.$$

将式(2)、(3)代入上式,经杂交后得到 H 模式在池中的生灭率为

$$\eta=[1+2\delta(H)(1-c(H,t))^2-\delta(H)+1-l(1-c(H,t))+(\delta(H)-1)c(H,t)^1+l(\delta(H)-1)c(H,t)^{l-1}(1-c(H,t))]/2(L-1). \tag{4}$$

现令 $b=\eta-1$,下面分析 b 变化的情况.

这样就得到经复制和杂交后,池中模式 H 浓度的递推公式:

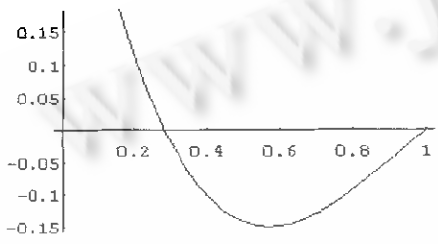
$$c(H,t+1)=c(H,t)\frac{f(H,t)}{\overline{f(t)}}(1+p_c b). \tag{5}$$

在式(4)中取 $l=3,\delta(H)=6,L=8$,得 b 随 c 变化的曲线,如图 1(a)所示.

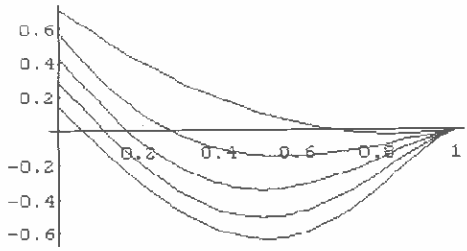
另外,图 1(b)为其他参数不变,阶不同时(5 条曲线自上至下其阶分别为 2,3,4,5,6)对应的 b 变化的情况.图 1(c)为其他参数不变,定义距不同时(5 条曲线自左至右其距分别为 3,4,5,6,7),b 变化的情况.

由图 1(a)可以看出,当 $c>c_0$ 之后,$b<0$,经杂交后,对适应度不大于平均适应度的模式 H 的个数将减少.从图 1(b)可以看出,低阶的模式杂交后其个数增加的可能性比高阶的模式大.从图 1(c)可以看出,当浓度小于 0.5 时,定义距长的模式其个数增加的可能性比定义距短的模式大;当浓度大于 0.5 时,情况正相反.其中各条曲线的交点为(0.5,−0.142857).

取 $l=3,\delta(H)=6,L=8$,按式(5)求浓度变化的情况,我们求出了 $c(H,t+1),c(H,t+2),c(H,t+3)$ 与 $c(H,t)$ 的关系,如图 2 所示.从图 2 可以看出,当 $c<c_1$ 时,H 的浓度呈增加趋势;当 $c>c_1$ 时,浓度呈下降趋势.其中 $c_1=0.28211$,此点是不动点.

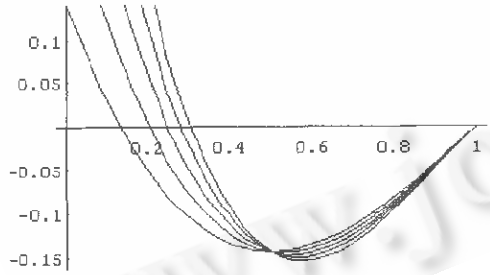


(a)



(b)

Fig. 1
图 1



(c)

Fig. 1
图 1

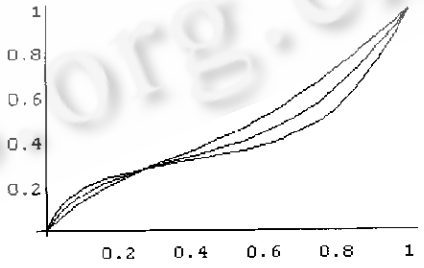


Fig. 2
图 2

在式(5)中取 $p_c=1$,当 $c_0<c<c_1<1$ 时,因子 $(1+b)<1-d(d>0)$.

另一方面,因子 $f(H,t)/\overline{f(t)}$,当 t 增加时,将趋向 1,即当 t 增加时,式(5)的左边将小于右边.

这个结果说明,若染色体池的容量无限,则经遗传算法作用后,池中原有的各模式的浓度均呈下降趋势(可能除个别适应度特别高以及浓度特别低的模式外),由此推得池中必有新的模式产生.

模式种类过多将影响遗传算法的效率,故在资源有限的情况下,还必须进行"优胜劣汰"的选择.如每次经过遗传操作后(一般染色体个数增加一倍),将池中适应度最差的一半的模式删去.通过这种"优胜劣汰"后,余下的模式其浓度一般就增加一倍.由此可见,"优胜劣汰"的操作对模式的浓度影响很大.下面简单加以讨论.

3.3 优胜劣汰的影响

优胜劣汰操作:取数 $A(0<A<1)$,设池中的染色体经复制、杂交、变异后,将池中适应度最差的 $A\times 100\%$ 的染色体淘汰掉,对余下的按比例伸缩成 N 个,称为淘汰率为 A 的优胜劣汰操作.

性质 2. 池 K 经淘汰率为 A 的优胜劣汰操作后,设一模式 H,若 H 的全体均未被淘汰,则淘汰后,H 在池中的浓度至少是原来的 $1/(1-A)$ 倍.

结论显然,如取 $A=0.5$,则未被淘汰的模式其浓度将增加一倍.由此可见,优胜劣汰操作对高适应度的模式的浓度增加,甚至比复制的影响更大(特别在遗传操作进行过多次之后,这时平均适应度值趋向最高适应度值,故复制对浓度的影响减弱),这个结果是原先未曾料到的.

现将遗传算法的全过程回忆一遍:经复制后,将池中的各模式的浓度调整成与其适应度成正比;其次,经杂交后,池中原有的模式的浓度普遍下降,一般会产生新的模式;进行"变异"操作,所引起的浓度变化是随机的,与一般的统计方法相似;最后,进行"优胜劣汰"后,适应度高的模式的浓度将成倍增高(这时,在一般情况下,不同的模式个数会减少).

由此可以看出,对浓度影响最大的是"复制"和"优胜劣汰"这两个操作,特别是后者.若不进行"优胜劣汰"操作,则模式个数过多,将延长求到最优解的时间,若按 50% 进行"优胜劣汰"操作,则使适应度高的模式的浓度过快地增大,容易引起"早熟".据此,似应采取"适度优胜劣汰"方法,如取百分比 A(开始时小些,以后逐步增大),将杂交后的染色体池,先淘汰掉百分比 A 适应度最差的模式,对余下的部分再按"按比例淘汰"法进行淘汰.这样既能保证资源的合理使用,又能保证模式种类的多样性,避免陷入"早熟"情况.从以上分析最后可得遗传算法的机理为:遗传算法是一个具有定向制导的随机搜索技术,其定向制导的原则是:导向以高适应度模式为祖先的"家族"方向.

从上面的分析得知,一个具体问题用遗传算法进行求解是否有效,就看它是否满足:以高适应度模式为祖先的"家族"中,其成员具有高适应度的概率比其他家族成员高.用一个谚语来描述,那就是:必须具有"龙生龙,凤生凤"的特点.在自然进化和在人工进行培育优良品种时,其根据的道理也在于此.若某个问题不满足上述特性时,遗传的定向制导就可能产生误导作用,如所谓的"欺骗问题",就是这种误导的结果.

根据上面的分析,我们可以对遗传算法进行如下的改进.

遗传算法的改进:

(1) 取原始池 K,其染色体个数为 N.

(2) 对池 K 进行遗传操作:复制、杂交、变异.

(3) 在杂交后,计算出当时理想浓度最高的 p 个值,并求出其对应的染色体(注意,这些染色体可能不在池中),然后将它们加到池中(p 是预告给定的整数).

(4) 给出一个数值 $A(t)$,对第 t 代池中的染色体,先将适应度最低的 $NA(t)$($A(t)$ 是一个关于 t 的递增函数,且 <0.5,N 是池中染色体的个数)染色体淘汰掉,然后再对余下的染色体进行"按比例淘汰",最后在池中留下 N 个染色体.若满足结束条件,则停止,不然,返回第(1)步.

上面的改进,主要是为了避免出现"早熟"现象,同时尽量保持模式的浓度与其适应度成正比.但对如何在以高适应度模式为祖先的家族中,选取个体以提高算法的效率的问题尚未涉及,我们将在另文讨论.

4 模拟例子

下面我们举两个十分典型的函数,用改进过的 GA 算法和一般的 GA 算法进行求解比较,结果十分令人满意.

例 1:

$$\max f_1(x)=|(1-x)\times x^2\times\sin(200\pi x)|,$$

$$\text{s.t.}\quad 0\leqslant x\leqslant 1.$$

这个函数在求解的定义域内有 200 个极大、极小值,用其他方法(如图 3 所示)很难求解.

例 2:

$$\max f_2(x)=(1-2\sin^{20}(3\pi x)-\sin^{20}(20\pi x))^{20},$$

$$\text{s.t.}\quad 0\leqslant x\leqslant 1.$$

这个函数在求解的定义域内有 20 个极大、极小值,且其变化速度非常大(如图 4 所示),用其他方法也是很难求解的.

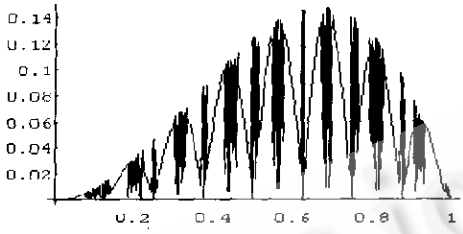


Fig. 3 The image of function f_1

图 3 函数 f_1 的图像

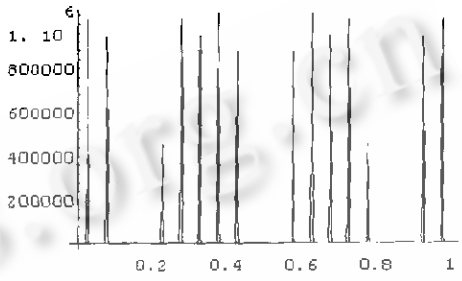


Fig. 4 The image of function f_2

图 4 函数 f_2 的图像

用 GA 算法和改进的 GA 算法解例 1,例 2 的结果见表 1(表 1 是解题 20 次得到的最大值的均值).

Table 1

表 1

Algorithm①	$f_1(x)$			$f_2(x)$		
	Times②	Time③	Maximum④	Times	Time	Maximum
GA⑤	80	0.659 3	0.128 699	120	1.112 6	454 176.219
Improved GA⑥	8	0.494 5	0.148 092	18	0.334 9	1 007 603.125

①算法,②次数,③时间,④最大值,⑤GA 法,⑥改进的 GA 法.

改进后的 GA 算法收敛速度(指迭代次数)比普通 GA 算法几乎快了一个数量级,精度也提高不少,特别是例 2 的最大值提高一倍多.速度提高这么快是未曾料到的(改进后的 GA 算法的参数,开始时取 $A(t)=0.2$,之后每 5 代增加 0.05,直到 $A(t)=0.5$ 为止).

5 结 论

本文首先分析了 GA 算法中的模式定理和隐性并行性质的两大定理,指出其不足之处.在此基础上,为搞清 GA 算法的机理,我们引入 GA 算法的理想浓度模型,用此模型对遗传算法的机理进行了深入地分析,给出遗传算法的运行机理及特点,并根据此机理提出一个稍加改进的遗传算法,最后对两个典型的函数优化问题用新、旧两种方法进行了模拟比较,结果令人满意.

参考文献

1 Radolph G. Convergence analysis of canonical genetic algorithms. IEEE Transactions on Neural Network, 1994,5(1):96~101

2 Perey C. Combinative optimization with use of guided evolutionary simulated annealing. IEEE Transactions on Neural Network, 1995,6(2):290~295

3 Qi X F. Palmieri theoretical analysis of evolutionary algorithms with an infinite population size in continuous space (Part I, II). IEEE Transactions on Neural Network, 1994,5(1):102~129

4 Fogal D B. An introduction to simulated evolutionary optimization. IEEE Transactions on Neural Network, 1994,5(1):3~14

5 Eiben A E, Aarts E H L, Van Hee K M. Global convergence of genetic algorithms: a markov chain analysis. In: Shwefel

H P, Manner R eds. Parallel Problem Solving from Nature. Berlin, Heidelberg: Springer-Verlag, 1991. 4～12

6 Goldberg D E. Genetic Algorithms. Reading MA: Addison-Wesley Publishing Company, Inc., 1989

7 Zhang Ling, Zhang Bo. The statistical genetic algorithms. Journal of Software, 1997,8(5):335～344
(张钤,张钹.统计遗传算法.软件学报,1997,8(5):335～344)

8 Chen Guo-liang, Wang Xu-fa, Zhuang Zhen-quan *et al*. Genetic Algorithm and Its Applications. Beijing: People's Post and Telecommunications Publishing House, 1996
(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,1996)

9 Holland J. Adaptation in Nature and Artificial System. Cambridge, MA: MIT Press, 1991

Research on the Mechanism of Genetic Algorithms

ZHANG Ling[1,3] ZHANG Bo[2,3]

[1](*Institute of Artificial Intelligence Anhui University Hefei* 230039)

[2](*Department of Computer Science and Technology Tsinghua University Beijing* 100084)

[3](*State Key Laboratory of Intelligence Technology and Systems Tsinghua University Beijing* 100084)

Abstract It's well known that the schemata theorem and the implicit parallelism are two basic theoretical foundations of genetic algorithms (GA). In this paper, the authors analyze the two basic principles and show that the two principles are not strict and have some disadvantages. That is, as the bases of GAs, the theorems are not perfect. In order to deepen the comprehension of GA, a new ideal density model of GA is presented in this paper. Based on the model, it's known that the GA is actually a guiding stochastic search. And the searching direction is guided onto the chromosome family whose ancestors belong to schemata with high fitness. Using the model to solve the typical function optimization problem, the simulation results show that the new GA has much better speed and can get more precise results. This shows that the new GA model has potential usage in practice.

Key words Genetic algorithm, schemata theorem, implicit parallelism, ideal density model of genetic algorithm.