蜜蜂进化型遗传算法

孟 伟[1], 韩学东[2], 洪炳镕[3]
(1. 北京林业大学信息学院, 北京 100083; 2. 中国航天科工集团七〇六所, 北京 100854;
3. 哈尔滨工业大学计算机科学与技术学院, 黑龙江哈尔滨 150001)

摘 要: 本文提出了一种蜜蜂进化型遗传算法. 在该算法中, 种群的最优个体作为蜂王与被选的每个个体(雄蜂)以概率进行交叉操作, 增强了对种群最优个体所包含信息的开采能力. 为了避免算法过早收敛, 在代进化过程中引入了一个随机种群, 提高了算法的勘探能力. 通过将该算法建模为齐次有限 Markov 链, 证明了它的全局收敛性. 实验结果表明, 蜜蜂进化型遗传算法是一种提高遗传算法性能的有效改进算法.

关键词: 遗传算法; 最优保留; 全局收敛性; Markov 链
中图分类号: TP18 文献标识码: A 文章编号: 0372-2112 (2006) 07-1294-07

Bee Evolutionary Genetic Algorithm

MENG Wei[1], HAN Xue-dong[2], HONG Bing-rong[3]
(1. *Information School of Beijing Forestry University, Beijing 100083, China*;
2. *No. 706 Institute of China Aerospace Science and Industry Corporation, Beijing 100854, China*;
3. *School of Computer Science and Technology, Harbin Institute of Technology, Harbin, Heilongjiang 150001, China*)

Abstract: This paper proposes a Bee Evolutionary Genetic Algorithm (BEGA). In BEGA, optimum individual being a queen-bee in population crossover with each selected individual (drone). As a result, it reinforces the exploitation of genetic algorithm. In order to avoid premature convergence, BEGA introduces a random population that extends search area. Consequentially, it enhances the exploration of genetic algorithm. By treating the collection of individuals in each generation as a state and modeling the algorithm as a homogeneous finite Markov chain, it is proven that BEGA can guarantee the convergence towards the global optimum of the problem. Experiments results show BEGA is an efficient and effective improved genetic algorithm.

Key words: genetic algorithm; elitist preserved; global convergence; Markov chain

1 引言

在人工智能领域中, 有不少问题需要在复杂而庞大的搜索空间中寻找最优解. 在求解此类问题时, 如果不能利用问题的固有知识来缩小搜索空间, 则可能产生搜索的组合爆炸. 因此, 研究能在搜索过程中自动获取和积累有关搜索空间的知识, 并自适应地控制搜索过程从而得到最优解的通用搜索算法一直是令人注目的课题. 遗传算法 (简称 GA) 就是这类算法之一[1]. GA 以达尔文的生物进化论 "适者生存、优胜劣汰" 和孟德尔的遗传变异理论 "生物遗传进化主要在染色体上, 子代是父代遗传基因在染色体上的有序排列" 为基础, 模拟生物界进化过程. 它是由美国密执安大学的 Holland 教授于 1975 年首先提出的[2], Goldberg[3] 的工作使遗传算法获得巨大发展. 由于 GA 简单、对函数没有可微性要求而在组合优化、神经网络、智能控制、系统仿真、人工智能等诸多领域得到了广泛的应用.

目前, GA 本身在理论和应用方法上仍有许多亟待完善之处, 一个突出问题是收敛速度和全局收敛性之间的矛盾. 研究发现, GA 可以用极快的速度达到最优解的 90% 左右, 但要达到真正的最优解则要花费很长的时间[4]. 为了提高遗传算法的性能, 研究人员从各个方面对其进行了改进, 例如, 从算子的角度, 文献 [5] 采用二元变异算子 (变异操作需要两个个体) 替代传统的变异算子, 从算法结构的角度, 文献 [6] 把一代种群分成了三部分: 保留种群、繁殖种群和随机种群, 通过调整它们各自的规模来协调开采 (exploitation) 和勘探 (exploration), 从借鉴其他机制的角度, 文献 [7] 将病毒机制引入遗传算法, 文献 [8] 将免疫原理引入遗传算法, 针对简单遗传算法局部搜索能力差的缺

收稿日期: 2004-02-16; 修回日期: 2005-12-25
基金项目: 国家 863 高科技发展计划 (No. 2001AA422270); 国家自然科学基金 (No. 69985002)

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

点,文献[9]将局部搜索算法与遗传算法结合,也有研究人员针对初始种群、交叉和变异概率对算法收敛速度的影响,提出了相应的改进措施[10],等等.这些方法都在一定程度上提高了遗传算法的搜索效率,加快了收敛的速度.

众所周知,利用遗传算法求解问题的根本目标是在尽可能短的时间内找到问题的全局最优解.因此,在算法结束时,我们最关心的往往是种群中最优个体的性能.从概率上来说,群体中的优秀个体和全局最优解之间的亲和度[11]要大于群体中其他个体和全局最优解之间的亲和度,而且,与优秀个体有较大亲和度的个体也应有较高的适应度.所以,优秀个体所体现的特征信息能否被充分利用将是决定该算法的优化能力的一个重要因素. Eiben[12]等用 Markov链证明了保留最优个体(elitist)的遗传算法的概率全局收敛性, Rudolph[13]用齐次有限 Markov链证明了带有选择、交叉、变异操作的经典遗传算法收敛不到全局最优解.但是,若在遗传算法中保留每一代的最优个体,则算法将收敛到全局最优解.可以说, Eiben和 Rudolph的上述工作,表明了种群最优个体对算法的重要意义.随后许多研究人员对保留最优个体算法进行了研究和应用[14~19].其中,文献[17]在对最优保留遗传算法较为深入分析的基础上指出,在加入最优保留操作之前,具有各态遍历性的遗传算法在加入最优保留操作后,若不改变(或影响)遍历性,则得到的遗传算是全局收敛的.由此可见,研究人员采取最优保留策略的主要目的是保证算法的全局收敛性.因此,如何利用保留下来的最优个体值得进一步研究.

本文从蜜蜂的繁殖进化过程获得启示,蜜蜂的繁殖方式是有性生殖,但它具有独特的方式,即子代具有共同的母亲——蜂王.蜂王在蜂群个体最大,它唯一的职能是产卵繁殖后代.而作为子代父亲的雄蜂通过竞争获得与蜂王的交配机会.这些特征对进一步改进遗传算法,开发设计新的计算智能模型,具有重要的现实意义和发展前景.

本文以遗传算法为基础,从蜜蜂的繁殖进化过程中抽取出适合改进遗传算法的一些机制并加以改进,提出了一种蜜蜂进化型遗传算法(BEGA).该算法充分利用了种群中的最优个体对种群的进化作用,同时,为维持个体的多样性,引入了蜜蜂繁殖过程中"外来种群"的思想.本文通过把 BEGA建模为齐次有限 Markov链,证明了该算法具有全局收敛性.最后,实验结果表明, BEGA通过增强算法的勘探和开采能力,提高了遗传算法的性能.

2　蜜蜂进化机制抽象模型

2.1　蜜蜂繁殖进化过程简介[20]

蜜蜂是一种社会群居性昆虫,蜂群由蜂王(后蜂)、工蜂和雄蜂组成.蜂王由受精卵发育而来,个体最大,体重约有工蜂的 2倍,它唯一的职能是产卵繁殖后代;雄蜂由不受精的卵发育而成,除了在繁殖季节与处女蜂王交尾外,不担任何其他工作;工蜂由受精卵发育而来,个体最小,生殖器官发育不完全,无生殖能力,是蜂巢内外一切繁重劳动的承担者.

蜂王性成熟后,出巢飞舞,一群雄蜂追随其后.它们在空中旋转飞翔,越飞越快,体弱的雄蜂相继掉落地面.只留最强壮的一只雄蜂在空中与蜂王交配,随后双双落地,旋即分开.雄蜂由于分开时交接器拔断而死去,蜂王则满纳精子飞归巢中.蜂王交尾后开始产卵,产卵蜂王不再离开蜂巢.有时,为了避免近亲繁殖,蜂王会飞出,寻找其他蜂群,与之交配.

一般情况下,同一个蜂巢中不允许两只蜂王,否则相遇就要互相斗杀,直到剩一只蜂王为止.

从上述过程可以看出,蜂王在蜂群的繁殖进化中发挥着主导作用,雄蜂通过竞争获得与蜂王的交配机会,而工蜂对蜜蜂繁殖不产生实质性作用.

2.2　蜜蜂进化过程的抽象模型

在遗传算法中引入蜜蜂繁殖进化机制,目的是提高遗传算法的性能,因此有必要对蜜蜂的繁殖进化机制进行取舍.

关于种群的构成:由于工蜂不对繁殖产生实质性影响,种群中只保留蜂王和雄峰即可.

关于雄蜂的选择:通过某种选择算法从种群中选择若干个体作为雄蜂,而不是仅仅一个.

关于外来雄蜂:采取随机产生新个体的方式来实现.

关于蜂王和雄蜂交配:作为蜂王的最优个体,以概率与每个被选出的雄蜂个体和随机产生的雄蜂个体交配.

关于蜂王相斗(即争夺蜂王):失败者不是消亡(死亡或逃走),而是被视为普通的雄蜂.

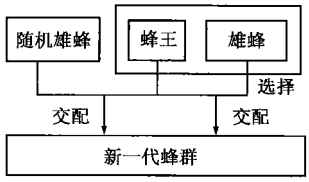

图 1　蜜蜂繁殖进化抽象模型

综上所述,我们得到蜜蜂进化机制在 GA中的抽象模型,如图 1所示.

3　蜜蜂进化型遗传算法

3.1　算法结构与算法描述

本文只考虑种群规模 N 恒定的遗传算法,并以适应度最大化为求解目的.给定初始种群 $A(0)$,则遗传算法的整个进化过程便生成一个种群序列 $\{A(0), A(1), \cdots, A(t), A(t+1), \cdots\}$,目的是使其最终包含全局最优个体.由于资源的限制,在实际应用遗传算法时,我们不可能让其无限地运行下去,总要设计某种停止准则来终止算法的运行,得到该无穷序列的一段 $\{A(0), A(1), \cdots, A(t)\}$.在遗传算法中,这是一个迭代过程,关键是如何从 t 时刻的种群 $A(t)$ 得到 $t+1$ 时刻的种群 $A(t+1)$,这就是代进化策略[6].本文提出的 BEGA,其基本思想是:找出种群 $A(t)$ 中的最优个体,与上一代蜂王比较,优胜者作为第 t 代蜂王,

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

记为 $Queen$,通过选择算法(本文采用比例选择)从种群 $A(t)$ 选出 $\frac{N}{2}\gamma(0\leqslant\gamma\leqslant1)$ 个个体,并随机产生 $\frac{N}{2}(1-\gamma)$ 个新个体,然后,$Queen$ 分别与上述 $\frac{N}{2}$ 个个体($\frac{N}{2}\gamma$ 个被选个体和 $\frac{N}{2}(1-\gamma)$ 个随机个体)配对,这 $\frac{N}{2}$ 对个体经交叉、变异后,得到子代种群 $C(t+1)$,找出 $C(t+1)$ 中的最优个体,记为 $Queen_New$,如果 $Queen_New$ 的适应度大于或等于 $Queen$ 的适应度,则 $C(t+1)$ 中的个体全部进入 $A(t+1)$,并且 $Queen_New$ 成为第 $t+1$ 代蜂王,否则,$Queen$ 替换 $C(t+1)$ 中的最差个体,得到种群 $A(t+1)$,$Queen$ 成为第 $t+1$ 代蜂王. 算法的代进化过程如图 2所示.

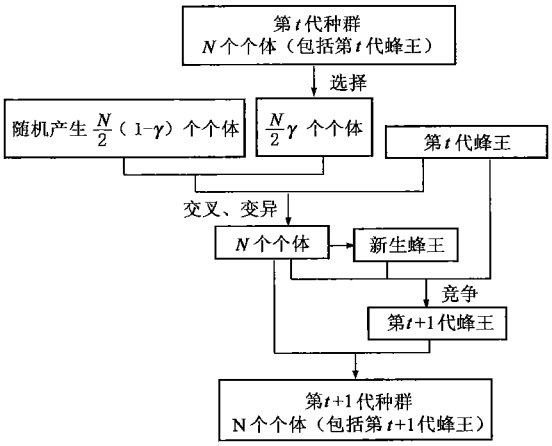

图 2 代进化过程示意图

在图 2所示的代进化过程中,参数 γ 具有调节算法勘探和开采的重要作用,本文第 4节将进一步分析 γ 对算法性能的影响.

在上述代进化策略的基础上,BEGA 整个过程可描述如下:

Step 1 随机生成初始种群 $A(0)$,$t=0$,分别对 γ、交叉概率 P_c 和变异概率 P_m 赋值.

Step 2 计算种群 $A(0)$ 中个体的适应度,将最优个体(即第 0代蜂王)保存到 $Queen$ 中.

Step 3 如果满足停止准则,算法输出结果并停止运行;否则,继续.

Step 4 $t=t+1$

Step 5 利用比例选择方式,从 $A(t-1)$ 中选出 $\frac{N}{2}\gamma$ 个个体.

Step 6 随机生成 $\frac{N}{2}(1-\gamma)$ 个个体.

Step 7 $Queen$ 分别与 Step 5和 Step 6得到的 $\frac{N}{2}$ 个个体进行交叉运算,得到子代种群,记为 $B(t)$.

Step 8 对 $B(t)$ 执行变异操作,得到种群 $C(t)$.

Step 9 计算种群 $C(t)$ 中个体的适应度,将适应度最大的个体记为 $Queen_New$.

Step 10 如果 $f(Queen_New)>f(Queen)$,$Queen=Queen_New$,$C(t)$ 即为第 t 代种群 $A(t)$;否则,用 $Queen$ 中的个体替代 $C(t)$ 中的最差个体,得到种群 $A(t)$.

Step 11 转 Step 3.

与简单遗传算法[3](简称 SGA)相比,BEGA 具有两个主要的特征:

(1)每对父本均包含蜂王(种群的最优个体).

(2)在代进化过程中引入一个随机种群,该种群的规模由参数 γ 确定.

第一个特征加强了遗传算法的开采能力,后代主要依赖交叉操作和种群的最优个体. 当然,这也增加了算法过早收敛的可能性. 第二个特征帮助遗传算法搜索新的空间,它通过引入随机种群提高了遗传算法的勘探能力. 这两个特征使算法进化加快,并保持优良的解. 因此,BEGA 有可能使遗传算法更快地接近全局最优解,同时防止过早收敛.

3.2 算法的收敛性分析

本小节讨论二进制编码时 BEGA 的收敛性问题. 此时个体为定义在集合 $\{0,1\}$ 上长度为 L 的串,这样的串共有 2^L 个,其全体组成集合 X. $f(\cdot)$ 为定义在 X 上的适应度函数,满足 $\forall x\in X$,$f(x)>0$. 设所有个体的适应度集合为 $Y=\{y\mid y=f(x),x\in X\}$,容易得到 $|Y|\leqslant|X|=2^L$,记 $b=|Y|$. 将集合 Y 中的元素按降序排列,可得到有序集合 $\{y_1,y_2,\cdots,y_b\}$,其中,$y_1>y_2>\cdots>y_b$.

记 $S=X^N$ 为所有规模为 N 的种群组成的集合. 由于种群中允许个体相同,因而,式(1)成立.

$$|S|=\binom{|X|+N-1}{N}=\binom{2^L+N-1}{N} \tag{1}$$

具体证明可参考文献[21]. 记 $M=\binom{2^L+N-1}{N}$.

为了比较种群的优劣,定义种群的适应度如下:

定义 1 $s\in S$ 是规模为 N 的种群,s 的适应度根据式(2)计算:

$$f(s)=\max_{x\in s}\{f(x)\} \tag{2}$$

定义 1表明,种群的适应度就是该种群中最优个体的适应度. 因此,种群适应度集合与个体适应度集合相同,从而有 $\forall s\in S$,$y_b\leqslant f(s)\leqslant y_1$. 根据适应度将种群集合 S 划分 b 个非空子集 S_i,$S_i=\{s\mid f(s)=y_i,s\in S\}$,记 $q_i=|S_i|$,$i=1,2,\cdots,b$. 显然,种群集合 S_1 包含所有适应度最高的种群,并且,$\sum_{i=1}^{b}q_i=M$.

为了便于描述,对符号作如下约定:$\Pr(\cdot)$ 表示事件发生的概率,$s_{i,j}$ 表示种群集合 S_i 中的第 j 个种群,在遗传算子的作用下,$s_{i,j}\to s_{k,l}$ 表示种群 $s_{i,j}$ 直接转变为种群 $s_{k,l}$,其状态转移概率用 $\Pr(s_{i,j}\to s_{k,l})$ 表示,$s_{i,j}\to S_k$ 表示种群 $s_{i,j}$ 直

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

接转变到种群集合 S_k 中的某一种群, 其状态转移概率用 $\Pr(s_{ij}\to S_k)$表示, $S_i\to S_k$ 表示种群集合 S_i 直接转变到集合 S_k, 其状态转移概率用 $\Pr(S_i\to S_k)$表示. 根据上述约定, 容易证明下列各式成立:

① $\Pr(s_{ij}\to S_k)=\sum_{t=1}^{q_k}\Pr(s_{ij}\to s_{kt})$;

② $\sum_{t=1}^{b}\Pr(s_{ij}\to S_t)=1$;

③ $P(s_{ij}\to S_k)\geqslant P(s_{ij}\to s_{kl})$.

其中, $j=1, 2, \cdots, q_i$, $i=1, 2, \cdots, b$, $l=1, 2, \cdots, q_k$, $k=1, 2, \cdots, b$.

定义 2　如果对于任意的初始种群 $A(0)$

$$\lim_{t\to\infty}\Pr(f(A(t))=y_1)=1 \tag{3}$$

则称算法全局收敛, 其中, y_1 为前述的全局最大适应度.

定义 2表明, 全局收敛是指算法经过无数次迭代后, 种群中包含全局最优个体的概率为 1, 即通常所说的以概率 1收敛.

以下证明 BEGA 具有定义 2给出的全局收敛性, 如不特殊说明, 均认为 $1>P_m>0$.

引理 1[13]　设 $\boldsymbol{P}$ 是可约化随机矩阵, 其中, $\boldsymbol{C}$ 是 m 阶随机正矩阵, 且 $\boldsymbol{R}, \boldsymbol{T}\neq 0$, 则

$$\boldsymbol{P}^{\infty}=\lim_{k\to\infty}\boldsymbol{P}^k=\lim_{k\to\infty}\begin{bmatrix}\boldsymbol{C}^k & 0\\ \sum_{i=0}^{k-1}\boldsymbol{T}^i\boldsymbol{R}\boldsymbol{C}^{k-i} & \boldsymbol{T}^k\end{bmatrix}=\begin{bmatrix}\boldsymbol{C}^{\infty} & 0\\ \boldsymbol{R}^{\infty} & 0\end{bmatrix} \tag{4}$$

是一个稳态随机矩阵.

引理 2　BEGA 中, $\forall i, k\in\{1, 2, \cdots, b\}$, $\forall j\in\{1, 2, \cdots, q_i\}$, 下式成立:

$$\Pr(s_{ij}\to S_k)\begin{cases}>0, & k\leqslant i\\ =0, & k>i\end{cases} \tag{5}$$

引理 3　BEGA 中, $\forall i, k\in\{1, 2, \cdots, b\}$, 下式成立:

$$\Pr(S_i\to S_k)\begin{cases}>0, & k\leqslant i\\ =0, & k>i\end{cases} \tag{6}$$

引理 2表明, 适应度较低的种群可以转移到适应度相同或更高的种群, 而适应度较高的种群不可能转移到适应度较低的种群. 引理 3表明, 当算法处于适应度较低的种群集合时, 它可以转移到适应度相同或更高的种群集合, 反之却不然.

定理 1　BEGA 的运行过程是一齐次有限 Markov 链.

证明　由算法的代进化过程可以看出, $t+1$时刻的种群 $A(t+1)$只与 t时刻的种群 $A(t)$有关, 而与种群 $A(0)$, $A(1)$, $\cdots$, $A(t-1)$无关, 因而, BEGA 的运行过程满足 Markov性. 由算法的整个进化过程可以看出, 各种遗传算子均不随时间发生变化, 因而, BEGA 的运行过程满足时间齐次性. 由式(1)可知, 种群集合是有限的, 因此, BEGA 的运行过程是一齐次有限 Markov 链.

定理 2　BEGA 具有全局收敛性.

证明　根据定理 1, 可以认为每一个种群集合 S_i ($i=1, 2, \cdots, b$), 都是齐次有限 Markov 链上的一个状态. 设 $_i$ 为处于状态 S_i 的概率, 易知, $\forall i\in\{1, 2, \cdots, b\}$, $_i>0$, 且 $\sum_{i=1}^{b}{}_i=1$.

根据引理 3, 可以写出该 Markov 链的状态转移矩阵为

$$\boldsymbol{P}=\begin{bmatrix}p_{11} & 0 & 0 & \cdots & 0\\ p_{21} & p_{22} & 0 & \cdots & 0\\ p_{31} & p_{32} & p_{33} & \cdots & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ p_{b1} & p_{b2} & p_{b3} & \cdots & p_{bb}\end{bmatrix}=\begin{bmatrix}\boldsymbol{C} & 0\\ \boldsymbol{R} & \boldsymbol{T}\end{bmatrix}$$

显然, $\boldsymbol{R}=(p_{21}, p_{31}, \cdots, p_{b1})'>0$, $\boldsymbol{T}\neq 0$, $\boldsymbol{C}=p_{11}=1$, 因此根据引理 1, 可得

$$\boldsymbol{P}^{\infty}=\lim_{k\to\infty}P^k=\lim_{k\to\infty}\begin{bmatrix}\boldsymbol{C}^k & 0\\ \sum_{i=0}^{k-1}\boldsymbol{T}^i\boldsymbol{R}\boldsymbol{C}^{k-i} & \boldsymbol{T}^k\end{bmatrix}=\begin{bmatrix}\boldsymbol{C}^{\infty} & 0\\ \boldsymbol{R}^{\infty} & 0\end{bmatrix}$$

其中, $\boldsymbol{C}^{\infty}=1$, $\boldsymbol{R}^{\infty}=(1, 1, \cdots, 1)'$, 从而 $\boldsymbol{P}^{\infty}$是一个稳定随机矩阵, 并且

$$\boldsymbol{P}^{\infty}=\begin{bmatrix}1 & 0 & 0 & \cdots & 0\\ 1 & 0 & 0 & \cdots & 0\\ 1 & 0 & 0 & \cdots & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ 1 & 0 & 0 & \cdots & 0\end{bmatrix}$$

这说明, 当迭代次数足够多时, 算法的种群将必定属于适应度最高的种群集合, 此时, 种群的适应度为 y_1, 即式(3)成立, 因而 BEGA 具有全局收敛性.

4　实验结果与分析

4.1　参数 γ对算法性能的影响

遗传算法中存在大量的随机操作, 因此, 对其评估应建立在统计规律的基础上. 本小节采用平均进化代数和平均最大适应度作为算法的两个性能指标, 以便考察参数 γ对算法性能的影响.

定义 3　算法对同一目标函数重复做 K 次实验, 每次实验结束时的进化代数为 t_k, $k=1, 2, \cdots, K$, 则 $\frac{1}{K}\sum_{k=1}^{K}t_k$ 称为平均进化代数.

定义 4　算法对同一目标函数重复做 K 次实验, 每次实验结束时种群的适应度为 f_k, $k=1, 2, \cdots, K$, 则 $\frac{1}{K}\sum_{k=1}^{K}f_k$ 称为平均最大适应度.

平均进化代数是体现算法效率(efficiency)的一个指标. 平均进化代数越少, 说明算法的优化效率越高, 反之, 优化效率越低. 平均最大适应度是体现算法效果(effeetiveness)的一个指标, 平均最大适应度越大, 说明算法的优化效果越好, 反之, 优化效果越差.

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

为了计算平均进化代数和平均最大适应度，应首先设计一个停止准则. 为遗传算法设计一个良好的停止准则是一个困难的任务，但往往被忽视[22]. 目前，常用的停止准则主要有以下几种：适应度边界法、时间边界法、个体数目边界法、适应度数目边界法、增长概率边界法以及基于Bayes方法的最小损失法等. 本文针对种群最优个体的进化能力，采用了如下停止准则：给定整数 $T\geqslant 1$，如果 $\sum_{i=1}^{T}|f_{t+i}-f_{t+i-1}|=0$，则算法停止. 其中，$f_{t+i}$ 和 f_{t+i-1} 分别表示第 $t+i$ 代和第 $t+i-1$ 代种群的适应度.

为了测试参数 γ 对算法的影响，将 f_1 作为测试函数.

$$\min f_1=100\times(x_2-x_1^2)^2+(1-x_1)^2 \quad s.t.\ -5\leqslant x_1,x_2\leqslant 5$$

f_1 是 Rosenbrock函数的特例 $(n=2)$，也是 De Jong五个测试函数之一. 由于该函数本身的特殊性，利用传统的函数优化方法很难得到满意解. 文献[23]在对 f_1 深入研究的基础上利用简单遗传算法对其进行求解，并探索了求解该函数合理的遗传操作.

实验中，适应度函数取为 $\frac{1}{1+f_1}$，交叉概率 $P_c=0.85$，变异概率 $P_m=0.01$，γ 由 0变到 1，步长为 0.01，$T=50$. 对同一个 γ 值，算法独立进行 1000次，计算其平均进化代数，结果如图 4所示，计算其平均最大适应度，结果如图 5所示.

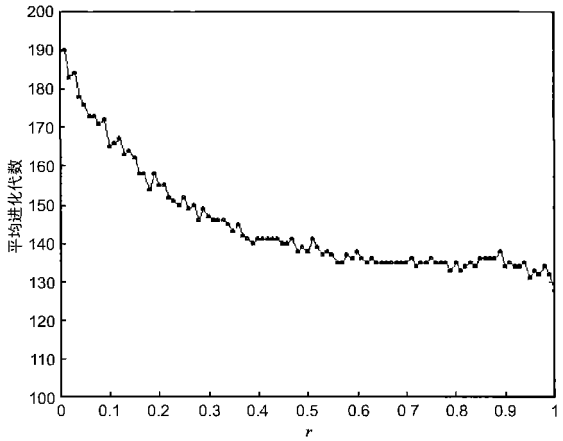

图 4 γ 与平均进化代数的关系

从图 4和图 5可以看出，随着 γ 的增大，平均进化代数和平均最大适应度都呈下降趋势，这是因为随着 γ 的增大，从上一代种群中选择的个体增多，而随机生成的个体减少，从而算法的开采能力提高，算法的勘探能力降低，算法加快了收敛的速度，但算法的优化效果却降低了. 综合考虑 γ 对平均进化代数和平均最大适应度的影响，γ 在 0.2附近效果比较理想.

4.2 BEGA与其他遗传算法的比较

为了验证 BEGA的有效性，选择了五个比较典型的函数（包括 4.1小节的 f_1）进行性能测试，并与简单遗传算法（SGA）和最优保持遗传算法（EGA）[13]进行了比较.

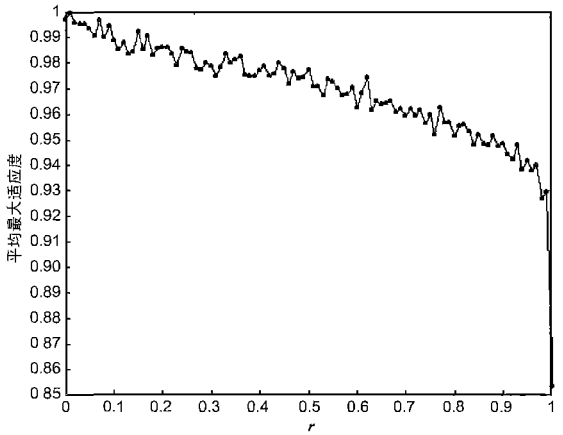

图 5 γ 与平均最大适应度的关系

$$\max f_2=|(1-x)\times x^2\times\sin(200x)| \quad s.t.\ 0\leqslant x\leqslant 1$$

$$\max f_3=(-1)\times(x^2+2y^2-0.3\times\cos(3x)-0.4\times\cos(3y))+4 \quad s.t.\ -1\leqslant x,y\leqslant 1$$

$$\max f_4=0.5-\frac{\sin^2\sqrt{x^2+y^2}-0.5}{[1+0.01\times(x^2+y^2)]^2} \quad s.t.\ -2\leqslant x,y\leqslant 2$$

$$\min f_5=\frac{x_1^2+x_2^2}{2}-\cos(2x_1)\cos(2x_2) \quad s.t.\ -10\leqslant x_1,x_2\leqslant 10$$

其中，f_1、f_2、f_3、f_4、f_5 的适应度函数分别是 $\frac{1}{1+f_1}$、f_2、f_3、f_4、$\frac{1}{2+f_5}$.

实验中，均采用二进制编码，除 f_2 中个体长度为 20，其余均为 40，种群规模 $N=50$，$P_c=0.85$，$P_m=0.01$，BEGA中，$\gamma=0.2$.

针对 4.1小节的两个性能指标，本小节做了两个实验.

实验一：收敛准则设为种群适应度与全局最大适应度的差小于等于 0.001，为了保证各种算法均在有限迭代次数内停止，设最大迭代次数为 2000，若算法在迭代 2000次后，仍然不收敛，则认为该次实验失败，否则认为成功. SGA、EGA和 BEGA分别对每个函数进行独立求解 1000次，计算他们的平均进化代数（成功实验的平均进化代数）和成功率，结果如表 1所示.

实验二：算法的停止准则设为算法运行 400毫秒，SGA、EGA和 BEGA分别对每个函数进行独立求解 1000次，计算其平均最大适应度，实验结果如表 2所示.

表 2 实验结果

	f_1	f_2	f_3	f_4	f_5
	平均最大适应度	平均最大适应度	平均最大适应度	平均最大适应度	平均最大适应度
SGA	0.737557	0.146702	4.698624	0.992972	0.847256
EGA	0.769190	0.147868	4.699996	0.999852	0.885756
BEGA	0.997381	0.148137	4.700000	1.000000	0.998723

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

表 1　实验结果

	f_1		f_2		f_3		f_4		f_5	
	平均进化代数	成功率	平均进化代数	成功率	平均进化代数	成功率	平均进化代数	成功率	平均进化代数	成功率
SGA	—	1.6%	6.4	90.8%	30.7	99.9%	24.5	94.2%	204.0	37.5%
EGA	—	2.1%	6.1	94.2%	19.5	100%	20.6	99.6%	173.3	50.1%
BEGA	559.8	91.4%	2.3	100%	6.9	100%	4.2	100%	125.4	100%

注:在求解 f_1,由于 SGA 和 EGA 成功率太低,因此计算其平均进化代数意义不大,故没有给出

实验一中的成功率本质上是对算法在规定时间内找到满意解的概率的估计. 从表 1不难看出,在给定解的精度要求下,BEGA的收敛速度(指平均进化代数)得到了成倍的提高. 因此,BEGA 的优化效率比 SGA和 EGA高.

与 SGA相比,EGA和 BEGA增加了寻找最佳个体和替换最佳个体操作,BEGA还增加了随机生成新个体操作,因此三种算法平均每次迭代所花费的时间不同,为了比较的合理性,实验二中设定算法的运行时间. 表 2的结果表明,在规定的时间内,BEGA得到的解优于 SGA和 EGA所得的解,因此,BEGA的优化效果比 SGA和 EGA好.

根据表 1和表 2的结果,我们可以看出,不论是优化效率还是优化效果,BEGA均优于 SGA和 EGA.

5　结论

受蜜蜂繁殖进化方式的启发,本文提出了一种新型的遗传算法——蜜蜂进化型遗传算法(BEGA). BEGA通过蜂王(最佳个体)与每个雄蜂(被选个体和随机个体)交配,提高了算法的开采能力,通过引入外来蜂群(随机种群)提高了算法的勘探能力. 实验结果表明,BEGA提高了遗传算法的优化效率和优化效果,因此 BEGA是一种提高遗传算法性能的有效改进算法.

参考文献:

[1] 陈国良,王煦法,庄镇泉,王东生. 遗传算法及其应用[M]. 北京:人民邮电出版社,1996.
Chen Guoliang, Wang Xufa, Zhuang Zhenquan, Wang Dongsheng. Genetic Algorithm and Its Application[M]. Beijing: People's Post and Telecommunications Publishing House, 1996. (in Chinese)

[2] Holland J H. Adaptation in Natural and Artificial System[M]. USA: University of Michigan Press, 1975.

[3] Goldberg D E. Genetic Algorithms in Search, Optimization and Machine Learning[M]. New York: Addison-Wesley, 1989.

[4] Kitano H. Empirical studies on the speed of convergence of the neural network training by genetical algorithm[A]. Proc of AAAI 90[C]. Menlo Park, USA: The AAAI Press, 1990. 881-890.

[5] 杨启文,蒋静坪,张国宏. 遗传算法优化速度的改进[J]. 软件学报. 2001, 12(2): 270-275.
Yang Qiwen, Jiang Jingping, Zhang Guohong. Improving optimization speed for genetic algorithms[J]. Journal of Software, 2001, 12(2): 270-275. (in Chinese)

[6] 江瑞,罗予频,胡东成,司徒国业. 一种协调勘探和开采的遗传算法:收敛性及性能分析[J]. 计算机学报. 2001, 24(12): 1233-1241.
Jiang Rui, Luo Yupin, Hu Dongcheng, Szeto Kworkyip. A genetic algorithm by coordinating exploration and exploitation: convergence properties and performance analyses[J]. Chinese Journal of Computer, 2001, 24(12): 1233-1241. (in Chinese)

[7] Kubota N, Shimojima K, Fukuda T. The role of virus infection in virus-evolutionary genetic algorithm envolutonary computation[A]. 1996 Proceeding of IEEE International Conference[C]. Nagoya, Japan: IEEE, 1996. 182-187.

[8] 王磊,潘进,焦李成. 免疫算法. 电子学报[J]. 2000, 28(7): 74-78.
Wang Lei, Pan Jin, Jiao Licheng. The immune algorithm[J]. Acta Electronica Sinica, 2000, 28(7): 74-78. (in Chinese)

[9] 彭伟,卢锡城. 一种函数优化问题的混合遗传算法[J]. 软件学报. 1999, (10)8: 819-823.
Peng Wei, Lu Xicheng. A hybrid genetic algorithm for function optimization[J]. Journal of Software, 1999, 10(8): 819-823. (in Chinese)

[10] 唐世浩,朱启疆. 遗传算法中初始种群与交叉、变异率对解的影响及其解决方案[J]. 科技通报, 2001, 17(3): 1-7.
Tang Shihao, Zhu Qijiang. Effects of the initial population, crossover and mutation rate to the results of genetic algorithms and a possible solution scheme[J]. Bulletin of Science and Technology, 2001, 17(3): 1-7. (in Chinese)

[11] 王煦法,张显俊,曹先彬,张军,冯雷. 一种基于免疫原理的遗传算法[J]. 小型微型计算机系统, 1999, 20(2): 117-120.
Wang Xufa, Zhang Xianjun, Cao Xianbin, Zhang Jun, Feng Lei. An improved genetic algorithm based on immune principle[J]. Mini-Micro Systems, 1999, 20(2): 117-120. (in Chinese)

[12] Eiben A E, Aarts E H, Van Hee K M. Global convergence of genetic algorithm: an infinite markov chain analysis[A]. In: Schwefel H P, Manner R. Eds. Parallel problem

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

Solving from Nature[C]. Heidelberg, Berlin: Springer-verlag, 1991. 4-12.

[13] Rudolph G. Convergence analysis of canonical genetic algorithms[J]. IEEE Transaction Neural Networks, 1994, 5(1): 96-101.

[14] Dinabandhu B, Murthy C A. Genetic algorithm with elitist model and its convergence[J]. Int J of Pattern Recognition and Artificial Intelligence, 1996, 10(6): 990-995.

[15] 彭宏, 王兴华. 具有 Elitist选择的遗传算法的收敛速度估计[J]. 科学通报, 1997, 42(2): 144-147.
Peng Hong, Wang Xinghua. Convergence speed estimate of genetic algorithm with elitist selection[J]. Chinese Science Bulletin, 1997, 42(2): 144-147. (in Chinese)

[16] 王宏刚, 曾建潮, 徐玉斌. 优良模式自学习遗传算法[J]. 自动化学报. 1999, 25(3): 375-379.
Wang Honggang, Zeng Jianchao, Xu YUbin. An excellent scheme as self-learning genetic algorithm[J]. Acta Automatica Sinica, 1999, 25(3): 375-379. (in Chinese)

[17] 何琳, 王科俊, 李国斌, 金鸿章. 最优保留遗传算法及其收敛性分析[J]. 控制与决策. 2000, 15(1): 63-66.
He Lin, Wang Kejun, Li Guobin, Jin Hongzhang. Elitist preserved genetic algorithm and its convergence analysis[J]. Control and Decision, 2000, 15(1): 63-66. (in Chinese)

[18] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, T Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. IEEE Transaction on evolutionary computation, 2002, 6(2): 182-197.

[19] Chang Wook Ahn, R S Ramakrishna. Elitism-based compact genetic algorithms[J]. IEEE Transaction on evolutionary computation, 2003, 7(4): 367-385.

[20] 刘凌云, 郑光美. 普通动物学(第三版)[M]. 北京: 高等教育出版社, 1997.
Liu Lingyun, Zheng guangmei. General Zoology (3rd)[M]. Beijing: Higher Education Press, 1997. (in Chinese)

[21] Melanie Mitchell. An Introduction to Genetic Algorithm[M]. Cambridge, MA: The MIT Press, 1996.

[22] 王正志, 薄涛. 进化计算[M]. 长沙: 国防科技大学出版社, 2000.
Wang Zhengzhi, Bo Tao. Evolutionary Computation[M]. Changsha: National University of Defense Technology Press, 2000. (in Chinese)

[23] 梁艳春, 周春光, 李寿范. 基于遗传算法的 Rosenbrock 函数优化问题的研究[J]. 软件学报. 1997, 8(9): 701-708.
Liang Yanchun, Zhou Chunguang, Li Shoufan. Optimization of Rosenbrock's function based on genetic algorithms[J]. Journal of Software, 1997, 8(9): 701-708. (in Chinese)

作者简介:

孟 伟 女, 1974年10月出生于山东德州, 2005年毕业于哈尔滨工业大学计算机科学与技术学院, 获工学博士学位, 研究领域为智能机器人、信息融合技术, 已发表论文10余篇.

韩学东 男, 1973年6月出生于山东泰安, 2004年毕业于哈尔滨工业大学计算机科学与技术学院, 获工学博士学位, 主要研究方向为嵌入式系统、智能机器人、多智能体系统、机器学习, 已发表论文20余篇. E-mail: xdhm@263.net

洪炳 男, 1937年8月出生于吉林延边, 教授, 博士生导师, 主要研究方向为空间机器人、智能机器人、虚拟现实等.

?1994-2015 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net