

文章编号: 1001-0920(2008)12-1417-06

# 一种基于双变异算子的免疫网络算法

薛文涛, 吴晓蓓, 徐志良

(南京理工大学 自动化学院, 南京 210094)

**摘要:** 针对遗传算法难以解决多峰函数优化的问题, 提出一种基于双变异算子的免疫网络算法. 该算法借鉴免疫系统的克隆选择和免疫网络理论, 采用双变异算子提高算法的全局和局部搜索能力. 利用动态网络抑制策略保持种群的多样性, 自适应地调节抗体群的规模. 仿真结果表明, 该算法能有效地改善种群的多样性, 较好地实现全局优化与局部优化的有机结合, 具有更强的多峰函数优化能力.

**关键词:** 多峰优化; 克隆选择; 免疫网络; 双变异算子

**中图分类号:** TP18      **文献标识码:** A

## Immune network algorithm based on double mutation operators

XUE Wen-tao, WU Xiao-bei, XU Zhi-liang

(Institute of Automation, Nanjing University of Science and Technology, Nanjing 210094, China. Correspondent: XUE Wen-tao, E-mail: wttiger@sina.com.cn)

**Abstract:** Aiming at the problem that genetic algorithm has difficulty in solving multimodal function optimization, an immune network algorithm is proposed based on double mutation operators. The algorithm refers to the clonal selection and immune network theory. Double mutation operators are adopted to improve global and local searching ability. The strategy of dynamic network suppression is used to maintain the diversity of population and adjust the scale of antibody population adaptively. Simulation results show that the algorithm can not only improve population diversity effectively, but also combine well global optimization with local optimization. Therefore, it has excellent optimization performance to multimodal function.

**Key words:** Multimodal optimization; Clonal selection; Immune network; Double mutation operators

### 1 引言

现实中很多实际优化问题, 如工程设计、组合优化和决策等, 常存在多个最优解或一个最优解和多个局部最优解, 它们都可归类为多峰优化问题. 遗传算法具有高效的全局搜索性和良好的鲁棒性, 与确定性算法相比, 在多峰函数优化上有更大的优势. 但它在进化中存在遗传漂移的现象, 这将导致种群容易早熟或收敛于单一极值<sup>[1]</sup>.

通过对遗传漂移成因的分析, 喻寿益等<sup>[2]</sup>指出, 在不增加变异概率的前提下, 维持物种的多样性, 以及在物种内部实行有效的局部搜索, 是确保遗传算法搜索到多个局部最优解和全局最优解的关键. 为了抑制遗传漂移, 相继提出许多改善种群多样性的方法, 其中大多研究的基础来源于小生境技术或其改进技术, 如适应值共享算法<sup>[3]</sup>、确定性拥挤算法<sup>[4]</sup>和物种保留算法<sup>[5]</sup>等. 这些算法在一定程度上保持

了种群的多样性, 对全局最优解具有较好的求解能力, 但它们存在适应性不强, 局部搜索能力较弱等缺点.

免疫系统具有较强的自适应性、多样性、学习和记忆等优点<sup>[6]</sup>, 使得免疫系统能产生适应多种抗原的抗体, 而且可用于解决基本遗传算法种群多样性不足和易早熟等问题. 因此, 许多学者利用免疫机理设计相应的策略解决多峰优化问题. De Castro 等<sup>[7]</sup>基于克隆选择原理提出一种免疫克隆选择算法 (CLONAL G), 该算法有较高的并行性和全局收敛能力, 但它仅对亲和度较高的抗体群进行克隆和突变, 这样造成了种群多样性较差, 以及对处理复杂的多峰函数优化问题能力有限. De Castro 等<sup>[8]</sup>基于免疫网络提出 opt-aiNet 算法, 该算法的种群规模动态可调, 能保持较多的局部最优解. 但当目标函数的峰值过于密集且大于种群规模时, 易发生过早收敛. 罗

收稿日期: 2007-08-19; 修回日期: 2007-10-30.

作者简介: 薛文涛 (1974—), 男, 郑州人, 博士生, 从事免疫算法、进化计算等研究; 吴晓蓓 (1958—), 女, 成都人, 教授, 博士生导师, 从事智能控制、网络控制等研究.

印升等<sup>[9]</sup>提出一种基于免疫机理的并行优化算法,该算法具有良好的多峰搜索性能,但算法中各抗体组变异率的大小对搜索到的峰值数目有一定影响,对不同的函数需合理地取值.

基于克隆选择和免疫网络理论,本文提出一种基于双变异算子的免疫网络算法(DMIN),目的是找到目标函数的全部最优解和尽可能多的局部最优解,并提高算法的收敛速度.该算法的核心在于保持种群的多样性,并利用双变异算子提高算法的全局和局部搜索能力.对多个典型的多峰函数进行仿真实验,结果表明此算法对多峰函数具有较好的优化性能.

## 2 基于双变异算子免疫网络的主要思想

### 2.1 算法思路

DMIN 算法主要由克隆选择、变异机制和克隆抑制等要素组成.算法利用克隆增殖对一定数量的随机抗体进行复制生成克隆群,通过变异机制产生子抗体,对每个克隆群中的子抗体与父本进行比较后,保留亲合度最高的抗体.克隆群中的个体组成免疫网络,网络在不断地进化中提高抗体群的亲和度.当网络无法进一步进化时,利用抑制阈值消除相似抗体,存活的抗体则作为记忆单元保留下来.最后,补充随机产生的新抗体以调节种群规模,并重复以上过程,直到记忆单元的抗体数目不再发生变化,最后记忆单元中的抗体即为多峰函数的优化解.

### 2.2 双变异算子

对于多峰优化问题,单一变异算子很难使算法同时具有良好的全局和局部搜索能力,因此 DMIN 采用双变异算子,分别加强算法的全局和局部搜索能力.变异机制遵循变异的程度与亲和度成反比的策略,这样可保证种群是向亲和度改善的方向进化,加快算法向具有高亲和度的峰值点收敛.

#### 1) 柯西变异算子

与高斯算子相比,柯西变异算子能以更大的概率产生大的变异值.这样,柯西变异有利于全局搜索,可保证算法在整个解空间内搜索,这对解空间较大的函数特别有利.但当算法到了迭代后期,对于多峰函数的搜索主要转为局部峰值点的搜索,柯西变异过大的步长很容易逃逸出峰值点的区域.因此,在本算子中加入自适应调节步长机制,每隔 50 代调节一下变异的步长.由于其步长的大小与运行代数成反比,此机制能逐渐提高算子后期的局部搜索能力和算法的搜索精度.在柯西变异算子的作用下,父代  $X_c$  生成子代  $X_c$ ,即

$$X_c = X_c + \mu C(0,1), \quad (1)$$

$$\mu = (1/\text{int}(\text{Ngen}/50)) \exp(-\mu k - f^*), \quad (2)$$

$$k = \text{int}(\text{Ngen}/50). \quad (3)$$

其中:  $C(0,1)$  是比例参数为 1 的柯西随机变量; 是控制指数函数衰减的参数;  $f^*$  是标准化后的抗体对抗原的亲合度;  $\mu$  是调节系数,本文设定为  $0.1 \sim 0.5$ ;  $\text{Ngen}$  为当前的迭代次数;  $\text{int}$  为取整函数.  $C(0,1)$  取值范围为  $(-\infty, +\infty)$ . 因此,必须限制柯西变异值的范围,控制其最大变异值在目标函数的解空间之内,以防出现无效解.

#### 2) 高斯变异算子

因为早期柯西变异算子产生的步长较大,很容易跳出好的区域而产生较差的子代,所以需引入高斯变异算子,利用高斯变异的小步长来加强算法在局部搜索极值的能力,通过步长不断地自适应调整,使算法逐渐地逼近峰值点.父代  $X_N$  和子代  $X_N$  的关系如下<sup>[8]</sup>:

$$X_N = X_N + \mu N(0,1), \quad (4)$$

$$\mu = (1/\text{int}(\text{Ngen}/50)) \exp(-\mu k - f^*). \quad (5)$$

为了防止变异后的子代超出目标函数的解空间  $[a_j, b_j]$ ,  $j = 1, 2, \dots, l$ ,  $l$  为目标解的维数,子代的取值原则<sup>[10]</sup>是:若  $x_i(j) > b_j$ ,则  $x_i(j)$  取  $2b_j - x_i(j)$  的值;若  $x_i(j) < a_j$ ,则  $x_i(j)$  取  $2a_j - x_i(j)$  的值.其中:  $i = 1, 2, \dots, N$ ,  $N$  为种群的规模.

### 2.3 克隆抑制

算法在克隆抑制阶段,保留了与当前抗体相似度小且亲和度大的抗体,这样可保证在抑制阈值  $s$  设定的范围内,亲和度大的抗体能有更多的机会存活,以利于搜索到的抗体是区域内的最好点.因为  $s$  依照经验而定,所以对不同的多峰函数,依靠相对固定的阈值区分峰值,会造成一些距离较近但可能是异峰的抗体被意外清除.因此,本文利用山谷聚类点判断低于阈值的抗体是否属于同一峰,从而保证绝大多数异峰的抗体能进入到下一代,有利于算法搜索到更多的峰值点.

山谷聚类点可用来模糊地确定搜索空间中任意两点是否属于相同的峰.取阈值内的两个抗体  $x_{e1}$  和  $x_{e2}$  的中点  $x_e$  为山谷聚类点<sup>[11]</sup>,即

$$x_e(j) = x_{e1}(j) + 0.5(x_{e2}(j) - x_{e1}(j)), \quad (6)$$

其中  $j = 1, 2, \dots, l$ ,  $l$  为目标解的维数.

克隆抑制的步骤如下:

Step1: 按照抗体对抗原的亲合度大小进行降序排列.

Step2: 将第 1 个抗体指定为第 1 个网络中心,并加入记忆单元.

Step3: 从第 2 个抗体开始顺序执行下列操作到最后一个抗体.

Step3.1: 若当前抗体到所有已指定网络中心

的距离均大于  $s$ , 则该抗体被指定为一个新的网络中心, 同样加入记忆单元。

Step3. 2: 若当前抗体到某个已指定网络中心的距离小于  $s$ , 并且其山谷聚类点的亲和度同时小于该抗体和网络中心的亲和度, 则该抗体作为另一网络中心, 并加入记忆单元。

Step3. 3: 不满足以上条件的抗体均被淘汰。

## 2.4 网络的动态平衡

DMIN 算法中种群的规模由克隆抑制和网络补充机制动态调节。在网络稳定后, 若记忆单元中的抗体数与上代相同, 则终止算法, 表明已找到问题的解; 若记忆单元规模继续扩大, 则说明搜索到了新的优化解, 函数还可能有新的峰值点, 此时需利用补充机制加入新抗体, 以促使算法获得更大的搜索空间。

当目标函数的峰值较密集且大于种群规模时, 一旦记忆单元的抗体都已收敛于峰值, 而新引入的随机个体只是已知峰值的相似个体, 变异的子代没有探索到新的区域, 则搜索将停滞在已找到的峰值上, 这将导致算法早熟收敛。为了避免早熟现象, 本文在第 1 次记忆单元数量不变化后, 计数 1 次, 继续引入数目为占群体  $d\%$  的随机个体进入种群, 一共计数 3 次。若最后记忆单元数量还是不变, 则可认为算法已搜索到目标函数的所有峰值, 终止算法。

## 2.5 算法的步骤

在处理多峰函数优化时, DMIN 采用实数编码, 用抗原对应目标函数, 抗体对应目标函数的解个体, 抗体群形成目标函数的解空间, 具体步骤如下:

Step1: 参数初始化。抗体的编码长度为  $l$ ,  $S$  为解空间, 迭代次数  $T = 0$ , 计数  $\text{count} = 0$ 。

Step2: 种群初始化。在解空间内随机生成规模为  $N$  的初始种群  $A(0)$ 。

Step3: 计算种群中每个抗体  $Ab_i(T)$  的亲和度  $f_i$ , 并对其进行标准化得到  $f_i^*$ ,  $i = 1, \dots, N$ 。

Step4: 克隆增殖。对种群中的每个抗体  $Ab_i(T)$  克隆  $N_c$  个个体组成克隆群  $B(T)$ 。

Step5: 种群变异。对产生的克隆抗体进行柯西变异和高斯变异, 并将父代保留。

Step6: 计算每个克隆群变异后  $Ab_j(T)$  抗体的亲和度  $f_j$ ,  $j = 1, \dots, N_c \times N$ 。

Step7: 群体更新。将每个克隆群的抗体同父代抗体比较, 取亲和度最高的抗体组成新种群  $C(T)$ , 并计算  $C(T)$  的平均亲和度。

Step8: 网络稳定性判断。若  $C(T)$  的平均适应度与  $A(T)$  相比小于误差  $\epsilon$ , 则继续; 否则, 返回到 Step3, 令  $T = T + 1$ 。

Step9: 收敛性判断。将  $C(T)$  进行克隆抑制, 得

到记忆单元  $M(T)$ , 比较记忆单元中抗体的数目  $N_m$  与上一代是否相同。若相同, 则  $\text{count} + 1$ ; 当  $\text{count} = 3$  或迭代次数达到最大迭代次数时, 结束算法。

Step10: 引入  $d\%N$  个随机抗体加入记忆单元, 组成下一代种群  $A(T + 1)$ , 返回 Step3。

## 3 算法的收敛性证明

利用文献 [12] 中的收敛分析方法, 可证 DMIN 依概率收敛函数的所有峰值。

对于  $l$  维空间的多峰函数  $f(x)$ ,  $x \in S \subseteq R^l$ , 目标函数的解空间  $S_j = [a_j, b_j]$  ( $j = 1, 2, \dots, l$ ) 为闭集,  $f$  为目标函数。设群体规模为  $N$  的抗体群构成的群体空间为  $S^N$ , 每一空间中的元素称为状态,  $S$  为空间所有元素的集合。设  $A_n$  为  $n$  时刻种群  $A(n)$  的状态空间,  $B_n$  和  $C_n$  分别为种群经克隆和更新后的状态空间,  $M_n$  表示  $n$  时刻记忆单元的状态空间,  $D_n^d$  表示  $n$  时刻产生  $N_1 = d\%N$  个新抗体的状态空间。

**定义 1** 峰值区域集合  $\text{pa}(x_i, r_i) = \{x_i \in S, 0 < r_i < r_m \mid x_i - x_i^* \leq r_i\}$ 。其中  $r_i$  为峰的区域半径,  $r_m$  为  $S$  空间内峰值最大的收敛半径,  $x_i^*$  为峰值点, 即  $f(x_i^*) = f(x_i)$ , 由所有  $x_i^*$  构成的集合为  $S^*$ 。

从 DMIN 的运行机理分析可知,  $A_{n+1}$  由两部分组成, 一部分是上一代记忆单元中的优秀抗体群  $M_n$ , 一部分是随机加入的新个体群  $D_n^d$ 。由双变异特性可知, 高斯变异算子产生的抗体多集中于上一代优秀抗体的附近, 而柯西变异算子使算法可以探索更大的区域。对于种群中某一抗体  $x_i \in \text{Pa}(x_i, r_i)$ , 经变异和更新后, 在峰值区域内可产生更优的抗体, 若存在某一时刻  $m$ , 随着代数的增加, 其子代  $x_i$  的亲和度无法进一步改善, 且在阈值范围内具有最好的亲和度, 说明算法已找到峰值, 则  $x_i$  满足记忆条件, 而被保留在记忆单元中。在  $\text{Pa}(x_i, r_i)$  内的其他抗体, 它们在双变异算子的作用下, 也将最终进化为  $x_i$ , 或进入到其他峰值区域。在这种机制下, 随着新个体的不断加入, 一旦某个峰的抗体满足记忆条件, 它将会保留在记忆单元中, 这样一直进化下去, 所有的峰值都将被找到。

由于 DMIN 中  $A_{n+1}$  所处的状态仅与  $A_n, M_n, D_n^d$  有关, 与过去的状态无关, 整个种群序列构成的随机过程为马氏链, 即

$$A_n \rightarrow B_n \rightarrow C_n \rightarrow M_n \rightarrow D_n^d \rightarrow A_{n+1}.$$

但由于柯西变异与进化代数有关, 该链为非齐次马氏链。在这一随机过程中, 用  $A_n^i$  表示在  $n$  时刻  $A_n$  处于状态  $s_i$ ,  $P(A_n^i)$  表示  $A_n$  处于状态  $s_i$  的概率,  $P(A_{n+1}^j / A_n^i)$  表示在  $A_n^i$  下  $A_{n+1}$  处于状态  $s_j$  的概率。

**定理 1** DMIN 对于任意初始分布,均以概率 1 弱收敛,即

$$\lim_n P(M_n \subseteq S^* | \phi) = 1.$$

**证明** 由算法中 Step7 可知,  $C_n$  中的抗体是  $B_n$  中同一抗体克隆群中亲和度最高的,且亲和度好于父代,则必存在

$$f(C_{n+1}) \leq f(C_n), n = 1, 2, \dots$$

而  $M_n$  中的抗体是  $C_n$  中抑制相似性后亲和度最好的抗体,则  $f(M_{n+1}) \leq f(M_n)$  成立,故可得

$$P(M_{n+1} \subseteq S^* | M_n \subseteq S^* | \phi) = 0.$$

经克隆抑制后,  $A_{n+1}$  中的新个体由  $D_n^d$  补充. 设  $M_{n+1}^d$  为  $D_n^d$  经 Step3 至 Step7 形成的记忆抗体,则

$$P(M_{n+1}^d \subseteq S^* | \phi) = 1 - \left( \frac{|S| - |S^*|}{|S|} \right)^{N_1} = \dots$$

对于集合  $M_{n+1} \supseteq M_{n+1}^d \cup M_n$ , 有

$$P(M_{n+1} \subseteq S^* | \phi | M_n \subseteq S^* | \phi) = \dots$$

由全概率公式可知

$$\begin{aligned} P(M_{n+1} \subseteq S^* | \phi) &= \\ (1 - \alpha) P(M_n \subseteq S^* | \phi) &+ \\ \alpha P(M_{n+1}^d \subseteq S^* | \phi) & \\ (1 - \alpha) P(M_n \subseteq S^* | \phi). & \end{aligned}$$

其中

$$\begin{aligned} \alpha &= P(M_{n+1} \subseteq S^* | \phi | M_n \subseteq S^* | \phi), \\ \alpha &= P(M_{n+1} \subseteq S^* | \phi | M_n \subseteq S^* | \phi). \end{aligned}$$

根据归纳法可得

$$P(M_{n+1} \subseteq S^* | \phi) = (1 - \alpha)^{n+1}.$$

因为  $0 < \alpha < 1$ , 故结论成立.

## 4 仿真实验及结果分析

### 4.1 典型测试函数

本文选用 4 个多峰函数作为算法的优化实例, 分别代表少量峰值点函数、大量峰值点函数和无穷多峰值点函数. 函数  $f_1$ <sup>[13]</sup> 是非均匀分布的一维多峰函数, 有 5 个不等高的峰; 函数  $f_2$ <sup>[8]</sup> 是非均匀分布多峰函数, 有 4 个全局极值点和较多的局部极值点; 函数  $f_3$ <sup>[6]</sup> 是 Schubert 函数, 有 9 个全局最优解, 共 760 个极大值解, 要求算法具有较好的维持种群多样性的能力; 函数  $f_4$ <sup>[8]</sup> 是 Schaffer 函数, 仅有一个最大值, 但此解附近有无穷多个局部极大值解, 一般算法只能搜索到次优解. 函数表达式分别如下:

$$\begin{aligned} f_1(x) &= \exp\left(-2(\ln 2) \left(\frac{x - 0.08}{0.834}\right)^2\right) \times \\ &\quad \sin^6(5(x^{0.75} - 0.05)), 0 \leq x \leq 1; \\ f_2(x, y) &= x \sin(4 - x) - y \sin(4 + y) + 1, \\ &\quad -2 \leq x, y \leq 2; \\ f_3(x, y) &= \sum_{i=1}^5 \cos[(i+1)x + i] \times \end{aligned}$$

$$\begin{aligned} &\quad \sum_{i=1}^5 \cos[(i+1)y + i], \\ &\quad -10 \leq x, y \leq 10; \\ f_4(x, y) &= 0.5 - \frac{\sin^2 \sqrt{x^2 + y^2} - 0.5}{[1.0 + 0.001(x^2 + y^2)]^2}, \\ &\quad -10 \leq x, y \leq 10. \end{aligned}$$

本文将用 SH<sup>[3]</sup>, CLONAL G<sup>[7]</sup>, opt-aiNet<sup>[8]</sup> 和 DMIN 算法对多峰函数求解, 通过 30 次仿真实验比较 4 个算法对多峰函数的优化能力.

SH 采用实数编码, 两点交叉,  $p_c$  和  $p_m$  分别设为 0.8 和 0.05; CLONAL G 的二进制变量位数为 20,  $\alpha = 0.1, d = 10$ ; opt-aiNet 的  $N_c = 10, \alpha = 20, d = 40\%$ ,  $N_{gen_{max}} = 500$ ; DMIN 中  $N_c = 5, \alpha = 20, d = 40\%, \mu = 0.001, \mu = 0.2, N_{gen_{max}} = 500$ . 4 个测试函数的优化结果如表 1 所示.

表 1 优化结果比较

函数	算法	种群规模	阈值	最大峰值个数	平均峰值个数	平均迭代次数	能否搜到所有峰值
$f_1$	SH	50	0.2	5	4.2	50	Y
	CLONAL G	50	—	5	4.8	50	Y
	opt-aiNet	20	0.1	5	5	28.2	Y
	DMIN	20	0.1	5	5	24.7	Y
$f_2$	SH	100	0.5	17	10.7	200	N
	CLONAL G	100	—	38	31.6	200	N
	opt-aiNet	40	0.2	90	76.9	123.6	N
	DMIN	40	0.2	100	96.2	146.2	Y
$f_3$	SH	100	0.5	14	10.6	200	N
	CLONAL G	100	—	18	18	200	N
	opt-aiNet	40	0.2	738	710.9	364.4	N
	DMIN	40	0.2	760	759.4	248.7	Y
$f_4$	SH	100	0.5	27	22.4	200	N
	CLONAL G	100	—	36	31.1	200	N
	opt-aiNet	40	0.2	386	361.3	354.8	N
	DMIN	40	0.2	1126	1098.8	500	N

表中 SH 和 CLONAL G 的平均迭代次数是指搜索到最大峰值数的最小指定代数; Y 代表能搜索到所有峰值, 而 N 代表不能搜索到所有峰值. 图 1 ~ 图 3 分别为 DMIN 与其他算法的寻峰比较, 其中  $f$  为目标函数的函数值.

### 4.2 结果分析

从收敛的结果可以看出, DMIN 的多峰寻优能力大大超过 SH 和 CLONAL G 算法, 且局部搜索能力和收敛速度优于 opt-aiNet.

1) SH 算法虽然通过适应度共享技术改善了遗传算法的多样性, 但适应度共享技术也使一些相距

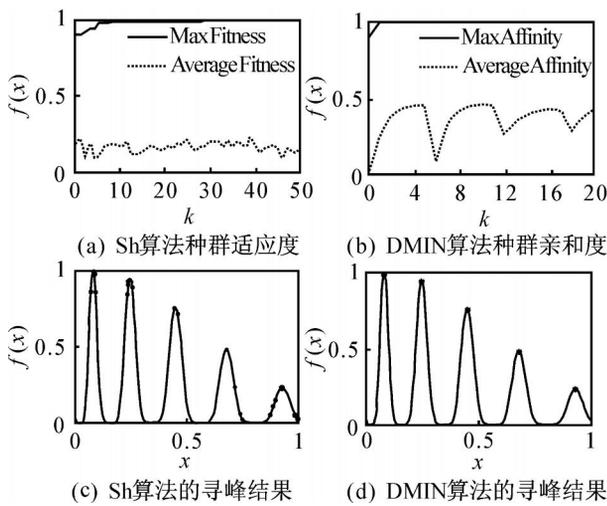


图 1 函数  $f_1$  的寻峰比较

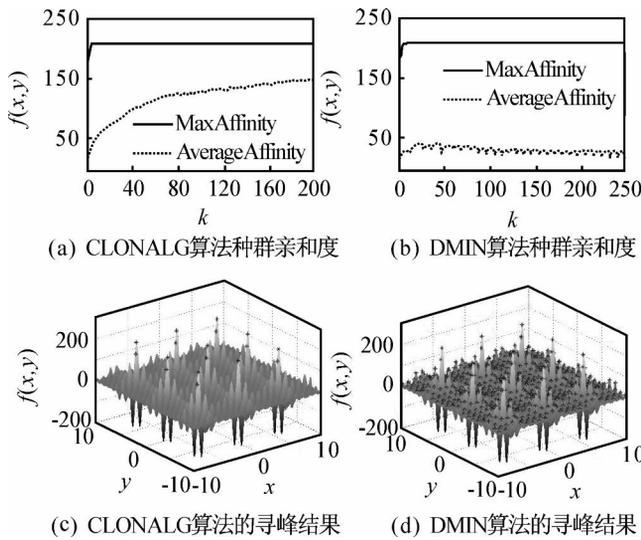


图 2 函数  $f_3$  的寻峰比较

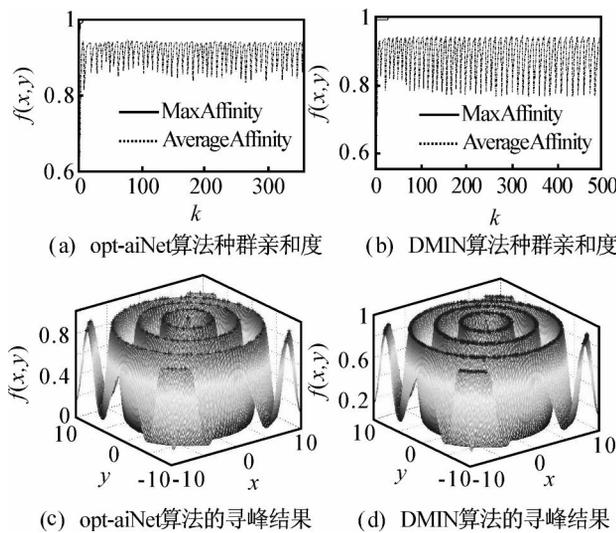


图 3 函数  $f_4$  的寻峰比较

较远但适应度较差的个体在下一代存活,若这些个体的成长性较差,将使种群的平均适应度维持在较低的水平,而且当迭代结束后这些个体无法收敛于峰值,如图 1(c) 所示. DMIN 算法采用抑制策略消

除相似抗体,淘汰阈值范围内亲和度较低的抗体,保持了种群的多样性,并使区域内最优的抗体进入到下一代. 通过网络的自适应动态调节不断地补充新抗体进入网络,使种群数量随迭代次数不断地增加,提高了算法搜索到更多峰值的概率.

2) CLONALG 算法由于仅对亲和度较高的抗体群进行克隆和突变,有效地提高了高亲和度抗体的竞争力,因此,算法对全局最优解有较好的搜索能力,如图 2(a) 所示. 算法迭代很少的次数就能收敛到全局最优,种群的平均亲和度随着代数增加不断地提高,说明个体是向全局最优解的方向进化,但这种进化机制也使亲和度低的峰值点不断地被淘汰,算法只能搜索到较高亲和度的峰值点. 而 DMIN 算法利用双变异算子,保证了算法在深度搜索和广度搜索之间的平衡,不但能搜索到所有全局最优解,而且能搜到尽可能多的局部最优解. 从图 2(b) 可以看出,种群的平均亲和度在局部极值附近波动,说明有更多的解趋向局部最优解.

3) opt-aiNet 具有较强的局部搜索能力,但开拓搜索空间的能力较差. 当目标函数的峰值较密集且大于种群规模时,易发生早熟收敛,如图 3(c) 所示. opt-aiNet 很难搜索到全局最优解区域以外的大部分局部极值. 通过柯西变异算子,DMIN 可在整个解空间内大范围地搜寻峰值,有利于算法的全局寻优,易于搜索到新的区域而找到更多的峰值. 从图 3(b) 可以看出,DMIN 算法有更多区域的抗体参与搜索,使平均亲和度的波动要大于 opt-aiNet. DMIN 算法通过多次网络抑制,加上柯西变异算子的全局搜索,避免了早熟,算法搜索到的最大峰值个数和平均峰值个数要明显优于 opt-aiNet,且搜索到同等数量峰值的迭代次数要远小于 opt-aiNet.

### 5 结 论

本文借鉴克隆选择和免疫网络理论,提出一种双变异算子的免疫网络算法(DMIN),DMIN 采取双变异算子,在保证算法具有良好的局部搜索能力的基础上,增强了算法全局搜索的能力. 构建的网络具有自适应调节功能,能动态地调节抗体群的规模,保证种群的多样性,并且可防止算法过早收敛. 几个典型多峰函数仿真实验表明,算法能适用于多种类型的多峰函数求解,且寻优能力大大超过 SH 和 CLONALG 算法,全局搜索能力和收敛速度优于 opt-aiNet 算法.

### 参考文献(References)

[1] Mahfoud S W. Genetic drift in sharing methods [C]. Proc of the 1st IEEE Conf on Evolutionary Computation. Piscataway: IEEE Service Center, 1994,

- 1: 67-72.
- [2] 喻寿益, 郭观七. 选择的遗传漂移分析[J]. 计算机研究与发展, 2004, 41(2): 346-351.  
(Yu S Y, Guo G Q. Genetic drift analysis of selection [J]. J of Computer Research and Development, 2004, 41(2): 346-351.)
- [3] Goldberg D E, Richardson J. Genetic algorithms with sharing for multimodal function optimization[C]. Proc of the 2nd Int Conf on Genetic Algorithms. NJ: Lawrence Erlbaum, 1987: 41-49.
- [4] Mahfoud S W. Crowding and preselection revisited[C]. Parallel Problem Solving from Nature. Amsterdam: Elsevier, 1992: 27-36.
- [5] Li J P, Balazs M E, Parks G T. A species conserving genetic algorithm for multimodal function optimization [J]. J of Evolutionary Computation, 2002, 10(3): 207-234.
- [6] 黄席樾. 现代智能算法理论及应用[M]. 北京: 科学出版社, 2005.  
(Huang X Y. Modern intelligent algorithms theory and applications[M]. Beijing: Science Press, 2005.)
- [7] De Castro L N, Von Zuben F J. Learning and optimization using the clonal selection principle [J]. IEEE Trans on Evolutionary Computation, 2002, 6(3): 239-251.
- [8] De Castro L N, Timmis J. An artificial immune network for multimodal function optimization[C]. Proc of IEEE Conf on Evolutionary Computation. Hawaii, 2002, 1: 699-704.
- [9] 罗印升, 李人厚, 张维玺. 基于免疫机理的多峰值函数并行优化算法[J]. 系统仿真学报, 2005, 17(2): 319-322.  
(Luo Y S, Li R H, Zhang W X. Multimodal functions parallel optimization algorithm based on immune mechanism[J]. J of System Simulation, 2005, 17(2): 319-322.)
- [10] 王向军, 向东. 一种双种群进化规划算法[J]. 计算机学报, 2006, 29(5): 835-840.  
(Wang X J, Xiang D. A novel bi-group evolutionary programming[J]. Chinese J of Computers, 2006, 29(5): 835-840.)
- [11] 谭竹梅, 余晓峰, 郭观七. 排挤小生态遗传算法的改进方法[J]. 控制理论与应用, 2004, 21(4): 651-654.  
(Tan Z M, Yu X F, Guo G Q. Improvement of niching genetic algorithms using crowding[J]. Control Theory and Applications, 2004, 21(4): 651-654.)
- [12] 张著洪, 黄席樾. 一种新的免疫算法及其在多模态函数优化中的应用[J]. 控制理论与应用, 2004, 21(1): 17-21.  
(Zhang Z H, Huang X Y. Novel immune algorithm and its application to multi-modal function optimization [J]. Control Theory and Applications, 2004, 21(1): 17-21.)
- [13] Sareni B, Krahenbuhl L. Fitness sharing and niching methods revisited [J]. IEEE Trans on Evolutionary Computation, 1998, 2(3): 97-106.

(上接第 1416 页)

- [5] Ho K J, Leung J T, Wei W D. Complexity of scheduling tasks with time-dependent execution times [J]. Information Processing Letters, 1993, 48: 315-320.
- [6] Potts C N, Kovalyov M Y. Scheduling with batching: A review[J]. European J of Operational Research, 2000, 120(2): 228-249.
- [7] Potts C N, Van Wanssenhove L N. Integrating scheduling with batching and lot-sizing: A review of algorithms and complexity[J]. J of Operational Research Society, 1992, 43(3): 395-406.
- [8] 张峰. 工件加工时间随工件开工时间线性增加的成组排序问题[J]. 上海第二工业大学学报, 2001, 2(2): 7-9.  
(Zhang F. Batching problem about increase of processing times[J]. J of Shanghai Second Polytechnic University, 2001, 2(2): 7-9.)
- [9] Cheng T C E, Janiak A. Resource optimal control in some single-machine scheduling problems [J]. IEEE Trans on Automatic Control, 1994, 39(6): 1243-1246.
- [10] Zhao C L, Tang H Y. Single machine scheduling with linear processing times [J]. Acta Automatica Sinica, 2003, 5(9): 703-708.
- [11] 赵传立, 唐恒永. 一类资源约束的单机排序问题[J]. 系统工程学报, 2004, 19(5): 451-456.  
(Zhao C L, Tang H Y. Single machine scheduling with resource constrains [J]. J of Systems Engineering, 2004, 19(5): 451-456.)
- [12] 闫杨, 赵传立. 安装时间受资源约束的单机成组调度问题[J]. 电机与控制学报, 2007, 11(1): 70-73.  
(Yan Y, Zhao C L. Single machine group scheduling with learning effect and resource dependent setup times [J]. Electric Machines and Control, 2007, 11(1): 70-73.)
- [13] 闫杨, 赵传立. 一类安装时间受资源约束的单机成组排序问题[J]. 系统工程与电子技术, 2007, 29(6): 938-941.  
(Yan Y, Zhao C L. Single machine group scheduling with resource dependent setup times [J]. Systems Engineering and Electronics, 2007, 29(6): 938-941.)