

文章编号:1005-0523(2012)04-0057-07

高维动态函数优化的免疫算法研究

何珍梅

(华东交通大学电气与工程学院,江西 南昌 330013)

摘要:生物免疫系统在识别和清除抗原的过程中,免疫细胞之间信息交互和协作,能够快速适应环境变化,具有很强的学习和自适应控制能力。基于此,本文提出了针对高维动态函数优化的免疫算法。该算法的主要特点是采用Gray码编码、采用不同的克隆繁殖策略、对抗体实施不同概率的超变异和多细胞编辑等操作,提高算法寻优能力和种群的多样性。通过与几种典型算法进行比较,仿真结果证明该算法对动态优化性能及跟踪能力有明显的改善。

关键词:免疫算法;高维动态函数;环境跟踪

中图分类号:TP301.6

文献标志码:A

在动态优化问题中,其决策或决策变量会随时间动态变化,因此要求其求解算法在复杂多变的环境下保持稳定的优化性能。遗传算法凭借它对函数类型、搜索空间的形状等没有任何限制的优势,成为求解复杂函数优化的常用方法之一,但当环境条件发生变化,其种群模式趋于单一,陷入局部极值的几率增大,寻优能力会受较大影响。为了克服这些问题,研究人员提出许多改进算法^[1-4],这些算法可以归结为如下几类:①增强种群模式多样性;②保持群体多样性;③记忆法;④多子群搜索策略。但这些方法仍面临诸多问题:增加模式多样性,其缺点是判别环境变化方面存在困难;保持多样性与解决收敛速度存在矛盾;如何合理设计记忆库引导算法的正面搜索效应及子群合理划分等,这些问题都有待进一步改善。

人工免疫优化算法是模仿生物免疫系统功能的一种智能方法,借鉴了生物免疫系统的自组织、自适应、记忆等进化学习机理特点,具有搜索效率高、群体多样性好的特点,为解决复杂多变的优化问题提供了新的方案。文献[5]比较了克隆选择算子和进化策略在处理动态优化问题时算法参数的敏感性;文献[6]针对动态变化优化问题提出了一种自适应免疫算法;文献[7]在处理高维全局优化问题存在多样性不足、个体间信息交流少等问题,提出一种免疫进化算法;文献[8]利用相似结构的小生境及近优淘汰算子对免疫算法加以改进,有效改善了群体的多样性等。这些方法在一定范围内取得了较好的效果。本文基于免疫反应、克隆选择原理及免疫网络理论,针对动态环境下高维函数设计一种免疫算法。算法具有以下特点:①抗体编码采用Gray码以提高算法的局部搜索能力;②免疫反应细胞具有增强学习能力,从而能更高效地消灭抗原;③抗体进化前后采用两种不同克隆策略,用来增加抗体多样性;④根据抗体的相似度实施不同概率的超变异,考虑抗体局部搜索同时也兼顾全局寻优;⑤利用多细胞随机编辑保持细胞多样性,提高算法跳出局部极值点能力。

1 免疫特征及算法的基本思想

生物免疫系统面对复杂的外部环境能够产生相应的抗体,快速消除对肌体的威胁,在变化的环境中体

收稿日期:2012-05-25

基金项目:江西省自然科学基金项目(2007GZS0883);江西省教育厅科技项目(GJJ08239)

作者简介:何珍梅(1968—),女,硕士,主要从事进化计算和智能控制研究。

现出很强学习和自适应能力。克隆选择学说认为:抗原与抗体间存在相互刺激和抑制关系,抗体对抗原的亲合度越大,抗体克隆细胞的能力就越强,免疫网络理论进一步完善抗体克隆选择原理,认为抗体克隆数量不仅由抗体对抗原的亲合度决定,而且取决于抗体的浓度,即当抗体浓度过高,会对抗体的克隆过程抑制,同时生物免疫系统具有增强学习机制,免疫系统遇到抗原发生免疫反应的时候,被激活的免疫细胞产生大量的抗体消灭抗原的同时也发生进化,变得能够更好地与对应的抗原相匹配,从而更高效的消灭抗原;另一方面,多细胞随机编辑通过进化过程中多细胞之间信息交互,对子细胞进行随机基因改造,能有效地丰富种群的多样性。设计算法时,主要考虑了免疫系统机理的以下方面:① 免疫细胞产生过程的基因多样性;② 抗原入侵后,免疫细胞在抗体诱导下克隆繁殖;③ 克隆扩增后免疫细胞发生高频变异及多细胞编辑;④ 记忆细胞的产生;⑤ 抑制浓度过高抗体;⑥ 免疫细胞的补充。

2 动态环境下的免疫算法

非约束极大值动态优化问题(DP)可描述为

$$DP: \max_{x \in D(t) \subseteq R^{p(t)}} f(x, t) \quad (1)$$

式中: $R^{p(t)}$ 是实数值空间; $D(t) = \prod_{i=1}^{p(t)} \langle a_i(t), p_i(t) \rangle$ 为定义域,是 $R^{p(t)}$ 的有界闭区域; $x = (x_1, x_2, \dots, x_{p(t)}) \in D(t)$; $p(t)$ 为环境变量的可变维数; $a(t)$ 为环境参数; t 表示时间, $0 < t \leq T$ (文中时间 t 取整数,第 t 时刻的优化问题称为第 t 环境, T 为环境总数); $f(x, t)$ 为目标函数。

根据上述免疫优化机理的特点,将免疫概念对应到进化优化问题中,特作如下定义:

① 抗原 Ag : 表示待求解的问题;② 抗体 Ab_i : 表示问题的候选解;③ 免疫细胞 $BN = \{Ab_i\}$, $i = (1, 2, \dots, N_1)$: 免疫系统中当前免疫细胞集合, N_1 是细胞总数;④ 免疫反应细胞 $SBN = \{Ab_j\}$, $j = (1, 2, \dots, N_2)$, $N_2 < N_1$: 用于搜索当前环境下最优模式的亚种群;⑤ 克隆子细胞 $CL = \{c_i\}$: 即由 SBN 克隆出的免疫细胞后代,通过搜索获得的备选模式;⑥ 亲和度 af : 抗体对抗原的刺激反映,表示候选解对目标函数评价标准。

本文算法的具体操作如下:

Step1: 初始化。随机生成初始种群 BN , 设定算法参数;

Step2: 重复执行本步,直到满足结束条件;

Step2.1: 确定当前环境下的抗原 Ag , 选择免疫反应细胞群 SBN ;

Step2.1.1: 计算所有抗体与抗原的亲和度,并按亲和度 af 由大到小排序;

Step2.1.2 选择前 N_2 个细胞 Ab_i 构成免疫反应细胞群 SBN ;

Step2.1.3: 对免疫反应细胞群实施奖励机制,调整免疫反应细胞亲和度大小;

Step2.2: 免疫反应细胞群进行克隆操作;

Step2.2.1: 被选择的 N_2 个抗体将被克隆,产生克隆抗体群 CL_1 。抗体克隆的方式取决于种群内具有最优亲和度抗体的浓度及阈值。当抗体浓度小于等于设定阈值,抗体克隆规模取决于调整后亲和度为刺激度,当抗体浓度大于设定阈值,抗体克隆规模由抗体浓度起主导作用;

Step2.2.2: 计算免疫反应细胞与最优抗体相似度,对 CL_1 实施不同概率的超变异。当相似度小于等于设定阈值克隆细胞实施大概率超变异,当相似度大于设定阈值克隆细胞将按小概率变异,生成为新的进化抗体群 CL_2 ;

Step2.2.3: 多受体编辑。新生子细胞 CL_2 与免疫反应细胞 SBN 实施多受体随机编辑,获得新的子细胞,最后将所有子细胞合并生成克隆子细胞群 CL ;

Step2.3: 对克隆子细胞进行免疫记忆操作,更新免疫网络。计算所有克隆子细胞 $cl \in CL$ 的亲和度

af ;然后根据亲和度及模式相似度对CL进行网络抑制;再从经过抑制后的子细胞中选择亲和度最高的部分子细胞构成备选细胞群;最后,从备选细胞群中选择适当子细胞更新B细胞群 BN ;

Step3:检查计算是否满足设定进化代数,如不满足进化条件,转Step2,进行下一轮进化,否则,输出结果。

3 免疫算子设计

3.1 编码、解码操作

1) 编码。抗体的编码与基本的遗传算法编码相同。对一般函数的优化问题来说,Gray 编码方案要优于普通二进制编码方案。文献[9]中比较了标准二进制编码和Gray 码对算法性能的影响,采用Gray 编码可以提高算法的局部搜索能力,能有效地克服普通二进制编码的Hamming 悬崖现象。Gray 码($\langle g_{l-1}g_{l-2}\cdots g_0 \rangle$)与普通二进制编码($\langle b_{l-1}b_{l-2}\cdots b_0 \rangle$)可以进行一一对应的转换。由Gray 码转换为二进制码,表达式如式(2):

$$b_k = \begin{cases} g_{l-1} & k=l-1 \\ b_k \oplus g_{k+1} & k < l-1 \end{cases} \quad (2)$$

2) 解码。解码时先将Gary 码转换为二进制,然后对其进行译码。假设二进制编码是 $b_{l-1}b_{l-2}\cdots b_0$,变量 x 取值范围在 $[x_{\min}, x_{\max}]$,则对应的解码公式如式(3)

$$x = x_{\min} + \left(\sum_{i=0}^{l-1} b_i \cdot 2^i \right) \cdot \frac{x_{\max} - x_{\min}}{2^{l+1} - 1} \quad (3)$$

3.2 亲和度

计算抗原与抗体的亲和力,根据式(2),(3)将Gary 码转换为二进制码,并解码为实际计算数据。高维动态环境下亲和度评价方案应随环境变化而动态调节,这有助于使算法在更短时间内定位下一环境解的位置。为了防止某些环境下因变量维数过高定义域内目标函数过大,造成抗体群中所有抗体的亲和度大小相距甚远的问题,抗体的亲和度的设计为

$$af = f(x, t) / p(t) \quad (4)$$

考虑免疫细胞具有增强学习的能力,因此对亲和度较大的抗体进行一定的鼓励,有助于提高它的期望生存率,加快算法的收敛速度。设 $af(Ab_i)$ 是免疫反应细胞 Ab_i 原有亲和度大小,对其进行调整后为 af_{Abi} 。

$$af_{Abi} = af(Ab_i) + W_{Abi} \times p(Ab_i)$$

其中

$$p(Ab_i) = \exp[-1/af(Ab_i)] \quad (5)$$

式中: $p(Ab_i)$ 表示对局部极值或极值附近点的奖励函数,是亲和度的增函数; W_{Abi} 为抗体 Ab_i 的奖励函数的权值系数,它根据抗体优劣排序来确定,越优秀的个体权值系数越大。

3.3 克隆操作

克隆操作主要通过高变异克隆操作对抗体实现群体进化。其过程是:抗体亲和度越大,则期望繁殖概率越大,抗体浓度越大则期望繁殖概率越小,这样既鼓励了亲和度高的抗体,又抑制了浓度高的抗体,期间与最优抗体相似度越大其变异概率小,反之亦然。同时多受体随机编辑加强了克隆细胞与父代信息交流与协作,提高种群的多样性。本文将这种思想与传统的克隆操作相结合,因此每个免疫反应细胞的克隆操作包括下面3个步骤:① 克隆扩增,② 高变异克隆,③ 多细胞编辑。

3.3.1 克隆扩增

1) 抗体相似性定义。抗体 Ab_i 、 Ab_j 的相似性定义为

$$S_{ij} = \left| af_{Abi} / (af_{Abj} + \varepsilon) \right| \quad (6)$$

式中: af_{Abi} , af_{Abj} 是抗体调整后的亲和度大小; af_{Abj} 是当代反应群中最优抗体 Ab_j 的亲和度; 常数 $\varepsilon > 0$, 用于防止除零。在此抗体相似度利用抗体亲和度来评价, 抗体 Ab_i 与最优抗体相似度越高, S_{ij} 越接近 1。

2) 抗体浓度定义。抗体的浓度 C_i 即群体中相似抗体所占的比例, 即

$$C_i = \frac{1}{N} \sum_{j=1}^N S_{ij}$$

其中

$$S_{ij} = \begin{cases} 1, & S_{ij} \geq \tau \\ 0, & \text{其它} \end{cases} \quad (7)$$

式中: N 为抗体总数; τ 为抗体相似度阈值, 当任意两个抗体间相似度大于等于 τ 时, 可将两个抗体看作同类抗体, 此时 $S_{ij} = 1$ 计算抗体浓度, 反之, $S_{ij} = 0$ 不计抗体浓度。抗体浓度 C_i 的大小取决于 S_{ij} 在抗体中出现的次数。

抗体 Ab_i 克隆扩增规模正比于抗体刺激度, 与抗体浓度成反比。具体可描述为

$$cl(Ab_i) = \begin{cases} \text{int}[\alpha_1(af_{Abi} - af_{\min}) / (af_{\max} - af_{\min} + \varepsilon) + \beta_1], & C_i \leq \rho \\ \text{int}[\alpha_2 \exp(-C_i / \beta_2)], & C_i > \rho \end{cases} \quad (8.1)$$

$$(8.2)$$

式中: $af_{\max}$, $af_{\min}$ 表示当代的最大、最小亲和度, α_1 , β_1 , α_2 , β_2 是比例调节因子, $\text{Int}()$ 是表示取整运算, ρ 为抗体浓度阈值。

进化初期, 当种群内最高亲和度抗体的浓度小于等于抗体浓度阈值 ρ 时 ($C_i \leq \rho$), 按抗体刺激作用(调整后亲和度大小)的方式进行抗体克隆, 抗体克隆的数量取决于抗体亲和度大小, 亲和度越大, 抗体克隆的数量就越多。抗体克隆的规模为式(8.1)所示; 进化后期, 种群内相似抗体的增多, 抗体 Ab_i 的浓度 C_i 增大, 抗体克隆的数量主要取决于抗体浓度的影响, 浓度越大的抗体克隆的数量越少, 当种群内最高亲和度抗体的浓度大于浓度阈值 ρ 时 ($C_i > \rho$), 按抗体浓度影响来进行抗体克隆; 抗体克隆的规模为式(8.2)所示。

3.3.2 高变异克隆

高变异克隆是指抗体复制过程中发生高概率基因突变, 生成克隆子细胞群体的过程。当抗体相似度 S_{ij} 大于某一阈值时, 以小概率变异在较小邻域内精细搜索局部最优解; 当相似度 S_{ij} 小于等于某一阈值时, 以大概率变异在较大邻域内进行粗搜索, 以达到增加群体多样性的目的。设被克隆的 B 细胞 Ab_i , Ab_i^m 是 Ab_i 的第 m 个基因, 表达式如(9)所示。

$$cl_i^m = \text{mod}(Ab_i^m + \text{int}(\text{rand} - \alpha(1 - \exp(\beta(af_{\min} - af_{Abi}) / (af_{\max} - af_{\min} + \varepsilon))))), 2) \quad (9)$$

式中: cl_i 为克隆 Ab_i 生成的子细胞, cl_i^m 是 cl_i 的第 m 个基因, $\text{mod}(Ab, 2)$ 是对 Ab 模 2 取余运算; α 、 β 是比例调节因子, 通过调节其大小实现不同概率的变异; rand 为 $[0, 1]$ 之间均匀分布的随机数。

3.3.3 多细胞编辑

多细胞编辑是对高变异克隆生成的子细胞进行随机的基因改造过程, 这样产生的子细胞有可能比父代细胞具有与特异抗原更高的亲和度, 提高算法的收敛速度。具体操作是: 设有 3 个参与编辑的细胞, 编辑过程是先随机选择子细胞为 cl_j , 两次从免疫反应细胞群中随机选择 B 细胞 Ab_j , 随机截取两次选择的反应细胞的基因片段替换 cl_j 相应基因完成克隆子细胞的编辑过程。多细胞随机编辑主要特点: ① 在整个算子参数是随机生成, 提高了算法的适应性; ② 编辑后子细胞包含多父代信息, 提高了群体的多样性。

3.4 免疫记忆操作

通过免疫记忆操作实现免疫网络更新, 构造该算子时不仅要注意记忆进化结果, 而且要兼顾保持种群的多样性。主要思想: 首先通过浓度控制实现网络抑制, 防止克隆操作后的子细胞造成超级抗体垄断种群; 其次选择候选子细胞, 根据计算克隆抗体与高亲和度的子细胞之间的距离, 当大于预先选定的阈值时

选择其作为更新免疫细胞群的候选子细胞;最后进行网络更新,将克隆选择的结果构造一些小生境环境,亲和度更大的子细胞按相似程度,替代亲和度低,相似度高的B细胞,其他子细胞不发生替代,实现网络更新,生成下一代进化抗体群。

具体操作过程是:

1) 计算克隆子细胞之间距离,使其距离大于设定阈值,否则为冗余子细胞。

2) 按条件 $number(CL') < \lambda$; $\min(D(cl, CL')) \geq \eta$ 选择候选解;式中: $number$ 是计算细胞的个数; $\min$ 是取最小值运算, CL' 为备选细胞群,令 CL'' 是 CL 中每次剔除备选细胞和冗余细胞外的细胞集合, cl 是 CL'' 中亲和度最大的细胞, D 为细胞间的距离, λ 为备选细胞规模, η 为预先选定的阈值。

3) 以备选子细胞中的 cl_j 为中心,计算 cl_j 与 SBN 的距离,将距离小于特定阈值,且亲和度小于 cl_j 的个体作为一个集合,用 cl_j 替代其中距离最小的B细胞,其他不变,实现B细胞网络更新。

4 性能测试准则

为了描述不同算法性能的差异性,在此选择环境适应性和性能的稳定性进行比较。

1) 环境适应性。采用文[10]中指标 Acc 和 Ada 对算法的环境适应能力进行比较。 Acc 反映算法环境变化的能力及各环境下的收敛性; Ada 测试算法的搜索效率,分别表达为

$$Acc = \frac{1}{T} \sum_{t=1}^T (opt_t - b_{Tr_t}), \quad Ada = \frac{1}{T} \sum_{t=1}^T \left[\frac{1}{r_t} \left(\sum_{j=1}^{r_t} opt_t - b_{tj} \right) \right]$$

式中: T 为环境类型总数; opt_t 为第 t 个环境的理论最优值; r_t 为第 t 个环境的总迭代数; b_{Tr_t} 为第 t 个环境最后一代所得的最优函数值; b_{tj} 为第 t 个环境第 j 代所得最优函数值。从上式可知:若 Acc , Ada 越小,则算法环境适应性能越好。

2) 性能的稳定性。采用统计方法进行分析运行性能的稳定性。每种算法均独立运行数次,考察测试过程中每种环境下所能达到的最大值,比较运行数次测试过程中,该值的最大值、最小值、平均值和方差,计算公式为

$$M1_T = \max(af_{i,r_T}), i \in [1, 2, \dots, n]; \quad M2_T = \min(af_{i,r_T}), i \in [1, 2, \dots, n]; \quad M3_T = \frac{1}{n} \sum_{i=1}^n af_{i,r_T}; \quad M4_T = \frac{1}{n} \sum_{i=1}^n (af_{i,r_T} - \overline{af_{i,r_T}})$$

式中: af_{i,r_T} 为第 i 次测试,第 T 种环境类型第 r 代最大亲和度, n 是独立测试次数(文中 $n=50$), $\overline{af_{i,r_T}}$ 为第 T 种环境类型 n 次运行的最大亲和度的平均值。 $\max$ 为求最大值运算, $\min$ 为求最小值运算。

5 算例

选择如下函数测试算法性能

$$\begin{aligned} \max_{x \in R^{p(t)}} f(x, t) &= a(t)p(t) + \sum_{i=1}^{p(t)} (x_i^2 - a(t)\cos(2\pi x_i)) \\ a(t) &= \sigma_1(t), p(t) = 10 + 10 \times \sigma_2(t), \quad -5.12 \leq x \leq 5.12 \end{aligned}$$

函数 $f(x, t)$ 是高维 Rastrigin 函数移动峰问题,跟踪与搜索其最优解具有较大难度。式中: $\sigma_1(t)$ 为 0~2 之间的随机数, $\sigma_2(t)$ 为 0~4 之间的随机整数,该函数最小自变量维数为 10,最大为 50。

为了测试本文算法的性能,选取简单遗传算法(GA)、一种简单免疫算法(Sais)及改进免疫算法(Yasais)参与比较,此4种算法均对上述测试函数进行50次独立性能测试实验。算法测试中,先随机生成该问题的7个环境参数 $a(t)$ 和维数值 $p(t)$,将参数调到适当,为了排除干扰因素,参与比较的

表1 环境适应性比较

Tab.1 Comparison of adaptability to environment

参数	算法			
	GA	Sais	Yasais	本文算法
Acc	1.617 2	0.205 2	0.184 0	-1.310 9e-5
Ada	2.702 7	0.778 0	0.691 1	0.189 2

算法的亲合力或适应度设计相同,见式(4)。采用相同的环境总数($T=7$)、种群规模($N_1=80$)、反应子群规模($N_2=20$),相同进化代数,每次运行2 800代,每个环境下进化400代。本文算法为Gray码,其它均为二进制编码,基因片段长均为6。其它算法参数为:GA的交叉率为0.8变异率为0.05,比较算法克隆操作方式,见式(8.1)。与文中算法进化初期一样, $\alpha_1=\beta_1=5$,文中算法当浓度大于阈值(ρ 取0.8)时采用克隆方式,见式(8.2)。 $\alpha_2=5$, $\beta_2=2$;当相似度大于阈值 τ (文中取0.9)时 $\alpha=0.45$, $\beta=0.05$,否则 $\alpha=0.3$, $\beta=0.2$;其它参数与比较算法相同。经由以上4种算法对上述函数各自独立运行50次,获得相应的测试平均结果分别见表1、表2。

表2 各环境下统计值比较

Tab.2 Comparison of statistical results in different environments

参数		环境总数 T						
		1	2	3	4	5	6	7
最大值	理论最优值	26.512 5	26.620 9	26.377	26.729 4	26.458 3	26.566 7	26.756 5
	GA	25.568 4	25.090 6	25.908 1	26.625 1	24.681 9	26.566 7	25.716 6
	Sais	26.512 5	26.211 6	26.377	26.729 4	26.365	26.566 7	26.756 5
	Yasais	26.512 5	26.244 7	26.377	26.729 4	26.384 3	26.566 7	26.756 5
	本文算法	26.512 5	26.620 9	26.377	26.729 4	26.458 3	26.566 7	26.756 5
最小值	GA	23.829 2	23.467 3	24.333 2	25.595 7	22.419 6	26.566 7	23.103 7
	Sais	26.215 7	25.249 7	26.317 7	26.729 4	25.675 8	26.566 7	26.512 7
	Yasais	26.086 8	25.492 5	26.278 6	26.729 4	25.959 9	26.566 7	26.658 9
	本文算法	26.512 5	26.620 9	26.377	26.729 4	26.458 3	26.566 7	26.756 5
平均值	GA	24.660 5	24.220 4	25.104	26.301	23.497 8	26.566 7	24.350 6
	Sais	26.457 0	25.729 1	26.375 8	26.729 4	26.055 5	26.566 7	26.671 3
	Yasais	26.300 0	25.873 1	26.352 9	26.729 4	26.197 5	26.566 7	26.713 5
	本文算法	26.512 5	26.620 9	26.377	26.729 4	26.457 2	26.566 7	26.756 5
方差	GA	0.146	0.171 7	0.123 3	0.053 5	0.308 4	0	0.242 6
	Sais	0.005 4	0.061 7	0.000 1	0	0.028 2	0	0.003 1
	Yasais	0.008 7	0.024 9	0.001	0	0.007 2	0	0.000 9
	本文算法	0	0	0	0	0	0	0

表1反映了4种算法的跟踪能力和优化性能。本文算法中,Acc,Ada均为最小,这说明了本文算法跟踪目标模式的能力和适应环境变化的速度比其它3种算法更好;表2反映了算法的运行性能的稳定性。环境中每次搜索的极值越接近其理论最优值越好,比较各环境中最大亲和度(或适应度)的最大值、最小值和平均值的结果都表明了文中算法的优化性能更好,从实验结果可以看出每次独立运行都能找到各个环境的最优值,说明算法寻优性能良好;方差反映的是各环境中最大亲和度(或适应度)的离散程度,其值越小说明算法越稳定。文中算法的方差均为最小也进一步表明该算法的性能更为稳定。

6 结论

本文借鉴生物免疫系统的免疫反应所蕴含的动态进化机制,提出了针对高维动态环境下函数优化的免疫算法。算法考虑了抗体在亲和度成熟过程的增加学习机制,用来加快算法收敛速度,同时从编码方式、抗体浓度对克隆增殖的影响、抗体相似度调节克隆超变异概率及多细胞编辑几个方面,增加抗体的多样性,同时利用免疫选择算子,免疫记忆算子及网络抑制等手段,较好地实现了免疫细胞群在动态环境下的模式搜索、模式进化和中间结果记忆。借助性能评价指标,通过与3种较为出色的算法比较,结果表明了该算法具有较强的模式跟踪能力,同时具备良好的适应环境变化能力以及稳定的优化性能。

参考文献:

- [1] GREFENSTETTE J J, RAMSEY C L. An approach to anytime learning[C]//San Mateo:Organ Kaufmann, 1992:189-195.
- [2] ARAGÓN V S, ESQUIVEL S C. An evolutionary algorithm to track changes of optimum value locations in dynamic environments[J]. Computer Science and Technolgy, 2004, 4(3):27-134.
- [3] BENDTSEN C N, KRINK T. Dynamic memory model for non-stationary optimization[C]//Proc of The Congress of Evolutionary Computation, 2002:145-150.
- [4] OPPACHER F, WINEBERG M. The shifting balance genetic algorithm: improving the GA in a dynamic environment[C]//In the Proc. of the Genetic and Evolutionary Computation Conference GECCO99, 1999:504-510.
- [5] WALKER J, GARRETT S. Dynamic function optimization : comparing the performance of clonal selection and evolutionary strategies[J]. Lecture Notes on Computer Science, 2003, 787(2):273-284.
- [6] 张著洪, 钱淑渠. 自适应免疫算法及其对动态函数优化的跟踪[J]. 模式识别与人工智, 2007, 20(1):85-94.
- [7] 刘星宝, 蔡自兴, 王勇, 等. 应用于高维优化问题的免疫进化算法[J]. 控制与决策, 2011, 26(1):59-64.
- [8] 曾毅. 一种免疫算法的改进[J]. 华东交通大学学报, 2007, 24(1):123-128.
- [9] 孙勇智. 人工免疫系统模型、算法及其应用研究[D]. 杭州:浙江大学, 2004:45-48.
- [10] SIMÕES A, COSTA E. Using GA to deal with dynamic environments: a comparative study of several approaches based on promoting diversity[R]. Proc of the Genetic and Evolutionary Computation Conference, New York, USA: MorganKaufman, 2002:9-13.

An Immune Algorithm for High-dimension Dynamic Function Optimization

He Zhenmei

(School of Electrical and Electronic Engineering, East China Jiaotong University, Nanchang 330013, China)

Abstract: In the process of identifying and removing antigens in the immune system, immune cells interact and collaborate between each other, which quickly adapt to the environment change and have very strong learning and self-adapting control ability. In this paper, by adopting Gray code, using different cloning strategies, implementing hyper-mutation and multiple- cell editing with different probabilities on antibodies, the author presents an immune algorithm for high-dimension dynamic function optimization to improve the tracking capacity and diversity of antibodies. Compared with several typical algorithms, the simulation results show the performance and tracking capacity of the proposed algorithm for dynamic function optimization is improved obviously.

Key words: immune algorithm; dynamic function optimization; environment tracking