

文章编号: 1001-0920(2007)12-1395-04

一种并行免疫进化策略算法研究

程 博, 郭振宇, 王军平, 曹秉刚

(西安交通大学 机械工程学院, 西安 710049)

摘 要: 基于克隆选择原理, 提出一种自适应并行免疫进化策略. 在算法中根据抗体抗原亲和度将初始抗体种群分为两个子群, 相应地提出了精英克隆算子和超变异算子. 通过精英克隆算子提高算法局部搜索能力, 同时利用超变异算子维持种群多样性, 通过这两个功能互补算子的并行操作实现种群进化. 仿真表明, 自适应并行免疫进化策略搜索效率高, 能有效抑制早熟收敛现象, 可用于解决复杂机器学习问题.

关键词: 免疫算法; 克隆选择; 进化策略; 并行进化

中图分类号: TP18 **文献标识码:** A

A parallel immune evolutionary strategy

CHENG Bo, GUO Zhen-yu, WANG Jun-ping, CAO Bing-gang

(School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an 710049, China. Correspondent: CHENG Bo, E-mail: chengbo7681@126.com)

Abstract: Based on clonal selection theory, an adaptive parallel immune evolutionary strategy (PIES) is presented. According to antigen-antibody affinity, the original antibody population can be divided into two subgroups. Correspondingly, two immune operators, elitist clonal operator (ECO) and super mutation operator (SMO), are proposed. The former is adopted to improve the local search ability while the latter is used to maintain the population diversity. Thus, population evolution can be actualized by concurrently operating ECO and SMO. Experimental results show that PIES is of high efficiency and can effectively prevent premature convergence. Therefore, it can be employed to solve complicated machine learning problems.

Key words: Immune algorithm; Clonal selection; Evolution strategy; Parallel evolution

1 引 言

工程实际中碰到的优化问题大多属于多模态组合优化, 即存在多个最优解, 或存在一个全局最优解与众多局部最优解. 针对这种类似多峰值函数问题的优化, 基于人工免疫与进化策略相融合的思想, 产生了许多免疫进化算法. 文献[1]采用免疫克隆机理实现了局部与全局寻优相结合, 不足之处是对于较劣等抗体, 采取克隆操作, 寻优效率较低. 文献[2]提出利用背景知识和局部特征信息实现疫苗提取, 以获取局部极优.

在并行多种群免疫进化算法的研究中, 常用的操作方法是: 同时利用几个初始种群, 采用几个相似功能的变异算子, 通过各个子种群间信息交流和竞争的方法获得最优个体^[3-5].

本文基于免疫克隆、多种群进化以及功能互补

的进化算子互相协作的思想, 提出一种自适应并行免疫进化策略(PIES). 新的子群竞争和信息交流方式的实现过程是: 根据抗体抗原亲和度差异, 以抗体种群平均亲和度为界, 将一个初始种群动态地划为两个子群; 然后由两个迥然不同的具有互补功能的变异算子协同操作来获得最优抗体个体.

2 克隆选择理论

当生物免疫系统受到外来抗原激励时, 与抗原亲和度高的 B 细胞被选出, 发生克隆扩增和超突变现象, 以在局部范围内搜索亲和度更高的 B 细胞; 同时, 受体编辑现象使亲和度低的子 B 细胞在形状空间中突变为离其较远的点, 以在全局范围内搜索亲和度更高的 B 细胞. 此外, 一些亲和度低的子 B 细胞死亡, 被骨髓产生的新 B 细胞所替代, 以增加群体多样性. 经过若干世代进化, 最终产生了亲和度

收稿日期: 2006-09-20; 修回日期: 2006-11-23.

基金项目: 国家星火计划项目(2004EA105003); 汽车安全与节能国家重点实验室开放基金项目(KF2005-6).

作者简介: 程博(1976—), 男, 陕西大荔人, 博士生, 从事计算智能、电动汽车电池管理系统的研究; 曹秉刚(1953—), 男, 甘肃会宁人, 教授, 博士生导师, 从事机电系统控制、电动汽车驱动控制等研究.

很高的B细胞,它进一步分化成浆细胞,产生大量与受体形状相同的抗体,以消灭抗原.

3 自适应并行免疫进化策略

连续对象待寻优问题的目标函数为

$$f^* = f(x_i^*) = \min f(x_i),$$

$$i = 1, 2, \dots, m, x_i \in [p_i, q_i].$$

对变量 x 进行编码,记为 $a, a = a_1, a_2, \dots, a_L, a \in I$. 抗体 a 是长度为 L 的字符串, a_i 为 a 的等位基因,其可能取值与编码方式有关. 本文采用实数编码方式, I 为抗体空间,变量 x 称为抗体 a 的解码. f 为 I 上的正实值函数,称为抗体抗原亲合度函数.

一般地,将抗体位串分为 m 段,每段长度为 l_i ,

则 $L = \sum_{i=1}^m l_i$, 每一段分别表示变量 x_i . 特别地,对于二进制编码方式, x_i 的译码表示如下:

$$x_i = p_i + \frac{q_i - p_i}{2^{l_i} - 1} \left(\sum_{j=1}^{l_i} a_j 2^{j-1} \right). \quad (1)$$

抗体种群表示为 $P = \{a_1, a_2, \dots, a_N\}$, 正整数 N 表示种群规模. 种群进化流程如图1所示. 当前抗体种群为 P_k , 下一代种群为 P_{k+1} , 从 A_k 到 I_k 均表示进化子种群.

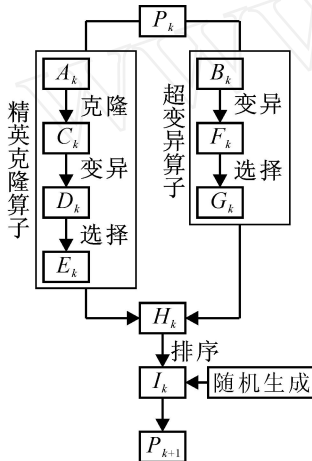


图1 抗体种群进化流程

操作机理的依据是:根据克隆选择理论,针对克隆扩增和超突变现象设计精英克隆算子,针对受体编辑现象应用超变异算子.

3.1 精英克隆算子

基于免疫学中的克隆操作机理,与自适应平均变异算子的优势特点相结合,设计出一种新的免疫算子——精英克隆算子(ECO).

设抗体抗原间亲和度函数为 $f(\cdot)$, 在当前种群 P_k 中,由高于平均亲合度函数的 M 个抗体组成种群 $A_k, A_k = \{a_1, a_2, \dots, a_M\}$. ECO分4个步骤:精英选取、克隆、变异以及克隆选择. 操作过程如下:

(1) 克隆

将第 i 个抗体 a_i 克隆为数目为 q_i 个相同的抗体

$a_{i1}, a_{i2}, \dots, a_{iq_i}, q_i$ 定义如下:

$$q_i = \text{Int}(C * p_i), i = 1, 2, \dots, M. \quad (2)$$

其中 p_i 为 a_i 产生新个体的概率,定义如下:

$$p_i = f_i / \sum_{l=1}^M f_l, i = 1, 2, \dots, M. \quad (3)$$

常数 C 为种群克隆规模, q_i 可依据 C 和 p_i 自适应调整, $\text{Int}(\cdot)$ 为上取整函数. 克隆过后,种群变为

$$C_k = \{A_1, A_2, \dots, A_M\}. \quad (4)$$

其中

$$A_j = \{a_{j1}, a_{j2}, \dots, a_{jq_j}\}, j = 1, 2, \dots, M. \quad (5)$$

(2) 克隆变异

对于多模态函数的优化,自适应平均变异比传统采用的高斯变异具有更好的搜索能力^[6]. 因此,本文提出以自适应平均变异算子结合克隆操作,形成一种新的免疫算子.

种群 C_k 中,抗体 a_i 的分量为 $a_i(j)$, 新抗体分量为 $A_i(j)$, 则相应位置更新方程如下:

$$A_i(j) = a_i(j) + i(j) [C_j(0, 1) + i(j) N_j(0, 1)]. \quad (6)$$

其中: $a_i(j), A_i(j), i(j)$ 和 $i(j)$ 相应地表示向量 a_i, A_i, i 和 i 的第 j 个分量; $C_j(0, 1)$ 表示一个中心值为0, 比例系数为1的一维柯西随机数; $N_j(0, 1)$ 表示均值为0, 方差为1的正态分布随机数; $i(j)$ 表示柯西变异和高斯变异的总体标准差; $i(j)$ 的变化决定了概率密度函数的形状.

自适应均匀变异的优点在于:随着被优化目标函数形状的变化,概率密度函数的形状在高斯与柯西概率密度函数之间能够自适应调整.

(3) 克隆选择

克隆变异后, D_k 取代种群 C_k . 对于任意 $i = 1, 2, \dots, M$, 若存在抗体

$$e_i = \{a_{ij} \mid \max f(a_{ij}), j = 1, 2, \dots, q_i\}, \quad (7)$$

使得 $f(e_i) > f(a_i), a_i \in A_k$, 则用抗体 e_i 代替抗体 a_i , 从而形成种群 E_k , 完成信息更新.

精英克隆算子的本质是:对于具有较高亲合度的抗体,由克隆自身产生多个相似的抗体组成种群,通过竞争的方式选出一个精英抗体为最优. ECO实质上是对点的搜索扩展为一个点附近临域的搜索,然后由临域中最优代替点最优.

3.2 超变异算子

由 P_k 中低于平均亲和度的抗体组成种群 $B_k, B_k = \{a_{ij}, i = 1, 2, \dots, S, j = 1, 2, \dots, L\}$. S 为种群规模, L 为编码长度.

超变异算子采用文献^[7]中提出的均匀变异,以代替高斯变异. 这是因为后者只能产生位于原始抗体附近的变异,而均匀变异则产生远离原始抗体

的变异,使得更大的解空间得到搜索,可以更好地维持种群多样性。

均匀变异的常用操作方式是,通过对 a_{ij} 加入来自某区间的随机数来产生变异,最简单的方法是使用如下规则:

$$a_{ij} = a_{ij} + j_j \text{Rand}(0,1), \quad j = 1, 2, \dots, L. \quad (8)$$

式中: j_j 是一个窄化搜索区域因子; j 是当前循环数; $\text{Rand}(0,1)$ 表示从 0 到 1 的均匀分布随机变量;

$$j_j = \begin{cases} a_{ij, \min} - a_{ij}, & \text{Rand}(0,1) < 0.5; \\ a_{ij, \max} - a_{ij}, & \text{Rand}(0,1) \geq 0.5. \end{cases} \quad (9)$$

式(9)表示每次随机选择一个 j_j , 每一个 a_{ij} 准确地位于一个对应区间 $[a_{ij, \min}, a_{ij, \max}]$ 内, 随着循环次数增加, j_j 减少, 搜索空间也逐渐减小。

经过 ECO 和 SMO 运算之后, 在 I_k 中由随机生成的新抗体代替亲和度最低的 $\text{Int}(N)$ 个抗体, 增加种群的多样性, 完成 P_k 到 P_{k+1} 的进化过程。 $\text{Int}(\ast)$ 为上取整函数, 预先设定, 一般为 $0.1 \sim 0.15$ 。

4 仿真实例

4.1 函数优化问题

用自适应并行免疫进化策略 (PIES) 优化以下 4 个典型的多峰值函数 (求最小值), 并与传统进化

策略 (CESA), 免疫单克隆策略 (IMSA) 进行比较^[8]。

$$f_1(x, y) = x^2 + y^2 - 0.3 \cos(3x) + 0.3 \cos(4y) + 0.3, \quad x, y \in [-1, 1];$$

$$f_2(x, y) = 4x^2 - 2.1x^4 + \frac{1}{3}x^6 + xy - 4y^2 + 4y^4, \quad x, y \in [-5, 5];$$

$$f_3(x) = 100 \sum_{i=1}^{n-1} (x_{i+1} - x_i^2)^2 + (x_i - 1)^2, \quad x_i \in [-10, 10];$$

$$f_4(x) = nA + \sum_{i=1}^n (x_i^2 - A \cos(2x_i)), \quad x_i \in [-5.12, 5.12].$$

为便于比较 3 种算法性能, 主要参数设置与文献[8]相同。对于函数 f_1 和 f_2 , CESA 的种群规模为 100, IMSA 和 PIES 的种群规模为 50, 克隆规模为 100, 变异概率为 0.1, 最大进化代数为 500; 对于函数 f_3 和 f_4 , 种群规模与二维函数一致, 最大进化代

表 1 函数 f_1 和 f_2 的优化结果

算法	最大进化代数		最小进化代数		平均进化代数		每代消耗时间	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
CESA	150	163	14	20	150	98.1	0.008 3	0.018 9
IMSA	32	90	11	65	22.6	81.3	0.008 0	0.021 9
PIES	22	46	9	22	14.5	26.7	0.007 6	0.037 6

表 2 函数 f_3 和 f_4 的优化结果

算法	优化指标	f_3			f_4		
		$n = 5$	$n = 10$	$n = 30$	$n = 5$	$n = 10$	$n = 30$
IMSA	最大进化代数	568	/	/	136	322	/
	最小进化代数	224	/	/	81	210	/
	平均进化代数	390.5	/	/	121.7	253.5	/
	每代消耗时间	0.238 2	/	/	0.336 2	0.402 0	/
PIES	最大进化代数	46	93	286	49	56	163
	最小进化代数	21	51	161	26	32	53
	平均进化代数	33.1	73.4	192.4	38.4	43.2	113.4
	每代消耗时间	0.472 2	0.672 3	1.654 3	0.568 2	0.765 6	1.656 4

表 3 3 种算法优化函数的平均最优值与标准方差比较

函数	精度	CESA		IMSA		PIES	
		平均最优值	标准方差	平均最优值	标准方差	平均最优值	标准方差
f_1	10^{-3}	- 0.180 8	8.6×10^{-3}	- 0.184 8	5.82×10^{-3}	- 0.184 8	4.63×10^{-3}
	10^{-3}	- 1.024 5	3.1×10^{-4}	- 1.031 6	2.74×10^{-4}	- 1.031 6	2.31×10^{-4}
f_3	10^{-2}			- 0.007	9.7×10^{-3}	- 0.006	8.4×10^{-3}
	10^{-3}			$6.605e-0.04$	1.12×10^{-2}	$5.384e-0.04$	1.01×10^{-2}
f_4	10^{-2}			- 0.006	1.1×10^{-2}	- 0.008	1.2×10^{-2}
	10^{-3}			$8.946e-0.04$	3.37×10^{-2}	$8.683e-0.04$	2.98×10^{-2}

数为 1 000, 优化精度分别为 0.01 和 0.001.

表 1 和表 2 是 3 种算法均采用不同初始种群重复实验 10 次的优化结果对比, 表 3 是 3 种算法优化函数的平均最优值和标准方差比较. 从表 1 可以看出, 对于函数 f_1 和 f_2 , PIES 与 CESA 和 IMSA 相比, 能够在更少的平均进化代数内取得函数最优值, 说明 PIES 的收敛速度比较快. 从表 2 可以看出, 对于函数 f_3 和 f_4 , CESA 在指定进化代数内无法取得符合条件的最优解; 利用 IMSA, 对于函数 f_3 , 当 $n = 10$ 和 $n = 30$ 时无法取得优化结果, 对于函数 f_4 , $n = 30$ 时无法取得优化结果, 说明 IMSA 的局部搜索能力需要加强; 利用 PIES, 对于函数 f_3 和 f_4 , $n = 5, 10$ 和 30 时, 均可取得优化结果.

图 2 和图 3 分别是函数 f_3 和 f_4 平均函数值相对于进化代数的优化结果. 从图可以看出, 随着进化代数增加, IMSA 和 PIES 最终得到一系列局部极值. 相比于 IMSA, PIES 获得了更多数量的局部极值, 说明优化多模态高维函数时, PIES 更具优势.

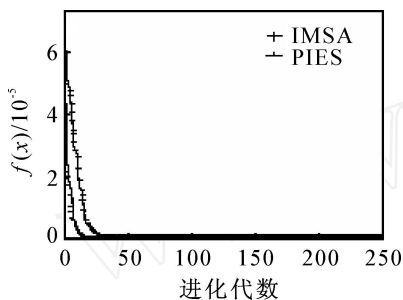


图 2 函数 f_3 的优化结果

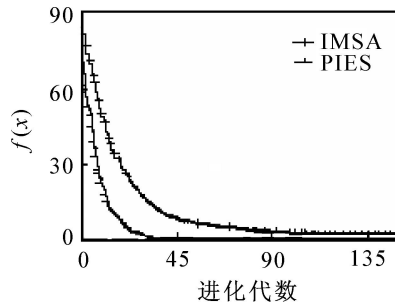


图 3 函数 f_4 的优化结果

f_3 和 f_4 的优化结果说明, 相比于 IMSA, PIES 采用两个功能互补的算子并行进化种群是有效的, 算法局部搜索能力得到加强, 多样性能得到更好的保持, 从而使算法的搜索性能得到提高.

从表 3 可以看出, PIES 和 IMSA 在优化 4 个函数时, 得到的平均最优值与标准方差比较接近, 都优于 CESA. 说明 PIES 的稳定性、搜索精度与 IMSA 的指标基本相当.

4.2 TSP 优化问题

传统进化策略(CESA)与 PIES 进行对比. 利用

PIES 优化 TSP 问题时, 每一个抗体代表一个可行路径, 即一个可行解; 抗原为目标函数; 亲和度代表可行路径接近最短路径的程度, 取为路径长度的倒数. 采用遍历城市的排列次序进行编码, 两种算法的群体大小均为 100. 以 20 个城市 TSP 优化问题为例, 两种算法分别经过 20 次重复迭代, 虽然路径的起始点不同, 但最终遍历全部 20 个城市的路径相同. 两种算法最优个体变化过程如图 4 所示.

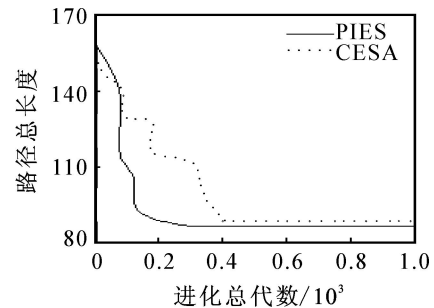


图 4 最优个体的变化

两种算法的最大进化代数设为 1 000 次, 各自独立运行 20 次. CESA 找到最佳路径的次数为 13 次, 平均最佳路径长度为 89.3; PIES 找到最佳路径的次数为 19 次, 平均最佳路径长度为 86.4. 结合图 4 可以看出, 本文算法找到最佳路径的次数和路径平均长度均优于 CESA.

5 结 论

本文提出的自适应并行免疫进化策略算法利用两个算子并行化操作, 通过空间的扩展和压缩加强较优秀个体周围的空间搜索, 并结合自适应变异操作, 提高了算法的局部搜索能力, 通过超变异算子有效维持种群的多样性, 提高了算法的搜索效率. 多模态数值函数测试和 TSP 优化表明, PIES 收敛速度快, 能有效防止早熟收敛, 可用于解决复杂的机器学习问题.

参考文献(References)

- [1] Liu Ruochen, Du Haifeng, Jiao Licheng. Immune clonal strategies [C]. 5th Int Conf on Computational Intelligence and Multimedia Applications (ICCIMA '03). Piscataway: IEEE Press, 2003: 290-295.
- [2] Zheng Jianguo, Liu Fang, Jiao Licheng. Immune evolutionary strategy and its application [C]. 6th Int Conf on Signal Processing. Beijing, 2002, 2: 1819-1822.
- [3] 杨孔雨, 王秀峰. 自适应多模态免疫进化算法的研究与实现[J]. 控制与决策, 2005, 20(6): 717-720.
(Yang Kong-yu, Wang Xiufeng. Research and implement of adaptive multi-modal immune evolution algorithm[J]. Control and Decision, 2005, 20(6): 717-720.)

(下转第 1402 页)

R_9 : If x is A_{13} or A_{17} , Then y is B_9 ;

R_{10} : If x is A_{14} or A_{16} , Then y is B_{10} .

系统函数 $y = G(x)$ 和待逼近函数 $y = f(x)$ 的比较如图 3 所示.

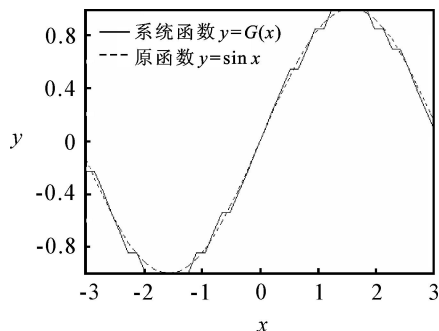


图3 系统函数与待逼近函数的比较

5 结 语

本文重点讨论了布尔模糊系统逼近的充分条件,并在此基础上给出了一个计算逼近给定函数到指定误差时所需要的模糊集个数的最小上界公式.最后以一个例子说明了设计布尔模糊系统的方法.

参考文献 (References)

- [1] Li Y M, Shi Z K, Li Z H. Approximation theory of fuzzy systems based upon genuine many-valued implication—SISO case [J]. Fuzzy Sets and Systems, 2002, 130: 147-197.
- [2] Li Y M, Shi Z K, Li Z H. Approximation theory of fuzzy systems based upon genuine many-valued implication —MIMO case[J]. Fuzzy Sets and Systems, 2002, 130: 159-174.
- [3] Bugarin A J, Barro S. Reasoning with truth values on compacted fuzzy chained rules [J]. IEEE Trans on Systems, Man and Cybernet, 1998, 28(1): 34-46.
- [4] Cao Z, Kandel A. Applicability of some fuzzy implication operators [J]. Fuzzy Sets and Systems, 1989, 31: 151-186.
- [5] Nguyen H T, Kreinovich V, Sirisauatkin O. Fuzzy control as a universal control tool [J]. Fuzzy Sets and Systems, 1996, 80: 71-86.
- [6] Cao Z, Park D, Kandel A. Investigations on the applicability of fuzzy inference [J]. Fuzzy Sets and Systems, 1992, 49: 151-169.
- [7] Cordon O, Herrera F, Peregrin A. Applicability of the fuzzy operators in the design of fuzzy logic controllers [J]. Fuzzy Sets and Systems, 1997, 86: 15-41.
- [8] Ying H. Sufficient condition on uniform approximation of multivariate functions by Takagi-Sugeno fuzzy systems with linear rule consequent [J]. IEEE Trans on Systems, Man and Cybernetics, 1998, 28(4): 515-520.
- [9] Wang L X. A course in fuzzy systems and control [M]. London: Prentice Hall PTR, 1997.
- [10] Zeng K, Zhang N Y, Xu W L. A comparative study on sufficient conditions for Takagi-Sugeno fuzzy systems as universal approximators [J]. IEEE Trans on Fuzzy Systems, 2000, 8(6): 773-780.
- [11] Lorentz G G. Approximation of functions [M]. London: Holt, Rinehart, 1966.
- [4] 李智勇, 童调生. 基于多种群进化小生境遗传算法的神经网络进化设计方法研究 [J]. 控制与决策, 2003, 18(5): 607-610.
(Li Zhi-yong, Tong Tiao-sheng. Research on ANN evolutionary design method based on populations evolution niche genetic algorithm [J]. Control and Decision, 2003, 18(5): 607-610.)
- [5] 王向军, 嵇斗, 张民. 一种多群竞争进化规划算法 [J]. 电子学报, 2004, 11(32): 1824-1828.
(Wang Xiang-jun, Ji Dou, Zhang Min. A multi-subgroup competition evolutionary programming algorithm [J]. Acta Electronica Sinica, 2004, 11(32): 1824-1828.)
- [6] Kumar Chellapillal, David Fogel. Two new mutation operators for enhanced search and optimization in evolutionary programming [C]. Applications of Soft Computing. Piscataway: IEEE Press, 1997, 3: 260-269.
- [7] 莫宏伟, 金鸿章. 用于函数优化的改进免疫克隆多样性算法 [J]. 哈尔滨工程大学学报, 2004, 25(1): 75-79.
(Mo Hong-wei, Jin Hong-zhang. The modified immune diversity algorithm used in function optimization [J]. J of Harbin Engineering University, 2004, 25(1): 75-79.)
- [8] 刘若辰, 杜海峰, 焦季成. 一种免疫单克隆策略算法 [J]. 电子学报, 2004, 11(32): 1880-1884.
(Liu Ruo-chen, Du Hai-feng, Jiao Li-cheng. An immune monoclonal strategy algorithm [J]. Acta Electronica Sinica, 2004, 11(32): 1880-1884.)
- [6] Kumar Chellapillal, David Fogel. Two new mutation

(上接第 1398 页)