

文章编号: 1001-0920(2004)03-0252-05

一类自适应免疫进化算法

左兴权, 李士勇

(哈尔滨工业大学 控制科学与工程系, 黑龙江 哈尔滨 150001)

摘要: 基于免疫系统中的进化机理, 提出一种自适应免疫进化算法。通过定义扩展半径和突变半径两个新算法参数构造了较小和较大两个邻域, 分别利用这两个邻域进行局部和全局搜索, 从而形成两层领域搜索机制, 以保证算法的全局和局部搜索能力。定义了群体的多样性, 并以此自适应调节算法参数以提高算法性能。给出了算法的全局收敛性证明。仿真结果表明, 该算法收敛速度快, 具有良好的全局寻优和局部求精能力。

关键词: 人工免疫系统; 免疫算法; 遗传算法; 优化计算

中图分类号: TP18

文献标识码: A

Adaptive immune evolutionary algorithm

ZUO Xing-quan, LI Shi-yong

(Department of Control Science and Engineering, Harbin Institute of Technology, Harbin 150001, China. Correspondent: ZUO Xing-quan, E-mail: zuoxingquan@vip.sina.com)

Abstract: An adaptive immune evolutionary algorithm is proposed based on the evolutionary principle in the immune system. Two new algorithm parameters of expansion radius and mutation radius are defined to construct a small and a large neighborhoods, then the two neighborhoods are used to perform local and global search respectively, so that two-level neighborhood search mechanism is realized to ensure the global and local search capabilities of the algorithm. The degree of diversity in population is defined and used to adjust the algorithm parameters adaptively, which improves the performances of the algorithm. The algorithm is proved to be convergent. The simulation results show that the algorithm converges quickly, and has satisfactory capabilities of global and local search.

Key words: artificial immune system; immune algorithm; genetic algorithm; optimization computation

1 引言

在管理科学、计算机科学、控制工程等领域存在着大量优化问题, 对于其中的很多 NP 完全问题, 传统优化方法易陷入局部最优解, 而且当问题规模增大时, 往往由于算法的时间复杂性太大而丧失了可行性。近年来, 以遗传算法(GA)、模拟退火(SA)和禁忌搜索(TS)为代表的智能优化算法为 NP 完全问题的解决提供了新的途径^[1]。然而 GA 的局部搜索能力不理想且常出现进化缓慢的现象, SA 和 TS

虽具有良好的局部搜索能力, 但其全局寻优能力弱于 GA, 且搜索效率较低。由此, 构造具有良好全局寻优和局部求精能力的高效智能优化算法, 已成为优化计算领域中的一个重要研究课题^[2]。目前广泛采用的方法是将 GA 与局部搜索算法相混合, 利用 GA 的全局探索能力和局部搜索算法的局部求精能力来改善混合算法的局部和全局寻优性能。然而混合算法仍难以完全克服 GA 和局部搜索算法的缺点, 如搜索效率不高, 可能出现早熟等。

收稿日期: 2003-02-23; **修回日期:** 2003-05-20

基金项目: 国家自然科学基金资助项目(50138010); 哈尔滨工业大学跨学科交叉性研究基金资助项目(HIT.MD 2001.02)。

作者简介: 左兴权(1971—), 男, 黑龙江伊春人, 博士生, 从事免疫优化计算、智能控制的研究; 李士勇(1943—), 男, 黑龙江哈尔滨人, 教授, 博士生导师, 从事模糊控制、智能控制理论及应用等研究。

生物免疫系统是一个智能信息处理系统^[3,4], 可为探索新的智能优化算法提供启发^[5-7]. 本文基于生物适应性免疫应答中的进化机理, 提出一种具有新颖的全局和局部搜索机制的优化算法——自适应免疫进化算法(A IEA). 该算法利用较小和较大两个邻域在解空间中进行多点并行邻域搜索, 在全局范围内搜索评价价值较高区域的同时, 也在该区域局部搜索最优解, 实现了从全局到局部的两层领域搜索机制, 从而使算法具有较强的全局和局部搜索能力. 定义了群体的多样性, 并以此自适应调节算法参数以进一步提高算法的全局寻优能力和收敛速度. 仿真结果验证了该算法的有效性.

2 生物免疫系统中的进化机理

与达尔文的生物进化原理相似, 免疫系统中也存在着进化现象^[8]. 免疫系统中有大量带有独特受体的 B 细胞, 每个 B 细胞受体的形状可用一个 n 维实向量描述, 因此可表示为 n 维欧几里德空间中一点, 称此空间为欧几里德形状空间. 两 B 细胞的受体形状越相似, 它们在形状空间中距离越近. 当抗原侵入机体时, B 细胞受体与抗原形状互补程度越大, 二者间亲和力越高, 从而更易结合. B 细胞群体通过如下进化过程产生抗体^[7]:

- 1) 选出与入侵抗原亲和力高的 B 细胞
- 2) 该 B 细胞分裂为若干子 B 细胞, 称为克隆扩增. 子 B 细胞受体形状在母细胞的基础上发生微小变异, 即超突变. B 细胞通过克隆扩增在形状空间中的小邻域内产生若干子 B 细胞, 以在局部范围内搜索亲和力更高的 B 细胞
- 3) 在克隆扩增生成亲和力更高的子 B 细胞的同时, 也产生了亲和力低的子 B 细胞. 一些亲和力低的子 B 细胞删除其受体并生成新受体, 称为受体修饰. 受体修饰使得子 B 细胞可能突变为形状空间中离其较远的点, 以避免在寻求高亲和力 B 细胞的过程中陷入局部最优
- 4) 一些亲和力低的子 B 细胞死亡, 同时骨髓产生一些新的 B 细胞加入群体, 以保持群体的多样性

经过若干世代的选择、克隆扩增、受体修饰和骨髓产生新 B 细胞的过程, 最终产生了亲和力很高的 B 细胞. 该 B 细胞分化为浆细胞, 产生与受体形状相同的抗体以消灭抗原, 发生适应性免疫应答.

3 自适应免疫进化算法

由上可知, 免疫系统通过无性繁殖来实现 B 细胞群体的进化, 以搜索亲和力最高的 B 细胞. 其中: 克隆扩增和受体修饰分别用于局部和全局搜索, 骨

髓产生新 B 细胞以增加群体多样性. 本文把 B 细胞及其亲和力分别看作问题的可行解(个体)和其评价价值, 模拟适应性免疫应答过程提出一种 A IEA. 下面给出 A IEA 的具体操作和计算步骤.

3.1 算法描述

- 1) 初始化: 随机产生 n 个实数编码的个体, 组成初始群体 A_k , 令 $k = 0$, 计算每个个体的评价价值
- 2) 选择操作: 选出群体 A_k 中评价价值最高的 $\text{integer}(n \cdot \alpha)$ 个个体组成群体 B_k , $0 < \alpha < 1$, 定义 α 为选择率
- 3) 调节算法参数: 根据群体 B_k 的多样性自适应调节算法参数(具体方法见第 3.2 节).
- 4) 扩展操作: 模拟克隆扩增和超突变过程. 本文构造了一个较小邻域, 群体 B_k 中每个个体在其小邻域内随机产生若干新个体, $\text{integer}(n \cdot \alpha)$ 个个体共产生 n 个新个体, 组成群体 C_k . 群体 B_k 中任一个体 v_i 的小邻域构造为

$$SN(v_i) = \{v \mid |v - v_i| \leq r, v \in \Omega, r > 0\}. \quad (1)$$

其中: Ω 为可行解空间; $|\cdot|$ 为欧几里德范数; $SN(v_i)$ 由与 v_i 的欧氏距离不大于常数 r 的所有可行解构成, 在可行解空间中是以 r 为半径的球形区域, 定义 r 为扩展半径.

在适应性免疫应答中, B 细胞与抗原的亲和力越高, 其分裂产生的子 B 细胞越多. 模拟这一现象, 本文用正比选择法^[9] 确定群体 B_k 中每个个体产生新个体的数目. 设群体 B_k 中有 m 个个体, 分别为 v_1, v_2, \dots, v_m , 其评价价值分别为 f_1, f_2, \dots, f_m , 则群体 B_k 中每个个体产生新个体的概率为

$$p_i = f_i / \sum_{i=1}^m f_i, \quad i = 1, 2, \dots, m. \quad (2)$$

累积概率为

$$q_i = \sum_{t=1}^i p_t, \quad i = 1, 2, \dots, m. \quad (3)$$

令 $q_0 = 0$, 在 $[0, 1]$ 上产生一个均匀分布的随机数 w . 若 $q_{i-1} < w < q_i$, 则 v_i 产生一个新个体, 产生 n 个这样的随机数, 就得到 n 个新个体. 扩展操作相当于在优秀个体的小邻域内搜索更优秀的个体. 个体评价价值越高, 其邻域内存在优秀个体的概率越大, 故其产生更多新个体的概率越大, 以在其邻域内搜索更多次.

- 5) 突变操作: 在该操作中, 本文构造了一个较大邻域. 群体 C_k 中评价价值最低的 $n - \text{integer}(n \cdot \alpha)$ 个个体中的每个个体突变为其较大邻域内的任一个

体突变后的个体与群体 C_k 中未突变的个体一起组成群体 D_k 。设个体 v_j 发生突变, 其较大邻域构造为

$$MN(v_j) = \{v \mid |v - v_j| \leq R, R > 0\} \quad (4)$$

$MN(v_j)$ 在解空间中是以 v_j 为中心, 以 R 为半径的球形区域, 定义 R 为突变半径。突变半径应远大于扩展半径。

6) 替换操作: 群体 D_k 中评价最低值的 L 个个体替换为随机个体形成群体 E_k 。替换操作模拟骨髓产生新 B 细胞过程以增加群体多样性。

7) 最优个体保留^[10]: 为了保证算法收敛以及避免当前群体中最优个体被扩展操作破坏掉, 将群体 E_k 中评价最低值的个体替换为 A_k 中评价最高的个体, 形成下一代群体 A_{k+1} 。

若满足收敛准则, 则算法结束, 否则 $k = k + 1$, 返回 2) 开始下一代演化。自适应免疫进化算法的群体演化过程如图 1 所示。

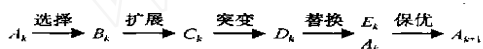


图1 群体演化过程

3.2 算法参数的自适应调节

用 A IEA 优化计算, 如果扩展半径和突变半径较大, 可保持群体多样性, 但群体进化缓慢; 如果扩展半径和突变半径较小, 算法可快速收敛, 但群体多样性减小很快, 算法容易陷入局部最优。因此算法快速收敛和保持群体多样性具有矛盾性。

当群体多样性较小时, 个体均集中在解空间某一小区域内, 算法接近局部或全局最优。此时扩展半径和突变半径应增大, 以使个体在解空间更广阔的范围内变异, 增大群体的多样性, 避免陷入局部最优或脱离局部最优。同时选择率应减小, 以使群体 B_k 中每个个体产生更多新个体, 即在其小邻域内搜索更多次, 使得扩展半径增大不会减弱算法的局部搜索能力。当群体多样性较大时, 群体正在进化中, 为使算法快速收敛, 扩展和突变半径应减小。由于不能确定哪一个个体的邻域内存在更优秀的个体, 选择率应增大以在更多个体的邻域内搜索。

如果算法参数根据群体的多样性适当变化, 则算法不仅能快速收敛, 而且能保持群体多样性以避免陷入局部最优解。本文用群体中所有个体相互间的平均距离来定义群体的多样性, 并以此自适应调节算法参数。假设第 k 世代中群体 B_k 包含 m 个个体, 则这些个体间的平均距离为

$$\bar{d}^{(k)} = \frac{1}{m(m-1)} \sum_{i=1}^m \sum_{j=1}^m d(v_i^{(k)}, v_j^{(k)}), \quad i \neq j \quad (5)$$

群体 B_k 的多样性定义为

$$D^{(k)} = \begin{cases} \bar{d}^{(k)} / d_{\max}, & \bar{d}^{(k)} < d_{\max}; \\ 1, & \text{else} \end{cases} \quad (6)$$

其中 d_{\max} 为给定常数。第 k 世代的算法参数按下式进行自适应调节:

$$\alpha^{(k)} = \alpha_0 + \eta_D^{(k)}, \quad (7)$$

$$r^{(k)} = r_0 + \eta_r(1 - D^{(k)}), \quad (8)$$

$$R^{(k)} = R_0 + \eta_R(1 - D^{(k)}), \quad (9)$$

其中: $\alpha^{(k)}, r^{(k)}, R^{(k)}$ 分别为第 k 世代的选择率、扩展和突变半径; α_0, r_0, R_0 分别为相应参数的最小值; $\eta_\alpha, \eta_r, \eta_R$ 分别为相应参数的调节范围。

4 算法的优化机理及收敛性

4.1 算法的优化机理

下面通过 A IEA 优化一个多峰函数来说明其优化机理。为说明方便, 设函数 $F(x)$ 只有一个自变量 x (如图 2), 算法参数固定不变, 群体规模为 10, 选择率为 0.1, 个体的函数值作为其评价。设初始群体中个体 a 的编码为 (x_0) , 其评价 $F(x_0)$ 最高而被选择操作选出。扩展操作在 x_0 的小邻域 $(x_0 - r, x_0 + r)$ 内随机产生 10 个新个体, 即在 x_0 的小邻域内随机搜索 10 次。新个体中评价最高的个体保留, 其他 9 个新个体进行突变操作, 即在 x_0 的大邻域 $(x_0 - R, x_0 + R)$ 内随机搜索 9 次。替换操作在解空间中随机搜索 L 次。因此, 在每一世代中, A IEA 能够在当前群体中最优个体的小邻域和大邻域内同时搜索评价更高的个体。

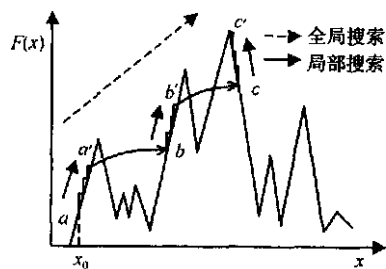


图2 A IEA 的优化机理

如果扩展操作在群体中最优个体的小邻域内找到更优秀的解, 而突变和替换操作没有找到更好的解, 则该优秀解被选择操作选入下一世代, 扩展操作又在其小邻域内搜索更优秀的解。因此扩展操作可被看作一种爬山算法^[1]。经过若干世代, 扩展操作向上“爬山”找到个体 a 。设个体 a 经扩展操作后, 突变操作找到评价更高的个体 b , 则 b 被选入下一

世代 经过若干世代后, 扩展操作向上“爬山”找到个体 b . 设 b 经扩展操作后, 突变操作找到评价值更高的个体 c , 则 c 被选入下一世代, 扩展操作最终找到全局最优解 c .

由上可知, 突变操作将整个解空间看作一个“山峰”, 通过大邻域搜索进行“爬山”以寻求评价值较高的区域 (a, b, c) . 突变操作用于搜索全局解空间, 是粗搜索. 如果突变操作找到评价值较高的区域, 扩展操作通过小邻域搜索在该区域进行局部细搜索以寻求高精度的解 (a, b, c) , 扩展操作用于搜索局部解空间. A IEA 具有从全局到局部的两层邻域搜索机制, 从而保证了其全局和局部寻优性能.

4.2 A IEA 的收敛性分析

利用文献[10]中的概率分析方法, 可证 A IEA 依概率收敛于全局最优解.

对于多维欧几里德空间中连续参数优化问题

$$\max f(x) \quad x \in S \subseteq R^n \quad (10)$$

其中: f 为评价函数, $S = \prod_{i=1}^n [a_i, b_i]$ 为搜索空间, $a_i < b_i, i = 1, 2, \dots, n$. 不失一般性, 假设搜索空间为 $S = I^n, I = [0, 1]$. 设 B_n 为 S 上的波莱尔域, 对于 B_n 中的集合, 其 Lebesgues 测度 $m(\bullet)$ 满足 $m(I^n) = 1$, 故 $P = (I^n, B_n, m)$ 构成一个概率空间. 假设 f 满足: 1) 全局最大值 $f^* = \max_x f(x)$ 存在; 2) $\forall \epsilon > 0, M_\epsilon = \{x \in S \mid |f(x) - f^*| \leq \epsilon\}$ 满足 $m(M_\epsilon) > 0$. 条件 1) 是理所当然的. 对于不满足条件 2) 的评价函数, 用任何优化方法求解都将是困难的. 因此, 对 f 的假设条件是合理的^[10].

定义 1 设第 k 世代群体中最优个体的评价值为 f_k , 若非负随机序列 $d_k = f^* - f_k$ 依概率收敛于 0, 则称 A IEA 收敛于全局最大值 f^* .

证明 设群体 D_k 中评价值最低的 L 个个体进行替换操作, 其中任一个体 x 被替换为 I^n 中的随机向量 x' . x' 的各分量是独立同分布的随机变量, 服从均匀分布 $U(0, 1)$, 因此其密度函数为

$$p_{x'}(t) = \begin{cases} 1, & t \in I^n; \\ 0, & \text{else} \end{cases} \quad (11)$$

对于 $\forall \epsilon > 0$, 有 $M_\epsilon \subseteq I^n$, 故

$$P(x \in M_\epsilon) = \int_{M_\epsilon} p_{x'}(t) dt = m(M_\epsilon) > 0 \quad (12)$$

成立. 令 $m(M_\epsilon) = \delta$, 则个体 x 经替换操作后进入 M_ϵ 的概率为 δ . 因此, D_k 中评价值最低的 L 个个体经替

换操作未产生进入 M_ϵ 的个体的概率为 $P_{\text{not}} = (1 - \delta)^L$. $1 - \delta$ 在前 k 代中共进行 k 次替换操作, 因此在前 k 代中没有经替换操作产生进入 M_ϵ 的个体的概率为

$$P_{\text{not}}(k) = \prod_{i=1}^k P_{\text{not}} = (1 - \delta)^k \quad (13)$$

由于在前 k 世代中可能存在通过扩展或突变操作而进入 M_ϵ 但没有被替换的个体, 并被最优个体保留策略所保留, 故

$$P(d_k > \epsilon) \leq P_{\text{not}}(k) \quad (14)$$

由于 $\delta > 0$, 由式(13) 有

$$\lim_{k \rightarrow \infty} P_{\text{not}}(k) = \lim_{k \rightarrow \infty} (1 - \delta)^k = 0 \quad (15)$$

由式(14) 和(15) 有

$$\lim_{k \rightarrow \infty} P(d_k > \epsilon) = \lim_{k \rightarrow \infty} P_{\text{not}}(k) = 0 \quad (16)$$

故 A IEA 收敛.

5 仿真实例

用 A IEA 优化以下 3 个复杂多峰函数, 并与标准遗传算法(SGA) 和自适应遗传算法(AGA)^[11] 的优化结果比较, 以验证其性能.

1) De Jong's F_5 函数:

$$F_5 = 0.002 + \frac{1}{\sum_{i=1}^{25} |j + (x_i - a_{ij})^6|} \quad (17)$$

其中 $|x_i| \in [65, 56]$, 全局最大值约为 1.002.

2) Schaffer's F_6 函数:

$$F_6 = 0.5 + \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1.0 + 0.001(x_1^2 + x_2^2)]^2} \quad (18)$$

其中: $-100 \leq x_1, x_2 \leq 100$, 全局最小值为 0.

3) Schaffer's F_7 函数:

$$F_7 = (x_1^2 + x_2^2)^{0.25} [\sin^2(50(x_1^2 + x_2^2)^{0.1}) + 1.0] \quad (19)$$

其中: $-10 \leq x_1, x_2 \leq 10$, 全局最小值为 0.

A IEA 的群体规模 $n = 100$, 算法参数见表 1.

表 1 A IEA 的参数

函数	α_0	η_l	r_0	η	R_0	η_k	d_{\max}	L
F_5	0.1	0.1	1	2	20	20	40	15
F_6	0.05	0.25	0.4	0.4	2	3	40	50
F_7	0.1	0.2	3e-4	4e-4	0.05	0.1	5	50

A IEA 取不同随机初始群体对每个函数进行 30 次优化计算. 在给定最大世代内, 若群体中最优个体的函数值达到给定阈值, 则认为算法收敛, 否则认为算法陷入局部最优解. 表 2 给出了 A IEA 对每个函数的 30 次优化计算中陷入局部最优次数, 算法收敛

表2 A IEA 与SGA 和AGA 的比较

函数	最大世代	陷入局部最优次数			平均世代数			平均评价次数			阈值
		SGA	GA	A IEA	SGA	AGA	A IEA	SGA	AGA	A IEA	
F_5	100	7	0	0	64.06	36.63	14.53	6406	3663	2937	1.000
F_6	200	23	6	0	173.90	106.56	35.43	17390	10656	8452	0.001
F_7	500	21	5	0	419.90	220.61	40.67	41990	22061	9581	0.005

所需平均世代数和平均函数评价次数。文献[11]用SGA 和AGA 优化以上3个函数的结果也列于表2中,以便进行比较。A IEA 在所有优化计算中均未陷入局部最优,而且收敛所需世代数和函数评价次数较少。这表明A IEA 具有更强的全局优化能力,而且计算量更小,即计算时间更少。

表3给出了用A IEA 优化每个函数得到的30个结果的平均函数值、平均误差和标准差。平均误差和标准差都很小,这表明A IEA 具有良好的局部搜索能力和稳定性。文献[11]未考虑SGA 和AGA 的局部搜索能力和稳定性,但由表2中二者陷入局部最优次数可知,A IEA 具有更好的稳定性。因为遗传算法的局部搜索能力弱于基于邻域的局部搜索算法,而A IEA 用邻域搜索来搜索局部解空间,因此具有更强的局部搜索能力。仿真中发现,当初始群体多样性很小时,A IEA 仍有很好的优化性能,而SGA 和AGA 在该情况下很容易早熟收敛。

表3 A IEA 的优化结果

函数	全局最优	平均函数值	平均误差	标准差
F_5	1.002	1.002	0	0
F_6	0	5.252e-7	5.252e-7	5.660e-7
F_7	0	7.620e-4	7.620e-4	1.229e-4

6 结 论

本文基于适应性免疫应答中的进化机理,提出一种自适应免疫进化算法。该算法通过对多个可行解进行大邻域和小邻域搜索实现了由全局到局部的两层邻域搜索机制,从而既保证了全局寻优和所求解的精度,又有较高的搜索效率。同时根据群体的多样性自适应调节算法参数,以进一步提高算法的全局寻优能力和收敛速度。仿真结果表明,该算法具有不易陷入局部最优、解的精度高、收敛速度快、稳定性好等优点。

参考文献(References):

- [1] 王凌. 智能优化算法及其应用[M]. 北京: 清华大学出版社, 2001.
- [2] Goldberg D E, Voessner S. Optimizing global-local search hybrids[A]. *Proc of Genetic and Evolutionary*

Computation Conf [C]. San Francisco, CA: Morgan Kaufmann, 1999. 220-228.

- [3] 丁永生, 任立红. 人工免疫系统: 理论与应用[J]. 模式识别与人工智能, 2000, 13(1): 52-59.
(Ding Y S, Ren L H. Artificial immune systems: Theory and applications [J]. *Pattern Recognition and Artificial Intelligence*, 2000, 13(1): 52-59.)
- [4] 左兴权, 李士勇. 人工免疫系统研究的新进展[J]. 计算机测量与控制, 2002, 10(11): 701-705.
(Zuo X Q, Li S Y. The new progresses in artificial immune system [J]. *Computer Measurement & Control*, 2002, 10(11): 701-705.)
- [5] 曹先彬, 刘克胜, 王煦法. 基于免疫进化规划的多层前馈网络设计[J]. 软件学报, 1999, 10(11): 1180-1184.
(Cao X B, Liu K S, Wang X F. Design multilayer feed-forward networks based on immune evolutionary programming [J]. *J of Software*, 1999, 10(11): 1180-1184.)
- [6] Jiao L C, Wang L. A novel genetic algorithm based on immunity [J]. *IEEE Trans on System, Man, and Cybernetics, Part A: Systems and Humans*, 2000, 30(5): 552-561.
- [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] Ada G L, Nossal G. The clonal selection theory [J]. *Scientific American*, 1987, 257(2): 50-57.
- [9] 李士勇. 模糊控制·神经控制和智能控制论[M]. 哈尔滨: 哈尔滨工业大学出版社, 1998. 74-127.
- [10] 林丹, 李敏强, 寇纪淞. 基于实数编码的遗传算法的收敛性研究[J]. 计算机研究与发展, 2000, 37(11): 1321-1327.
(Lin D, Li M Q, Kou J S. On the convergence of real-coded genetic algorithm [J]. *J of Computer Research & Development*, 2000, 37(11): 1321-1327.)
- [11] Srinivas M, Patnaik L M. Adaptive probabilities of crossover and mutation in genetic algorithm [J]. *IEEE Trans on Systems, Man, and Cybernetics*, 1994, 24(4): 656-667.