

# 基于免疫识别的免疫算法

张四海,曹先彬,王煦法

(中国科学技术大学计算机科学技术系,安徽合肥 230026)

**摘要:** 免疫算法已开始成为计算智能研究的一个热点.本文提出了一种基于免疫识别的免疫算法,该算法的设计借鉴了生物免疫识别中的非选择机制,其求解思想与现有优化方法不同,它是通过在解空间中不断识别、选择出劣解并加以排除来获得优解.本文详细介绍了该算法的设计,从理论上分析了算法的收敛性能和时间复杂度,并针对 TSP 问题设计了相应的验证实验.理论分析和实验结果都表明该算法具有很强的问题求解能力.

**关键词:** 免疫算法; 免疫识别; 非选择; TSP 问题

**中图分类号:** TP18 **文献标识码:** A **文章编号:** 0372-2112 (2002) 12-1840-05

## Immune Algorithm Based on Immune Recognition

ZHANG Si-hai, CAO Xian-bin, WANG Xu-fa

(Department of Computer Science and Technology, University of Science and Technology of China, Hefei, Anhui 230026, China)

**Abstract:** Immune algorithm has become a hotspot in the area of computational intelligence. We proposed an immune algorithm based on immune recognition (IAIR). The design of IAIR was inspired by the negative selection mechanism in the biological immune system and its idea was different with current optimization methods in that it got the good solution by excluding bad solutions. This paper introduced in detail the design of IAIR, analyzed its ability and time complexity. We also designed experiments to solve the TSP problem. The theoretical analysis and experimental results showed that IAIR had a good ability of problem solving.

**Key words:** immune algorithm; immune recognition; negative selection; TSP

### 1 引言

近年来,人们意识到生物免疫机制对开发新的计算智能的启示意义,基于人工免疫的计算模型及应用已开始成为当前的一个研究前沿<sup>[1]</sup>.典型工作如:S Forrest 模拟免疫系统的自我非我识别机制,提出一个简单的非选择模型用于反病毒研究<sup>[2]</sup>;Watanabe 利用免疫网络提出机器人的仲裁机制<sup>[3]</sup>;Fukuda 借鉴免疫学习中较好的多样性来克服遗传算法(GA: Genetic Algorithm)易于陷入局部极优的缺点<sup>[4]</sup>.在优化领域,目前的相关工作还集中在以 GA 为基础,通过增加体现免疫特点的操作算子来提高算法的性能,一般称为免疫遗传算法(IGA: Immune GA),或简称为免疫算法(IA: Immune Algorithm).如焦李成等提出的基于免疫接种的 IA<sup>[5]</sup>,作者等也提出了基于浓度调节的 IGA<sup>[6]</sup>.

人工神经网络、GA 等优化方法的求解思想本质上都是“好上加好”,即不断向优解爬山来得到最终解,其局部和全局搜索能力在很多情况下都不能令人满意.在 GA 中增加免疫算子的作用也是有限的.S Forrest 提出的非选择(Negative Selection, NS)模型实质上是一种与“好上加好”完全不同的求解思路.在本文,我们借鉴生物免疫识别机制,提出了一种基于 NS 模型的免疫算法:基于免疫识别的免疫算法(IAIR:

Immune Algorithm Based on Immune Recognition). IAIR 将优化问题的解空间对应为全空间,而非优解空间对应为自我空间,其补集则对应为非我空间;在求解过程中,不断地用自我空间去检测非我空间,减小非我空间,最终导致在非我空间中都是比较好的解.这样我们通过不断地排除劣解来得到优解.

本文后面的内容安排如下:第 2 节介绍 IAIR 的设计;第 3 节对其收敛性能进行分析;第 4 节说明我们设计的实验及结果分析;第 5 节是和其它几种典型优化算法的对比;第 6 节是结束语.

### 2 算法设计

#### 2.1 算法的生物免疫基础

生物免疫系统的基本功能是区分自我和非我,免疫识别是生物免疫系统各种功能的基础.根据 N K Burnet 的细胞克隆选择学说,生物免疫系统能够识别和抵抗各种抗原的袭击,是因为在胚胎期由于免疫细胞在增殖中发生基因突变,形成了免疫细胞的多样性<sup>[7]</sup>.在抗体形成过程中,免疫系统采用了一系列的学习和记忆机制:通过基因重组得到未成熟免疫细胞;通过 NS 机制在未成熟免疫细胞集合中清除自我成分;对组成有效抗体的基因提高其在基因重组时被选择的概率,

收稿日期:2001-10-28;修回日期:2002-04-22

基金项目:国家自然科学基金(No.69971022);国家博士点基金;安徽省自然科学基金项目(No.00043106)

达到增殖的目的.这些机制使得生物免疫系统表现出很强的识别能力.

借鉴上述机制,着重利用 NS 思想,我们提出了和现有的优化求解策略不同的 IAIR,将待解决的问题看作抗原,问题的解看作抗体.在不断通过基因重组生成未成熟抗体后,利用 NS 机制排除这些抗体中属于自我的部分得到成熟抗体;如果成熟抗体能很好地识别该抗原,我们就通过提高组成该抗体的基因的浓度来奖励该抗体;反之则将对应的基因的浓度降低,甚至将该基因加入自我来完全消除该基因,达到惩罚的目的.

### 2.2 算法设计

图 1 是 IAIR 的基本流程.和 GA 类似,IAIR 也不能直接处理解空间的解数据,必须通过编码将它们表示成基因型串结构数据.一般采用 2 进制编码.我们把待求解问题的组成要素作为基因段,把基因段所有可能的取值初始化到基因库中.因此,基因库的所有组合结果就是待求解问题的全部解空间. IAIR 在求解过程中一般不需要其他外部信息,仅用评估函数值来评估抗体或解的优劣,并作为以后免疫操作的依据.基因重组,非选择和基因库调整是 IAIR 的 3 个主要免疫算子.

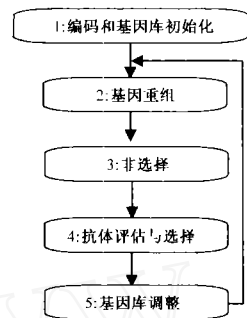


图 1 IAIR 的基本流程图

假设问题规模为  $N$ ,于是基因段的个数也为  $N$ .又假设每个基因段所有的可能取值数目为  $M$ ,则其具体操作过程如下:

#### (1) 基因重组

从基因库中的各个基因段随机选择相应的等位基因,构成一个合法的抗体(问题的解).时间复杂度为  $O(N \cdot M)$ .这借鉴了生物免疫机制中抗体的生成机制,目的是为了提高算法的全局搜索性能.

#### (2) 非选择

基因重组得到的新抗体,可能是较优解,也可能是较差解. NS 就是用自我来排除较差解,从而提高搜索速度,提高算法的时间性能.与基因库不同,自我库不是初始生成的,而是在求解过程中动态生成的.它是 NS 的基础,并可以有效缩小搜索空间.记自我规模为  $SL$ ,则  $SL \leq N \cdot M$ ,时间复杂度为  $O(SL)$ ,即  $O(N \cdot M)$ .

#### (3) 基因库调整

基因库调整是利用成熟抗体来修正基因库中相应基因的浓度,提高较好抗体中的基因浓度,降低较差抗体中的基因浓度,从而期待下次基因重组能得到更好的抗体.易知其时间复杂度也为  $O(N \cdot M)$ .

### 3 算法性能分析

统计确定理论中的双角子机问题表明:要获得最优的可行解,则必须保证较优解的样本数呈指数级增长<sup>[8]</sup>.下面考虑

IAIR 的情况.

令串长为  $l$ ,不失一般性,考虑定位为两位的模式  $P: * \dots * 0 * \dots * 1 * \dots *$ ,其中 0 出现的位置为  $i$ ,1 出现的位置为  $j(1 \leq i, j \leq l, i \neq j)$ ,假设在第  $n$  代,基因浓度矩阵为  $C_{ij}(n)$ ,则模式  $P$  出现的概率为:

$$m(P, n) = \frac{c_{i1}(n)}{c_{i1}(n) + c_{i2}(n)} \frac{c_{j2}(n)}{c_{j1}(n) + c_{j2}(n)}$$

若  $P$  的样本实际出现,且该样本评价值  $f_p <$  抗体的历史平均评价值.则在基因库调整时,基因浓度修改如下( $\rho$  为衰减系数):

$$c_{i1}(n+1) = \rho * c_{i1}(n) + \Delta c(f_p);$$

$$c_{j2}(n+1) = \rho * c_{j2}(n) + \Delta c(f_p); (\Delta c(f_p) > 0)$$

则当下一次重组时,  $P$  出现的概率为:

$$m(P, n+1) = \frac{c_{i1}(n+1)}{c_{i1}(n+1) + c_{i2}(n+1)} \frac{c_{j2}(n+1)}{c_{j1}(n+1) + c_{j2}(n+1)}$$

$$= \frac{c_{i1}(n) + \Delta c(f_p)}{c_{i2}(n+1) + c_{i1}(n+1) + \Delta c(f_p)}$$

$$\cdot \frac{c_{j2}(n) + \Delta c(f_p)}{c_{j1}(n+1) + c_{j2}(n+1) + \Delta c(f_p)} \quad (\text{if } \rho = 1)$$

$$= m(P, n)(1 + C)(C > 0, \text{ if } c_{i2}(n+1) = c_{i2}(n), c_{j1}(n+1) = c_{j1}(n))$$

所以从概率意义上,如果模式评价值大于抗体的历史平均评价值,该模式在子代中得以指数级增长.此结论可以推广到定位为多位的情况,于是得到定理 1:

**定理 1** 在调整算子的作用下,如果衰减系数为 1,那么平均评价值高于抗体历史平均评价值的模式将得以指数级增长.

IAIR 可被描述为一个马尔可夫链;状态空间  $S = IB^l = 2^l$ ,其中  $l$  表示串个体的长度,状态空间的元素是一个二进制表示的整数.空间中的状态由于免疫算子引起的变化由矩阵  $P$  描述,而  $P$  又可以很自然地分解为随机矩阵  $R, N$  和  $A$  的乘积,即有  $P = RFA$ .其中,  $R, N$  和  $A$  分别描述重组算子, NS 算子和调整算子引起的变化.

**定理 2** 采用重组算子(比例选择法), NS 算子和调整算子的 IAIR,其概率转移矩阵  $P$  是基本随机矩阵.

**证明** 重组算子是在状态空间  $S$  中的一个随机满射函数,即  $S$  的每个状态都随机地映射到另一状态.因此  $R$  是随机矩阵.又易知,在重组算子的作用下,从某状态  $i$  到另一状态  $j$  的概率  $P_{ij} > 0$ ,因此  $R$  是定正的,所以  $R$  是基本的.

NS 算子的作用仅仅是直接排除差解,提高时间性能,对  $R$  不产生影响.

调整算子的作用是提高某些基因被选择的概率,并不影响  $R$  是基本随机矩阵的性质,所以算法的概率转移矩阵是基本随机矩阵.证毕.

**推论 1** 具有如定理 2 所示的标准免疫算法是一个遍历马尔可夫链,即对于任意初始分布,在任意时刻,任意状态的概率非零的链存在唯一的极限分布(证明略).

类似遗传算法的相关证明<sup>[8]</sup>,可以得到定理 3:

**定理 3** 具有定理 1 所示参数及在三种免疫算子的作用下, IAIR 不能收敛到全局最优解.

**证明** 设  $i$  是状态空间中的一个状态, 且  $i < i^*$ ,  $P_i^t$  是时间步  $t$  IAIR 进入状态  $i$  的概率. 显然,  $P\{i \neq i^*\} \geq P_i^t \Leftrightarrow P\{i = i^*\} \leq 1 - P_i^t$ . 由定理可知, IAIR 在状态  $i$  的概率收敛至  $P_i^\infty > 0$ . 则有:  $\lim_{t \rightarrow \infty} P\{i = i^*\} \leq 1 - P_i^\infty < 1$ . 因此, 标准 IAIR 是不收敛的. 证毕.

下面对标准 IAIR 作一些改进, 即不按比例进行选择, 而是保留当前所得的最佳值 (称作超个体). 对马尔可夫链的描述也作一定的扩展, 即该超个体不参加免疫算子的作用. 状态空间  $S$  的基从  $2^l$  扩展到  $2^{2l}$ . 为了描述方便, 将该超个体置于最左的位置, 并能用  $\pi_0(i)$  从状态  $i$  加以访问. 由于最佳个体不受免疫算子作用, 扩展的  $P^+$  可被写成块对角矩阵, 即  $P^+$

$$= \begin{pmatrix} P & & & \\ & P & & \\ & & \ddots & \\ & & & P \end{pmatrix}, \text{其中 } P \text{ 为 } 2^l \text{ 方阵, } P^+ \text{ 为 } 2^{2l} \text{ 方阵.}$$

调整操作由“改良”矩阵  $U$  表示, 其作用就是将一个包含优于当前超个体的个体的中间状态变换至超个体等于该个体的状态. 特别地, 设  $b$  表示除超个体之外任意状态之中的最佳个体. 如果  $f(\pi_0(i)) < f(b)$ , 则  $U_{ij} = 1$ , 其中  $j^{def}(b, \pi_1(i)) \in S$ ; 否则  $U_{ij} = 0$ . 这样, 每一行中恰好只有一个元素. 换句话说, 一个状态要么被改进, 要么保持不变. 因此, 改良矩阵的结构可写成:

$$U = \begin{pmatrix} U_{11} & & & \\ U_{21} & U_{22} & & \\ \vdots & \vdots & \ddots & \\ U_{2^l 1} & U_{2^l 2} & \cdots & U_{2^l 2^l} \end{pmatrix}, \text{其中 } U_{a,b} \text{ 大小为 } 2^l \times 2^l.$$

为了简单起见, 假设只有一个全局最优解. 则只有  $U_{11}$  为单位矩阵, 而其他所有  $U_{aa} (a \geq 2)$  是单位矩阵且具有对角零值. 则有:

$$P^+ = \begin{pmatrix} P & & & \\ & P & & \\ & & \ddots & \\ & & & P \end{pmatrix} \begin{pmatrix} U_{11} & & & \\ U_{21} & U_{22} & & \\ \vdots & \vdots & \ddots & \\ U_{2^l 1} & U_{2^l 2} & \cdots & U_{2^l 2^l} \end{pmatrix} = \begin{pmatrix} PU_{11} & & & \\ PU_{21} & PU_{22} & & \\ \vdots & \vdots & \ddots & \\ PU_{2^l 1} & PU_{2^l 2} & \cdots & PU_{2^l 2^l} \end{pmatrix}$$

其中  $PU_{11} = P > 0$ . 子矩阵  $PU_{a1} (a \geq 2)$  可以集中在一起以长

方矩阵  $R \neq 0$  表示, 则有如下定理:

**定理 4** 具有定理 1 所示参数及在三种免疫算子的作用下, 如果采取最佳保留策略, 则 IAIR 最终能收敛到全局最优解.

**证明** 子矩阵  $PU_{11} = P > 0$  中集中了包含全局最佳个体的状态 (称为全局最优状态) 的变换概率. 因为  $P$  是一个基本随机矩阵且  $R \neq 0$ , 则所有包含在非全局最优状态中的概率收敛于 0. 则所有包含在全局最优状态中的概率收敛于 1, 因此有

$$\lim_{t \rightarrow \infty} P\{i = i^*\} = 1$$

即算法收敛于全局最优解. 证毕.

## 4 实验设计及结果分析

### 4.1 实验设计

以 TSP 为例设计了相应的验证实验. 首先引进如下记号:

$N$  为问题规模;  $M$  为抗体规模; 城市集合  $V = \{a_i | a_i = (x_i, y_i), 1 \leq i \leq N, x_i, y_i \in R\}$ ;  $E = \{e_{ij} | e_{ij} = (a_i, a_j), 1 \leq i \leq N, i \neq j\}$  为城市间边集合; 自我集合 SELF 初始为  $\{\}$ , 规模为  $S$ ;

$$D = \{d_{ij} | d_{ij} = \text{dist}(a_i, a_j), 1 \leq i \leq N, i \neq j\}$$

为城市间距离集合; 浓度衰减系数为  $\rho$ ; 自我阈值  $\alpha$ ; 非自我阈值  $\beta$ ; 城市间距离的浓度集合  $C = \{c_{ij}, 1 \leq i, j \leq N, i \neq j\}$ .

$E$  作为基因集合, 通过将  $E$  中各基因重组, 得到抗体 (路径). 生成抗体的过程是: 从起点  $a_1$  (可随机得到) 出发, 按照一定的概率选择出下一个城市, 直至得到一条合法抗体. 基因重组算子的动力学方程为:

$$P_i = (c_{ij}/d_{ij}) / \sum_{j=1}^N c_{ij}/d_{ij}$$

其中  $a_j$  为可取点,  $a_i$  为当前城市; (1)

NS 算子的排除策略为: 如果该抗体中包含  $\beta$  个以上自我集合中的基因, 则认为该抗体已经被识别, 则排除该抗体. (本文只考虑  $\beta = 1$  的情况)

评价函数  $f$  定义如下:

$$f = 1/L, L \text{ 表示路径的长度} \quad (2)$$

如果  $f > \bar{f}$  (历史平均值), 则修改  $\bar{f}$ , 并按照公式 (3) 修改抗原中基因的浓度:

$$c_{ij}(n+1) = \rho * c_{ij}(n) + \sum_{k=1}^M P/L_k; \\ P = 1, \text{ if } e_{ij} \text{ 在第 } k \text{ 条路径中, else } P = 0 \quad (3)$$

对于  $f < \bar{f}$  的抗体, 统计在其中出现的所有的基因  $e_{ij}$ , 并按照公式 (4) 计算这些基因的自我度量  $s_{ij}$ :

$$s_{ij} = d_{ij} / \sum_{j=1}^N d_{ij} \quad (4)$$

如果  $s_{ij} > \alpha$ , 那么将该边添加到自我集合 SELF 中. 如果求解过程中出现的非自我数目超过自我集合规模  $S$ , 那么将基因长度最短的自我替换为当前自我.

选择文献 [9] 中给出的标准 TSP 数据 Oliver30 来验证 IAIR 的求解性能, 目的是: (1) 考察与分析关键参数对算法性能的影响; (2) 验证算法的求解能力.

### 4.2 关键参数对算法性能的影响

对算法性能有影响的关键参数有:

- $\alpha$ : 自我阈值,  $\alpha > 0$ ;
- $S$ : 自我集合规模,  $0 \leq S \leq n$ ;
- $\rho$ : 衰减系数,  $0 < \rho \leq 1$ ;

表 1 是针对自我阈值  $\alpha$  的实验结果(重复 10 次实验的结果).

表 1 自我阈值对算法的影响 ( $\beta=1, S=15$ )

$\rho$	$\alpha$	平均结果	平均代数	最好结果	最好代数
0.9	0.0004	424.270	120.5	423.741	114
	0.0008	424.572	100.3	423.741	101
	0.0015	424.648	75.8	423.741	56
	0.0025	425.525	93.6	423.741	141
1.0	0.0004	424.270	120.5	423.741	114
	0.0008	424.572	100.3	423.741	101
	0.0015	424.648	75.8	423.741	56
	0.0025	425.525	93.6	423.741	141

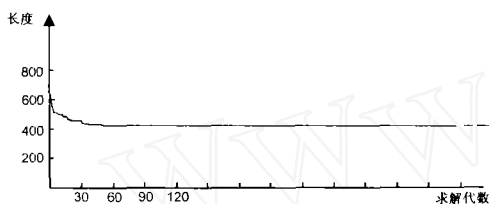


图 2 最优解的求解曲线(Oliver30),最终收敛到 423.741

可以看到,最好的实验结果都收敛到了 423.741,这也是文献[10]中得到的最优解.图 2 是 IAIR 的求解曲线,图 3 是求得的最优解.本算法的收敛性很好,几乎每次都能得到最优解.另外,当  $\rho$  为 0.9 时,算法收敛速度较快,在 100 代左右就得到最优解,这是算法的一大优点,在实际应用中有着很大优势.

由公式 4 可知,自我度量值是基因长度与所有等位基因长度之和的比值.显然,某基因的长度越长,则该基因成为自我的概率就越大.表 1 中的结果证明: $\alpha$  对实验结果具有重大影响.虽然在每种情况下,算法都收敛到最优解,但是随着  $\alpha$  的降低,解的平均质量提高.这是因为  $\alpha$  较大时,自我数目就会降低,从而不能有效地减小优解空间,导致解的质量相应下降.

表 2 收敛速度实验结果( $\rho=0.9, \alpha=0.0015$ , 重复 10 次实验的结果)

$S$	平均结果	平均代数	最好结果	最好代数
6	424.414	133.6	423.912	304
15	424.648	75.8	423.741	56
18	425.074	44	424.692	24

表 2 是参数  $S$  取不同值时的实验结果.可以看出,当  $S$  较小时,收敛速度较慢,但解的质量较高;当  $S$  较大时,收敛速度较快,但是有可能得不到优解.这是因为:当排除了较多的非我后,搜索空间减小,因此算法的时间复杂度降低,收敛较快,但有可能排除了较好的基因,导致不能得到最优解.  $S$  不能过大,否则由于非我集过小,导致获得有效解的能力下降.在实验中我们发现,当  $S > 20$  时,求解效果显著下降.图 4

是三种  $S$  下的求解曲线图.

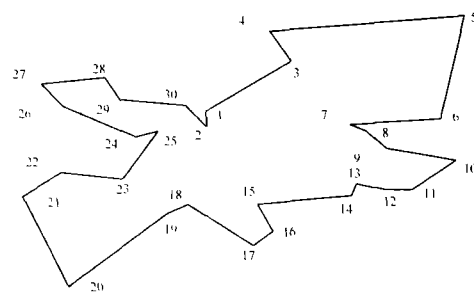


图 3 得到的最佳路径(Oliver30),长度为 423.741

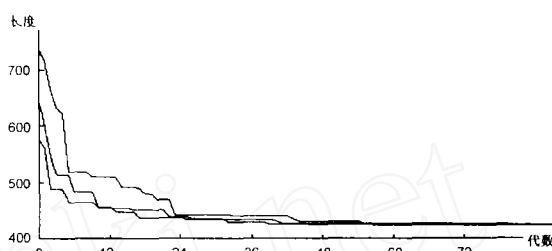


图 4 自我数量对比求解曲线图(从上到下分别为  $S=6, S=15, S=18$ )

对参数  $\rho$ ,第 3 节中的理论分析已经证明:当  $\rho=1$  时,平均评价价值高于抗体平均评价价值的模式得以指数级增长,由表 1 可知,虽然每次的最优解相同,但  $\rho=1$  的平均结果比  $\rho=0.9$  的好.另外,当  $\rho=1$  时,求解时间远远高于  $\rho=0.9$  的.这是因为在有关衰减的情况下,那些未被选择的、较差的基因浓度迅速降低,使得较好基因被选择的概率加大,因此求解速度加快.但如果较好基因未被选择,则该基因的浓度将自然衰减,导致下次被选择的概率更小.这就导致在有关衰减的情况下,平均结果比无衰减时要差.

### 4.3 多样性分析

这里定义了一个简单的多样性评价函数,来表征抗体多样性.定义如下:

$$DIV(n) = \frac{\sum_{k=1}^{M(n)} (L_k(n) - avg(L(n)))^2}{M(n)} \quad (5)$$

其中

$$avg(L(n)) = \frac{\sum_{k=1}^{M(n)} L_k(n)}{M(n)}, M(n) \text{ 为第 } n \text{ 代时的抗体数目} \quad (6)$$

图 5 是求解过程中的多样性曲线.到 600 代时,多样性依然很好;随着求解的进行,总体上多样性降低(这是合理的),但依然保持较高的多样性,因此具有不断获得新解的能力,而不易陷入局部极优.

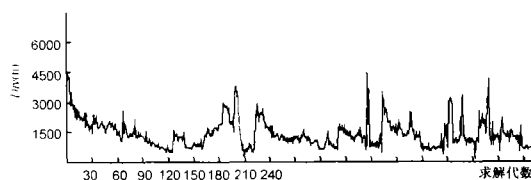


图 5 抗体多样性曲线( $\alpha=0.008, \beta=1, S=15, \rho=0.9$ )

表 3 IAIR 和 GA、AS 的实验对比 (\* : 文献[10]未提供)

优化算法	最好结果	最好代数	平均结果	平均代数
IAIR	423.741	56	424.270	120.5
GA	423.741	553	425.680	850
AS	423.741	342	424.250	*

## 5 和其他算法的对比

本节,将 IAIR 和 GA 以及蚂蚁系统(AS)作对比分析.一般认为 GA 的求解时间较长<sup>[8]</sup>,AS 的时间复杂度为  $O(NC \cdot N^3)$ <sup>[10]</sup>.由 2.2 节可知 IAIR 的时间复杂度为  $O(NC \cdot M \cdot N)$ ,比上述算法要好.另外,由于 NS 算子能排除一部分差基因,因此,实际的时间性能要好于  $O(NC \cdot M \cdot N)$ .我们认为,NS 机制起了决定性的作用. IAIR 和现有的优化算法思想的不同在于,利用 NS 机制将搜索空间大大缩小,这是 IAIR 的时间性能较好的根本原因.

表 3 是 IAIR、GA 和 AS 求解 Oliver30 的实验对比.在解的质量和求解时间上,IAIR 都要好于 GA;解的质量和 AS 相当,但是 IAIR 的收敛速度比 AS 要快许多.

## 6 结束语

借鉴生物免疫机制,本文提出了一种基于免疫识别的免疫算法,并进行了理论分析和实验验证.还与其他典型优化算法进行了对比,结果证明具有很好的效果,且具有很好的时间复杂度,对于解决实际中的大规模问题尤其适合.

### 参考文献:

- [1] D Dasgupta. An overview of artificial immune systems and their applications[A]. Artificial Immune Systems and Their Applications [C]. Springer, 1998. 3-21.
- [2] S Forrest, et al. Self-nonsel self discrimination in a computer [A]. Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy [C]. Los Alamitos, CA: IEEE Computer Society Press, 1994.
- [3] Watanabe, Ishiguro, Uchikawa. Decentralized behavior arbitration mechanism for autonomous mobile robot using immune network [A].

Artificial Immune Systems and Their Applications [C]. Springer, 1998. 3-21.

- [4] Fukuda, Mori, Tsukiyama. Parallel search for multi-modal function optimization with diversity and learning of immune algorithm [A]. Artificial Immune Systems and Their Applications [C]. Springer, 1998. 3-21.
- [5] 王磊,潘进,焦李成.免疫算法[J].电子学报,2000,28(7):74-78.
- [6] 曹先彬,罗文坚,王煦法.基于免疫网络调节的改进遗传算法[J].高技术通讯,2000,(10):23-27.
- [7] F M Burnet. The Clonal Selection Theory of Acquired Immunity [M]. Vanderbilt University Press, 1959.
- [8] 陈国良,王煦法等.遗传算法及其应用 [M]. 人民邮电出版社, 1996.
- [9] D Whitley, T Starkweather, D Fuquay. Scheduling problems and traveling salesman: The genetic edge recombination operator [A]. in Proc. Third Int. Conf. On Generic Algorithms [C]. San Mateo, CA: Morgan Kaufmann, 1989.
- [10] Dorigo M, Maniezzo V, Colnani A. Ant system: Optimization by a colony of cooperating agents [J]. IEEE Trans on SMC, 1996. 26(1):28-41.

### 作者简介:



张四海 男,1974年2月生于安徽省淮南市,1996年毕业于青岛海洋大学计算机应用专业,2002年获中国科技大学计算机应用专业硕士学位,现为中国科技大学博士生.主要研究领域为智能计算,人工免疫,网络安全.



曹先彬 男,1969年1月生于安徽省巢湖市,1996年获中国科技大学智能信息处理专业博士学位,现为中国科技大学计算机系副教授,主要研究方向为信息安全、计算智能,已在国内外重要学术刊物和学术会议上发表论文近40篇,合作出版专著一部,获省部级科技进步二、三等奖两项.