基于人工免疫响应的线性系统逼近*

公茂果[1] 杜海峰[1,2] 焦李成[1]

(1. 西安电子科技大学智能信息处理研究所雷达信号处理国家重点实验室, 西安 710071;
2. 西安交通大学机械工程学院, 西安 710049)

摘要 提出一种基于人工免疫响应的线性系统逼近算法. 给出了人工免疫响应的四元组模型, 为免疫响应过程建立了一个可用于工程计算的数学模型; 设计了克隆选择、免疫记忆和免疫调节等具体操作, 模拟了抗体克隆选择、免疫记忆、基因免疫、免疫耐受等现象, 实现了人工免疫响应的记忆学习. 基于抗体群的随机状态转移过程, 证明了新算法具有全局收敛性. 基于两个典型的稳定或非稳定线性系统逼近问题的数值试验表明, 无论在固定的区间内搜索还是在动态扩展的区间内搜索, 人工免疫响应算法都能得到线性系统的最优逼近模型, 算法是有效的.

关键词 线性系统逼近 人工免疫系统 免疫响应 克隆选择 免疫记忆

线性系统逼近问题具有广泛的工程应用背景. 例如, 在模拟复杂动态系统或设计其控制器时, 重要的任务之一就是求解非线性系统的最优逼近模型[1]. 为了解决这类模型逼近问题, 国内外学者已经提出了很多种方法[2~6]. 其逼近模型由最小化特定性能评价标准而得到, 常用的如H^2-范数, L^2-范数, H^∞-范数和L^∞-范数等. 评价标准和最优化方法对这类方法的性能影响很大. 当采用H^2-范数和L^2-范数时, 常用基于梯度的优化方法, 但是这种方法往往只能求出局部最优解而不是全局最优解; 而当采用H^∞-范数和L^∞-范数时, 由于H^∞-范数和L^∞-范数均不是处处连续可导, 传统的优化方法存在更大的困难[7]. 因此, 需要寻找更好的优化方法.

人工免疫系统是受免疫学启发, 模拟免疫学功能、原理和模型来解决复杂问题的自适应系统[8], 已经成为人工智能领域理论和应用研究的热点[9,10]. 近几年, 国内外很多学者相继提出了自己的人工免疫系统算法或模型[11~14]. 免疫响应是指免疫细胞对抗原分子的识别、活化、分化和产生免疫效应的全过程. 其对相应

2005-07-05 收稿, 2005-08-29 收修改稿
* 国家“863”(2002AA135080), “973”(2001CB309403), 国家自然科学基金(批准号: 60133010, 60372045)和西安电子科技大学研究生创新基金资助项目

抗原的特异性识别及特异性记忆, 遵循再次响应规律[15]. 免疫系统具有识别各种抗原并将特定抗原排斥掉的学习记忆机制, 这是与神经网络和生物进化系统不同的记忆机制. 在人工免疫系统算法中体现免疫响应的再次响应过程, 将会成为人工免疫系统算法区别于进化算法和神经网络的重要亮点之一[14].

本文基于解释生物免疫响应过程的相关理论, 提出了用于线性系统最优逼近的人工免疫响应算法. 具体地, 第一部分阐述线性系统最优逼近问题, 描述免疫系统的免疫响应过程; 第二部分提出一种可用于工程计算的人工免疫系统模型——人工免疫响应的四元组模型, 给出克隆选择、免疫记忆和免疫调节的具体操作, 并设计针对线性系统逼近问题的人工免疫响应支配算法; 第三部分对算法的全局收敛性进行分析; 第四部分基于两个被广泛采用的线性系统逼近问题, 通过与微分进化算法[2]、多智能体遗传算法[4]等其他线性系统逼近算法的数值实验比较, 来验证算法的优越性; 最后对全文进行总结.

1　问题阐述及免疫学背景

1.1　线性系统逼近问题的数学模型

在控制领域, 一个复杂的动态系统完全可以由一个具有时延的低阶传递函数表达. 因此, 给定一个高阶有理或者无理的传递函数$G(s)$, 可以找到形如(1)式的逼近模型, 以使得$H_m(s)$具有$G(s)$的特性:

$$H_m(s)=\frac{b_0+b_1s+\cdots+b_{m-1}s^{m-1}}{a_0+a_1s+\cdots+a_{m-1}s^{m-1}+s^m}\cdot e^{-\tau_d s}. \tag{1}$$

通过求解最优逼近模型$H_m(s)$, 使得下式中所示的频率域L^2-误差J最小化, 其中所用的频率点为 ω_i, i = 0, 1, 2, 3, ⋯, N, N为预先设定的参数.

$$J=\sum_{i=0}^{N}\left|G(j\omega_i)-H_m(j\omega_i)\right|^2. \tag{2}$$

当原始系统 $G(s)$是渐进稳定时, 令 $H_m(0)=G(0)$, 以保证原始系统与逼近模型对于单位阶跃输入具有相同的稳态响应.

1.2　免疫响应过程描述

免疫响应一般是由抗原引发、由多种免疫细胞参加的一系列反应, 首先是抗原递呈细胞把经过加工处理的抗原递呈给相应的T细胞(抗原特异性T细胞克隆), 活化以后的细胞通过细胞膜上的分子或分泌的细胞介素进一步活化其他免疫细胞, 产生相应的免疫效应, 并通过补体成分从细胞因子介导炎症反应, 促进对抗原和靶细胞的吞噬, 最终清除抗原, 维持机体内部的平衡和稳定[15].

抗体克隆选择是生物体免疫系统自适应抗原刺激的动态过程. 当免疫系统

遇到以前未遇到的抗原类型时, 它会产生初次响应, 用几个星期来消除感染; 在初次响应期间, 免疫系统学习识别以前未见到的外部模式, 抗体基因频繁的变异和编辑, 抗体反应在经过免疫后得到改善, 即亲和度成熟. 淋巴细胞除了扩增或者分化成浆细胞以外, 也能分化成生命期较长的 B 记忆细胞. 这些记忆细胞中的信息编码组成免疫系统记忆, 使它能学习、记忆蛋白质结构. 再次遇到相应的抗原后, 为清除已经刺激初次响应的相应的特异抗原, 记忆细胞将预先被免疫系统选择出来, 并迅速活化、增殖、分化为效应细胞, 执行高效而持久的免疫功能. 免疫响应过程中所体现出的学习、记忆、抗体多样性等, 正是人工免疫系统所借鉴的可用于解决工程问题的特性.

2 用于线性系统逼近的人工免疫响应算法设计

上述生物免疫响应过程可以简化为如下过程: 由抗原引发, 在免疫系统的控制下多种免疫细胞(包括抗体和 T 细胞)经过一系列反应, 逐步亲和度成熟, 产生相应的免疫效应, 清除抗原. 因此, 在本节中我们将生物免疫响应抽象为一个四元组, 并且针对线性系统逼近问题设计出相应的规则集合和动态支配算法.

2.1 人工免疫响应的四元组模型

人工免疫响应可以被描述为一个四元组 $\boldsymbol{AIR} = (\boldsymbol{G}, \boldsymbol{I}, \boldsymbol{R}, \boldsymbol{A})$, 其中 $\boldsymbol{G}$ 为引发免疫响应的外界刺激, 即抗原; $\boldsymbol{I}$ 为所有可能抗体(本文将抗体和 T 细胞不做区分, 统一抽象为抗体)的集合; $\boldsymbol{R}$ 为抗体间相互作用的规则集合; $\boldsymbol{A}$ 为支配抗体反应、指导抗体进化的算法.

1) 抗原 $\boldsymbol{G}$. 在免疫学中, 抗原是一类能够诱导机体免疫响应并能与相应抗体或 T 细胞受体发生特异反应的物质. 在人工免疫响应四元组模型中, 抗原一般指问题及其约束. 针对线性系统逼近问题, 抗原 $\boldsymbol{G} = J = \sum_{i=0}^{N} \left| G(j\omega_i) - H_m(j\omega_i) \right|^2$. 与免疫学中抗原的作用类似, 它是诱导人工免疫响应的始动因子.

2) 抗体空间$\boldsymbol{I}$. 集合$\boldsymbol{I} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, \cdots, \boldsymbol{b}_n\}$被称作抗体空间, 是针对抗原$\boldsymbol{G}$所有可能出现的抗体的集合, 其中$n$可以为无穷大的整数. 针对不同的抗原$\boldsymbol{G}$, 抗体$\boldsymbol{b}$的表现形式不同, 例如可以是二进制码串、实数序列、抽象的符号序列、特征序列等. 抗体是人工免疫响应的基础, 其表现形式对抗体间相互作用的规则集合$\boldsymbol{R}$的设计起着决定作用.

以字符串$\boldsymbol{b} = b_1 b_2 \cdots b_l$为例, 依生物学术语, 抗体$\boldsymbol{b}$中, b_i被视为遗传基因, 被称为等位基因, 其可能取值与编码方式有关. 在实践中, 常用的抗体编码方式有二进制和十进制. 例如, 抗体结构为八位二进制数, 则位串“0-1-1-1-0-1-0-0”即代表一个抗体. 抗体群$\boldsymbol{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, \cdots, \boldsymbol{b}_m\}$为抗体$\boldsymbol{b}$的$m$元组, 是抗体空间$\boldsymbol{I}$的一个

子集, 正整数m称为抗体群规模.

3) 规则集合$\boldsymbol{R}$. 抗体间相互作用的规则集合$\boldsymbol{R}$ = {R_1, R_2, R_3, …, R_l}描述了抗体空间$\boldsymbol{I}$中所有抗体之间可能存在的作用形式. 一个规则$R_i \in \boldsymbol{R}$可以从生物免疫系统中抗原与抗体间、抗体与抗体间的相互作用中启发得到. 对抗体群$\boldsymbol{B}$ = {$\boldsymbol{b}_1$, $\boldsymbol{b}_2$, $\boldsymbol{b}_3$, …, $\boldsymbol{b}_n$}, 一个规则$R_i \in \boldsymbol{R}$可以简略地表示为

$$R_i(\boldsymbol{B}) = R_i(\boldsymbol{b}_1 + \boldsymbol{b}_2 + \cdots + \boldsymbol{b}_n) = \boldsymbol{b}_1' + \boldsymbol{b}_2' + \cdots + \boldsymbol{b}_m', \tag{3}$$

其中 n, m 为正整数, m 的大小由规则 R_i 决定, 在这里‘+’并不是传统意义上的操作算子, 只是为了对(3)式中各抗体之间进行分隔(除特殊说明外, 下文中‘+’均为此义).

(3)式表示n个抗体经过规则R_i的作用, 演变成m个抗体. 为了细致地模拟生物免疫响应过程, 应该尽量详尽地设计足够多的规则, 从而实现对免疫响应过程的模拟.

4) 支配算法 $\boldsymbol{A}$. 动态支配算法 $\boldsymbol{A}$ 是模拟免疫系统中抗体进化过程以及支配人工免疫响应过程中抗体相互作用的算法, 包括规则集合 $\boldsymbol{R}$ 作用在抗体空间 $\boldsymbol{I}$ 中某一抗体群 $\boldsymbol{B}$ 上的具体方式、抗体-抗原亲和度、抗体-抗体亲和度的计算法则以及人工免疫响应终止条件的判断等.

在免疫学中, 抗体-抗原亲和度反映整分子抗体与抗原之间总的结合力. 在人工免疫响应模型中, 指候选解所对应的目标函数值或候选解对问题的适应性度量. 抗体-抗体亲和度反映抗体与抗体间的结合能力. 在人工免疫响应模型中, 指抗体间的距离度量, 用来度量抗体间的疏密程度. 例如, 对于二进制编码可以采用海明距离, 而实数编码可以采用范数, 如 Euclid 距离. 终止条件一般设置为最大迭代次数或求解精度或者二者的结合.

算法 $\boldsymbol{A}$ 可以是任何迭代算法, 但是一般需要根据具体的问题进行设计.

2.2　规则集合设计

人工免疫响应模型$\boldsymbol{AIR}$ = ($\boldsymbol{G}$, $\boldsymbol{I}$,$\boldsymbol{R}$, $\boldsymbol{A}$)的四个元素中, 抗体空间$\boldsymbol{I}$和支配算法$\boldsymbol{A}$要根据不同的抗原$\boldsymbol{G}$进行设计, 而规则集合$\boldsymbol{R}$中每条规则的具体操作和规则的选择是由抗原$\boldsymbol{G}$和抗体决定的. 本文设计的规则集合$\boldsymbol{R}$由三个子集组成, 分别是克隆选择规则集$\boldsymbol{R}_{CS}$, 免疫记忆规则集$\boldsymbol{R}_M$和免疫调节规则集$\boldsymbol{R}_A$, 即$\boldsymbol{R}$ = $\boldsymbol{R}_{CS} \cup \boldsymbol{R}_M \cup \boldsymbol{R}_A$, 具体设计如下.

2.2.1　克隆选择规则集

1958 年 Burnet 等提出了著名的克隆选择学说, 其中心思想为: 抗体是天然产物, 以受体的形式存在于细胞表面, 抗原可与之选择性地反应. 抗原与相应抗体受体的反应可导致细胞克隆增殖, 该群体具有相同的抗体特异性, 其中某些细胞

克隆分化为抗体生成细胞, 通过抗体基因频繁的变异和编辑, 在经过免疫后得到改善. 另一些形成免疫记忆细胞以参加之后的再次免疫响应.

1) 克隆增殖操作R_C^P. 在免疫学中, 克隆增殖是指通过无性繁殖(如细胞丝分裂)可连续传代并形成群体. 在人工免疫响应中, 对抗体种群$\boldsymbol{B}=\{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, \cdots, \boldsymbol{b}_n\}$的克隆增殖操作$R_C^P$表示为

$$\begin{aligned} &R_C^P(\boldsymbol{b}_1+\boldsymbol{b}_2+\cdots+\boldsymbol{b}_n)=R_C^P(\boldsymbol{b}_1)+R_C^P(\boldsymbol{b}_2)+\cdots+R_C^P(\boldsymbol{b}_n)\\ &=\{\boldsymbol{b}_1^1+\boldsymbol{b}_1^2+\cdots+\boldsymbol{b}_1^{q_1}\}+\{\boldsymbol{b}_2^1+\boldsymbol{b}_2^2+\cdots+\boldsymbol{b}_2^{q_2}\}+\cdots+\{\boldsymbol{b}_n^1+\boldsymbol{b}_n^2+\cdots+\boldsymbol{b}_n^{q_n}\}, \end{aligned} \tag{4}$$

其中 $R_C^P(\boldsymbol{b}_i)=\{\boldsymbol{b}_i^1+\boldsymbol{b}_i^2+\cdots+\boldsymbol{b}_i^{q_i}\}$, $\boldsymbol{b}_i^j=\boldsymbol{b}_i$, $j=1,2,\cdots,q_i$, $i=1,2,\cdots,n$, $q_i\in[1,n_c]$ 为一自适应参数, 也可以设定为一常数, n_c为设定的克隆比例上限, $q_i=1$ 表示对抗体没有进行克隆增殖操作. 可见, 上述克隆增殖过程与免疫学中的克隆增殖类似, 是简单的无性繁殖过程. 同一个抗体$\boldsymbol{b}_i$经过克隆增殖后形成的亚群体 $\boldsymbol{B}_i=(\boldsymbol{b}_i^1, \boldsymbol{b}_i^2,\cdots,\boldsymbol{b}_i^{q_i})$ 中的所有抗体与抗体$\boldsymbol{b}_i$具有完全相同的属性.

2) 亲和度成熟操作R_A^M. 在免疫学中, 亲和度成熟指免疫系统学习识别外部模式, 抗体基因频繁的变异和编辑, 抗体反应在经过免疫后得到改善. 亲和度成熟操作R_A^M是对免疫系统亲和度成熟过程的模拟.

对克隆增殖操作后的抗体群 $\boldsymbol{B}^{(1)}=R_C^P(\boldsymbol{B})=\{\boldsymbol{b}_1^1,\boldsymbol{b}_1^2,\cdots,\boldsymbol{b}_1^{q_1},\boldsymbol{b}_2^1,\boldsymbol{b}_2^2,\cdots,\boldsymbol{b}_2^{q_2},\cdots, \boldsymbol{b}_n^1,\boldsymbol{b}_n^2,\cdots,\boldsymbol{b}_n^{q_n}\}$ 的亲和度成熟操作R_A^M表示为

$$\begin{aligned} R_A^M(\boldsymbol{B}^{(1)})&=R_A^M(\{\boldsymbol{b}_1^1+\boldsymbol{b}_1^2+\cdots+\boldsymbol{b}_1^{q_1}\}+\{\boldsymbol{b}_2^1+\boldsymbol{b}_2^2+\cdots+\boldsymbol{b}_2^{q_2}\}\\ &\quad+\cdots+\{\boldsymbol{b}_n^1+\boldsymbol{b}_n^2+\cdots+\boldsymbol{b}_n^{q_n}\})\\ &=\{R_A^M(\boldsymbol{b}_1^1)+R_A^M(\boldsymbol{b}_1^2)+\cdots+R_A^M(\boldsymbol{b}_1^{q_1})\}\\ &\quad+\cdots+\{R_A^M(\boldsymbol{b}_n^1)+R_A^M(\boldsymbol{b}_n^2)+\cdots+R_A^M(\boldsymbol{b}_n^{q_n})\}\\ &=\{\boldsymbol{b}_1^{1'}+\boldsymbol{b}_1^{2'}+\cdots+\boldsymbol{b}_1^{q_1'}\}+\{\boldsymbol{b}_2^{1'}+\boldsymbol{b}_2^{2'}+\cdots+\boldsymbol{b}_2^{q_2'}\}\\ &\quad+\cdots+\{\boldsymbol{b}_n^{1'}+\boldsymbol{b}_n^{2'}+\cdots+\boldsymbol{b}_n^{q_n'}\}, \end{aligned} \tag{5}$$

其中 $R_A^M(\boldsymbol{b}_i^j)=\boldsymbol{b}_i^{j'}$, $i=1,2,\cdots,n$, $j=1,2,\cdots,q_i$. 免疫学中, 亲和度成熟操作主要是超变异(Hypermutation)[16]. 受此启发, 亲和度成熟操作的基本内容是以较小概率对抗体的某些基因位置上的基因值作变动, 但对$\boldsymbol{B}^{(1)}$中的每个抗体以概率 1 执行. R_A^M可以采用进化算法中已有的任何一种变异方式. 例如, 就基于字符集{0, 1}的二值编码串而言, R_A^M就是把某些基因位置上的基因位以概率取反, 即 1→0 或 0→1. 对实数编码串而言, 可以采用均匀变异, Gauss变异, Cauchy变异等随机变异策略.

3) 克隆选择操作R_C^S. 与进化算法中的选择操作不同, 克隆选择操作R_C^S是从抗体各自克隆增殖后的子代中选择优秀的个体, 从而形成新的种群, 是一个无性

选择过程. 对抗体群 $\boldsymbol{B}^{(2)}=R_A^M(\boldsymbol{B}^{(1)})=\{\boldsymbol{b}_1^{1'},\boldsymbol{b}_1^{2'},\cdots,\boldsymbol{b}_1^{q_1'},\boldsymbol{b}_2^{1'},\boldsymbol{b}_2^{2'},\cdots,\boldsymbol{b}_2^{q_2'},\cdots,\boldsymbol{b}_n^{1'},\boldsymbol{b}_n^{2'},\cdots,\boldsymbol{b}_n^{q_n'}\}$, 克隆选择操作$R_C^S$表示如下:

$$\begin{aligned}&R_C^S(\{\boldsymbol{b}_1^{1'}+\boldsymbol{b}_1^{2'}+\cdots+\boldsymbol{b}_1^{q_1'}\}+\{\boldsymbol{b}_2^{1'}+\boldsymbol{b}_2^{2'}+\cdots+\boldsymbol{b}_2^{q_2'}\}+\cdots+\{\boldsymbol{b}_n^{1'}+\boldsymbol{b}_n^{2'}+\cdots+\boldsymbol{b}_n^{q_n'}\})\\&=R_C^S(\{\boldsymbol{b}_1^{1'}+\boldsymbol{b}_1^{2'}+\cdots+\boldsymbol{b}_1^{q_1'}\})+R_C^S(\{\boldsymbol{b}_2^{1'}+\boldsymbol{b}_2^{2'}+\cdots+\boldsymbol{b}_2^{q_2'}\})\\&\quad+\cdots+R_C^S(\{\boldsymbol{b}_n^{1'}+\boldsymbol{b}_n^{2'}+\cdots+\boldsymbol{b}_n^{q_n'}\})\\&=\boldsymbol{b}_1'+\boldsymbol{b}_2'+\cdots+\boldsymbol{b}_n',\end{aligned}\tag{6}$$

其中 $R_C^S(\boldsymbol{b}_i^{1'}+\boldsymbol{b}_i^{2'}+\cdots+\boldsymbol{b}_i^{q_i'})=\boldsymbol{b}_i'$. $\forall i=1,2,\cdots n$, $\exists j\in\{1,2,\cdots,q_i\}$, 若抗体 $\boldsymbol{b}_i^{j'}$ 为亚群体 $\boldsymbol{B}_i'=(\boldsymbol{b}_i^{1'},\boldsymbol{b}_i^{2'},\cdots,\boldsymbol{b}_i^{q_i'})$ 中亲和度最高的抗体, 则在亚群体 $\boldsymbol{B}_i'$ 中 $\boldsymbol{b}_i^{j'}$ 的选择压力最大, 即 $\boldsymbol{b}_i'=\boldsymbol{b}_i^{j'}$ 的概率最大, 本文设置为 1.

可见, 克隆选择操作是克隆增殖操作的逆操作. 一个抗体经过克隆增殖后形成一个亚群体, 再经过亲和度成熟操作后通过克隆选择操作实现局部的亲和度升高.

2.2.2　免疫记忆规则集

现代免疫学认为, 淋巴细胞除了扩增或者分化成浆细胞以外, 也能分化成生命期较长的B记忆细胞. 这些记忆细胞中的信息编码组成免疫系统记忆, 使它能学习、记忆蛋白质结构. 再次遇到相应的抗原后, 为清除已经刺激初次响应的相应的特异抗原, 记忆细胞将预先被免疫系统选择出来, 并迅速活化、增殖、分化为效应细胞, 执行高效而持久的免疫功能. 本文通过设置抗体记忆单元来模拟免疫记忆过程. 具体操作包括记忆单元的强迫学习操作R_M^N和记忆单元的自学习操作R_M^S.

1) 记忆单元强迫学习操作R_M^N. 设当前抗体记忆单元$\boldsymbol{M}$ = ($\boldsymbol{m}_1$, $\boldsymbol{m}_2$, $\boldsymbol{m}_3$, ⋯, $\boldsymbol{m}_s$), 抗体$\boldsymbol{b}_0$为记忆单元 $\boldsymbol{M}$ 之外的一个强势抗体(在与其他同源抗体竞争中胜出的抗体), 则记忆单元的强迫学习操作R_M^N表示如下:

$$R_M^N(\boldsymbol{M}+\boldsymbol{b}_0)=\{\boldsymbol{m}_1'+\boldsymbol{m}_2'+\boldsymbol{m}_3'+\cdots+\boldsymbol{m}_s'\}=\boldsymbol{M}',\tag{7}$$

其中强迫学习操作R_M^N设计如下: 计算抗体$\boldsymbol{b}_0$与当前记忆单元$\boldsymbol{M}$中的所有抗体的抗体-抗体亲和度, 如上文所述, 抗体-抗体亲和度体现了两个抗体在空间的相对分布情况, 其值越大, 表示两个抗体的基因差异越大; 其值越小, 则两抗体的基因差异越小. 设记忆单元$\boldsymbol{M}$中抗体$\boldsymbol{m}_i\neq\boldsymbol{M}$与抗体$\boldsymbol{b}_0$的抗体-抗体亲和度最小, 且最小值为$\delta_{i0}$. 计算记忆单元$\boldsymbol{M}$中任意两个抗体间亲和度的最小值, 标记为$\delta_0$. 如果$\delta_{i0}\leqslant\delta_0$, 则比较两抗体$\boldsymbol{b}_0$和$\boldsymbol{m}_i$的抗体-抗原亲和度$\boldsymbol{G}(\boldsymbol{b}_0)$和$\boldsymbol{G}(\boldsymbol{m}_i)$的大小, 令

$$\boldsymbol{m}_i=\begin{cases}\boldsymbol{b}_0, & \boldsymbol{G}(\boldsymbol{b}_0)>\boldsymbol{G}(\boldsymbol{m}_i),\\ \boldsymbol{m}_i, & \text{其他}.\end{cases}\tag{8}$$

记忆单元抗体数目$N_m = s\ N_m = s$；如果$\delta_{i0} > \delta_0$，则令$\boldsymbol{m}_{s+1} = \boldsymbol{b}_0$，记忆单元抗体数$N_m = s + 1$，若$N_m$大于指定的记忆单元最大规模$N_{\max}$，则删除当前记忆单元中的最差抗体.

2) 记忆单元的自学习操作R_M^S. 记忆单元在通过强迫学习R_M^N不断增加自体信息的同时，其中的抗体基因也在频繁的变异和编辑，实现对记忆信息的自身优化，逐步实现亲和度成熟. 记忆单元的自学习操作R_M^S是对这一过程的模拟，表示如下:

$$R_M^S(\boldsymbol{m}_1 + \boldsymbol{m}_2 + \boldsymbol{m}_3 + \cdots + \boldsymbol{m}_s) = \boldsymbol{m}_1' + \boldsymbol{m}_2' + \boldsymbol{m}_3' + \cdots + \boldsymbol{m}_s'. \tag{9}$$

具体地，各抗体以自身为中心的克隆区间大小为S_m(S_m一般取为变量区间大小的$\frac{1}{s}$)，在克隆区间内对$\boldsymbol{M}$执行克隆增殖操作R_C^P，亲和度成熟操作R_A^M和克隆选择操作R_C^S. 为了保持算法操作的多样性，互补各种算子的长短，对记忆单元的亲和度成熟操作可以设计不同于常规抗体亲和度成熟操作的方法.

2.2.3 免疫调节规则集

免疫作为维持生物体的防御和自身稳定的重要功能必然要受机体的调节，然而免疫系统调节过程既不像神经系统调节那样有明晰的反射通路，也不像内分泌系统那样仅通过释放一些激素分子来完成生理机能的调节. 免疫系统功能受多种因素调节，有抗原、抗体调节也有细胞及细胞分泌的各种因子的调节. 各调节因素之间关系也错综复杂，有正的调节，也有负的调节.

1) 基因免疫操作R_G^I. 基因免疫的主要原理是将抗原编码基因插入带有强启动子的质粒载体，然后用物理方法将此重组质粒导入体内细胞，抗原编码基因即在细胞内合成抗原蛋白，诱发机体产生保护性免疫反应. DNA疫苗实质是基因免疫. 本文设计的基因免疫操作R_G^I如下:

首先，要提取 DNA 疫苗形成疫苗库 vaccine-storage, vaccine-stroage 内存在三种疫苗: 完整的整分子抗体、抗体基因片段、单个基因位. 一般情况下，疫苗库内要充分包含问题的先验知识和已得知识. 实际的操作过程中，首先要对所求解的问题(抗原)进行具体分析，从中提取出最基本的特征信息，作为疫苗；然后，对此特征信息进行处理，将其转化为问题的可行解，使其成为整分子抗体，同样放入 vaccine-storage. 另外，除了从先验知识中获取疫苗外，还可以从已得知识中获得疫苗. 例如，从记忆单元中获取疫苗，即以概率抽取当前记忆单元中的某些基因片断，或者整分子抗体放入疫苗库.

对抗体群$\boldsymbol{B} = \{\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3, \cdots, \boldsymbol{b}_n\}$进行基因免疫操作 R_G^I 的一种方式是在$\boldsymbol{B}$中按比例随机抽中$\alpha(0<\alpha\leqslant n)$个抗体. 设抗体$\boldsymbol{b}_s$被抽中，则用疫苗库vaccine- storage中的基因或基因片段替代抗体$\boldsymbol{b}_s$的某些基因位上的基因，使得新抗体以较大概率具

有更高的抗体-抗原亲和度. 另一种方式是用疫苗库vaccine-storage中的整分子抗体疫苗以一定概率替代抗体群$\boldsymbol{B}$中的某些或某个抗体. 上述模型可以用下式简略表示:

$$R_G^I(\boldsymbol{B}+\text{vaccine-storage})=\boldsymbol{B}'+\text{vaccine-storage}. \tag{10}$$

2) 克隆死亡R_C^D. 细胞死亡(apoptosis)是免疫响应和免疫调控的重要形式之一, 免疫细胞的增殖和分化过程是与免疫细胞的正常死亡相统一的, 免疫响应的效应过程也是与靶细胞的死亡相伴随的. 克隆死亡操作R_C^D是对上述细胞死亡现象的简单模拟.

对抗体种群$\boldsymbol{B}=\{\boldsymbol{b}_1,\boldsymbol{b}_2,\boldsymbol{b}_3,\cdots,\boldsymbol{b}_n\}$的克隆死亡操作$R_C^D$表示为

$$R_C^D(\boldsymbol{B})=R_C^D(\{\boldsymbol{b}_1+\boldsymbol{b}_2+\cdots+\boldsymbol{b}_n\})=\alpha_1\boldsymbol{b}_1+\alpha_2\boldsymbol{b}_2+\cdots+\alpha_n\boldsymbol{b}_n, \tag{11}$$

其中$\alpha_i\boldsymbol{b}_i$表示抗体$\boldsymbol{b}_i$经过克隆死亡操作后在抗体群中存在α_i个$\boldsymbol{b}_i$, 系数α_i是大于等于零的整数, $\alpha_i=0$表示抗体$\boldsymbol{b}_i$在人工免疫响应过程中呈现细胞死亡, 即克隆死亡. 造成抗体$\boldsymbol{b}_i$克隆死亡的原因可能是因为抗体$\boldsymbol{b}_i$具有较低的抗体-抗原亲和度值, 也有可能是因为抗体$\boldsymbol{b}_i$与其他抗体具有相同的基因表现型, 而造成抗体群的整体多样性下降. $\alpha_i>0$表示伴随死亡现象所呈现的靶细胞正反馈免疫调节作用, 促使亲和度高的抗体(靶细胞)进行局部增殖. 在人工免疫响应模型中, 为了保持算法的可持续性, 一般有$\alpha_1+\alpha_2+\cdots+\alpha_n=n$.

与生物进化系统有性别之分不同, 免疫系统是一个无性系统, 通过上述描述可以看出, 规则集$\boldsymbol{R}=\boldsymbol{R}_{CS}\cup\boldsymbol{R}_M\cup\boldsymbol{R}_A$中的操作均为无性操作过程, 可以理解为人工免疫响应对免疫系统这一特点的简单刻画.

2.3　动态支配算法设计

针对线性系统最优化逼近问题, 本文设计的动态支配算法的主要步骤为:

第 1 步: 设定算法终止条件, 给定抗体群规模n, 记忆单元规模s, 克隆规模n_c, 记忆单元克隆规模n_m, 变异概率p_m, 基因免疫概率p_c, 克隆死亡比例T%; 设定初始搜索空间$[\underline{\boldsymbol{x}},\overline{\boldsymbol{x}}]$, 并在$[\underline{\boldsymbol{x}},\overline{\boldsymbol{x}}]$区间内随机产生初始抗体群$\boldsymbol{B}(1)=\{\boldsymbol{b}_1(1),\boldsymbol{b}_2(1),\cdots,\boldsymbol{b}_n(1)\}\in\boldsymbol{I}^n$, 从$\boldsymbol{B}(1)$中选取$s$个抗体构成记忆单元, $\boldsymbol{M}(1)=\{\boldsymbol{m}_1(1),\boldsymbol{m}_2(1),\cdots,\boldsymbol{m}_s(1)\}\in\boldsymbol{I}^s$, 进化代数$k=1$.

第 2 步: 对抗体群$\boldsymbol{B}(k)$执行克隆死亡操作: $\boldsymbol{B}^{(1)}(k)=R_C^D(\boldsymbol{B}(k))$.

第 3 步: 对$\boldsymbol{B}^{(1)}(k)$进行克隆增殖操作: $\boldsymbol{B}^{(2)}(k)=R_C^P(\boldsymbol{B}^{(1)}(k))$.

第 4 步: 对$\boldsymbol{B}^{(2)}(k)$进行基因免疫操作: $\boldsymbol{B}^{(3)}(k)=R_G^I(\boldsymbol{B}^{(2)}+\text{vaccine-storage})$, 其中 vaccine-storage 为从当前记忆单元$\boldsymbol{M}(k)$中随机抽取的基因片断组成的疫苗库.

第 5 步: 对$\boldsymbol{B}^{(3)}(k)$进行亲和度成熟操作: $\boldsymbol{B}^{(4)}(k)=R_A^M(\boldsymbol{B}^{(3)}(k))$.

第 6 步: 对$\boldsymbol{B}^{(4)}(k)$进行克隆选择操作: $\boldsymbol{B}^{(5)}(k)=R_C^S(\boldsymbol{B}^{(4)}(k)\bigcup\boldsymbol{B}(k))$.

第 7 步: 对记忆单元$\boldsymbol{M}(k)$进行记忆单元强迫学习操作: $\boldsymbol{M}^{(1)}(k)=R_M^N(\boldsymbol{M}(k)+\boldsymbol{b}_0)$, 其中$\boldsymbol{b}_0$为抗体群$\boldsymbol{B}^{(5)}(k)$中的最优抗体.

第 8 步: 对记忆单元$\boldsymbol{M}^{(1)}(k)$进行记忆单元自学习操作: $\boldsymbol{M}^{(2)}(k)=R_M^S(\boldsymbol{M}^{(1)}(k))$.

第 9 步: 终止条件判断, 如果满足, 则输出$\boldsymbol{M}^{(2)}(k)$中的最优抗体, 算法结束; 否则, 令 $\boldsymbol{B}(k+1)=\boldsymbol{B}^{(5)}(k)$, $\boldsymbol{M}(k+1)=\boldsymbol{M}^{(2)}(k)$, $k = k + 1$, 如果k为N_E的倍数, 执行搜索区间动态扩展操作, 返回第 2 步.

对于上述算法, 有以下 5 点需要说明:

1) 算法采用实数编码, 即用一定精度的实数向量表示单一抗体, 本文精确到小数点后 9 位;

2) 算法中抗体的优劣仅由抗体-抗原亲和度决定, 而抗体-抗原亲和度设计为抗体对应的频率域L^2-误差J的值, 抗体对应的J值越小越优秀;

3) 算法亲和度成熟操作时采用按位变异的策略, 即对种群$\boldsymbol{B}^{(3)}(k)$的新抗体$\boldsymbol{b}_i^{(3)}(k)$, 以变异概率$p_m$用 0~9 的随机整数代替其某些基因位上的值. 例如, 若抗体$\boldsymbol{b}_i^{(3)}(k)=(1.233567,12.33\underline{4}567,0.123356)$, 并且该抗体的第二个变量的第 5 个基因位被选中变异, 则随机产生一个 0~9 的整数, 代替$\boldsymbol{b}_i^{(3)}(k)$中画下划线的基因位上的数字 4.

4) 本文设定最大迭代次数作为算法的终止条件.

5) 算法中搜索区间动态扩展操作采用文献[2]中的方法.

3 算法分析

用于线性系统逼近的人工免疫响应算法(AIRA)本质上是并行地运行在两个种群上, 即抗体种群和记忆单元. 抗体种群和记忆单元分别构成了不同的随机状态转移过程. 抗体种群的演化可以描述为

$$\boldsymbol{B}(k)\xrightarrow{R_C^D}\boldsymbol{B}^{(1)}(k)\xrightarrow{R_C^P}\boldsymbol{B}^{(2)}(k)\xrightarrow{R_G^I}\boldsymbol{B}^{(3)}(k)\xrightarrow{R_A^M}\boldsymbol{B}^{(4)}(k)\xrightarrow{R_C^S}\boldsymbol{B}^{(5)}(k)\rightarrow\boldsymbol{B}(k+1). \tag{12}$$

记忆单元的演化可以描述为

$$\boldsymbol{M}(k)\xrightarrow{R_M^N}\boldsymbol{M}^{(1)}(k)\xrightarrow{R_M^S}\boldsymbol{M}^{(2)}(k)\rightarrow\boldsymbol{M}(k+1). \tag{13}$$

这两个过程并行地进行: 抗体种群是与抗原作用(解空间搜索)的基本种群, 强调全局搜索; 而记忆单元主要是在抗体种群的基础上, 进行定义域变尺度的局部搜索, 一方面记忆搜索过程, 加快收敛速度; 另一方面通过记忆学习, 保持种

群的多样性. 另外, 正是由于存在记忆单元, 基因免疫操作才更好地实现了抗体间的相互协作.

下边主要分析算法的收敛性.

定义 1　定义问题的全局最优解集为

$$\boldsymbol{B}^* \triangleq \{\boldsymbol{B}\in\boldsymbol{I}: f(\boldsymbol{B}) = f^* \triangleq \max(f(\boldsymbol{B}'): \boldsymbol{B}'\in\boldsymbol{I})\}. \tag{14}$$

对于抗体种群 $\boldsymbol{B}$, $\mathcal{G}(\boldsymbol{B})\equiv|\boldsymbol{B}\cap\boldsymbol{B}^*|$ 表示抗体种群 $\boldsymbol{B}$ 中包含最优解的数目. 称 $\boldsymbol{I}^*=\{\boldsymbol{B}\in\boldsymbol{I}^*\,|\,\mathcal{G}(\boldsymbol{B})\geqslant 1\}$ 为最优抗体种群空间.

定义 1 表明, 最优抗体种群空间中的抗体种群至少含有 1 个最优抗体. 进一步称 $\overline{\boldsymbol{I}}^*=\boldsymbol{I}^n-\boldsymbol{I}^*$ 为普通抗体种群空间. 另外, 称克隆增殖操作导致变化的抗体空间为克隆抗体种群空间, 记为 $\boldsymbol{I}^{\sum q_i}$, 这是一个变化的空间.

定义 2　如果对于任意的初始状态$\boldsymbol{B}_0$, 均有

$$\lim_{k\to\infty} P\{\boldsymbol{B}(k)\cap\boldsymbol{B}^*\neq\varnothing\,|\,\boldsymbol{B}(0)=\boldsymbol{B}_0\} = \lim_{k\to\infty} P\{\boldsymbol{B}(k)\in\boldsymbol{I}^*\,|\,\boldsymbol{B}(0)=\boldsymbol{B}_0\}=1. \tag{15}$$

也就是

$$\lim_{k\to\infty} P\{\mathcal{G}(\boldsymbol{B}(k))\geqslant 1\,|\,\boldsymbol{B}(0)=\boldsymbol{B}_0\}=1, \tag{16}$$

则称算法以概率 1 收敛到最优种群集.

定理 1　用于线性系统逼近的人工免疫响应算法以概率 1 收敛到最优种群集.

证　记 $P_0(k)=P\{\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k)=0\}=P\{(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))\cap\boldsymbol{B}^*=\varnothing\}$, 由Bayes条件概率公式[17]有

$$\begin{aligned}&P_0(k+1)=P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=0\}\\&=P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=0\,|\,\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))\neq 0\}\times P\{\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))\neq 0\}\\&+P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=0\,|\,\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\times P\{\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}.\end{aligned} \tag{17}$$

由克隆选择操作的性质可知, 克隆选择获得的最优解至少比克隆增殖操作之前的好, 因此, $P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=0\,|\,\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))\neq 0\}=0$, 所以

$$P_0(k+1)=P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=0\,|\,\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\times P_0(k). \tag{18}$$

又

$$\begin{aligned}&P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=1\,|\,\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}_{\min}\\&=\min_i\left\{\frac{1}{n+s}\left[\frac{1}{\lambda-1}p_m^{d(\boldsymbol{b},\boldsymbol{b}_i(k))}(1-p_m)^{l-d(\boldsymbol{b},\boldsymbol{b}_i(k))}\right]^{q_i(k)}\right\}>0,\end{aligned} \tag{19}$$

其中对本文中基于实数编码的按位变异策略, (19)式中λ = 10.

记: $\zeta=\min_k P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=1\,|\,\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}_{\min}, k=0,1,2,\cdots,$

$$P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=1\big|\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\geqslant\zeta>0\,, \tag{20}$$

所以

$$\begin{aligned}&P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=0\big|\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\\&=1-P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))\neq 0\big|\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\\&=1-P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))\geqslant 1\big|\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\\&\leqslant 1-P\{\mathcal{G}(\boldsymbol{B}(k+1)\cup\boldsymbol{M}(k+1))=1\big|\mathcal{G}(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))=0\}\leqslant 1-\zeta<1.\end{aligned} \tag{21}$$

因此

$$0\leqslant P_0(k+1)\leqslant(1-\zeta)\times P_0(k)\leqslant(1-\zeta)^2\times P_0(k-1)\cdots\leqslant(1-\zeta)^{k+1}\times P_0(0)\,. \tag{22}$$

因为 $\lim\limits_{k\to\infty}(1-\zeta)^{k+1}=0$，$1\geqslant P_0(0)\geqslant 0$，所以

$$0\leqslant\lim_{k\to\infty}p_0(k)\leqslant\lim_{k\to\infty}(1-\zeta)^{k+1}P_0(0)=0\,, \tag{23}$$

故 $\lim\limits_{k\to\infty}p_0(k)=0$，因此 $\lim\limits_{k\to\infty}P\{(\boldsymbol{B}(k)\cup\boldsymbol{M}(k))\cap\boldsymbol{B}^*\neq\varnothing\big|\boldsymbol{B}(0)=\boldsymbol{B}_0\}=1-\lim\limits_{k\to\infty}P_0(k)$ $=1$，即 $\lim\limits_{k\to\infty}p\{\boldsymbol{B}(k)\cup\boldsymbol{M}(k)\in\boldsymbol{I}^*\big|\boldsymbol{B}(0)=\boldsymbol{B}_0\}=1$.

于是定理 1 得证，即用于线性系统逼近的人工免疫响应算法以概率 1 收敛.

4 算法的有效性验证

本节选用两个线性系统逼近问题来验证人工免疫响应算法(AIRA)的有效性. 第一个是稳定线性系统的最优化逼近问题，第二个是非稳定、非最小相位线性系统的最优化逼近问题. 对每种情况随机独立运行 30 次，并将结果与文献[2~4]中的相应结果进行比较. AIRA算法的运行参数设置如下：抗体群规模 10，记忆单元规模 5，克隆规模 4，记忆单元克隆规模 4，变异概率 0.25，基因免疫概率 1，克隆死亡比例 0.5，最大迭代次数 300. 在此参数设置下，AIRA算法 30 次试验的平均评价次数仅为 18015，与同类算法相比(例如，求解稳定线形系统逼近问题时，文献[2]需要评价 19800 次，文献[4]需要评价 19735 次)，计算量相当.

4.1 稳定线性系统逼近问题

该系统来源于文献[2]，其传递函数为

$$G(S)=\frac{Y(s)}{U(s)}=\frac{k_d k_{r1}(\tau_{od}s+1)e^{-\theta_d s}}{(\tau_r s+1)(\tau_1 s+1)(\tau_2 s+1)-k_{r2}k_d(\tau_{od}s+1)e^{-\theta_d s}}\,, \tag{24}$$

其中k_{r1}= 0.258，k_{r2}= 0.281，k_d= 1.4494，θ_d = 0.2912，τ_r = 0.3683，τ_1 = 1.9624，τ_2 = 0.43256. 用下式所示的 2 阶模型来逼近原系统:

$$H_2(s)=\frac{k_{2,p}(s+\tau_{2,z})}{a_{2,0}+a_{2,1}s+s^2}\cdot e^{-\tau_{2,d}s}\,. \tag{25}$$

采用 51 个频率值, $\omega_i = 10^{-2+0.1i}, i = 0,1,\cdots N = 50$, 来最小化(2)式. 由于要满足 $H_2(0) = G(0)$, 因此参数 $a_{2,0}$ 可以用下式表示:

$$a_{2,0} = \frac{k_{2,p}(1-k_{r2}k_d)}{k_d k_{r1}}\tau_{2,z}, \tag{26}$$

因此需要确定的参数为$a_{2,1}$, $k_{2,p}$, $\tau_{2,z}$, $\tau_{2,d}$. 由于原始系统$G(s)$是稳定的, 每个参数都位于区间$[0,+\infty)$上.

首先用固定搜索区间的AIRA同时搜索参数$a_{2,1}$, $k_{2,p}$, $\tau_{2,z}$, $\tau_{2,d}$, 分别令搜索区间为$[0,10]^4$, $[0,50]^4$, $[0,100]^4$, $[0,150]^4$进行测试. 然后采用搜索区间动态扩展的AIRA来搜索参数$a_{2,1}$, $k_{2,p}$, $\tau_{2,z}$, $\tau_{2,d}$, 其中初始搜索区间为$[0, 0.1]^4$, 扩展周期N_E = 10, 区间扩展参数为 2. AIRA与其他算法的最优结果比较如表 1 所示.

表 1　不同算法逼近稳定线性系统的最优结果比较

搜索区间	方法	逼近模型	J
$[0, 10]^4$	DEA[2]	$H_2(s) = \frac{0.0266114(s+6.10646821)e^{-0.4894422s}}{s^2+1.65105264s+0.25757166}$	1.4308×10^{-4}
	AIRA	$H_2(s) = \frac{0.021430231(s+6.130164886)e^{-0.454259829s}}{s^2+1.226172374s+0.208228229}$	4.4889×10^{-5}
$[0, 50]^4$	DEA	$H_2(s) = \frac{0.00495562(s+35.0505313)e^{-0.390779375s}}{s^2+1.7606862s+0.27531759}$	6.2568×10^{-5}
	AIRA	$H_2(s) = \frac{0.006473125(s+20.835707811)e^{-0.360175108s}}{s^2+1.254970252s+0.213777917}$	8.2269×10^{-6}
$[0, 100]^4$	DEA	$H_2(s) = \frac{0.00415886(s+65.9114728)e^{-0.5907097s}}{s^2+2.7140333s+0.43448690}$	1.0021×10^{-3}
	AIRA	$H_2(s) = \frac{0.003374488(s+39.802784056)e^{-0.332274274s}}{s^2+1.250285692s+0.212893270}$	8.5147×10^{-6}
$[0, 150]^4$	AIRA	$H_2(s) = \frac{0.006163363(s+21.914785100)e^{-0.357316587s}}{s^2+1.256812898s+0.214089609}$	7.7369×10^{-6}
动态扩展区间	DEA	$H_2(s) = \frac{0.0025538(s+68.1706405)e^{-0.3787929s}}{s^2+1.7606862s+0.2759465}$	6.1129×10^{-5}
	MGA[4]	$H_2(s) = \frac{0.0172520(s+7.7668024)e^{-0.4346119s}}{s^2+1.2483717s+0.2123842}$	2.7180×10^{-5}
	AIRA	$H_2(s) = \frac{0.004717881(s+28.822747965)e^{-0.351042212s}}{s^2+1.264931127s+0.215537542}$	6.5526×10^{-6}

从表 1 可以看出, 在搜索区间为$[0,10]^4$, $[0,50]^4$, $[0,100]^4$以及采用区间动态扩展时, AIRA均能找到比DEA更优的逼近模型, 即使搜索区间为$[0,150]^4$时, AIRA同样能够逼近到10^{-6}的误差. 因此, AIRA算法性能受搜索区间扩大的影响明显弱于DEA, 甚至可以说, 在一定范围内, 搜索区间的增大, 不会使AIRA的搜索结果

变差. 同时, AIRA得到的逼近误差也明显小于MGA的逼近误差. 图 1 所示AIRA得到的逼近系统的频率响应曲线与原系统的频率响应曲线几乎重合(在图中已经无法区分), 从而说明了AIRA优良的逼近性能.

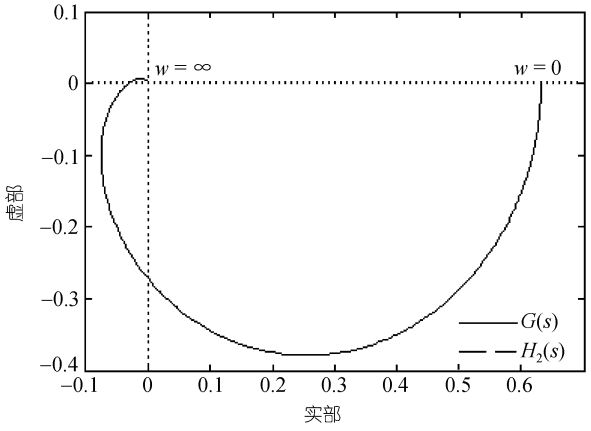


图 1 原系统与逼近系统的频率响应

4.2 非稳定线性系统逼近问题

考虑一个四阶非稳定传递函数[2~4]

$$G(s)=\frac{60s^3+25850s^2+685000s-2500000}{s^4+105s^3+10450s^2+45000s-500000}, \tag{27}$$

用下式所示的 2 阶系统来逼近原系统:

$$H_2(s)=\frac{c_{2,1}s+c_{2,0}}{s^2+b_{2,1}s+b_{2,0}}. \tag{28}$$

采用 61 个频率值, $\omega_i=10^{-2+0.2i}, i=0,1,\cdots,N=60$, 来最小化(2)式, 因此需要确定的参数为$c_{2,1}$, $c_{2,0}$, $b_{2,1}$, $b_{2,0}$, 每个参数都位于区间 $(-\infty,+\infty)$ 上.

同样, 首先用固定搜索区间的AIRA来搜索参数$c_{2,1}$, $c_{2,0}$, $b_{2,1}$, $b_{2,0}$, 分别令搜索区间为$[-500,500]^4$, $[-1000,1000]^4$, $[-2000,2000]^4$, $[-5000,5000]^4$进行测试, 上述四种情况J值的最优结果如表 2 所示.

然后采用搜索区间动态扩展的AIRA来搜索参数$c_{2,1}$, $c_{2,0}$, $b_{2,1}$, $b_{2,0}$, 其中, 初始搜索区间为$[-0.1, 0.1]^4$, 扩展周期$N_E=10$, 区间扩展参数为 2. AIRA得到最优结果与其他方法的比较如表 3 所示, 表中还给出了逼近误差$\xi=|G(jw)-H_2(jw)|$的L^∞-范数的值.

从表 2 可以看出, 对四组不同搜索区间, AIRA 都能逼近到误差为 8.795. 因此, 在逼近非稳定线形系统时, 搜索区间的增大, 对 AIRA 性能的影响也不是很明显.

表 2　AIRA 搜索区间固定时逼近非稳定线性系统的最优结果

方法	逼近模型	J	ξ
$[-500,500]^4$	$H_2(s)=\dfrac{129.283987966s-451.269244983}{s^2+13.078500583s-91.851175413}$	8.795	1.381
$[-1000,1000]^4$	$H_2(s)=\dfrac{129.283365165s-450.817260700}{s^2+13.085060426s-91.764682717}$	8.795	1.380
$[-2000,2000]^4$	$H_2(s)=\dfrac{129.330759446s-450.748056298}{s^2+13.090054195s-91.755967406}$	8.795	1.379
$[-5000,5000]^4$	$H_2(s)=\dfrac{129.251155710s-450.318766288}{s^2+13.082602535s-91.687472588}$	8.795	1.381

表 3　不同算法逼近非稳定线性系统的最优结果比较

探索区间	逼近模型	J	ξ
AIRA	$H_2(s)=\dfrac{129.310457004s-450.941328300}{s^2+13.088391129s-91.794882948}$	8.795	1.380
MGA[4]	$H_2(s)=\dfrac{129.315s-451.012}{s^2+13.092s-91.8129}$	8.795	1.380
DEA[2]	$H_2(s)=\dfrac{129.297s-450.711}{s^2+13.089s-91.7495}$	8.795	1.449
GS[3]	$\hat{G}_{2,H}(s)=\dfrac{160.1(s-3.074)}{(s+25.84)(s-5.000)}$	26.46	1.194
GL1[18]	$\hat{G}_{2,g1}(s)=\dfrac{161.0(s-3.155)}{(s+24.7)(s-5)}$	18.21	1.122
GL2[18]	$\hat{G}_{2,g2}(s)=\dfrac{177.6(s-7.183)}{(s+24.7)(s-12.8)}$	25.21	1.599
BT1[19]	$\hat{G}_{2,\alpha}(s)=\dfrac{183.5(s-3.099)}{(s+27.1)(s-5.21)}$	23.79	1.278
BT2[19]	$\hat{G}_{2,n}(s)=\dfrac{205.5(s-2.680)}{(s+31.9)(s-4.73)}$	32.26	1.445

从表 3 可以明显看出AIRA得到的L^2-误差J的值明显优于文献[3, 18, 19]中的最优结果. 虽然AIRA、MGA和DEA都能达到同样的L^2-误差, 但是AIRA逼近误差的L^∞-范数的值要小于DEA. 图 2 所示为以上 8 个逼近模型的误差大小随频率的变化情况, 从图 2(a)中可以看出AIRA, MGA和DEA获得的模型在频率范围为$[10^{-4},10^{0.6}]$和$[10^2,10^{10}]$内具有最好的逼近性能(在图 2(a)中三条曲线几乎重合, 无法区分). 图 2(b)的放大图显示AIRA获得的模型的逼近误差要小于MGA和DEA获得的模型. 因此, AIRA能够得到原系统更好的逼近模型.

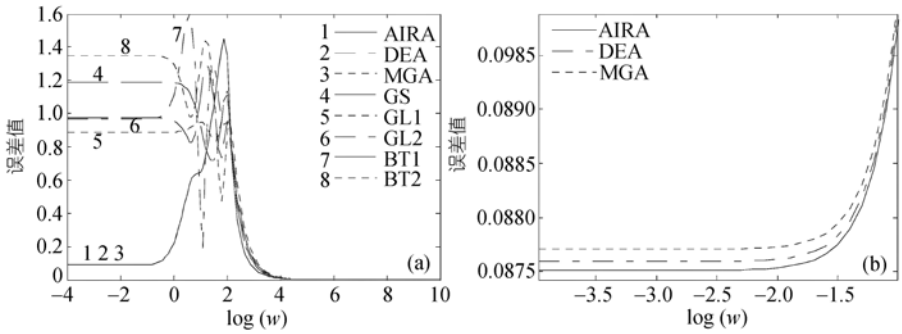


图 2 逼近误差大小随频率的变化情况

5 结论与展望

本文提出了人工免疫响应的四元组模型，给出了克隆选择、免疫记忆和免疫调节的具体操作，设计了用于线性系统逼近的人工免疫响应算法，证明了算法的全局收敛性. 新算法具有以下特点:

1) 提出了人工免疫响应的四元组模型，为免疫响应过程建立了一个可用于工程计算的数学模型;

2) 模拟了抗体克隆选择过程;

3) 有效模拟了免疫系统再次响应现象，实现了人工免疫响应的记忆学习;

4) 模拟了基因免疫、免疫耐受等现象.

新算法通过最小化频率域L^2-误差来优化逼近参数，从而获得了稳定线性系统或非稳定线性系统的最优降阶模型. 基于两个典型的稳定或非稳定线性系统的逼近问题，通过与文献[2~4]中的相应算法的数值试验比较，本文验证了新算法的有效性，得出以下结论: 无论在固定的区间内搜索还是在动态扩展的区间内搜索，人工免疫响应算法都能得到线性系统的最优逼近模型，从而很好地解决线性系统逼近这一复杂问题.

工程应用是人工免疫系统研究的最终目标. 人工免疫系统从产生到现在，不到 20 年的时间内已经广泛应用于优化、设计、控制、诊断、计算机安全等许多领域，然而当冷静分析这些应用时，可以发现: 与神经网络等相对成熟的人工智能方法的应用相比，人工免疫系统的应用范围依旧较小，而且有关应用比较简单，甚至很多只是在原有的应用方法中引入了一些免疫机理; 再者，多数应用还停留在实验室，并没有进入工程实践. 然而，人工免疫系算法所具有的优良特性，使其具有非常广泛的应用前景. 本文的相关成果不仅可以用于线性系统逼近问题，也应该可以很好地应用于包括组合优化、图像处理、人工生命、决策优化、通讯、故障诊断等许多领域. 同时，我们认为，有关人工免疫系统的应用研究不

但要进一步扩大应用领域, 而且要在已有的应用领域中, 根据新的需求改进策略, 使之体现出更大的经济效益和社会效益.

参　考　文　献

1 Fortuna L, Nunnari G, Gallo A. Model Order Reduction Techniques with Applications in Electrical Engineering. London: Springer-Verlag, 1992

2 Cheng S L, Huang C Y. Optimal approximation of linear systems by a differential evolution algorithm. IEEE Transactions on systems, man, and cybernetics-part A, 2001, 31(6): 698~707[DOI]

3 Guo T Y, Huang C Y. Optimal reduced-order models for unstable and nonminimum-phase systems. IEEE Transactions on Circuits and Systems-I, 1996, 43(9): 800~805[DOI]

4 Zhong W C, Liu J, Xue M Z, et al. A multiagent genetic algorithm for global numerical optimization. IEEE Transactions on systems, man, and cybernetics-part B, 2004, 34(2): 1128~1141[DOI]

5 钟伟才, 刘 静, 焦李成. 多智能体遗传算法用于线形系统逼近. 自动化学报, 2004, 30(6): 933~938

6 Spanos J T, Milman M H, Mingori D L. A new algorithm for L^2 optimal model reduction. Automatics, 1992, 28(5): 897~909[DOI]

7 钟伟才. 多智能体进化模型和算法研究. 西安电子科技大学博士学位论文, 2004

8 Leandro N, de Castro, Jonathan T. Artificial Immune Systems: A New Computational Intelligence Approach. Springer Verlag, 2002

9 莫宏伟. 人工免疫系统原理与应用. 哈尔滨: 哈尔滨工业大学出版社, 2002

10 焦李成, 杜海峰. 人工免疫系统进展与展望. 电子学报, 2003, 31(10): 1540~1548

11 de Castro L N, Von Zuben F J. Learning and optimization using the clonal selection principle. IEEE Transactions on Evolutionary Computation, Special Issue on Artificial Immune Systems, 2002, 6(3): 239~251

12 Dasgupta D. Artificial Immune Systems and Their Applications. Springer-Verlag, 1999

13 Jiao L C, Wang L. A novel genetic algorithm based on immunity. IEEE Transactions on Systems, Man and Cybernetics, Part A, 2000, 30(5): 552~561[DOI]

14 Du H F, Gong M G, Jiao L C, et al. A novel artificial immune system algorithm for high-dimensional function numerical optimization. Progress in Natural Science. Taylor & Francis Ltd press, 2005, 15(5): 463~471

15 Abul K A, Andrew H L, Jordan S P. Cellular and Molecular Immunology. 4th edition. New York: W B Saunders Co, 2000

16 Berek C, Ziegner M. The maturation of the immune response. Immune Today, 1993, 14(8): 400~402[DOI]

17 Swinburne R. Bayes's Theorem. Oxford: Oxford University Press, 2002

18 Glover K. All optimal Hankel-norm approximations of linear multivariable systems and their L^{∞}-error bounds. International Journal of Control, 1984, 39(6): 1115~1193

19 Parker P J, Anderson B D O. Unstable rational function approximation. International Journal of Control, 1987, 46(5): 1783~1801