文章编号: 1001-0920 (2012) 06-0819-08

基于多尺度并行免疫克隆优化聚类算法

陶新民[1], 付丹丹[1], 刘福荣[2], 刘 玉[1]

(1. 哈尔滨工程大学 信息与通信工程学院，哈尔滨 150001；
2. 哈尔滨电力职业技术学院 信息工程系，哈尔滨 150030)

摘 要: 针对无监督分类问题, 提出一种多尺度并行免疫克隆优化聚类算法. 算法中, 进化在多个子群之间并行进行, 不同子群的抗体根据子群适应度采用不同变异尺度. 进化初期, 利用大尺度变异子群实现全局最优解空间的快速定位, 同时变异尺度随着适应值的提升逐渐降低; 进化后期, 利用小尺度变异子群完成局部解空间的精确搜索. 将新算法与其他聚类算法进行比较, 所得结果表明新算法具有较好的聚类性能和鲁棒性.

关键词: 聚类算法；免疫克隆优化；变异算子；子群适应度

中图分类号: TP18 **文献标识码:** A

Multi-scale parallel immune clone optimization clustering algorithm

TAO Xin-min[1], *FU Dan-dan*[1], *LIU Fu-rong*[2], *LIU Yu*[1]

(1. College of Information and Communication Engineering，Harbin Engineering University，Harbin 150001，China;
2. Department of Information Engineering，Harbin Power Vocational Technology College，Harbin 150030，China.
Correspondent：TAO Xin-min，E-mail：taoxinmin@hrbeu.edu.cn)

Abstract: A novel multi-scale parallel artificial immune clone algorithm for unsupervised clustering(MSPAICC) is presented, in which, evolutions of subgroups are performed in parallel with the different mutation strategies. The mutation capability of an individual is determined by the competition among subgroups and subgroup fitness value. The larger mutation operator is used to quickly localize the global optimal space at the early evolution, while the smaller mutation operator whose scale gradually reduces are adopted to improve the local search ability at the later evolution. The experimental results show the proposed method can improve clustering performance and the robustness compared with other clustering algorithms.

Key words: clustering algorithm；immune clone optimization；mutation operator；subgroup fitness value

1 引 言

聚类问题被认为是最重要的无监督学习问题, 作为一种重要的统计分析方法, 受到了国内外学者的广泛关注, 研究成果涉及了数据处理、机器学习、数据挖掘等领域. 在现有聚类方法中, K-均值算法[1]是最常用的, 属于最陡下降算法, 其目标函数是高度非线性和多峰的函数, 若初值设定不理想, 则其分类结果会陷入局部极值点.

进化算法是一种成熟的具有高鲁棒性和广泛适用性的全局优化方法, 由于其形式简单和群体进化等优点, 已被很多学者用于聚类问题. 如文献 [2] 利用遗传优化方法求解聚类问题; [3] 提出了基于遗传算法的聚类技术; [4] 将遗传算法应用于基于基因表述数据的多尺度聚类. 但是由于遗传算法本身存在容易早熟收敛以及变异算子和交叉算子的无方向性操作等问题, 使得聚类结果准确率不高. 免疫算法[5-10]是继遗传进化算法之后发展起来的全局优化搜索算法, 其独有的自组织、自学习、自记忆的能力使其在很多问题上与遗传算法相比表现出较好的性能[11-17], 因此, 有人将免疫算法用于解决聚类问题. 其中文献 [18-19] 提出了基于免疫克隆优化的聚类算法. 尽管这些算法部分解决了基于遗传算法的聚类算法的缺点, 但由于上述算法都是以一个种群参与进化, 无法保证在基因变异操作时选择合适的变异算子, 使算法在有限次迭代内快速寻到全局最优解.

鉴于此, 本文提出一种基于多尺度并行免疫克隆

收稿日期: 2010-12-21；**修回日期:** 2011-06-13.

基金项目: 国家自然科学基金项目(61074076)；中国博士后科学基金项目(20090450119)；中国博士点新教师基金项目(20092304120017)；黑龙江省博士后基金项目(LBH-Z08227).

作者简介: 陶新民(1973−), 男, 副教授, 从事智能信号处理、智能计算等研究；付丹丹(1985−), 女, 硕士生, 从事智能计算的研究.

优化的聚类算法(MSPAICC). 该算法中, 进化在采用不同变异尺度的子群间进行, 其中群体适应度的大小直接影响了子群的进化; 适应度好的子群为保持优势将以较小的尺度进行变异, 有利于局部精确解的开采; 适应度差的子群使用大尺度变异来逃离当前位置, 有利于算法初期逃逸局部极值, 实现最优解区域的快速定位. 这样不同大小尺度的变异使得整个种群对解空间进行了充分探索, 与单一尺度相比具有更详尽的解空间探索能力. 由于子空间的重新组合使得抗体在子群间转移, 从而实现了子群间信息的交换. 通过对不同形状的人工数据及不同维度的UCI数据进行聚类仿真实验, 结果表明本文算法具有很强的全局收敛性和鲁棒性.

2 人工免疫克隆优化聚类算法

将免疫克隆优化应用于聚类问题, 需要解决抗体表示、抗体亲和度定义、算法流程设计等关键技术, 其中抗体表示及抗体亲和度定义将在第3节介绍. 传统人工免疫克隆优化聚类算法(AICC)[18-19]的核心机制主要包括克隆增殖、基因变异和克隆选择操作, 它们构成了克隆算子的主体. 具体算法如下:

1) 参数初始化. 随机产生 N 个抗体, 组成初始抗体种群 $B^{(0)}$, 并计算它的亲和度.

2) 克隆增殖. 根据亲和度, 对抗体种群 $B^{(K)}$ 中的抗体进行克隆增殖操作, 得到扩增后的种群 $C^{(K)}$, 克隆规模是抗体亲和度的单调递增函数.

3) 基因变异. 对 $C^{(K)}$ 中抗体进行基因变异操作, 变异率为 p_m, 生成一个成熟的抗体种群 $M^{(K)}$.

4) 克隆选择. 计算 $M^{(K)}$ 中抗体的亲和度, 对 $M^{(K)}$ 进行克隆选择操作, 得到新的抗体种群 $B'^{(K)}$.

5) 判断是否满足终止条件. 若满足, 则程序结束, 输出最优解; 否则, 转至2).

克隆算子的实质是在抗体进化过程中, 在候选解的附近, 根据亲和度的大小进行克隆, 产生一个变异解的群体, 从而扩大了搜索范围, 有助于防止进化早熟和搜索陷入局部极值, 同时通过克隆选择来加快收敛速度. 克隆算子还实现了子群之间的信息交换, 提高了克隆的多样性和种群的亲和度. 其中基因变异操作利用均匀变异算子来提高算法抗体种群的多样性.

均匀变异算子是通过对克隆后的种群 $C^{(K)}$ 中的某个 b_i 抗体第 l 个分量 $b_{i(l)}$ 加入来自某区间的随机数进而产生变异, 具体公式如下:

$$b''_{i(l)} = b'_{i(l)} + \Delta_l \text{Rand}(0,1). \tag{1}$$

其中: $\text{Rand}(0,1)$ 表示从0到1均匀分布的随机变量, 而

$$\Delta_l = \begin{cases} {b'_{i(l)}}^{\min} - b'_{i(l)}, & \text{Rand}(0,1) < 0.5; \\ {b'_{i(l)}}^{\max} - b'_{i(l)}, & \text{Rand}(0,1) \geqslant 0.5. \end{cases} \tag{2}$$

式(2)表示每次随机选择一个 Δ_l, 保证每一个 $b_{i(l)}$ 准确地位于一个对应区间 $[{b_{i(l)}}^{\min}, {b_{i(l)}}^{\max}]$ 内. 由于均匀变异能产生远离原始抗体的变异, 使得算法在迭代过程中通过较大均匀变异算子保持了解空间的勘探能力, 维护了种群的多样性.

3 多尺度并行免疫克隆聚类算法

由上可知, 传统的AICC中[18-19]只有一个种群参与进化, 在种群中的抗体只能遵循唯一的规律进化(均匀变异算子). 任何一种进化算法都存在勘探和开采两类不同的操作, 如果这两类操作协调不好, 则很容易使算法陷入局部最优解或降低其收敛性能. 上述传统人工免疫克隆优化聚类算法中实现基因变异操作的均匀变异虽然具有很强的逃逸局部极值点的能力, 但因事先无法预知函数局部极值间的距离, 故无法保证逃逸后的新位置位于全局最优解的附近, 即所逃离到的新位置的适应值不一定优于现有的最优解, 尤其在进化后期最优解可能就存在于现有最优解周边区域, 这样通过均匀变异将无法保证最优解的精度和算法收敛性能.

为保证增加算法种群多样性, 防止算法陷入局部最优的同时, 使算法在进化后期仍具有较强的局部精确解的开采能力, 提高聚类性能, 本文提出一种基于多尺度并行免疫克隆优化的聚类算法(MSPAICC). 该算法首先根据适应度大小将种群分为多个, 对于适应度较差的子群中抗体, 通过采用具有振荡性质大尺度变异有利于新空间的勘探, 使算法在初期阶段快速逃出局部最优解; 适应值较好的子群采用小尺度变异可使算法在进化后期实现局部精确解的搜索以提高解的精度, 保证算法收敛性能. 这样可在保证算法种群多样性的同时提高算法收敛性能以及所求解的精度.

3.1 聚类算法描述

3.1.1 抗体表示

人工免疫系统中, 抗原对应于优化问题的输入数据, 如目标函数和各种约束条件; 抗体对应于优化问题的优化解. 聚类问题中, 把要分类的数据对象视为免疫系统中的抗原, 把聚类中心视为抗体, 数据对象的聚类过程即是免疫系统产生出可以捕获抗原的最佳抗体的过程. 对于 D 维聚类问题, 聚类个数为KN, 则抗体编码长度为 $D \times \text{KN}$, 采用实数编码的方式.

3.1.2 抗体亲和度的定义

抗体亲和度是一个抗体对一个相同链长的抗原产生识别的程度, 是对抗体结合强度的评估. 人工免疫系统中, 一般指优化问题的解与目标函数的匹配程

度. 聚类问题中, 抗体的亲和度值即为以抗体为聚类中心的类别划分所对应的目标函数值. 首先, 以抗体所代表的各类别的聚类中心为基准, 将所有未被分类的样本数据点 $x_i(i=1,2,\cdots,n)$ 按下式划分到不同的类别 $R_j(j\in\{1,2,\cdots,\mathrm{KN}\})$ 中, 并以欧氏距离作为抗体与抗原的亲和力指标:

$$j=\arg\min_{j=1,2,\cdots,\mathrm{KN}}(D(x_i,m_j)). \tag{3}$$

其中: m_j 为第 j 类的聚类中心, R 为抗体 p 的类别划分. 则定义抗体 p 的亲和度为

$$f(p)=\frac{1}{1+\sum\limits_{R_k\in R}\sum\limits_{i\in R_k}D(x_i,m_k)}. \tag{4}$$

其中: m_k 为类别 R_k 的中心, $D(x_i,m_k)$ 为类别 R_k 中的第 i 个样本与 m_k 之间的欧氏距离. 为了简化计算, 采用如下与抗体亲和度成反比的计算公式作为抗体的适应度:

$$\mathrm{Aff}(p)=\sum_{R_k\in R}\sum_{x_i\in R_k}D(x_i,m_k). \tag{5}$$

3.1.3　算法流程

本文提出的并行多尺度免疫克隆聚类算法是将整个种群按照适应值划分为多个子群, 利用不同变异尺度在多群间并行进化. 具体流程如下:

1) 初始化参数. 在候选解中选择 N 个抗体组成初始种群, 这些抗体被划分为 M 个子群, 其中 N 能被 M 整除. $P=N/M$ 为每一个子群的抗体数. 循环次数 $K=0$, 初始高斯变异标准差设置为 $\sigma_0=[\sigma_1^0,\sigma_2^0,\cdots,\sigma_{M-1}^0]$, $\sigma_1^0<\sigma_2^0<\cdots<\sigma_{M-1}^0$. 通过增加不同初始值的高斯变异在提高解精度的同时, 也可以防止算法对初始种群敏感的问题, 即如果算法初始种群中的任何一个个体都距离全局最优解很远, 而高斯变异标准差又很小, 则算法收敛于局部极值的机会将会很大, 而这种缺陷只能依靠均匀变异算子来克服, 因此具有很强的随机性, 同时随着维度的增多, 这种逃逸能力便显得微不足道, 最终导致算法需要很多的进化代数才能收敛到全局最优解. 通过增加不同初始值 σ_0 的高斯变异算子可以使得算法在进化前期, 通过较大初始值 σ_0 的高斯振荡变异, 使种群尽快地从局部极值的邻域变异到全局最优解的邻域, 提高了算法收敛速度.

2) 子群的划分. 子群成员的划分是以抗体适应度的好坏为依据的. 每个抗体适应度的好坏直接反映了其在种群中的竞争力, 适应度越好竞争力越强, 而竞争力的强弱决定于抗体变异能力的大小. 具体划分方法如下: 在问题的可行解空间中随机产生 N 个抗体, 并根据抗体适应度从小到大的顺序排列, 作为初始种群 $B^{(0)}=\{b_1,b_2,\cdots,b_N\}$. 按照排序结果将整个种群分为 M 个子群, 排在前 P 个的抗体组成子群 $B_1^{(0)}$, 以此类推, 排在最后的 P 个抗体组成子群 $B_M^{(0)}$; 前 $M-1$ 个子群参与高斯变异, 为进一步维护种群多样性, 最后一个子群按式 (1) 进行均匀变异. 这样划分子群的优势在于对子群适应度进行排序能够针对不同竞争能力的子群采用不同尺度的变异算子, 竞争中胜出的子群为保持优势将以较小的概率进行变异, 有利于局部精确解的开采: 竞争失败的抗体使用大尺度震荡变异来改变自己的生存条件, 有利于算法初期局部极值的逃逸.

3) 子群间的竞争. 对于高斯变异子群而言, 对环境适应能力强的子群使用较小的变异可以提高算法对解空间精确的局部搜索能力, 对环境适应能力弱的子群使用较大的变异可以在较大范围内对目标进行搜索, 防止算法陷入局部极小点. 计算高斯变异子群 $B_m^{(K)}$ 中每一个抗体 b_i^m 的适应度 $\mathrm{Aff}(b_i^m)$, 其中 $m\in[1,M-1], i\in[1,P]$, 则第 m 个子群 $B_m^{(K)}$ 的适应度 $\mathrm{Aff}B_m^{(K)}$ 为

$$\mathrm{Aff}B_m^{(K)}=\sum_{i=1}^{P}\mathrm{Aff}(b_i^m)/P. \tag{6}$$

不同子群在相互竞争中根据其适应环境能力的好坏获得了不同的变异能力. 第 m 子群的变异所采用的高斯算子标准差为

$$\begin{aligned}&\sigma_m^{(K)}=\\&\sigma_m^{(K-1)}\exp\Big(\frac{M\cdot\mathrm{Aff}B_m^{(K)}-\sum\limits_{m=1}^{M-1}\mathrm{Aff}B_m^{(K)}}{\mathrm{Aff}B_{\max}-\mathrm{Aff}B_{\min}}\Big),\\&\mathrm{Aff}B_{\max}=\max_{1\leqslant i<M}(\mathrm{Aff}B_i^{(K)}),\\&\mathrm{Aff}B_{\min}=\min_{1\leqslant i<M}(\mathrm{Aff}B_i^{(K)}).\end{aligned} \tag{7}$$

由式 (7) 可看出, 随着迭代次数的增加算法会逐渐搜索到适应度更好的抗体, 从而每个子群 $B_M^{(K)}$ 的适应度 $\mathrm{Aff}B_M^{(K)}$ 将随着算法的进化逐渐提升, 子群间适应度差距呈递减趋势, exp( ) 函数值变小, 因此 $\sigma_m^{(K)}$ 随迭代次数 K 的增加逐渐减小, 继续参与下代进化中变异算子的计算. 这样设计变异算子可以达到变异尺度随适应度自适应调节的目的, 有利于进化初期逃逸局部极值和后期精确解的搜索.

4) 子群的克隆增殖. 克隆增殖是指通过无性繁殖连续传代并形成群体. 对第 m 个抗体子群 $B_m^{(K)}=\{b_1^m,b_2^m,\cdots,b_P^m\}$ 中每个抗体进行克隆增殖产生新的 $b_i^{m(j)}=b_i^m, i=1,2,\cdots,P,\ j=1,2,\cdots,q_i$. 其中 q_i 定义如下:

$$q_i=\mathrm{Int}(\mathrm{CN}*p_i),\ i=1,2,\cdots,P. \tag{8}$$

p_i 为 b_i^m 产生新个体的概率, 定义如下:

$$p_i = \mathrm{Aff}(b_i^m)\Big/\sum_{i=1}^{P}\mathrm{Aff}(b_i^m),\ i=1,2,\cdots,P. \tag{9}$$

其中: $\mathrm{Aff}(b_i^m)$ 为第 i 个抗体 b_i^m 的适应度, 常数 CN 为子群克隆规模, q_i 可依据 CN 和 p_i 自适应调整, $\mathrm{Int}(\cdot)$ 为上取整函数.

5) 基因变异操作. 对不同的子群使用不同的变异算子来繁殖后代. 种群 $B_m^{(K)}$ 的基因变异操作是对抗体的某些基因位置上的基因值变动, 本文中, 针对高斯变异子群 $B_m^{(K)}(m=1,2,\cdots,M-1)$ 中抗体 b_i^m 的分量为 $b_{i(l)}^m$, 其更新方程为

$$b_{i(l)}^m = b_{i(l)}^m + \delta_i^{(K)}, \tag{10}$$

其中 $\delta_i^{(K)}$ 为服从 $N(0,\sigma_m^{(K)^2})$ 分布的高斯白噪声. $B_M^{(K)}$ 中的抗体利用式 (1) 进行均匀变异. 如此设置有利于保证算法种群多样性, 提高算法收敛性能及所求解的精度.

6) 克隆选择操作. 克隆选择操作是将克隆增殖和基因变异操作后形成的新子群中亲和度最好的个体选择出来从而形成下一代种群. 对于抗体子群 $B_m^{(K)} = \{b_1^{m(1)}, b_1^{m(2)},\cdots,b_1^{m(q_1)};\ b_2^{m(1)}, b_2^{m(2)},\cdots, b_2^{m(q_2)};\cdots;b_P^{m(1)}, b_P^{m(2)},\cdots,b_P^{m(q_P)}\}$, 抗体 b_i^m 经克隆增殖和基因变异操作后形成了新的抗体, 通过克隆选择操作在这些新抗体中选择出亲和度最好的一个, 可实现局部亲和度的升高. $\forall i=1,2,\cdots,P$, $\exists j\in\{1,2,\cdots,q_i\}$, 若抗体 $b_i^{m(j)}$ 为子群体 $\{b_i^{m(1)}, b_i^{m(2)},\cdots,b_i^{m(q_i)}\}$ 中亲和度最高的抗体, 则在子群体 $\{b_i^{m(1)}, b_i^{m(2)},\cdots,b_i^{m(q_i)}\}$ 中抗体 $b_i^{m(j)}$ 被选择代替 b_i^m 成为 $b'_{i^m}=b_i^{m(j)}$.

7) 种群重组. 种群重组的规则是选择 $\{b_i^{m(1)}, b_i^{m(2)},\cdots,b_i^{m(q_i)}\}(\forall i=1,2,\cdots,P)$ 中抗体适应度最好的 $b_i^{m(j)}(\exists j\in\{1,2,\cdots,q_i\})$ 来替换原抗体群中的 b_i^m, 生成新的种群 $I^{(K)}$, 由随机产生的新抗体代替种群 $I^{(K)}$ 中适应度差的 $\mathrm{Int}(\beta N)$ 个抗体, 增加种群的多样性. β 是随机替代比例, 一般取值为 $0.1\sim0.15$. 最后将生成的抗体重新按照适应度从小到大排序并均分为 M 个子群, 排在前 P 个的抗体组成子群 $B_1^{(K+1)}$, 以此类推, 排在最后的 P 个抗体组成子群 $B_M^{(K+1)}$, 作为均匀变异子群.

在利用 MSPAICC 进行聚类优化时需要注意的是, 因变异算子的进化是一个递归过程, 后面的子群高可能会采用很大的高斯变异算子, 故对变异算子的标准差作如下限制: 设 B 为目标优化变量空间的宽度, 如果 $\sigma_i^{(k)}>B/4$, 则

$$\sigma_i^{(k)} = |B/4-\sigma_i^{(k)}|, \tag{11}$$

直到满足 $\sigma_i^{(k)}<B/4$. 不同尺度变化情况如图 1 所示, 不同子群通过采用不同尺度变异能实现整个搜索空间的覆盖, 大尺度具有振荡性质, 有利于解空间的粗搜索, 可以快速定位到最优解区域; 逐渐减小的小尺度能在进化后期实现局部精确解的搜索. MSPAICC 流程如图 2 所示.

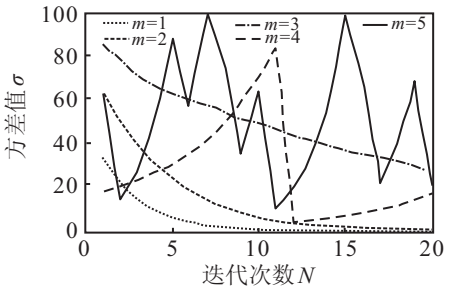

图 1 $M=5$ 时不同尺度方差随迭代数的变化示意

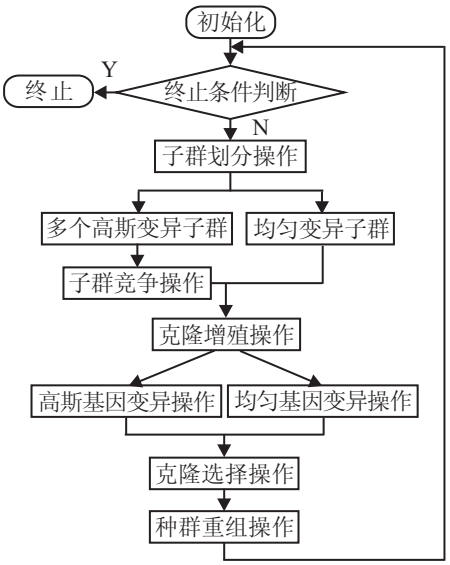

图 2 并行多尺度免疫克隆优化聚类算法流程

3.2 算法的收敛性能分析

与文献 [15] 相同, 在多尺度并行免疫克隆优化聚类算法中, 抗体种群序列 $B^{(K)}$ 仍是一个有限齐次可约马尔可夫链. 虽然将一个种群划分为多个子群, 但由克隆选择操作的保优性可得

$$P\{\vartheta(B^{(k+1)})=0|\vartheta(B^{(k)})\neq0\}=0, \tag{12}$$

其中 $\vartheta(B^{(K)})$ 为抗体种群 $B^{(K)}$ 中包含最优抗体的个数. 因此对于任意初始状态 $B^{(0)}$, 并行多尺度免疫克隆优化聚类算法仍以概率 1 收敛于最优抗体集合 B^*. 证明与文献 [15] 类似, 此略.

4 对比实验及结果分析

4.1 实验设置

为了测试本文提出的新算法对数据的分类性能, 首先将新算法应用于来自 UCI 的不同维度的实际数据[20]以及 3 个不同形状的人工数据集的聚类问题, 具体设置见表 1. 将新算法同基于 K-均值聚类算法 (KM)[1], 基于遗传的聚类算法 (GAC)[4]以及文献 [18]

表 1　实验中所用的数据集及其特征

数据集	种类	属性	类别	样本大小
Square1	人工	2	4	500
Size1	人工	2	4	500
Square5	人工	2	4	500
Iris	UCI	4	3	150
Glass	UCI	9	6	214
Wine	UCI	13	3	178
Ionosphere	UCI	34	2	351
Vehicle	UCI	18	4	846

基于传统克隆优化的聚类算法(AICC)进行比较. 其中本文算法MSPAICC及AICC的算法设置如下: 亲和度成熟条件为迭代次数500, 抗体种群的规模50, 子群的个数5, 克隆规模比例10, CN(每个子群共需克隆个数)$=(50/5)\times 10=100$, 随机替代比例0.1; GA的参数设置如下: 算法终止条件为迭代次数500, 种群规模50, 交叉概率0.8, 变异概率0.1; KM的最大迭代次数设为500, 停止阈值设为10^{-10}.

4.2　全局收敛性能对比实验

为了测试MSPAICC算法的全局收敛性能, 将3种进化聚类算法应用于上述每个数据集中, 每个数

表 2　4种优化聚类算法全局解性能与稳定性比较

数据	K-均值算法	GAC聚类	AICC聚类	本文算法
Square1	13. 205±10.306	12.665 9±6.321	10.898 3±5.211	**8.750 4±1.264 5**
Size1	18.102 2±9.566 8	10.815 3±5.120	13.852 9±4.128	**10.774 1±2.147**
Square5	23.031±13.126	12.292 4±7.311	12.066 3±6.431	**10.599 9±2.789**
Iris	13.485 3±5.174	19.45±7.321	9.663 8±2.311	**7.060 5±0.567**
Glass	54.312 9±24.356	100.265 8±39.121	31.511 2±17.673	**21.2485±6.132**
Wine	143.731±54.311	122.092 1±38.123	98.267 0±26.797	**60.009 1±16.866**
Ionosphere	1 211.326±171.212	1 031.768±123.31	731.486±116.11	**618.939 3±68.122**
Vehicle	822.43±153.210	726.8±103.321	343.631 9±67.576	**265.749 5±24.31**

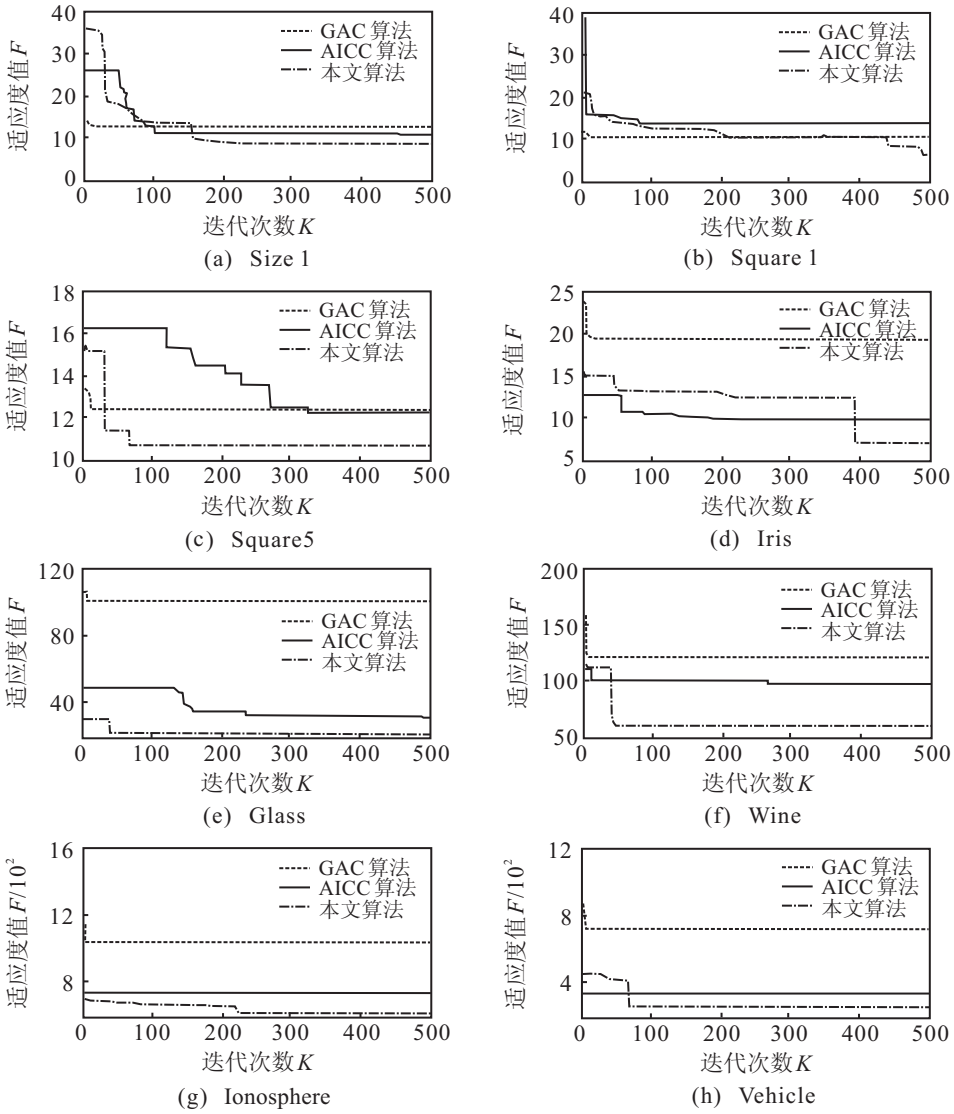

图 3　3种进化聚类算法对各数据集聚类收敛情况对比

据集独立运行 30 次. 为了便于比较收敛性能, 将每种算法的适应度设置为欧氏距离之和, 具体如式 (5), 计算各个算法得到的适应度的平均值及方差如表 2 所示. 为了更直观地显示 3 种算法的收敛情况, 在图 3 中展示了 3 种算法的收敛效果.

从表 2 的统计数据及图 3 中的聚类收敛结果可以看出, 本文的基于并行多尺度免疫克隆优化的聚类算法的全局收敛能力优于其他基于传统免疫克隆和基于遗传的聚类算法. 这是由于本文算法通过将抗体种群分为多个子群, 不同子群按照各自的高斯变异算子进行进化, 增强了算法的种群多样性, 克服了原有算法陷入局部最优的不足; 同时, 本文算法在保证算法全局解性能的前提下, 通过多个大小不一的高斯变异算子变异以及子群间的协作竞争, 提高了全局解的精度. 尤其当数据的维度增高时, 传统免疫克隆和遗传的聚类算法很快陷入局部最优解且无法逃逸; 而本文算法能够在有限的迭代次数下成功逃出局部极小点. 这是由于本文算法的子群采用了大尺度震荡变异算子, 使得算法初期快速逃离局部最优解. 实验结果也进一步说明了本文算法对不同维度不同形状数据集的聚类问题都具有较好的全局解搜索能力且算法稳定.

4.3 聚类性能对比实验

为了验证本文聚类算法的聚类性能, 将本文算法同 K-均值聚类算法 (KM), 基于遗传的聚类算法 (GAC)[4] 以及文献 [18] 传统克隆优化的聚类算法 (AICC) 进行了比较. 为评价聚类算法性能, 本文采用指标 adjusted rand index 来衡量, 它把样本类别的划分看作是它们之间的一种关系, 或者将样本分在同一类, 或者在不同类, 通过统计正确聚类对数来评价算法性能. 对于有 n 个样本的数据集, adjusted rand index 可按下式计算:

$$R(U,V)=\frac{\sum_{lk}\begin{bmatrix}n_{lk}\\2\end{bmatrix}-\frac{1}{2}\left[\sum_{l}\begin{bmatrix}n_{l\cdot}\\2\end{bmatrix}+\sum_{k}\begin{bmatrix}n_{\cdot k}\\2\end{bmatrix}\right]\rightarrow}{\leftarrow\frac{\left[\sum_{l}\begin{bmatrix}n_{l\cdot}\\2\end{bmatrix}\cdot\sum_{k}\begin{bmatrix}n_{\cdot k}\\2\end{bmatrix}\right]\Big/\begin{bmatrix}n\\2\end{bmatrix}}{\left[\sum_{l}\begin{bmatrix}n_{l\cdot}\\2\end{bmatrix}\cdot\sum_{k}\begin{bmatrix}n_{\cdot k}\\2\end{bmatrix}\right]\Big/\begin{bmatrix}n\\2\end{bmatrix}}}. \tag{13}$$

其中: n_{lk} 为被划分到类属 l 和类属 k 的样本个数; $R(U,V)\in(0,1]$, 其数值越大, 说明聚类划分的正确率越高. 对上述数据每一种算法分别运行 30 次, 计算其 adjusted rand index 的平均值如表 3 所示. 从实验结果可以看出, 本文聚类算法的聚类性能以及稳定性都优于其他 3 种聚类算法, 这是由于本文提出的优化算法全局收敛能力强且稳定的缘故.

表 3 4 种算法对数据聚类结果性能与稳定性比较

数据	K-均值算法	GAC 聚类	AICC 聚类	本文算法
Square1	0.947 4± 0.004 3	0.956 3± 0.065 6	0.957 9±0.027	**0.978 1±0.022 13**
Size1	0.555 3± 0.038 6	0.580 0± 0.031 2	0.564 7±0.022 8	**0.673 6±0.011 9**
Square5	0.652 0±0.0	0.699 1±0.009 1	0.712 1±0.017 1	**0.821 5±0.001 6**
Iris	0.684 5± 0.09	0.543 2±0.08	0.690 0± 0.105 5	**0.762 1±0.031 1**
Glass	0.160 4± 0.039 7	0.151 3± 0.023 2	0.164 4± 0.025 9	**0.432 0±0.012**
Wine	0.787 4± 0.162 1	0.819 5±0.135	0.867 0±0.191	**0.901±0.042 7**
Ionosphere	0.210 7± 0.081 8	0.272 8±0.031	0.486±0.021	**0.739 3±0.019**
Vehicle	0.079 8±0.008 9	0.18±0.003 21	0.312±0.0176	**0.641 7±0.002 3**

4.4 算法的鲁棒性分析

为了考察 4 种算法的鲁棒性, 采用文献 [18] 中的鲁棒性分析方法对 4 种算法在求解以上 8 个问题时的鲁棒性进行比较. 具体地, 算法在某一特定数据集上的相对性能用该算法在求解该问题时得到的 adjusted rand index 的值与最大 adjusted rand index 值的比值来衡量, 即

$$b_m=\frac{R_m}{\max\limits_k R_m}. \tag{14}$$

因此, 在某个数据集上表现最好的算法 m^* 的相对性能 $b_{m^*}=1$, 而其他算法的相对性能 $b_m\leqslant 1$. b_m 值越大, 表明算法 m 在所有算法中的相对性能越好. 算法 m 在所有数据集上的 b_m 值的总和可以用来客观评价算法的鲁棒性, 总和越大, 鲁棒性越好. 图 4 为 4 种算法的鲁棒性比较结果, 每一个算法对应的柱状图顶部所标数值为对应算法在所有 8 个问题上的 b_m 值的总和.

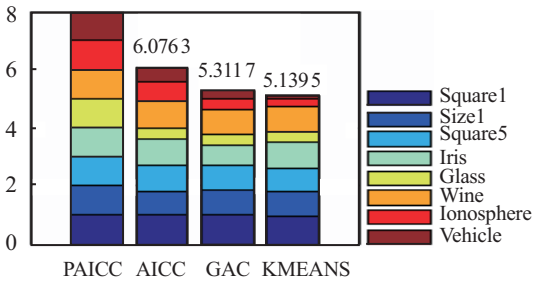

图 4 4 种算法的鲁棒性比较

从图 4 中可以看出, 本文算法获得了最高的总和值, 达到了 8; AICC 次之, 达到了 6.076 3. 这充分说明了基于多尺度并行免疫克隆优化聚类算法对于无监督聚类问题具有很好的鲁棒性. 实际上, MSPAICC 算法的 b_m 值对测试的 8 个问题均为 1, 因此, MSPAICC 对不同空间结构以及不同维度的数据聚类问题均表现出很好的性能, 其在所有比较的 4 种算法中具有最好的鲁棒性. 这是由于通过大小不一的多高斯变异算子变异可有效防止算法对初始种群敏感的不足, 提高了算法的稳定性.

4.5　子群个数对算法聚类性能的影响

为了考察 MSPAICC 算法中子群个数对算法聚类性能的影响, 取 100 个抗体, 将子群个数分别设置为 2, 5, 10, 20, 25, 实验数据为 Square1, Size1, Square5, Iris 四个数据集, 算法的其他设置同上, 实验结果如图 5 所示. 从实验可以看出, MSPAICC 算法开始时随着子群个数的增大聚类性能呈上升趋势, 这是由于子群个数的增加, 抗体的高斯变异算子不同, 增强了种群多样性, 提高了全局解的精度. 然而当子群个数进一步增加时, 由于每个子群中抗体数减少, 使得每个子群的局部搜索能力降低, 算法无法在有限的迭代次数内找到全局最优解, 因此在实际应用中需选择适度的子群个数以在增加多样性的同时保持子群的局部搜索性能. 另外, 从实验中可以看出, 多尺度并行免疫克隆优化聚类算法较单一变异克隆优化算法效果好, 这也验证了本文提出的多子群变异并行免疫克隆思想的可行性和有效性.

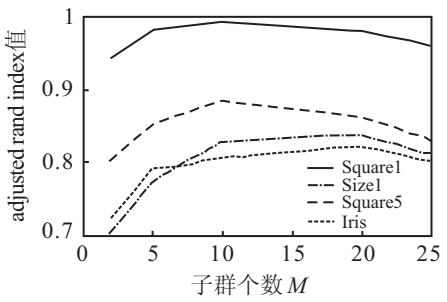

图 5　子群个数对 MSPAICC 算法聚类性能影响

5　结　论

本文提出了一种多尺度并行免疫克隆优化聚类算法. 进化在采用不同变异尺度的子群间进行, 进化不仅取决于群体适应度的大小, 同时也考虑了不同子群间的竞争和交流. 利用不同空间特征和维度的数据集进行测试, 实验结果表明, 本文算法能够在算法初期成功逃出局部最优解, 快速定位到最优解区域, 使得算法在进化后期, 通过采用逐渐减低的小尺度变异算子的子群抗体寻找精确解, 提高了全局解搜索能力, 并且算法稳定. 此外, 利用实验分析了不同子群数对本文算法聚类性能的影响, 结果表明, 在选取子群个数时种群多样性和子群的局部搜索性能应折中考虑. 需要指出的是, 本文算法的计算复杂度高于传统克隆优化算法, 如何在聚类性能和计算复杂度之间进行折中, 以及考虑在类别数目不确定时的算法聚类性能, 将是下一步研究的重点.

参考文献(References)

[1] Macqueen J B. Some methods for classification and analysis of multivariate observations[C]. Proc of the 5th Symposium on Mathematical Statistics and Probability. Berkeley, 1967, 12: 281-297.

[2] Hall L O, Ozyurt I B, Bezdek J C. Clustering with a genetically optimized approach[J]. IEEE Trans on Evolutionary Computation, 1999, 3(2): 103-112.

[3] Maulik U, Bandyopadhyay S. Genetic algorithm-based clustering technique[J]. Pattern Recognition, 2000, 33(9): 1455-1465.

[4] Pan H, Zhou J, Han D. Genetic algorithms applied to multiclass clustering for gene expression data[J]. Genomics, Proteomics and Bioinformatics, 2003, 1(4): 279-287.

[5] 杜海峰, 公茂果. 适应混沌克隆进化规划算法[J]. 中国科学 E 辑: 信息科学, 2005, 35(8): 817- 829.
(Du H F, Gong M G. Adaptive chaos clonal evolutionary programming algorithm[J]. Science in China Series E: Information Sciences, 2005, 35(8): 817-829.)

[6] Jiao L C, Liu J, Zhong W C. An organizational coevolutionary algorithm for classification[J]. IEEE Trans on Evolutionary Computation, 2006, 10(1): 67-80.)

[7] 刘若辰, 杜海峰, 焦李成. 一种免疫单克隆策略算法[J]. 电子学报, 2004, 32(10): 1880-1884.
(Liu R C, Du H F, Jiao L C. An immune monoclonal strategy algorithm[J]. Acta Electronica Sinica, 2004, 32(10): 1880-1884.)

[8] Gong M G, Jiao L C. Multiobjective immune algorithm with nondominated neighbor-based selection[J]. Evolutionary Computation, 2008, 16(2): 225-255.

[9] Handl J, Knowles J. An evolutionary approach to multiobjective clustering[J]. IEEE Trans on Evolutionary Computation, 2007, 11(1): 56-76.

[10] 陶新民, 刘福荣. 定向多尺度变异克隆选择优化算法[J]. 控制与决策, 2011, 26(2): 175-181.
(Tao X M, Liu F R. Clone selection optimization algorithm with directional multi-scale mutation[J]. Control and Decision, 2011, 26(2): 175-181.)

[11] Geng X, Zhan D C, Zhou Z H. Superivised nonlinear dimensionality reduction for visualization and

classification[J]. IEEE Trans on Systems, Man, and Cybernetics, Part B, 2005, 35(6): 1098-1107.

[12] Gong M G, Jiao L C. Multiobjective optimization using an immuno-dominance and clonal selection inspired algorithm[J]. Science in China Series F: Information Sciences, 2008, 51(8): 1064-1082.

[13] Liu J, Zhong W C, Jiao L C. An organizational coevolutionary algorithm for numerical optimization[J]. IEEE Trans on System, Man, and Cybernetics, Part B, 2007, 37(4): 1052-1064.

[14] Gong M G, Jiao L C. A population-based artificial immune system for numerical optimization[J]. Neurocomputing, 2008, 72(12): 149-161.

[15] Jian L C, Li Y Y, Gong M G. Quantum-inspired immune clonal algorithm for global numerical optimization[J]. IEEE Trans on System, Man, and Cybernetics, Part B, 2008, 38(5): 1234-1253.

[16] 尚荣华, 焦李成, 公茂果. 免疫克隆算法求解动态多目标优化问题[J]. 软件学报, 2007, 18(11): 2700-2711.
(Shang R H, Jiao L C, Gong M G. An immune clonal algorithm for dynamic multi-objective optimization[J]. J of Software, 2007, 18(11): 2700-2711.)

[17] 马文萍, 焦李成. 基于量子克隆优化的SAR图像分类[J]. 电子学报, 2007, 35(12): 2241-2246.
(Ma W P, Jiao L C. SAR image classification based on quantum clonal optimization[J]. Acta Electronica Sinica, 2007, 35(12): 2241-2246.)

[18] 程博, 郭振宇. 一种并行免疫进化策略算法研究[J]. 控制与决策, 2007, 22(12): 1395-1398.
(Cheng B, Guo Z Y. A parallel evolutionary strategy[J]. Control and Decision, 2007, 22(12): 1395-1398.)

[19] 公茂果, 焦李成, 马文萍. 基于流行距离的人工免疫无监督分类与识别算法[J]. 自动化学报, 2008, 34(3): 367-375.
(Gong M G, Jiao L C, Ma W P. Unsupervised classification and recognition using an artificial immune system based on manifold distance[J]. Acta Automatica Sinica, 2008, 34(3): 367-375.)

[20] 马文萍, 尚荣华, 焦李成. 免疫克隆优化聚类技术[J]. 西安电子科技大学学报, 2007, 34(6): 911-921.
(Ma W P, Shang R H, Jiao L C. Immune clone optimization clustering technique[J]. J of Xidian University, 2007, 34(6): 911-921.)

(上接第818页)

[6] 王辉, 钱峰. 基于拥挤度与变异的动态微粒群多目标优化算法[J]. 控制与决策, 2008, 23(11): 1238-1242.
(Wang H, Qian F. Improved PSO-based multi-objective optimization by crowding with mutation and particle swarm optimization dynamic changing[J]. Control and Decision, 2008, 23(11): 1238-1242.)

[7] Coello C A C, Pulido G T, Lechuga M S. Handling multiple objectives with particle swarm optimization[J]. IEEE Trans on Evolutionary Computation, 2004, 8(3): 256-279.

[8] Lechuga M S, Rowe J. Particle swarm optimization and fitness sharing to solve multi-objective optimization problems[C]. IEEE Congress on Evolutionary Computation. Edinburgh: IEEE Press, 2005: 1204-1211.

[9] Kennedy J, Eberhart R C. Particle swarm optimization[C]. Proc of the IEEE Int Conf on Neural Networks. Piscataway, 1995: 1942-1948.

[10] Eberhart R C, Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization[C]. Proc of the Conf on Evolutionary Computation. San Diego, 2000: 84-88.

[11] 张勇, 巩敦卫, 张婉秋. 一种基于单纯形法的改进微粒群优化算法及其收敛性分析[J]. 自动化学报, 2009, 35(3): 289-298.
(Zhang Y, Gong D W, Zhang W Q. A simplex method based improved particle swarm optimization and analysis on its global convergence[J]. Acta Automatica Sinica, 2009, 35(3): 289-298.)

[12] Fan S K S, Zahara E. A hybrid simplex search and particle swarm optimization for unstrained optimization[J]. European J of Operational Research, 2007, 181(2): 527-548.

[13] 陈宝林. 最优化理论与算法[M]. 第2版. 北京: 清华大学出版社, 2005: 281-359.
(Chen B L. Optimization theory and algorithms[M]. 2nd ed. Beijing: Tsinghua University Press, 2005: 281-359.)

[14] Balling R. The maximin fitness function: Multiobjective city and regional planning[C]. Proc of EMO 2003. Faro, 2003: 1-15.

[15] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-Ⅱ[J]. IEEE Trans on Evolutionary Computation, 2002, 6(2): 182-197.

[16] Zitzler E, Deb K, Thiele L. Comparison of multiobjective evolutionary algorithms: Empirical results[J]. Evolutionary Computation, 2000, 8(2): 173-195.