基于并行遗传算法的 K-means 聚类研究*)

戴文华[1] 焦翠珍[1] 何婷婷[2]

(咸宁学院计算机系 湖北咸宁 437100)[1] (华中师范大学计算机科学系 武汉 430079)[2]

摘 要 针对传统 K-means 聚类算法对初始聚类中心的选择敏感,以及聚类数 K 难以确定的问题,提出一种基于并行遗传算法的 K-means 聚类方法。该方法采用一种新型的可变长染色体编码方案,随机选择样本点作为初始聚类中心形成染色体,然后结合 K-means 算法的高效性和并行遗传算法的全局优化能力,通过种群内的遗传、变异和种群间的并行进化、联姻,有效地避免了局部最优解的出现,同时得到了优化的聚类数目和聚类结果。实验表明该方法是一种精确高效的聚类方法。

关键词 并行遗传算法,可变长染色体编码,K-means 算法,聚类

Research of K-means Clustering Method Based on Parallel Genetic Algorithm

DAI Wen-hua[1] JIAO Cui-zhen[1] HE Ting-ting[2]

(Department of Computer, Xianning College, Xianning 437005, China)[1]

(Department of Computer Science, Huazhong Normal University, Wuhan 430079, China)[2]

Abstract As K-means Clustering Algorithm is sensitive to the choice of the initial cluster centers and it's difficult to determine the cluster number, we propose a K-means Clustering Method Based on Parallel Genetic Algorithm. In the method, we adopt a new strategy of Variable-Length Chromosome Encoding and randomly chose initial clustering centers to form chromosomes among samples. Combining the efficiency of K-means Algorithm with the global optimization ability of Parallel Genetic Algorithm, the local optimal solution is avoided and the optimum number and optimum result of cluster are obtained by means of heredity, mutation in the community, and parallel evolution, intermarriage among communities. Experiments indicated that this algorithm is efficient and accurate.

Keywords Parallel genetic algorithm, Variable-length chromosome encoding, K-means algorithm, Clustering

1 引言

聚类是一种有效的数据挖掘方法。典型的聚类方法有多种,其中 K-means 算法[1-3]因其简单和高效性,在聚类中占有重要地位。K-means 算法在聚类中心的计算过程中采用了启发式方法,因而有效地降低了算法复杂度,提高了运算速度。也正是因为同样的原因,使得该算法对初始聚类中心的选择较为敏感,易于陷入局部最优解。

同时,传统的 K-Means 算法是在聚类数 K 确定的前提下进行的。然而,实际聚类问题中 K 值的确定往往非常困难,只能根据经验大致确定。这种估值方法必将带来算法精确度的下降。

为了避免聚类算法对初始聚类中心选择的敏感性和聚类数 K 难于确定的问题,我们提出一种基于并行遗传算法的 K-means 聚类方法。通过该方法,能在达到聚类目的的同时得到经过优化的聚类数目,因此聚类的精确度也将得到极大改善。

2 并行遗传算法

并行遗传算法(Parallel Genetic Algorithm, PGA)[4-6]是一种适用复杂优化问题的多种群并行进化的遗传算法。该算法能有效克服标准遗传算法(Genetic Algorithm, GA)的早熟收敛问题,具有较强的全局搜索能力。

并行遗传算法可分为三种类型:主从式模型、粗粒度模型和细粒度模型。其中粗粒度模型易于实现,既能在多处理机系统中运行,也能单机模拟,算法效率也较高,是适应性最强、应用最广的并行遗传算法模型。本文采用粗粒度模型。

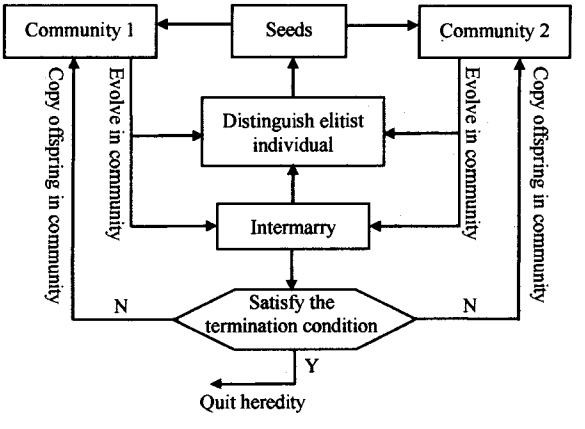

图 1 两种群联姻并行遗传模型

在进行聚类分析时,根据实际情况可选择基于"联姻"策略的并行遗传算法。这种并行遗传算法模仿人类的联姻策略,尽可能防止具有相同基因结构的个体进行交配,以避免算法的早熟。该算法以 $M(M\geqslant 2)$ 个子种群并行进化,当种群之

*)国家自然科学基金(No. 60442005,No. 60673040);国家社会科学基金(No. 06BYY029);教育部重点研究项目(No. 105117);湖北省教育厅科研重点项目(No. D200728002)。**戴文华** 副教授,硕士。

间满足联姻条件时，不同种群之间的当代精英个体两两联姻，并将联姻后代中的精英个体复制到相关的源种群。

在遗传过程中，为了保留优良基因，我们采用精英个体保留策略，将联姻后代和源种群中的精英个体进行比较，保留优者，作为种子参与下一代遗传。具体模型如图 1 所示(以两种群联姻并行遗传为例)。

3 基于并行遗传算法的 K-means 聚类方法

虽然并行遗传算法能有效地避免早熟，加快了系统的速度，但是其局部搜索能力仍然没能得到充分解决。而 K-means 算法是一种快速高效的聚类方法，具有较强的局部搜索能力，然而该算法对初始聚类中心的选择较为敏感，同时聚类数 K 的选择也较为困难。

通过分析上述两种算法，我们考虑使用并行遗传算法对 K-means 算法的初始聚类中心和聚类数进行动态优化，从而得到一种高精度的聚类方法——基于并行遗传算法的 K-means 聚类(PGAKClust)方法。具体模型如图 2 所示(以两种群并行遗传的 K-means 聚类为例)。

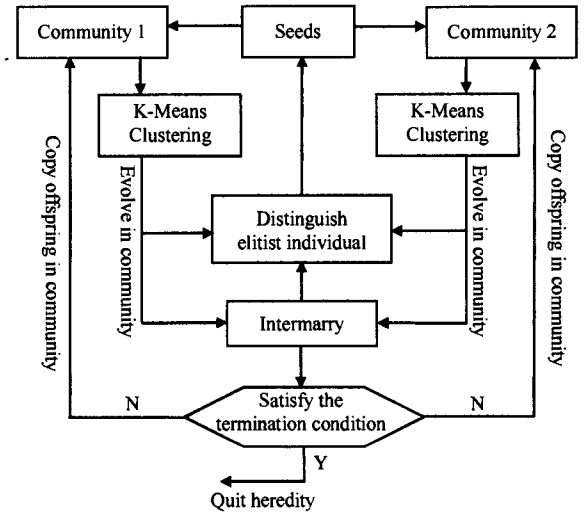

图 2 基于并行遗传算法的 K-means 聚类模型

将并行遗传算法应用于 K-means 聚类时，必须考虑到在算法实现过程中，编码方案、适应度函数、遗传算子和种群的初始化等都是影响算法效率的非常关键的因素。下面将就这些问题进行讨论。

3.1 可变长染色体编码方案

在聚类问题中，由于聚类中心数难以确定，只能凭经验设置，这种凭经验确定的聚类中心数会对聚类结果产生偏差，因此我们采用并行遗传算法以动态方式来确定聚类中心数，相应的染色体采用可变长染色体编码方案[7]。

可变长染色体编码有两种编码方式：一种方式是染色体的基因由初始聚类中心对应的样本点在样本集中的编号表示，我们称这种染色体编码为 PGAK-I 型染色体。另一种方式是染色体的基因由初始聚类中心对应的样本点数据直接表示，我们称这种染色体编码为 PGAK-Ⅱ型染色体。

(1)PGAK-Ⅰ型染色体

其编码形式为

$$CH=\{ch_1,ch_2,\cdots,ch_t\} \tag{1}$$

其中 t 为某条染色体的编码长度，对不同的染色体，t 的值是在变化的。$ch_i(i=1,2,\cdots,t)$ 为第 i 个聚类中心对应的样本在样本集中的编号，为一个[1，N]之间的自然数(N 为样本个数)。

(2) PGAK-Ⅱ型染色体

其编码形式为

$$EH=\{eh_1,eh_2,\cdots,eh_t\} \tag{2}$$

其中 t 为某条染色体的编码长度，对不同的染色体，t 的值是在变化的。$eh_i(i=1,2,\cdots,t)$ 为第 i 个聚类中心对应的样本点数据，由 n 维数据组成(n 为样本数据维数)。

3.2 适应度函数

由于染色体采用可变长编码，因此聚类中心的个数并不固定，适应度函数与定长编码时的适应度函数有所区别。具体定义如下：

$$Fit(Ind)=\frac{1}{1+\sum_{j=1}^{len(Ind)}\sum_{X_i\in C_j}Dis(X_i,Z_j)} \tag{3}$$

其中 $Len(Ind)$ 为个体 Ind 的染色体长度。公式的含义是：计算各类中的样本到该类中心的距离，并求这些距离之和，得到各类的适应度。所有类的适应度之和加上 1 并求倒数，得到染色体 Ind 的适应度。

样本间的距离采用公式(4)进行计算：

$$Dis(S_i,S_j)=\|S_i-S_j\|=\sqrt{\sum_{k=1}^{n}(x_{ik}-x_{jk})^2} \tag{4}$$

其中 x_{ik}，x_{jk} 分别表示样本 S_i 和 S_j 的第 k 维的值，n 为样本数据维数。

3.3 插入删除交叉算子

针对可变长染色体编码，我们特意设计了插入删除交叉算子，以适应遗传进化过程中染色体长度的变化。

插入删除交叉算子的主要思想是：将一个染色体的一段基因删除，并将这段基因插入另一个染色体的某一位置。

(1)PGAK-Ⅰ型染色体插入删除交叉算子

PGAK-Ⅰ型染色体插入删除交叉算子操作过程可用图 3 表示。具体操作步骤如下：

① 以 CH_1 作为被删除染色体，以 CH_2 作为被插入染色体，计算染色体 CH_1 和 CH_2 的长度 t_1 和 t_2；

② 如果 $t_2=\lceil\sqrt{N}\rceil$，则重新选择染色体 CH_2，直到 $t_2<\lceil\sqrt{N}\rceil$；

其中 N 为待聚类样本数，$\lceil\sqrt{N}\rceil$ 为聚类个数的经验值[8]，其实设置该值的目的只是为了加快算法速度，如果对算法精度要求较高，则可适当放宽该值的尺度。要求 $t_2\lhd\lceil\sqrt{N}\rceil$ 是为了防止插入操作后染色体 CH_2 的基因由于超长截断而无变化。

③ 随机生成插入点位置 Ins、删除点位置 Del 和插入(或删除)长度 $DLen$；

其中插入长度与删除长度相等，均为 $DLen$。要求满足如下条件：

$0\leqslant Del<t_1$，$0\leqslant Ins\leqslant t_2$ 且 $Len<t_1$

④ 将染色体 CH_1 从删除点开始，将长度 $DLen$ 的基因段删除，并将该基因段插入染色体 CH_2 中；

⑤ 将染色体 CH_2 中的重复基因去除；

⑥ 如果染色体 CH_2 的长度超长，则对其进行截尾操作。

(2)PGAK-Ⅱ型染色体插入删除交叉算子

PGAK-Ⅱ型染色体插入删除交叉算子操作过程可用图 4 表示。

从图中可以看出，PGAK-Ⅱ型染色体和 PGAK-Ⅰ型染色体插入删除操作步骤几乎一样，只是前者操作的是聚类中心

对应的样本点数据，而后者操作的是聚类中心对应的样本点编号。

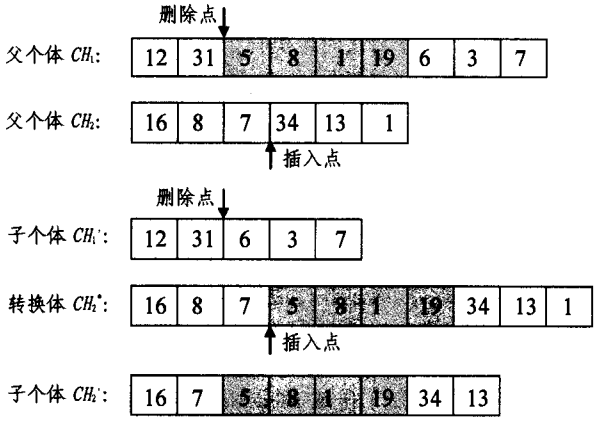

图 3 PGAK-Ⅰ型染色体插入删除操作示意图

经过插入删除操作后，染色体长度将发生变化，这种变化显然保证了进化过程中染色体的多样性，同时聚类数目也在进化过程中得到了动态改变，有利于遗传算法的优化和搜索。

针对染色体长度的动态变化，以及插入删除操作的频繁性，我们采用动态链表的方式存储染色体。这种存储方式插入删除操作快速，且具有链表长度的可动态变化性。

3.4 变异算子

(1)PGAK-Ⅰ型染色体变异算子

PGAK-Ⅰ型染色体的变异操作步骤如下：

① 计算染色体长度 *Len*；

② 随机产生一个[1，*Len*]之间的自然数 *C*，作为变异点个数；

③ $c=1$；

④ 随机产生一个与上一轮不重复的[1，*Len*]之间的自然数，作为变异点；

⑤ 随机产生一个[0，1]之间的数 r，如果 $r \leqslant P_m$（P_m 为变异概率），则转⑥，否则直接转⑦；

⑥ 随机产生一个[1，N]之间的在染色体中不存在的自然数，将父个体在变异点处的基因用这个自然数取代；

⑦ $c=c+1$；

⑧ 如果 $c>C$，退出变异，否则转④。

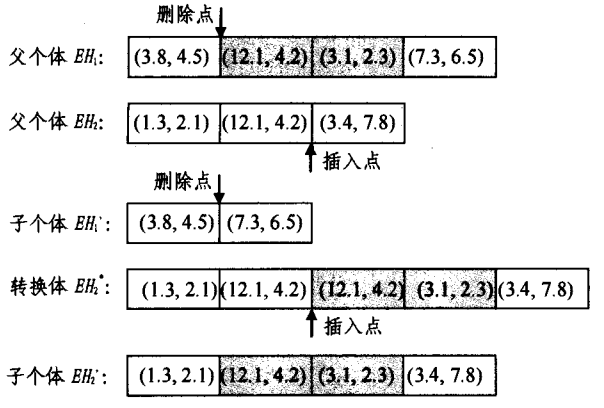

图 4 PGAK-Ⅱ型染色体插入删除操作示意图

(2)PGAK-Ⅱ型染色体变异算子

PGAK-Ⅱ型染色体的变异操作与 PGAK-Ⅰ型染色体变异操作几乎相同，只是在步骤⑥不用生成的自然数取代染色体基因，而是用这个自然数作为样本编号，将该样本编号对应的样本点数据代替染色体基因。

3.5 种群初始化

由于染色体体长度可变，因此其种群初始化方法具有自身的一些特点。

(1)PGAK-Ⅰ型染色体的种群初始化

PGAK-Ⅰ型染色体的种群初始化具体步骤如下：

① 设置种群规模 *Gsize*；

② $I=1$；

③ 如果 $I \leqslant Gsize$，则转④，否则结束初始化；

④ 随机设置染色体长度 $Len(Len \leqslant \lceil \sqrt{N} \rceil)$；

⑤ 随机产生 *Len* 个[1，N]之间的不重复的自然数，形成一条染色体 *Ind*；

⑥ 判断染色体 *Ind* 是否已经在种群中存在，如果存在则转④，否则转⑦；

⑦ $I=I+1$；

⑧ 转③。

(2)PGAK-Ⅱ型染色体的种群初始化

PGAK-Ⅱ型染色体的种群初始化也与 PGAK-Ⅰ型染色体的种群初始化操作几乎相同，只是在步骤⑤不用生成的一组自然数组成一条染色体，而是用这组自然数作为样本编号，将这些样本编号对应的样本点数据代替各基因点的染色体基因，形成一条染色体。

4 评价标准

为了评价聚类结果，我们采用的评价指标为平均准确率[9]，其计算公式如下：

$$aa=(pa+na)/2 \tag{5}$$

其中 na，pa 分别称为消极准确率和积极准确率，计算公式如下：

$$na=d/(b+d),\ pa=a/(a+c) \tag{6}$$

任意两样本之间的关系，按照手工分类的标准和自动聚类的标准可以有表 1 中列出的 4 种情况。

公式(6)中 a，b，c，d 的计算方法为：如果聚类结果属于第一种情况，则将 a 加 1；如果属于第二种情况，则将 b 加 1；如果属于第三种情况，则将 c 加 1；如果属于第四种情况，则将 d 加 1。

表 1 样本间的类属关系

情况	自动聚类中属于同一类	手工分类中属于同一类
1	是	是
2	是	否
3	否	是
4	否	否

5 实验设置及结果分析

为了验证本文所提聚类算法的可行性，我们进行了多项实验。实验平台为 Windows XP，使用 Visual C++6.0 进行开发，以多线程的方式模拟并行计算。实验参数设置为：并行种群数 $M=2$，种群规模 $Gsize=100$，最大进化代数 $Gnum=100$ 代，交叉概率 $P_c=0.86$，变异概率 $P_m=0.02$，精英个体数 $Elite=4$。

5.1 样本数对可变长染色体编码的并行遗传算法的影响

为了比较样本数量对 PGAK-Ⅰ型和 PGAK-Ⅱ型染色体并行遗传效率的影响，我们在 20 个二维数据点周围分别产生正态分布的大量样本点，并在这些样本点中随机抽取样本组成多个随机样本点集合，采用 PGAK-Ⅰ型和 PGAK-Ⅱ型染色体并行遗传算法分别对这些样本集进行聚类，实验所得聚类时间与样本数之间的关系如图 5 所示。

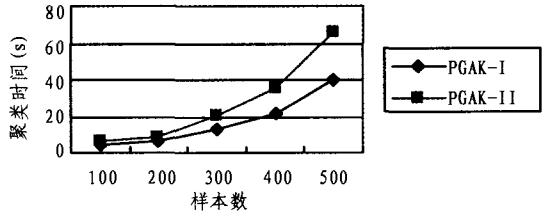

图 5　样本对可变长染色体编码的并行遗传算法的影响

从图 5 可以看出，样本数对 PGAK-Ⅰ型染色体的影响远小于 PGAK-Ⅱ型染色体，PGAK-Ⅰ型染色体在性能上要优于 PGAK-Ⅱ型染色体。这是因为 PGAK-Ⅱ型染色体中的基因是由聚类中心对应的样本点数据组成，而 PGAK-Ⅰ型染色体中的基因则只是由聚类中心对应的样本点编号组成，而且在遗传过程中，染色体长度在动态变化，在样本数增多的时候，染色体长度也会加长。因此 PGAK-Ⅱ型染色体占用的内存空间必然比 PGAK-Ⅰ型染色体大。随着样本数的增加，这种差别将越来越大，具体的性能差别当然会表现得较为明显。

5.2　算法性能测试

为了验证本文所提出的基于并行遗传算法的 K-means 聚类(PGAKClust)方法的实际性能，我们随机生成 12 个二维样本集，分别使用 K-Means 算法和本文提出的 PGAKClust 聚类方法对上述 12 个样本集进行聚类(采用 PGAK-Ⅰ型染色体编码)。聚类结果如表 2 所示。

表 2　算法性能测试结果

样本集	样本	实际	K-Means 聚类平均准确率			PGAKClust 聚类结果	
编号	数	类别数	K=5	K=10	K=15	聚类数	平均准确率
1	100	4	62%	54%	41%	4	94.8%
2	100	6	61%	57%	43%	6	94.6%
3	100	8	54%	59%	48%	8	94.5%
4	100	10	51%	66%	52%	10	94.6%
5	200	8	53%	57%	46%	8	94.1%
6	200	10	50%	64%	49%	10	94.0%
7	200	12	47%	56%	54%	12	93.8%
8	200	14	41%	51%	59%	14	93.7%
9	500	10	48%	61%	47%	10	93.0%
10	500	12	44%	54%	51%	12	93.1%
11	500	14	38%	49%	55%	14	92.9%
12	500	16	31%	42%	56%	16	92.8%

从表 2 可以发现：对于 K-Means 算法，由于聚类前无法获知样本类别数，只能通过大致估计类别数来进行聚类运算，当估值的聚类数目与实际类别数差别较大时，聚类精确度将显著下降；同时，由于算法对聚类初始中心选取的敏感性，使得聚类结果精确度相对较低；此外，随着样本数的增加和样本类别数的增加，聚类精确度也会明显下降。

对于本文所提出的 PGAKClust 聚类方法，由于采用可变长染色体编码的并行遗传算法，通过进化计算，能动态挖掘样本聚类数目，同时优化了初始聚类中心的选择，因此在避免聚类数盲目估计的同时能获取最优初始聚类中心，从而保证了聚类的精确度；此外，通过并行遗传算法的合理优化，使得样本数和样本类别数对聚类结果的影响得以有效降低。

结束语　本文通过提出一种基于并行遗传算法的 K-means 聚类方法，克服了传统 K-means 算法对初始聚类中心选择的敏感，以及聚类数 K 难以确定的问题，充分发挥了 K-means 算法的高效性和并行遗传算法的全局优化能力。有效地均衡了算法对聚类空间的探索和开发能力，实验证明该算法是一种高精度的聚类方法。

参 考 文 献

[1] Larsen B, Aone C. Fast and effective text mining using linear-time document clustering[A]// Proc. of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, August 1999:16-22

[2] MacQueen J B. Some methods for classification and analysis of multivariate observations// Proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probability. 1967:281-297

[3] Zhao Y, Karypis G. Criterion Functions for Document Clustering: Experiments and Analysis[R]. Technical Report. # 01-04. Department of Computer Science, University of Minnesota, 2001

[4] Steinbach M, Karypis G, Kumar V. A comparison of Document Clustering Techniques[R]. Department of Comp Sci & Eng University of Minnesota, 2000:1-20

[5] Salton G, Wang A, Yang C S. A vector space model for automatic indexing[J]. Communication of the ACM, 1975, 18(11): 613-620

[6] Mühlenbein H. Evolution in time and space-the parallel genetic algorithm . In Rawlins, Foundations of Genetic Algorithms: Morgan Kaufmann, 1991

[7] Hung S L, Adeli H. A parallel genetic/neural network learning algorithm for MIMD shared memory machines. IEEE Transactions on Neural Networks, 1994, 5(6):900-908

[8] Liu Juan, Iba H. Selecting informative genes with parallel GA in tissue classification. Genome Informatics, 2001, 3(12): 14-23

[9] Goldberg D E, Deb K, Korb B. Don't Worry Be Messy// Proc. of ICGA. 1991:24-30

[10] Ramze R M, Lelieveldt B P F, Reiber J H C. A new cluster validity indexes for the fuzzy c-mean[J]. Pattern Recognition Letters, 1998, 19:237-246

[11] Makoto I, Takenobu T. Hierarchical Bayesian clustering for automatic text classification[R]. Tech Rep. TR95-0015. Department of Computer Science Tokyo Institute of Technology, 1995

更　正

我刊 2008 年 Vol. 35No. 3“基于免疫重构的阴性选择算法”一文中，图 1 更正如右图。由此给读者带来的不便，表示歉意。

《计算机科学》编辑部

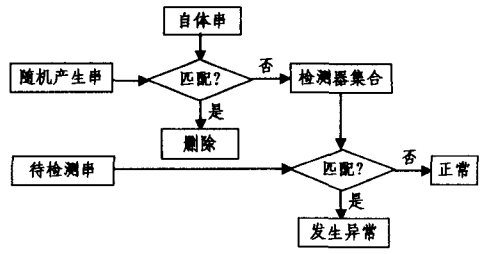

图 1　阴性选择算法流程图