

基于演化算法的带故障约束空间聚类分析

王媛妮^{1,2} 边馥苓¹

(武汉大学空间信息与数字工程研究中心 武汉 430079)¹ (中国地质大学计算机学院 武汉 430074)²

摘 要 现实世界当中的各种约束条件限制了空间聚类必须考虑这些限制条件的存在。主要研究带障碍物的空间聚类,采用 K-中心点算法进行聚类分析,在解决空间对象绕过障碍物的最短距离时引进改进的郭涛算法进行求解,对于中小规模数据体现了较高的执行效率。通过理论分析和实验验证,该算法是可行的。

关键词 空间聚类,障碍约束,演化算法

Clustering Based on Evolutionary Algorithm in the Presence of Obstacles

WANG Yuan-ni^{1,2} BIAN Fu-ling¹

(Research Center of Spatial Information & Digital Engineering, Wuhan University, Wuhan 430079, China)¹

(College of Computer, China University of Geosciences, Wuhan 430074, China)²

Abstract In the real-world, constraints limits the spatial clustering must take into account the conditions of these restrictions, this paper studied the spatial clustering with obstacles. It mainly used the K-medoid algorithm to cluster, and it introduced an improved algorithm Guo Tao to solve the distance of spatial objects in the presence of obstacles. It is higher efficiency for small and medium-sized data. Through theoretical analysis and experimental, the algorithm is feasible.

Keywords Spatial clustering, Obstacle constrain, Evolutionary algorithm

1 引言

空间聚类是指将数据对象集分为由类似的对象组成的簇,使得在同一簇中的对象之间具有较高的相似度,而不同簇中的对象差别较大,即相异度较大。由于空间数据库包含的大量数据与空间有关,并且数据来自不同的应用领域,比如土地利用、居住类型的空间分布、商业区位分布等,因此,根据空间数据库中的数据,运用空间聚类来提取不同领域的分布特征,是空间数据挖掘的一个重要部分。

空间聚类方法通常可以分为 4 大类:划分法、层次法、基于密度的方法和基于网格的方法。然而,在实际的应用中,河流、桥梁等障碍物的存在破坏了空间的连续性,常常影响聚类结果的应用,常常需要考虑各种约束条件才能使聚类的结果更加实用,这就是障碍约束下的空间聚类。

针对障碍约束下的聚类问题,主要是在聚类算法基础上增加障碍存在情况进行处理。COD-CLARANS^[1] 最早提出基于障碍约束的聚类算法,该算法以 CLARANS 聚类算法为基础,采用剪枝策略实现含障碍物数据的聚类,但算法的整体效率较低;AUTOCLUST+^[2] 将 Delaunay 图引入到聚类算法,它是在 AUTOCLUST 算法的基础上进行改进,聚类结果比较好,但其空间复杂度较高,在处理多维数据时尤甚;DB-CLUC^[3] 以 DBSCAN 算法为基础,将障碍物模型化为一组多边形,该算法的不足是预处理代价大, I/O 消耗较高,对于大

型数据库和高维数据库较难处理,另外虽然处理凸多边形时效果良好,但是对于凹多边形就比较麻烦,需要增加计算量和代码难度;DBRS+^[4],其算法以 DBRS 算法为基础,它对障碍物的处理极为细致,可以处理连接型障碍约束等多种形式,能够处理大型数据库,但也具有预处理运算量大等不足。

针对上述算法存在的各种问题,本文结合演化算法和 K-中心点的方法,在带障碍约束的 K 中心点算法的基础上设计了一种新的障碍约束下的聚类算法,该方法不仅兼顾了局部收敛和全局收敛性能,又充分考虑到了现实障碍物对聚类结果的影响。理论分析和实验证明该算法效果良好。

2 带障碍约束的 K 中心空间聚类

K 中心(K-Medoids)算法是一种典型的划分算法,与其它两种划分算法 K 平均(K-Means)算法和 CLARANS 算法相比,在实际应用中有其优势。K 平均算法以簇中对象的平均值作为中心,采用这种算法时,实际应用中簇中心可能刚好落在障碍物上,此时实施是不切实际的。而 K 中心算法是选择簇中的一个对象作为中心点,簇中心不会落在障碍物上。另外 CLARANS 算法主要处理较大数据集, K 中心算法对小数据的集合非常有效。

2.1 传统 K 中心算法

K 中心算法基本策略:首先为每个簇随意选择一个代表对象;剩余的对象根据其代表对象的距离分配给最近的一

到稿日期:2009-03-14 返修日期:2009-06-01

王媛妮(1980—)女,博士生,讲师,研究方向为空间数据挖掘、地理信息系统, E-mail: ynwang2005@163.com; 边馥苓(1941—),女,教授,博士生导师,研究方向为地理信息系统等。

个簇;然后反复地用非代表对象来代替代表对象,以改进聚类的质量,聚类结果质量用一个代价函数来估算。该函数用来度量对象与其参照对象之间的平均相异度。主要采用平方误差准则:

$$E = \sum_{i=1}^k \sum_{p \in C_i} |p - O_i|^2$$

其中, E 是所有对象的平方误差总和, p 是空间中给定的数据对象, O_i 是簇 C_i 的中心点。

其算法描述如下:

- (1)随机选择 k 个对象作为初始的中心点;
- (2)repeat;
- (3)指派每个剩余的对象给离它最近的中心点所代表的簇;
- (4)随机地选择一个非中心点对象 Orandom;
- (5)计算用 Orandom 代替 O_j 的总代价 S ;
- (6)if $S < 0$, then Orandom 替换 O_j , 形成新的 k 个中心点的集合;
- (7)until 不发生变化。

2.2 带障碍约束的 K 中心空间聚类算法

K 中心算法中的两点距离一般采用两点直线距离,即没有考虑有障碍物的情况,实际中这种做法得到的结果并不合理。因此,采用障碍距离代替直线距离,即将上述准则中的欧氏距离改为计算障碍距离。

假设两个数据对象为 a, b , 计算 a 到 b 的距离分两种情况:

- (1) a 与 b 之间无障碍物,其距离 $d(a, b)$ 即为两点之间的直线距离。
- (2) a 与 b 之间存在障碍物,其距离 $d(a, b) = \min(d(a, s), b)$, 即 a 绕过障碍物到 b 的最短距离, s 为 a 到 b 障碍物可见部分顶点的集合。

3 基于演化算法的带障碍约束空间聚类

采用带障碍约束的 K 中心空间聚类算法进行空间聚类存在着这样一个问题,即对数据对象 a, b 之间最短距离的确定。考虑到中小规模数据时算法的执行效率,本文主要采用演化算法中的郭涛算法来解决这个问题。

3.1 GT 算法

郭涛算法(GT)是武汉大学郭涛博士 1996 年提出的一种演化算法,主要用来求解 TSP 问题。该算法的特点是一个体只与自己的儿子竞争,只有一个自适应的演化算子,演化过程是一个并行爬山的过程。该算法保证了群体的多样性,同时也保证了适应性最好的个体可以一直存活下来,最后找到最优解。

3.2 基于郭涛算法求最短避障路径

针对郭涛算法求解 TSP 问题的特点,将其改进应用到求解两个数据对象 a, b 之间的最短避障路径问题,从而解决带障碍空间聚类问题中的空间对象最短路径问题。主要体现在两个方面:

- (1)代价矩阵的设计,即在遗传算法染色体的计算中点 i 到其他各个点的距离的表示。如果 a, b 两点在同一个障碍多边形,而且相邻;或者两点不在同一多边形,且两点的连线线段与所有多边形不包括这两点的所有边所在的线段没有交

点,则用两点距离公式计算 $d(a, b)$, 否则设定 $d(a, b) = \text{MAX}$, MAX 是一个远大于两点距离数量级的常数,表示 a, b 两点连线穿过了障碍物而不能直接相连。如果在问题求解中出现这两点相邻,则该路径为非法路径。

(2)TSP 问题的求解是寻找经过所有的点的最短路径,本文求解的空间对象的最短避障路径不需要经过所有的点,只要能够找到最短的通路即可。如果利用郭涛算法进行求解空间对象的最短避障路径,适应度在计算时只需考察这两对象之间的一段距离的倒数的倒数。适应度的评价函数可定义为:

$$F = \frac{1}{\sum_{i=a}^{b-1} d(i, i+1)}$$

其中, a, b 表示起点和终点在染色体中的位置。在 TSP 问题中,染色体的适应度与染色体各段距离的和成反比,但是最短避障路径中一个染色体的适应度其实就是起点和终点之间的一段有效染色体的适应度。

利用改进的郭涛算法求最短避障路径的步骤如下^[6]:

- (1)计算所有点的代价矩阵,得到邻接矩阵;
- (2)随机生成种群 $P = \{S_0, S_1, \dots, S_{N-1}\}$, 其中每一条染色体是一条遍历所有点的随机路径,利用适应度评价函数公式计算每条染色体适应度,初始化各参数:变异算子、映射算子;
- (3)初始化 $i=0$;
- (4)将种群 P 中的染色体 S_i 赋值给空染色体 S' , 找到起点和终点的位置 a, b , 从 S' 的 a 和 b 之间的点中随机选择一个基因点 C ;
- (5)变异算子。如果 $0 \sim 1$ 之间的随机数小于变异概率 P_i , 在 S' 中指定与 C 相邻的下一个基因位 C' , 从 S' 中任意选择一不同于 C' 的基因 C'' , 并逆转 C' 和 C'' 之间的基因片断, 转至(7);
- (6)杂交算子。从 C 的基因库中随机指定一个基因作为 C' , 判断:如果 S' 中 C' 的位置与 C 相邻, 转至(8), 将 S' 中从 C' 到 C'' 之间的基因片断进行倒位操作。计算逆转前后路径改变值 d , 如果小于 0 且演化速度小于临界速度, 转至(8);
- (7)将 C' 赋值给 C , 重复执行(5)开始的操作, 直到 C 已经取过 S' 中 a 和 b 之间的每一个基因位;
- (8)计算 S' 的适应度, 判断:如果 $F(S') > F_i$, 将 S' 替换掉 S_i , F_i 赋新值 $F(S')$;
- (9) i 增 1, 重复执行(4)开始的操作直到 $i=N$;
- (10)更新变异概率;
- (11)优化算子。选择目前最优个体中 a 和 b 之间的若干个基因片断, 分别判断其余个体中是否有选取的点相同、基因长度相同、片断适应值更高的片断, 如果有, 则替换掉最优个体中相应的基因片断;
- (12)重复执行(3)~(11), 直到满足结束条件, 则输出最优染色体。

3.3 基于改进 GT 算法的带障碍约束空间聚类

针对中小规模数据, K 中心点算法非常有效, 在此基础上, 考虑实际中的障碍物的存在, 在聚类过程中, 计算两个对象之间的距离问题就转换成两点之间的避障最短路径问题, 改进的 GT 算法在求解时比改进的 Dijkstra 算法效率更

(下转第 222 页)

- [3] Fitting M C, Mendelsohn R. First-order modal logic[M]. Kluwer, 1998
- [4] Fitting M C. On Quantified Modal Logic[J]. Fundamenta Informaticae, 1999, 39(1/2): 105-121
- [5] Fitting M C. First-order intensional logic[J]. Annals of Pure and Applied Logic, 2004, 127: 171-193
- [6] Fitting M C. FOIL Axiomatized. Unpublished Manuscript[OL]. <http://comet.lehman.cuny.edu/fitting/bookpapers/modal.html>, 2006
- [7] Garson J W. Quantification in modal logic[M]. Handbook of Philosophical Logic. Dordrecht: Kluwer Academic Publishers, 2001: 267-323
- [8] Hilken B P. Sheaf models of predicate intuitionistic modal logic I: Relational sheaves[D]. In preparation: Department of Computer Science, Manchester University, 1999
- [9] Hughes G E, Cresswell M J. An New Introduction to Modal Logic[M]. London: Routledge, 1996
- [10] Kripke S. Semantical considerations on modal logics[J]. Acta Philosophica Fennica, Modal and Many-valued Logics, 1963, 16:

83-94

- [11] Lewis D. Counterpart theory and quantified modal logic[J]. Journal of Philosophy, 1968, 65: 113-126
- [12] McKay T. Propositional attitude reports[J]. Stanford Encyclopedia of Philosophy, 2000
- [13] 弓肇祥. 可能世界理论[M]. 北京: 北京大学出版社, 2003
- [14] 江峰. 基于本体的一阶模态逻辑研究[D]. 北京: 中科院计算所, 2007
- [15] 王震宇. 可能世界与智慧[M]. 上海: 机械工业出版社, 2005
- [16] 周北海. 模态逻辑导论[M]. 北京: 北京大学出版社, 1997
- [17] 周礼全. 模态逻辑引论[M]. 上海: 上海人民出版社, 1986
- [18] Guarino N. The Ontological Level[M]//Casati R, et al., eds. Philosophy and the Cognitive Science, Holder-Pichler-Tempsky, Vienna, 1994: 443-456
- [19] Gruber T. Ontolingua: a Translation Approach to Portable Ontology specifications[J]. Journal of Knowledge Acquisition, 1993, 5(2): 199-200
- [20] 顾芳. 多学科领域本体设计方法的研究[D]. 北京: 中科院计算所, 2004

(上接第 198 页)

高^[6], 为此, 本文将改进的 GT 算法应用到 K 中心点聚类算法中, 其算法流程如图 1 所示。

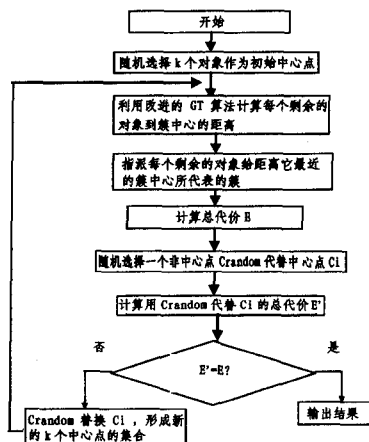


图 1 改进 GT 算法 K 中心点空间聚类流程图

4 实验结果与分析

本文实验选用了 400 个随机正态分布二维数据, 设置了线形障碍物, 假设聚类中心为 4, 最大迭代次数设置为 10000, 实验分析了在无障碍约束和有障碍约束情况下的空间聚类结果。实验结果表明, 障碍物的存在影响了数据空间的连续性, 直接导致了聚类结果的分布不同。

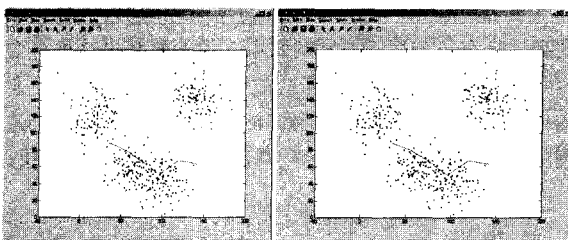


图 2 无障碍约束空间聚类

图 3 障碍约束空间聚类

图 2 的聚类结果并没有考虑障碍物的存在, 图 3 采用考虑障碍物的一般聚类思想, 图 4 采用本文算法思想进行聚类。对比图 2 和图 3 的聚类结果, 可以看出, 图 3 和图 4 较好地反

映了空间障碍对聚类的影响, 聚类结果更有实际意义。

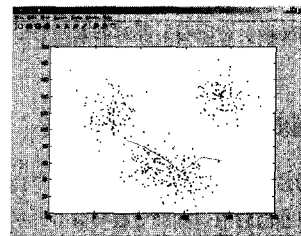


图 4 基于演化计算的障碍约束空间聚类

结束语 本文在 K 中心点空间聚类的基础上引入改进的 GT 算法求解带障碍物的数据对象点间距离, 执行结果与实际相符合, 对于中小规模数据体现了较高的执行效率。聚类过程考虑到了现实障碍物对聚类结果的影响, 使得聚类结果更具有实际意义。理论和实验分析说明此算法是可行的。

参考文献

- [1] Tung AKH, Hou J, Han J. Spatial clustering in the presence of obstacles[A]//Proceeding of International Conference on Data Engineering[C]. Heidelberg: IEEE Computer Society, 2001: 359-367
- [2] Estivill - Castro V, Lee I J. AUTOCLUST+: Automatic clustering of point-data sets in the presence of obstacles[A]//Proceeding of International Workshop on Temporal, Spatial and Spatial temporal Data Mining [C]. Berlin: Springer-Verlag, 2000: 133-146
- [3] Zaiane O R, Lee C H. Clustering spatial data when facing physical constraints [A]//Proceeding of the IEEE International Conference on Data Mining[C]. Los Alamitos: IEEE Computer Society, 2002: 737-740
- [4] Wang X, Rostoker C, Hamilton H J. Density-based spatial clustering in the presence of obstacles and facilitators[A]//Proceeding of the 8th European International Conference on Principles and Practice of Knowledge Discovery in Databases [C]. Berlin: Springer-Verlag, 2004: 446-458
- [5] 郭涛. 演化计算与优化[D]. 武汉: 武汉大学, 1999
- [6] 李庆华, 戴光明, 弓晨. 基于演化计算的最短避障路径算法设计[J]. 小型微型计算机系统, 2005, 26(3)