

多对象 RkNN 查询在空间数据库中的应用

刘金岭¹, 杨凤霞², 刘国香²

(1 淮阴工学院 计算机工程学院, 江苏 淮安 223003; 2 沧州师范学院 计算机系, 河北 沧州 061001)

摘 要: 针对空间数据库的 RkNN 查询概念和多个对象的查询特点设计了相关算法. 对于一组查询 Q , 先是生成查询 Q 的最小圆域 $DOM(O, R)$, 然后求出 RkNN 查询的初始化区域, 最后通过消除查询错误数据点得到 RkNN 查询结果. 通过对真实数据集进行的大量实验表明, 文中提出的算法效率明显优于目前最好的反 RkNN 算法.

关键词: RkNN 查询; 空间数据库; 算法

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2012)01-0018-05

RkNN of a Group of Object are Queried in the Application of Spatial Database

LIU Jin-ling¹, YANG Feng-xia², LIU Guo-xiang²

(1 Computer Engineering Faculty, Huaiyin Institute of Technology, Huaian 223003, China;

2 Department of Computer, Cangzhou Teachers College, Cangzhou 061001, China)

Abstract: According to the concept of RkNN in the spatial database and the features of RkNN of a Group of object are queried and design the corresponding algorithm. Given a group of nearby space objects as the query input, the author compute the smallest circle enclosing the the query objects, secondly compute the regional of initialization are queried. In the end the author eliminate wrong data to get the query's final RkNN query results. A large of experiments according to real data sets, show that the efficiency of the proposed method for RkNN outperform the state-of-the-art algorithm.

Key words: RkNN query; spatial database; algorithm

1 引言

在空间信息科学发展的过程中,用到了大量的地理信息等方面的系统,同时,地理信息系统在空间科学的发展中又占有非常重要地位,而对给定对象求出 RkNN (Reverse k-Nearest Neighbor) 查询结果在地理信息系统中有着广泛的应用. 而在气象预报、现代通讯、交通控制、智能导航等各个领域都对 RkNN 查询又有着广泛的需求. 空间数据库 (Spatial Databases) 就是在这些空间数据的有效处理中应运而生的. 空间数据库的处理的对象数据结构很复杂,数据对象的描述既有一些几何信息又包含着一些非空间属性,其空间数据对象常常是动态变化的,管理

难度很大. 所以对于空间数据库的各种搜索还存在许多困难. RkNN 查询是对空间数据库一种重要操作^[1-2],通常用来发现查询对象的影响集. 例如,出租车司机可以根据车辆中 GPS 导航系统随时检索他的 RkNN 查询对象, RkNN 查询对象分为双色 RkNN (Bichromatic RNN) 查询和单色 RkNN (Monochromatic RNN) 查询^[1],在双色 RkNN 查询中,对于不属于同一数据集的两个子集 A 和 B ,若子集 B 中的点 o_2 与子集 A 中的查询点 q_1 满足关系 $\text{dist}(o_2, q_1) < \text{dist}(o_2, o'_1)$, 则 o_2 是查询点 q_1 的 RkNN 对象, o'_1 为 A 中不同于查询点 q_1 的对象. 单色 RkNN 查询要求数据集 S 和查询点 q 均属于同一数据集,若数据集 S 中的对象 o 满足 $\text{dist}(o, o') >$

收稿日期: 2011-05-05; **修回日期:** 2011-06-30

基金项目: 河北省科技支撑计划项目 (10213581); 淮安科技计划项目 (HAG09061)

$\text{dist}(o, q)$, 则查询点 q 的 RkNN 对象为 o , 在这里 o' 与 o 是数据集 S 中不相同的对象. 本文只讨论 RkNN 的单色查询. 由于目前 RkNN 单色查询的解决方法都是基于单个点处理的^[3-4], 其实在实际应用中常常需要查询一组对象的 RkNN 对象及评价它们之间的相互影响, 如出租车之间、商业区中的空间对象、多个连锁店等, 它们之间都存在着相互的影响. 如果对这些空间对象进行单个 RkNN 查询, 再取各个查询结果的并集作为结果集, 由于组内对象较多, 相互之间也有近邻关系, 所以效率会较差. 本文针对空间数据库中的一组空间对象作为一个整体考虑, 首先设计相关算法, 确定 RkNN 查询结果的候选集, 然后对 RkNN 查询结果的候选集进行优化, 最终得到 RkNN 查询的结果集.

2 候选搜索区域的确定

为了更好地讨论多个查询点的 RkNN 结果集, 下面给出覆盖这多个查询点的最小圆域的概念, 本文是将一组查询作为一个整体考虑的.

定义 1 设组查询 $Q = \{q_1, q_2, \dots, q_s\}$ 是一组查询点集合, 称包含 Q 且半径最小圆为组查询 Q 的最小圆域, 记作 $\text{DOM}(O, R)$, 其中 O 为圆心, R 为半径.

文献[5]中已给出计算 $\text{DOM}(O, R)$ 的方法, 这里不再累赘. 在空间数据集中, 如果记数据点 P_1 到数据点 P_2 的距离为 $\text{dist}(P_1, P_2)$, 则定义 1 中的以 O 为圆心 R 为半径的圆域表示为 $\text{DOM}(O, R) = \{P \mid \text{dist}(P, O) \leq R, P \text{ 是数据点}\}$. 下面给出组查询 Q 的搜索区域算法.

为了使问题具有直观性, 本文在二维空间数据集 R^2 上进行讨论.

2.1 搜索区域初始化

本文参照文献[4]中空间缩减的算法给出搜索区域初始化的算法, 区别在于文献[4]中针对单一查询点进行讨论, 本文是对一组查询进行讨论. 搜索区域初始化实际上就是确定 $\text{DOM}(O, R)$ 内所有点的 RkNN 查询的搜索区域, 由于 $Q \subset \text{DOM}(O, R)$, 所以 Q 的 RkNN 查询区域含于该搜索区域. 为了方便搜索区域初始化算法的叙述, 下面给出几个概念:

定义 2 设 $P \in R^2, \text{DOM}(O, R) \subset R^2, P, O$ 连线与 $\text{DOM}(O, R)$ 圆周交点 M , 则称线段 PM 的垂直平分线为数据点 P 与 $\text{DOM}(O, R)$ 的垂直平分线.

定义 3 P 与 $\text{DOM}(O, R)$ 的垂直平分线将 R^2

分为两部分, 称包含数据点 P 的半平面为 P -半平面.

由定义 2 及文献[2], 显然有如下结论:

命题 1 设 $P \in R^2, \text{DOM}(O, R) \subset R^2$, 则在 P -半平面内的点(不包括 P)不可能是圆 $\text{DOM}(O, R)$ 的 RkNN 的结果.

事实上, 相对于圆 $\text{DOM}(O, R)$ 来说, P -半平面内的点离 P 更近.

定义 4 如果多边形 POL 含于 m 个 P -半平面中, 则称 POL 的层数值为 m , 记 $\text{layer}(\text{POL}) = m$, 特别地, 对于数据点 $P \in \text{POL}$, 也称数据点 P 的层数为 m , 记作 $\text{layer}(P) = m$.

定义 2、定义 3 和定义 4 的几何意义如图 1 所示.

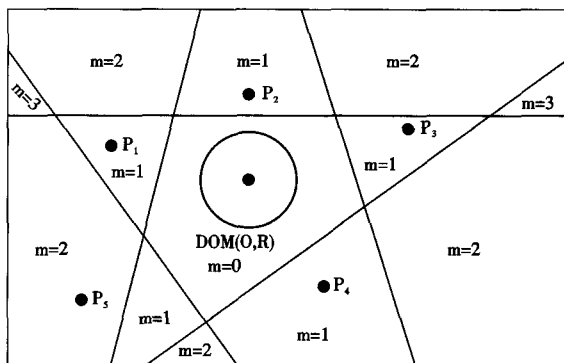


图 1 R^2 中各多边形层数值

为了叙述方便本文利用图 1 给出 $\text{DOM}(O, R)$ 的 RkNN 搜索区域初始化算法:

输入: $\text{DOM}(O, R)$, 给定一组数据对象 $S \subset R^2$, RkNN 值 k .

输出: 由凸多边形区域表示的查询搜索区域初始化.

算法 1:

(1) 对于给定的 R^2 的对象集 $S = \{P_1, P_2, P_3, P_4, P_5\}$, 对 $\forall P_i \in S$, 做出 P_i 与 $\text{DOM}(O, R)$ 的垂直平分线, 这样可以把 R^2 分割成若干个多边形, 如图 1 所示;

(2) 计算每个多边形层数值 m , 如图 1 所示;

(3) 将 $k \geq m$ 的所有多边形作成集合 SET ;

(4) 利用 SET 的各个顶点构造一个包含 SET 的凸多边形区域.

为了说明算法 1 的正确性, 将文献[3]中的定理 2 叙述如下:

命题 2 算法 1 所确定的搜索区域(凸多边形

区域)记为 QRS, 数据对象集 $S \subset R^2$, 对于 $\forall P \in R^2 - S$, 当且仅当 P 在 QRS 内部或边界上时, 有 $\text{layer}(P) < k$, 其中 k 为 RkNN 查询值 k .

由命题 1 和命题 2, 该凸多边形区域即为 $\text{DOM}(O, R)$ 搜索区域初始化区域, 由于 $Q \subset \text{DOM}(O, R)$, 因此, 该凸多边形区域也是 Q 搜索区域初始化区域. 如图 2 所示.

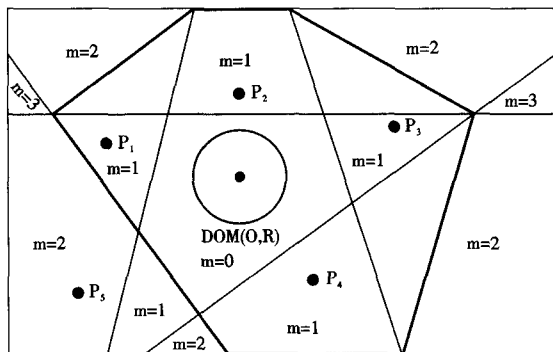


图 2 $\text{DOM}(O, R)$ 的 RkNN 搜索初始化区域

由图 2 可以看出如下命题 3 成立:

命题 3 给定一组数据对象 $S \subset R^2$, 随着 S 中对象的增加, 搜索区域 QRS 会越来越缩减, 特别地, 当向 S 中加入的数据对象距离圆越近时, 则 QRS 会越小.

2.2 确定候选区域

若将一组查询点分开考虑, 这样会使得问题更加复杂化, 为了使得问题便于讨论, 将查询组的所有查询点的最小圆域 $\text{DOM}(O, R)$ 作为整体考虑会使问题的处理更为便利. 为了确定查询组 Q 的候选集合, 由 RkNN 查询的概念给出下面的结论.

命题 4 给定查询组 $Q = \{q_1, q_2, \dots, q_s\}$ 和数据集 $S \subset R^2$, $\forall$ 数据点 $P \in S$, 如果 P 在 S 上的 kNN 查询的结果中包含 $q_i (1 \leq i \leq s)$ 时^[6], 则 P 就是查询组 Q 的 RkNN 一个查询结果.

由命题 4 可以看出, 查询组 Q 的 RkNN 查询的结果集即为 Q 中每个查询点的 RkNN 查询结果的并集. 对于任一组数据对象 $S \subset R^2$, 由命题 3、算法 1 和文献[4]可知, 在 $\text{DOM}(O, R)$ 搜索区域初始化区域内的所有数据对象和 S 都是查询候选者, 因此可以将 $S \cup (QRS \cap (R^2 - S))$ 作为查询 Q 的候选区域, 记为 QSSC. 对于确定 QSSC 的算法可以根据命题 3 采用迭代法来完成, 该算法的思想是: 先令 S 为空, 再将 $\text{DOM}(O, R)$ 外距离点 O 最近的数据对象逐步依次加入到点集 S 中, 这样可以使得搜索空间逐步缩减. 因此每当有新的数据加入到点集 S 时, 就必

须更新和维护搜索区域, 同时也必须及时维护搜索区域的相关信息, 如交点、垂直平分线以及多边形层数值等, 并且需要随时优化计算量.

3 RkNN 查询结果集的确定

RkNN 查询结果集是在查询 Q 候选搜索区域 QSSC 的基础上通过消除查询出的错误数据点而得到. 本文采用 Range-K 验证^[3]以排除错误结果的数据点, 最终得到 RkNN 查询结果集合.

具体算法如下:

输入: $\text{DOM}(O, R)$, RkNN 查询值 k , 第 2.2 节中形成的数据集 S , 候选搜索区域 QSSC.

输出: RkNN 查询结果集.

算法 2:

- (1) 设 $S = \{P_1, P_2, \dots, P_n\}$;
- (2) Do While $QRS \cap (R^2 - S) \neq \emptyset$; // 验证每一个候选对象排除错误
- (3) 取 $P_{n+1} \in QRS \cap (R^2 - S)$; // 由于 $P_{n+1} \in QRS$, 则 P_{n+1} 至少在 $k-1$ 个 P -半平面中
- (4) $S = S \cup \{P_{n+1}\}$;
- (5) If P_{n+1} 所在的 P -半平面数 $m > k-1$ then
- (6) $QSSC = QSSC - \{P_{n+1}\}$; // 由于 $P_{n+1} \in$ 自己的 P -半平面, 因此数据点 P_{n+1} 是错误项, 删除掉
- (7) End if
- (8) End Do
- (9) For $\forall P \in QRS \cap (R^2 - S)$; // 利用 Range-K 进一步验证, 以排除错误项
- (10) If (P 到查询点的距离比到其他候选点的距离小的个数) $\geq k$ then
- (11) $QSSC = QSSC - \{P\}$
- (12) End if
- (13) End For
- (14) 返回 QSSC

算法 2 对 RkNN 查询结果集的确定分为两个阶段, 一是利用逐步添加新数据点计算新的区域来排除错误项, 二是使用 Range-K 验证法分别验证每一个候选对象来进一步排除错误项. 对于步骤(10)的条件, 由于候选集 $QRS \cap (R^2 - S)$ 中含有 Q 中的一个或多个, 而 Q 中某一查询的 RkNN 查询点也可能是另一个查询的 RkNN 查询点, 因此在考虑到利用 Range-K 验证法时必须计算出 P 到查询点的距离比到其它候选点的距离小的个数是否小于 k , 若为真, 则 P 就是这个查询点的 RkNN 查询结果.

4 实验结果与分析

实验采用的2个数据集都是从美国地质调查局(U. S. Census Bureau) Tiger/Line 数据集^[7]中提取的,一个是加州公路(CAR)数据,另一个是洛杉矶地区公路和河流(RR)数据. 全部算法采用 C++ 实现,测试数据集采用 R 树索引,内存缓存区采用 LRU 管理,实验硬件环境为 PentiumIV 1.8 GHz CPU,512MB DDR 内存和 7200 RPM 硬盘. 本实验中取运行时间作为该算法的性能测试指标,这里的运行时间包括 I/O 和 CPU 的时间. 并将每个查询点的运行时间之和作为该算法的运行时间. 在实验中本文 RkNN 查询算法记为 D_RkS,文献[3]中 RkNN 查询算法为 S_RkS. 并用 5 个查询点的组合进行实验.

4.1 k 值对算法执行时间的影响

当 $Q = \{q_1, q_2, q_3, q_4, q_5\}$ 的查询点之间距离较近时,用 D_RkS 算法思想解决要比用 S_RkS 算法思想解决的性能更好. 对于例子当中给定的五个查询点,若用 S_RkS 算法思想解决,就必须对每一个查询点都要求出它的搜索域,然后再计算出它的候选集合 $QSSC_i (i=1, 2, 3, 4, 5)$,同时也可以得到搜索序列 Q 的搜索区域 $\bigcup_{i=1}^5 QSSC_i$. 然而如果对于较近的五个查询点,其中可能有很多重叠的搜索区域,这样也会有很多重复的查询点的候选集合出现,会使算法的性能大大的降低. 而本文的算法 D_RkS 算法是将达到相对接近程度的若干个查询点作划定在查询组 Q 所生成的最小圆域 $DOM(O, R)$ 的范围内,并将 $DOM(O, R)$ 作为一个整体,这样的处理思想就会大大减少了对相同候选者重复查找的可能性. 一般地,对于相对位置较接近的若干个查询点, RkNN 查询问题的处理时用 D_RkS 算法思想要比 S_RkS 算法思想解决会更有效. 为了将两种算法进行比较试验,对于给定的 5 个查询点,在 1~10 的 k 值上进行比较,算法在数据集 CAR 上对执行的性能做对比试验,试验结果如图 3 所示.

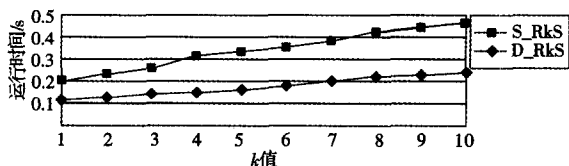


图3 k 值对算法执行时间的影响

由图 3 可以看出,若用 S_RkS 算法思想解决给定的五个查询点,就必须对每个查询点都进行计算

它的搜索区域,并求得每一个查询点的候选集合. 在多个查询点的距离较近时,查询点的搜索区域就会相重叠,也会出现很多重复的查询点的候选集合,尤其随着 k 值的增大, S_RkS 更加延长了算法执行的时间.

4.2 k 值对算法性能的影响

下面实验是在相同查询条件下,比较得到的候选者的数量,一般来说,得到的候选者数量越少,计算量就越小,效率越高. 由图 4 可以看出, S_RkS 算法得到的候选集合随着 k 值的增大比 D_RkS 算法更多.

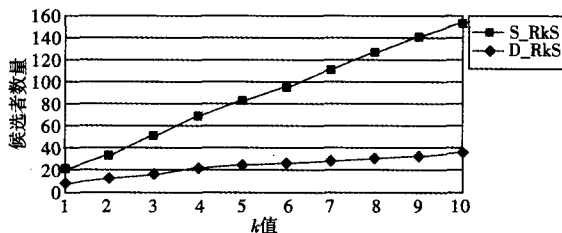


图4 k 值对候选者数量的影响

在处理查询点间相距较近的 RkNN 问题时, D_RkS 算法的性能要比 S_RkS 更有效. 事实上,对于给定的五个查询点序列,若用 S_RkS 算法解决,必须单个的去解决,假若这五个查询点的距离较接近时,会出现搜索区域的重叠和候选集合的重复,从而使得算法性能降低. 对于 D_RkS 算法,越是当查询点的距离想接近时,可将其查询组划定在一个小圆域内,算法当中可以将这个小圆域作为一个整体去考虑,这样就尽最大可能的避免了搜索区域的重叠和候选集合的重复. 通过试验可知,对于距离较近的多个查询点的情况下, RkNN 查询问题处理时, D_RkS 要比 S_RkS 算法具有更高的性能.

4.3 k 值对查询效率的影响

实验对相同的 $k (k=5)$ 值,在测试数据集 CAR 上测试算法 D_RkS 与算法 S_RkS 的运行时间,如图 5 所示. 该试验所取得查询点是比较接近的一组数据,这样就会使得 S_RkS 的查询点搜索区域有较多情况的频繁重叠,而从计算时间上来看,对计算每一个查询点的搜索区域的时间基本相同,因此对于多个查询点搜索区域的计算机时间成查询点的个数的倍数. 而 D_RkS 算法中将若干个距离较近的查询点用一个最小的圆域覆盖,综合计算它们的查询搜索区域,因此当查询点的数量增多时,对算法的性能没有太大的影响. 因此对于较近距离的多个查询点来说处理 RkNN 查询问题, D_RkS 算法比 S_RkS 算法查询效率高得多.

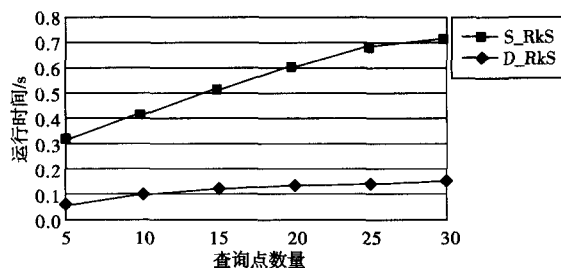


图5 k 值对候选者数量的影响

5 结束语

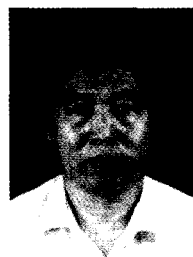
根据问题的实际需求提出了对于多对象综合考虑的 RkNN 查询算法. 本文对于一组查询点集合 $Q=\{q_1, q_2, \dots, q_s\}$, 先是生成覆盖查询 Q 的最小圆域 $DOM(O, R)$, 然后利用算法 1 输出 RkNN 查询搜索区域的初始化凸多边形区域, 并进一步确定其查询结果的候选区域, 然后对非法的数据点进行排除, 最后求得 RkNN 的查询结果. 本文的最后, 在测试数据集 CAR 上进行了算法 D_RkS 与算法 S_RkS 算法性能和效率测试, 结果表明, 对于一组空间中较近距离的若干个查询点的 RkNN 查询数据处理问题, D_RkS 算法的性能和效率要优于单个查询 S_RkS 算法的性能和效率.

参考文献:

- [1] Korn F, Muthukrishnan S. Influence sets based on reverse nearest neighbor queries [C]//Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data. Dallas, New York, USA: ACM Press, 2000: 201-212.

- [2] 陈丽, 陈静, 高新涛, 等. 基于支持向量机与 RkNN 的分类算法研究 [J]. 计算机工程与应用, 2010, 46(24): 135-137.
- [3] Wu Wei, Chee Fei Yang, Chan Yong, et al. Continuous reverse k-nearest-neighbor monitoring [C]//Proceedings of the 9th International Conference on Mobile Data management. Beijing, 2008: 132-139.
- [4] Tao Y, Papadias D, Lian X. Reverse kNN search in arbitrary dimensionality//Proceedings of the Very large Data Bases. Toronto, Canada, 2004: 744-755.
- [5] 汪卫, 王文平. 求一个包含点集所有点的最小圆的算法 [J]. 软件学报, 2000, 9(11): 1237-1240.
- [6] 余小高. P2P 环境中 k 最近邻搜索算法研究 [J]. 微电子学与计算机, 2009, 26(9): 61-63.
- [7] Census Bureau U S. Census, TIGER/Line Shapefiles [EB/OL]. [2008-07-12]. <http://www.census.gov/geo/www/tiger/tgrsh2008/tgrshp2008.html>.

作者简介:



刘金岭 男, (1958—), 教授, 硕士生导师. 研究方向为数据库、数据仓库、数据挖掘.

杨凤霞 女, (1976—), 硕士, 讲师. 研究方向为数据库、数据仓库.

刘国香 女, (1978—), 硕士研究生, 讲师. 研究方向为数据库.

(上接第 17 页)

- [2] 于江德, 王希杰, 余正涛. 基于最大熵模型的语义角色标注 [J]. 微电子学与计算机, 2010, 27(8): 173-180.
- [3] 周兰凤, 徐芳. 一种考虑不确定性的移动机器人路径规划方法 [J]. 微电子学与计算机, 2010, 27(7): 86-89.
- [4] 陈火旺, 罗朝晖, 马庆鸣. 程序设计方法学基础 [M]. 长沙: 湖南科学技术出版社, 1987: 18-38.
- [5] 莫绍揆, 徐永森, 沈百英. 数理逻辑 [M]. 北京: 高等教育出版社, 1984: 117-118.
- [6] 陈意云. 程序设计语言理论 [M]. 北京: 高等教育出版社, 2004: 16-60.

作者简介:



苗德成 男, (1979—), 博士研究生, 讲师. 研究方向为软件系统形式化理论、形式语义学.

奚建清 男, (1962—), 博士后, 教授, 博士生导师. 研究方向为软件系统形式化理论.