移动对象的连续最近邻查询算法

于忠诚　王金慧　郭景峰
（燕山大学信息科学与工程学院，秦皇岛 066004）
E-mail：wangjinhui363@163.com

摘　要　介绍了一种索引结构——TPR 树和静态环境中基本的最近邻查询算法，并提出了影响时间这一概念，将其运用到最近邻查询算法中，可以完成移动对象的连续最近邻查询。

关键词　移动对象　TPR 树　连续最近邻查询

文章编号 1002-8331-(2004)33-0198-03　**文献标识码** A　**中图分类号** TP311.13

Continuous Nearest Neighbor Queries for Moving Objects

Yu Zhongcheng　Wang Jinhui　Guo Jingfeng
（College of Information Science & Engineering，Yanshan University，Qinhuangdao 066004）

Abstract：An indexing structure-TPR tree（time-parameterized R tree）and basic nearest neighbor queries in static environment are introduced in this paper.The concept of "influence time"is given and it is used in nearest neighbor queries algorithm to implement continuous nearest neighbor queries for moving objects.

Keywords：moving objects，TPR tree，continuous nearest neighbor query

1　引言

随着卫星定位系统（例如：GPS）和无线通讯技术的快速发展，跟踪并记录移动对象的位置成为可能，移动对象的连续最近邻查询算法也成为研究的重点和难点。例如："依照 A 车当前的方向和速度，查询在未来 5 分钟内距离其最近的汽车。"

尽管连续查询占据着很重要的位置，然而有效处理这种查询的研究是有限的。在文[1]中，作者致力于建模和查询语言的研究，没有提出具体的存取和处理方法。文[2]采用取样的方法处理 R 树中的最近邻查询。这种方法递增地计算预定位置的结果，采用前一个点的计算结果计算后点，可以避免重复运算。然而，这种取样方法有着不可避免的缺点：如果采样频率很低，那么得到的结果可能会不正确；如果采样频率过高，则会产生很大的计算量。在任何情况下，都无法保证结果是精确的，因为甚至较高的采样频率也会丢失一些结果。文[3]讨论了另一种解决这个问题的方法（移动查询点，用 R 树索引的静态目标），采用 VD 模型，只适用于查找一个最近邻。然而，上述这些方法都不能够处理动态目标。

在静态环境中，有两种基本的最近邻查询算法：BAB（branch-and-bound）算法[4]和 BF 算法[5]。

BAB 算法通过深度优先遍历 R 树查询最近邻。首先，从根结点开始，所有的结点根据它们距离查询点的 MINDIST 值排序（MINDIST 是指查询点 q 和结点 E 的子树中所包含目标的距离的最小值），具有最小值的结点首先被访问。这一过程不断重复直到叶子层中第一个可能的最近邻被找到。然后在向上一层返回的过程中，这个算法只访问那些 MINDIST 值小于当前被找到的最近邻的距离值的结点。

BF 算法采用了一个堆（HEAP）保存到目前为止已经访问过的结点。开始时，这个堆只保存根结点的孩子，并根据 MINDIST 值排序。当一个结点被访问时，被从堆中移走，它的孩子根据它们的 MINDIST 值被添加到堆中。这一过程不断重复直到找到最近邻。BF 是一个较为优越的算法，因为这种方法从所有结点中选择一个要处理的结点，可以从全局进行控制，从而只访问必要的结点。

然而，上述两种算法都是基于 R 树的、适用于静态环境下的最近邻查询算法。该文提出的算法采用 TPR 树这一索引结构对动态目标进行索引，可以有效地完成移动环境下的连续最近邻查询。

2　TPR 树（Time Parameterized R-tree）

TPR 树[6]是 R 树的扩展，它能够索引动态目标并回答未来查询。一个动态目标的表示如下：（1）当前时刻绑定这个动态目标的最小边界矩形 MBR；（2）一个速度矢量。图 1 表示了两个移动对象 u 和 v，以及包含他们的边界矩形，箭头表示每个边的速度方向，数字对应于速度的值，坐标轴负方向的速度是负值。

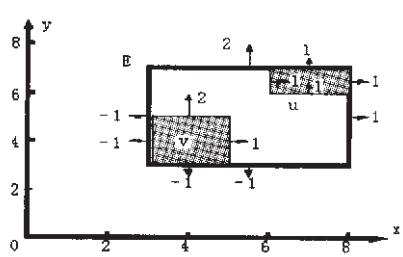

图 1　TPR 树中结点的表示

一个非叶结点也存储了一个 MBR 和它的速度矢量，MBR

作者简介：于忠诚，男，副教授，主要从事数据库应用技术、移动对象数据库、智能交通系统的研究。王金慧，女，硕士。主要从事移动对象数据库的研究。郭景峰，男，博士，教授，主要从事数据库应用技术，移动对象数据库，粗糙集，智能交通系统等的研究。

中紧紧包含着当前时刻的所有实体。非叶结点中 MBR 的速度矢量原则如下:(1)上(右)边的速度是在同一方向下,当前结点的子树中所有实体速度的最大值。(2)下(左)边的速度是在同一方向下,当前结点的子树中所有实体速度的最小值。这种方法可以保证 MBR 一直包含着这些目标。

3 基于 TPR 树的连续最近邻查询

这一查询算法的结果有如下形式:$<R,T,C>$,R 是满足查询条件的最近邻,T 是 R 的过期时间,即下一个目标比当前的最近邻更接近查询点的时刻。C 是在 T 时刻影响到 R 的目标,即 T 时刻后的下一个最近邻。例如:在(0,3)时刻中,若 A 是查询点 q 的最近邻,但是在 3 时刻,B 将成为 q 的最近邻,那么这一结果将表示为$(A,3,B)$。从当前结果 R 和 C,可以递增地计算下一个结果,从而完成连续的最近邻查询。

3.1 影响时间

定义 1:将一个目标 o 对于查询点 q 的影响时间定义为 $T_{INF}(o,q)$,为该目标比当前的最近邻距查询点更近的时刻。例如:当前时刻(0 时刻)q 的最近邻为 a,在 2 时刻,b 到 q 的距离要小于 a 到 q 的距离,那么 b 的影响时间为 2。如果一个移动目标到查询点的距离永远不会大于当前最近邻距查询点的距离,那么它的影响时间为 ∞。

定义 2:对于非叶结点 E,将在 E 的子树中所包含目标的最小影响时间定义为 $T_{INF}(o,q)$。

根据上面的定义,最近邻查询中当前最近邻的过期时间即为所有目标中最小影响时间,下一个最近邻即为具有最小影响时间的目标。因此,这种查询可以通过寻找具有最小影响时间值的目标来完成。

3.2 移动环境下影响时间的计算

在移动环境中,查询点和查询目标都是移动的,其运动轨迹可以定义为时间的线性函数。假设 P_{NN} 是 q 的当前最近邻,那么 $T_{INF}(o,q)$是指满足 $o(t)$与 $q(t)$的距离小于 $P_{NN}(t)$与 $q(t)$的距离这一条件的未来某一时刻,其中 $P_{NN}(t),o(t),q(t)$是在 t 时刻 P_{NN},o,q 的位置。也就是说,$T_{INF}(o,q)$是满足下个式子的最小时间:$\| o(t),q(t) \| \le \| P_{NN}(t),q(t) \|$ 而且 $t \ge 0$。如果不存在 t 满足这个不等式,那么 $T_{INF}(o,q)$的值设为 ∞。

如果一个移动点 o 的坐标为$(o_1,\dots,o_n)$,在维度 $i=1,\dots,n$ 上的速度为$(o.V_1,\dots,o.V_n)$,查询点 q 的坐标为$(q_1,\dots,q_n)$,在维度 $i=1,\dots,n$ 上的速度为$(q.V_1,\dots,q.V_n)$。那么上面的不等式可以转换成如下形式:

$$At^2+Bt+C\le 0,A=\sum_{i=1}^{n}[(o.V_i-q.V_i)^2-(P_{NN}.V_i-q.V_i)^2]$$

$$C=\sum_{i=1}^{n}[(o_i-q_i)^2-(P_{NN_i}-q_i)^2]$$

$$B=\sum_{i=1}^{n}2[(o_i-q_i)(o.V_i-q.V_i)-(P_{NN_i}-q_i)(P_{NN}.V_i-q.V_i)^2]$$

在非叶结点中,$T_{INF}(E,q)$表示在 E 的子树中的目标与 q 的距离小于 P_{NN} 与 q 的距离的最早时间。也就是说,$T_{INF}(E,q)$是满足下一不等式的最小时间:$\min dist(E(t),q(t)) \le \| P_{NN}(t),q(t) \|$ 而且 $t \ge 0$。

这个不等式需要进一步讨论,因为 $\min dist(E(t),q(t))$的计算要取决于 E 和 q 的位置。

图 2 给出了一个查询点 q 和 MBR E,它们之间的 $\min dist$ 有以下几种不同情况:当 q 在 E 内部时,$\min dist(q,E)=0$。初始时,$\min dist(q,E)$为 q 与 a 的距离,随着 q 点的运动,$\min dist(q,E)$逐渐变为 q 与 ab 边的距离、0、q 与 bc 边的距离和 q 与 c 的距离。因此,在计算时,需要根据 q 的运动情况将其分为不同的时间段,每个时间段中对应的不同。在维度为 n 的情况下,最多可以将其分成 $2n+1$ 个互不相交的时间段。根据 q 的运动情况,最多可以分成 5 个不同的时间段。

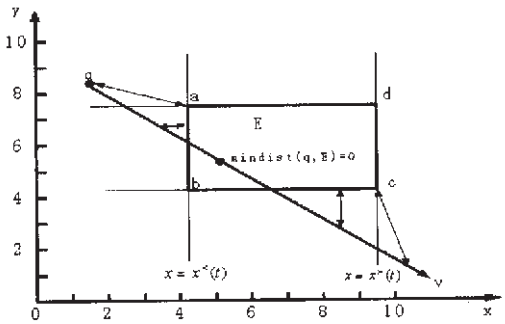

图 2 不同情况下的 $\min dist(q,E)$

如果查询点 q 的坐标为$(q_1,\dots,q_n)$,在维度 $i=1,\dots,n$ 上的速度为$(q.V_1,\dots,q.V_n)$。以时间为参数的矩形 E 在维度 $i=1,\dots,n$ 上的边的坐标为$([E_1^<,E_1^>],\dots,[E_n^<,E_n^>])$,在维度 $i=1,\dots,n$ 上的边的速度为$([E_1^<,E_1^>],\dots,[E_n^<,E_n^>])$。通过如下算法,可以计算出不同时间段的 $\min dist(E,q)$。在算法第二步中,计算了查询点 q 与 E 的 $\min dist(E,q)$值改变的时刻。在算法第三步中,计算了不同时刻的 $\min dist(E,q)$的值。

Mindist(E,q)

1.Set $T \to \emptyset$

2.For each dimension $i=1,\dots,n$ do

If $t_i^< = (q_i - E_i^<)/(E.V_i^< - q.V_i)$ and $q.V_i \ne E.V_i^<$

add $t_i^<$ to T

If $t_i^> = (q_i - E_i^>)/(E.V_i^> - q.V_i)$ and $q.V_i \ne E.V_i^>$

add $t_i^>$ to T

3.Sort T

The elements of T have at most $2n+1$ intervals.For each such interval T_j:

$$\min dist(E,q)=\sum_{i=1}^{n}\min dist_i(E,q) \text{ where}$$

$$\min dist_i^2(q,E)=\begin{cases} t^2(E.V_i^<-q.V_i)^2+2t(E_i^<-q_i)(E.V_i^<-q.V_i)+(E_i^<-q_i)^2 \\ \quad \forall t \in T_j(q_i+q.V_i t \le E_i^<+E.V_i^< t) \\ t^2(E.V_i^>-q.V_i)^2+2t(E_i^>-q_i)(E.V_i^>-q.V_i)+(E_i^>-q_i)^2 \\ \quad \forall t \in T_j(q_i+q.V_i t \le E_i^>+E.V_i^> t) \\ 0 \qquad otherwise \end{cases}$$

将上述算法计算得到的结果代入下式:$\min dist(E(t),q(t)) \le \| P_{NN}(t),q(t) \|$,也可以转换成如下形式:$At^2+Bt+C \le 0$,求出满足此不等式的最小时间 t,即为 $T_{INF}(E,q)$。如果没有 t 满足这个不等式,那么 $T_{INF}(E,q)$的值设为 ∞。

3.3 移动对象的连续最近邻查询算法

下文将 BAB 算法(DF 算法)和 BF 算法改进,使其应用于

移动对象数据库中的连续最近邻查询：

3.3.1 BAB 算法(DF 算法)

```
TP_NN(current node N)
/initially:T=∞,C=Ø/
1.if N is a leaf
2.for each object o
3.  if T_INF(o,q)<T
4.      C={o}
5.      T=T_INF(o,q)
6.  else if T_INF(o,q)=T
7.      C=C∪{o}
8.else /N is an intermediate node/
9.  sort all the entries E by their T_INF(E,q)
10.   for each entry E
11.     if(T_INF(E,q)≤T)
12.       TP_NN(e.childnode)
13.end TP_NN
```

3.3.2 BF 算法

```
TP_NN()
1.initialize a heap H that accepts <key,entry>
2.retrieve the root node R
3.for each entry E in R
      insert <T_INF(E,q),E> to H
4.while(H is not empty)
5.de-heap <key,E>/E has the minimum key in H/
6.if E points to a leaf node
7.      for each object o in E.childnode
8.        if T_INF(o,q)<T
9.          C={o}
10.         T=T_INF(o,q)
11.       else if T_INF(o,q)=T
12.         C=C∪{o}
13.else/E points to a non-leaf node/
14.  if T_INF(o,q)≤T
15.  for each entry E' in E.childnode
16.    insert <T_INF(E',q),E'> to H
17.end TP_NN
```

这两种算法分别通过深度优先和宽度优先搜索，在已知当前最近邻的情况下找到下一个最近邻和最近邻改变的时刻。将这种算法应用于连续最近邻查询的具体方法如下：首先，执行一次当前时刻的基本最近邻查询，得到第一个最近邻 R；然后，执行 TP_NN，得到下一个最近邻 C 和当前最近邻的过期时间 T；这时得到的 C 为下一次查询时的 R，再反复使用 TP_NN，直到得到所有满足连续查询条件的最近邻。

4 实验

该实验动态数据集的选取采用满足高斯分布的点，密度为0.5，移动点的速度从[-0.1,0.1]不等。磁盘页大小设为 1k，采用 TPR 树对 100k 个点建立索引，TPR 中结点最大数量为 26，通过产生 200 个查询来衡量性能。如图 3 所示，横轴为最近邻产生变化点的数量，纵轴表示磁盘存取次数。可以看出，BF 算法较 DF 算法更为优越。

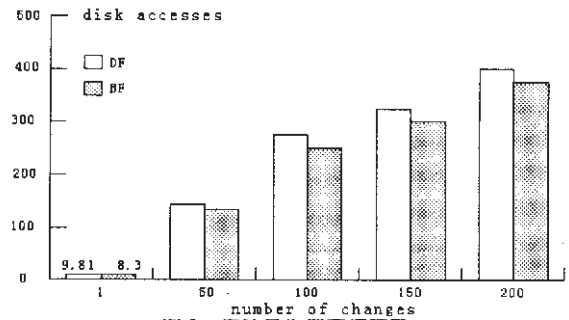

图 3 连续最近邻查询性能

5 结束语

文章在 TPR 树这一索引结构的基础上，提出了影响时间这一概念，将其运用到最近邻查询算法中，并对两种算法进行了分析比较，可以完成对移动对象未来最近邻的连续查询。未来的工作将致力于算法的改进以及 k 个连续最近邻的研究。

(收稿日期：2004 年 4 月)

参考文献

1.P Sistla,O Wolfson,S Chamberlain et al.Modeling and Querying Moving Objects[C].In:Proc 13th International Conference on Data Engineering,Birmingham,U K,1997:422~432

2.Z Song,N Roussopoulos.K-Nearest Neighbor Search for Moving Query Point[C].In:Proceedings of the 7th International Symposium on Spatial and Temporal Databases,2001:79~96

3.B Zheng,D Lee.Semantic Caching in Location-Dependent Query Processing[J].SSTD,2001

4.N Roussopoulus,S Kelley,F Vincent.Nearest neighbor queries[C].In:Proceedings of ACM SIGMOD international Conference on Management of Data,SanJose,USA,1995

5.G R Hjaltason,H Samet.Distance Browsing in Spatial Databases[J].ACM Transactions on Database Systems,1999;24(2):265~318

6.S Saltenis,C S Jensen,S T Leutenegger et al.Indexing the Positions of Continuously Moving Objects[C].In:Proceedings of the 2000 ACM SIGMOD international Conference on Management of Data,2000:331~342

(上接 143 页)

善，其对进一步研究多信道随机争用多址系统的信源接入、冲突分解控制和提高系统性能都将是有意义的。

(收稿日期：2004 年 6 月)

参考文献

1.ROM R,SIDI M.Multiple access protocols[M].New York Berlin Heidelberg:Springer-Verlag,1989:5~30

2.Gulko E.Tree-based multi-access protocols where collision multiplicities are known[J].IEEE Tran Commun,1985;33:999~1001

3.Capetanakis J I.Tree algorithms for packet broadcast channels[J].IEEE Trans Info Theory,1979;25(5):505~515

4.赵东风，李必海，郑苏民.二叉树形冲突分解算法研究[J].电子科技大学学报，1996;25(8):260~264

5.赵东风，李必海，郑苏民.一种新的时隙式随机多址系统分析方法研究[J].电子科学学刊，1997;19(6):814~819

6.黄华伟，赵东风，候芬.二叉树形冲突分解改进算法分析.云南大学学报(自然科学版)，1999;21(1):67~70

7.梁平原，赵东风，丁洪伟.多通道多业务优先级控制随机多址系统研究[J].云南民族学院学报(自然科学版)，2002;11(3):144~149

移动对象的连续最近邻查询算法

作者： 于忠诚， 王金慧， 郭景峰
作者单位： 燕山大学信息科学与工程学院,秦皇岛,066004
刊名： 计算机工程与应用 ISTIC PKU
英文刊名： COMPUTER ENGINEERING AND APPLICATIONS
年，卷(期)： 2004,40(33)
被引用次数： 5次

参考文献(6条)

1. P Sistla;O Wolfson;S Chamberlain Modeling and Querying Moving Objects[外文会议] 1997

2. Z Song;N Roussopoulos K-Nearest Neighbor Search for Moving Query Point[外文会议] 2001

3. B Zheng;D Lee Semantic Caching in Location-Dependent Query Processing 2001

4. N Roussopoulus;S Kelley;F Vincent Nearest neighbor queries 1995

5. G R Hjaltason;H Samet Distance Browsing in Spatial Databases[外文期刊] 1999(02)

6. S Saltenis;C S Jensen;S T Leutenegger Indexing the Positions of Continuously Moving Objects 2000

本文读者也读过(10条)

1. 黄敬良.郝忠孝.HUANG Jing-liang.HAO Zhong-xiao 移动对象的K个连续最近邻查询算法[期刊论文]-哈尔滨理工大学学报2007,12(6)

2. 刘彬.万静.LIU Bin.WAN Jing 基于R-树的连续最近邻查询算法优化研究[期刊论文]-信息技术2008,32(1)

3. 郭景峰.王金慧.侯爽.孙浩.GUO Jing-feng.WANG Jin-hui.HOU Shuang.SUN hao 连续最近邻查询方法研究[期刊论文]-现代计算机（专业版）2004(7)

4. 冯惠妍.郭俊凤.FENG Hui-yan.GUO Jun-feng 道路网络中的连续最近邻查询[期刊论文]-计算机工程2010,36(8)

5. 黄敬良 动态环境下移动对象连续最近邻查询研究[学位论文]2008

6. 孙冬璞.郝忠孝.SUN Dong-pu.HAO Zhong-xiao 移动对象历史轨迹的连续最近邻查询算法[期刊论文]-计算机工程2009,35(1)

7. 丁治明.孟小峰.王珊 A Transactional Asynchronous Replication Scheme for Mobile Database Systems [期刊论文]-计算机科学技术学报(英文版)2002,17(4)

8. 刘小峰.陈传波.刘云生.LIU Xiao-feng.CHEN Chuan-bo.LIU Yun-sheng 移动对象全局K最接近邻居查询研究[期刊论文]-微电子学与计算机2007,24(9)

9. 郭景峰.孙浩.王金慧.GUO Jing-feng.SUN Hao.WANG Jin-hui 移动对象数据库的查询方法研究[期刊论文]-现代计算机（专业版）2004(8)

10. HUANG Yan-yan 探索班主任与学生交流的艺术[期刊论文]-宿州学院学报2008,23(2)

引证文献(5条)

1. 刘彬.王建国 基于几何特征的连续最近邻查询方法代价研究[期刊论文]-唐山师范学院学报 2007(5)

2. 宋晓宇.孙业挺.孙焕良 支持动态负载的移动对象最近邻查询算法[期刊论文]-计算机工程与应用 2007(27)

3. 闵寻优.郝忠孝 三维空间中的连续最近邻查询[期刊论文]-软件 2011(2)

4. 刘彬.王建国 动态环境中连续K近邻查询的边界线方法[期刊论文]-计算机工程与设计 2008(18)

5. 王建朝 基于路网的移动对象数据库索引机制研究[学位论文]硕士 2006