移动对象查询处理技术:研究综述^{*})

丁晓锋 卢炎生

(华中科技大学计算机学院 武汉 430074)

摘要 近年来,随着无线通信技术的迅速发展和个人终端设备功能的逐步完善,移动计算较传统的分布式计算有了更广阔的应用背景和发展空间,特别是针对移动对象的查询处理研究引起了业界和学术界越来越多的兴趣和关注。移动对象相关的查询主要包括:最近邻查询、反最近邻查询、空间连接以及最近对查询等。本文从这些查询的基本定义开始,介绍了这些查询相关的基本处理技术,总结了近年来学者们提出的有效解决方法和研究进展,并在最后探讨了移动对象查询处理技术的未来发展趋势。

关键词 移动计算,查询处理,位置相关服务,移动对象

Techniques for Query Processing of Moving Objects: A Survey

DING Xiao-Feng LU Yan-Sheng

(Department of Computer Science, Huazhong University of Science and Technology, Wuhan 430074)

Abstract With the rapid development of wireless communications and advances in personal information terminals in recent years, mobile computing is emerging as a more powerful and promising trend than the traditional distributed computing, especially, techniques for efficiently answering queries about moving objects are gaining interest. Queries about moving objects mainly include: nearest neighbor, reverse nearest neighbor, spatial join, and closest pairs etc. The basic conceptions of these queries is defined in the beginning of the paper, then the related technologies about such query processing and new developments are concluded, and finally the future study directions about query processing of moving objects is pointed out.

Keywords Mobile computing, Query processing, Location-based services, Moving object

1 引言

在移动计算环境下,大量移动对象的定位和管理给传统的数据处理技术带来了新的挑战和机遇。在传统的数据处理中,数据对象的移动特性通常被人们忽略了。但在移动计算环境下,这种“移动透明”性则常常被“移动相关”的特性所取代^[1,2]。移动性是移动计算系统有别于传统分布式系统的最为重要的特征,而移动对象相关的查询则是移动信息系统最重要的功能之一。移动对象的查询属于位置相关的查询,它的查询结果依赖于移动对象的位置和查询时间。同一个查询请求,其提交地点与查询时间的不同,产生的查询结果也将不同。例如,在航海导航系统中,高速运行的舰船查询距离自己最近的 k 个舰船,该查询结果将随着舰船的移动而不断发生变化。

移动对象相关的查询在实际应用中充当了重要的角色,并由此激发了一类新的实用系统,如军事指挥系统、智能运输系统以及自动导航系统等^[2]。在这类系统中,移动主机位置的不透明性是实现位置相关查询的基本条件之一。由于移动对象的位置属性具有连续变化的特性^[3],它可以看成是特定轨迹上的运动或随时间变化的函数。所以,移动对象提交每一个查询时,系统都需要根据当前所有移动对象的位置信息返回一定的查询结果。

移动对象相关的查询与静止对象相关的查询是密不可分

的,它在有关静止对象的查询处理技术基础之上,结合移动计算环境下特有的应用需求,诠释了新的查询理念和技术要求。传统查询处理技术的不断发展以及各种新需求和新技术的不断涌现,使得移动对象查询处理技术得到了长足的发展。本文第2节详细总结了几种主要的移动对象相关的查询处理技术:最近邻查询、反最近邻查询、空间连接以及最近对查询技术等。在介绍了最新的研究进展之后,本文第3节讨论了查询处理技术未来的发展趋势。

2 移动对象查询及处理技术

为便于陈述,对系统中移动对象、移动对象的运动方式以及对象间距离计算方式假定如下:

- 系统中的移动对象都具有唯一标识符,所有对象的集合表示为 $M = \{M_1, M_2, \dots, M_n\}$, 其中 n 是系统中所有移动对象的个数, $M_i (1 \leq i \leq n)$ 为第 i 个移动对象的标识符。

- 系统中所有的移动对象均取自二维空间, M_i 在更新时刻 t_0 的位置表示为 (x, y) , 速度为 $v = (v_x, v_y)$, 则 M_i 在此后未更新前的任意时刻 $t > t_0$ 所在的位置为 $(x + (t - t_0)v_x, y + (t - t_0)v_y)$ 。

- 移动对象 M_i 和 M_j 之间的距离为欧几里德距离的平方, 表示为 $D(i, j) = (x_i - x_j)^2 + (y_i - y_j)^2$ 。

2.1 最近邻查询

最近邻查询(Nearest Neighbor, NN)就是对于一个给定

^{*})国家“十一五”国防预研基金资助项目(513150402)。丁晓锋 博士研究生,主要研究方向为移动数据库、时空数据库;卢炎生 教授,博士生导师,主要研究方向为数据库理论、数据挖掘、软件工程。

的移动查询对象 M_i , 在一定时间段 T 内找出距离 M_i 最近的移动对象 M_j , 其形式化定义如下:

定义 1 给定查询对象 $M_i \in M$, 在时间段 T 内, 其最近邻查询结果为

$$NN(M_i) = \{M_j, D(i, j) \leq D(i, k)\}, \forall M_k \in M \setminus \{M_i\}$$

事实上, 最近邻查询是 k 最近邻查询 (k Nearest Neighbor, kNN) 在 $k=1$ 时的特殊情况。更一般的 kNN 查询定义为, 在一定时间段 T 内返回距离查询对象 M_i 最近的 k 个移动对象, 有如下形式化定义:

定义 2 给定查询对象 $M_i \in M$, 在时间段 T 内, 其 k 最近邻查询结果为

$$kNN(M_i) = M' = \{M_{j_1}, M_{j_2}, \dots, M_{j_k}\}$$

且满足条件, 对 $\forall M_r \in M \setminus M'$ 有:

$$D(i, j_1) \leq D(i, j_2) \leq \dots \leq D(i, j_k) \leq D(i, r)$$

作为传统的对象查询处理技术, 最近邻查询已经成为移动计算系统中最为重要的应用之一。近年来针对最近邻查询的新需求, 学者们进行了大量研究, 取得了一系列研究成果^[4~20]。纵观这一领域的研究进展, 大部分研究工作都是围绕空间索引 R-tree^[5]展开的。

基于索引结构的分支定界 (branch-and-bound) 算法为此类查询提供了有效的解决平台。其中最具影响力的算法当属文^[6]中提出的深度优先 (Depth-First, DF) 遍历, 它利用 R-tree 索引所有的查询对象, 在若干启发式规则的约束下对索引树进行深度优先遍历。文^[7, 8]论证了 DF 算法的不足之处, 并在 R*-tree^[11] 索引结构的基础上提出了最好优先 (Best-First, BF) 遍历算法, 试验结果表明 BF 算法的查询效率优于 DF 算法。文^[9, 10]利用分支定界算法对索引结构进行了定制修改, 在解决高维空间中的最近邻查询中表现出很好的性能。文^[12, 13]中提出的多步 (Multi-Step) 最近邻查询算法也在高维空间中得到了较好的应用。

上述的各种查询算法, 严格意义上并没有很好的解决查询对象与目标对象集移动性的问题。对此, 学者们也进行了大量研究, 其中可以分为三类解决方法^[14~21]: (1) 利用泰森图 (Voronoi Diagram); (2) 对偶转换 (Duality Transformation); (3) 基于 TPR-tree。文^[14]通过计算移动查询对象 M_i 的泰森多边形来求解最近邻查询, 并提供动态重组算法以保证其泰森多边形的时效性。文^[16]利用对偶转换将形如 $x(t) = x_0 + v_x t$ 的一维移动对象, 转换成其对偶空间中的一个点 (x_0, v_x) 之后再行最近邻计算, 然而对于高维空间中的最近邻查询却不能达到预期的效果。文^[17, 18]充分考虑了高维空间中最近邻查询的问题, 特别是文^[18]基于 TPR-tree^[20] 分别提出了时间参数化 (time parameterized) 最近邻查询和持续 (continuous) 最近邻查询, 并给出了相应查询算法 TP_KNN 和 C_KNN 以及准确的代价模型。文^[19]将移动对象的查询时间段 T 分成 n 个等长的时间间隔, 使用周期取样 (periodical sampling) 技术重复利用上次的取样结果, 计算本次取样的最近邻。文^[38]提出了与最近邻查询功能相似的全部最近邻查询 (All-Nearest-Neighbor, ANN) 算法, 并通过试验证明了 ANN 算法优于一般的最近邻查询算法。此外, 针对在移动对象不断更新环境下, 如何保持最近邻查询结果一致性的问题, Benetis 等人^[3]基于 TPR-tree 提出了持久 (persistent) 最近邻查询方法, 算法利用最小优先级 (min-priority) 队列控制索引树的遍历顺序, 根据不同的查询时间段动态组建结果集, 并用试验证明了算法的健壮性。

2.2 反最近邻查询

作为一个相对于最近邻查询较新的问题, 反最近邻查询近年来得到许多研究者的关注^[22~30]。反最近邻查询 (Reverse Nearest Neighbor, RNN) 就是对于一个给定的移动查询对象 M_i , 在一定时间段 T 内找出以 M_i 为最近邻对象的移动对象 M_j , 其形式化定义如下:

定义 3 给定查询对象 $M_i \in M$, 在时间段 T 内, 其反最近邻查询结果为

$$RNN(M_i) = \{M_j, D(i, j) \leq D(i, k)\}, \forall M_k \in M \setminus \{M_i\}$$

与最近邻查询类似, 反最近邻查询是 k 反最近邻查询 (Reverse k Nearest Neighbor, RkNN) 在 $k=1$ 时的特殊情况, 更一般的 RkNN 查询定义为, 查找以查询对象 M_i 为其 k 最近邻查询结果的移动对象 M_j , 有如下形式化定义:

定义 4 给定查询对象 $M_i \in M$, 在时间段 T 内, 其 k 反最近邻查询结果为:

$$RkNN(M_i) = M' = \{M_{j_1}, M_{j_2}, \dots, M_{j_m}\}$$

且满足条件, 对 $\forall M_j \in M', \forall M_r \in M$ 有:

$$|D(j, r) - D(i, j)| < k$$

反最近邻查询最直接的解法就是转换成最近邻查询: 对于数据集中的每一个移动对象 M_j , 检查其是否以查询对象 M_i 作为自己的最近邻节点, 如果是, 则 M_j 可作为 M_i 的反最近邻节点之一, 否则不然。然而, 当系统中有大量的移动对象存在时此方法显然是不可接受的。

Stanoi 等人^[23]就反最近邻查询到最近邻查询的转换问题做了更深入的研究。文^[23]将位于二维空间中的查询对象 M_i 占有的空间 R , 通过 3 条相交直线分为 6 个相等的部分 R_i ($1 \leq i \leq 6$), 在每个 R_i 内计算 M_i 的最近邻节点 M_j , 如果 M_j 的最近邻节点是 M_i , 则 M_j 是 M_i 的反最近邻节点之一, 否则 M_i 在 R_i 区域内不存在反最近邻节点, 并通过理论推理和试验证明了该观点的正确性。然而该算法的效率在高维空间中成指数级下降。针对此问题, 文^[27]利用 kNN 算法直接搜索 RkNN 的预备候选集, 虽然这种方法不能总是找到所有的 RkNN 结果集, 但试验表明算法的效率可以得到很大的提高。

另外一种路径是基于 R-tree 索引方法。文^[22]利用两种索引结构来解答反最近邻查询: 一种是 RNN-tree, 它包含节点的 MBR (minimal bounding rectangle) 是这样一个特殊圆圈, 它以节点 M_i 为中心, M_i 到其最近邻节点的距离为半径; 另一种索引结构是 NN-tree, 严格意义上讲它是存储数据集的 R*-tree^[11]。此方法通过同时遍历两个索引树为最近邻查询提供了有效的解决路径。然而, 两种索引结构的使用势必影响查询效率。对此, Yang 等人^[24]在此基础上利用单一的索引结构 Rdn-tree 可以同时解决 RNN 和 NN 查询, Rdn-tree 在结构上与 R*-tree 基本相同, 不同之处在于其叶节点项存储的是节点到其最近邻节点的距离。此外, 文^[26]提出了一种基于主存结构的算法来回答二维空间中的 RNN 查询问题, 对于每一个空间索引对象, 算法都会在主存结构中记录其到最近邻节点的距离, 由于大部分操作都在主存中完成, 磁盘 I/O 数可以大大减少。

Tao 等人^[28~30]针对动态环境下任意维空间中的反最近邻查询进行了大量研究。其中文^[28]提出了 TPL 算法, 它的工作流程可以分作两个阶段: 过滤阶段 (filtering phase) 和精炼阶段 (refinement phase)。TPL 算法与 Stanoi 的解法有相似之处, 不同之处是其没有对空间进行平均分割, 从而很容易扩展到高维空间。然而, 对于移动对象大量更新的问题, 上述

各种算法都不能达到很好的求解效果。文[3]利用最近邻查询解法,分别提出了 FindRNN 和 FindRkNN 算法。理论分析表明,它们有助于解决移动计算环境下大量对象并发更新所带来的查询结果不一致的问题,并用试验证明了算法的有效性。

2.3 空间连接

空间连接(Spatial Join, SJ)就是对于给定的两个数据集 A 和 B ,从它们的笛卡尔(Cartesian)集 $A \times B$ 中找出满足一定空间谓词逻辑的数据对。一般的空间连接是 e 距离连接(e Distance Join, eDJ),它定义为从两个数据集 A 和 B 中各取一个对象,使得它们之间的距离在一定时间段 T 内等于 e ,其形式化定义如下:

定义5 给定数据集 $A \subset M, B \subset M \setminus A$,在时间段 T 内,其 e 距离连接的查询结果为

$$eDJ(A \times B) = \{(M_i, M_j), D(i, j) = e\}, M_i \in A, M_j \in B$$

对具有空间范围的数据对象, e 距离连接通常定义为交集连接(Intersection Join, IJ),它是查询在空间范围内有交集的对象对。IJ 是 $e=0$ 的特殊空间连接。

目前为止,大部分的空间连接算法都是基于 R-tree 提出的。最原始的连接算法 RJ(R-tree Join)是在文[31]中由 Brinkhoff 等人提出。RJ 算法分别对数据集 A 和 B 建立 R-tree 索引 R_A 和 R_B ,然后同时从根节点开始遍历索引树 R_A 和 R_B ,对于遍历得到的节点项 E_A 和 E_B ,如果它们的最小边界区域(MBR)相交,则节点项 E_A 和 E_B 分别指向的空间对象 M_i 和 M_j 可作为连接集的一部分而被保留,否则不然。同时索引两种数据集的方法对算法执行效率有一定的影响,文[32]在 RJ 算法基础上,通过建立单一数据集索引 R_A 或 R_B 对数据集的遍历方式进行了改进,并且取得了良好的试验效果。另外,文[33]也利用单一数据集索引的方式遍历索引树以得到连接结果,所不同的是它采用基于哈希(hash-based)的算法来动态确定遍历次序。

除了对数据集 A 和 B 建立索引的解法,研究者还提出了另外两种方法^[34~37];基于空间哈希的连接(spatial hash join)算法和 plane-sweep 技术^[15]。文[35]中提出的 PBSM(partition based spatial merge)连接算法就是具有代表性的典范,PBSM 将对象的移动空间 R ,利用垂直相交的直线分成均匀的网格,然后将数据集 A 和 B 中的所有数据利用哈希算法映射到这些均匀分布的网格中。在数据集的映射过程中,对于任何的 $M_i \in A, M_j \in B$,如果它们映射到相同的网格中,那么它们组成的数据对就会被加载,然后在主存当中以 plane-sweep 技术进行连接。如果划分的网格偏大,此时映射到网格中的数据可能引起主存溢出,在这种情况下,通过循环调用分割算法,将网格进一步的划分成相对更小的区域以使问题得到有效的缓解。

针对空间连接在实际应用中的新需求,最近有研究者^[18]提出了基于时间参数的空间连接(Time Parameterized Spatial Join, TP_SJ)和持续空间连接(Continuous Spatial Join, C_SJ)查询,并给出了相应求解算法 TP_Join 和 C_SJ。其中 TP_Join 采用较为传统的双索引树技术进行求解,不同之处是用时空索引 TPR-tree 代替了传统的空间索引 R-tree。C_SJ 可以看作是简化了的增量最近对查询解法,本文将在下一节中进行详细的介绍。

2.4 最近对查询

最近对查询(Closest Pairs, CP)就是对于给定的两个数

据集 A 和 B ,在一定时间段 T 内,从它们的笛卡尔(Cartesian)集中 $A \times B$ 找出 k ($k \geq 1$)对距离最短的数据对。通常情况下也叫做 k 最近对查询(k Closest Pairs, k CP),一般的形式化定义如下:

定义6 给定数据集 $A \subset M, B \subset M \setminus A$ 在时间段 T 内,其 k 最近对查询的结果为

$$kCP(A \times B) = \{(M_k, M_{jk})\}, M_k \in A', M_{jk} \in B'$$

其中 $A' \subset A, B' \subset B$,且满足条件,对 $\forall M_i \in A \setminus A', \forall M_j \in B \setminus B'$,有

$$D(i1, j1) \leq D(i2, j2) \leq \dots \leq D(ik, jk) \leq D(i, j)$$

纵观近年来有关 k 最近对查询的研究,学者们提出的解决方法大概可以分为两类^[38~46]:增量(incremental)查询方法和非增量(non-incremental)查询方法。在增量方法中,算法核心思想是利用一个基于距离查询的优先级队列来控制结果的生成,因此执行参数 k 在增量算法执行之前可以是未知的,系统可以根据自己的需求动态确定 k 的大小。增量算法的查询结果是通过管道(pipelined)方式以距离递增的顺序形成的,这样做的好处是在算法尚未结束之前就能得到部分查询结果。在查询结果集较小的情况下,增量算法表现出良好的系统稳定性和高效的查询性能。与此相反,在非增量方法中,执行参数 k 在算法执行之前必须是已知的,并且 k 对查询结果在算法执行结束后一次性输出。试验表明,在执行参数 k 较大的情况下,非增量方法的执行效率要优于增量方法。但非增量方法不能像增量方法那样重用结果集:在算法得到了 k 个最近对之后,若有新任务请求计算 m ($m > k$)个最近对,非增量算法就要重新启动以计算 m 个最近对,而增量算法只需继续计算再得到 $m-k$ 个最近对即可。这是非增量方法亟待解决的问题。

文[39]首先提出了基于 R-tree,利用增量算法对 k 最近对查询进行求解的方法,并且用扩展的增量算法较好的解决了半距离(semi-distance)连接查询问题。文[42]在增量算法的基础上,综合了自适应多级和 plane-sweep 技术^[15],对 e 距离连接和增量连接查询提出了新的解法。Corral 等人^[43~44]针对如何高效的利用非增量算法解答最近对查询问题进行了较多研究。文[44]在文[43]的基础上,利用递归分支定界算法,在非增量求解方式下对数据集 A 和 B 分别建立两个 R-tree 索引,通过同时遍历两个索引树的方法为最近对查询提供了有效的求解算法,并通过试验证明了算法的高效性。此外,与 k CP 相关的三种重要查询在文[44]中也做了相关算法研究以及性能对比试验,它们分别为:1. 自 k 最近对查询(k Self-Closest Pair, k -self-CP); k 最近对查询的数据集 A 和 B 具有相同的数据对象;2. 半最近对查询(Semi-Closest Pair, Semi-CP);对于数据集 A 或 B 中的每一个对象,查询它在另外一个数据集的最近距离对象;3. k 最远对查询(k Farthest Pair, k FP);与 k 最近对查询相反,从数据集 A 和 B 中查询具有最远距离的 k 个数据对。目前为止,此三种查询相关的理论研究还不成熟,尤其是对高效解决算法的研究还相当少,但是我们相信随着移动计算理论本身的不断发展和完善,在今后几年内此类查询必将成为学者们的研究热点并在相关领域得到广泛的应用。

小结和展望 移动计算环境的迅速发展,在催生了移动计算领域许多新应用需求的同时,移动对象相关的查询处理技术亦成为近年来学者们的研究热点。在实际应用当中,针对非移动对象的各种查询处理技术已经有了相当深厚的研究

基础,此正为移动计算环境下的对象查询处理技术奠定了基石。

从基础定义到最新的研究进展,本文总结了四种主要的移动对象相关的查询处理技术:最近邻查询、反最近邻查询、空间连接以及最近对查询。纵观上述各种查询处理技术的发展历程,我们可以发现高效的查询处理算法与有效的移动对象索引技术是密不可分的,特别是如何针对频繁更新的移动对象建立高效的索引,以支持各种查询处理高效的执行仍是尚未解决的难题。此外,文中提及的时间参数化查询(time parameterized query)、持久查询(persistent query)、持续查询(continuous query)以及基于位置的查询(location-based query)^[47, 48],作为新兴的查询应用需求,在研究中尚处于起始阶段,如何将这几种查询同传统的查询处理技术相结合以满足新的用户需求,将是一个充满挑战和极具潜力的研究方向。

参 考 文 献

- 1 Wolfson O, Xu B, Chamberlain S, et al. Moving object databases: Issues and solutions. In: The 10th Int'l Conf. on Science and Statistical Database Management, Capri, Italy, 1998, 11~122
- 2 Wolfson O, Chamberlain S, Dao S, et al. Location management in moving objects databases. In: The Second Int'l Workshop on Satellite-Based Information Services, Budapest, Hungary, 1997
- 3 Benetis R, Jensen C S, Karcauskas G, et al. Nearest and reverse nearest neighbor queries for moving objects. In: The VLDB Journal, 2006, 15 (3): 229~250
- 4 Papadias D, Zhang J, Mamoulis N, et al. Query Processing in Spatial Network Databases. In: Proceeding of the 29th VLDB conference, Berlin, Germany, 2003, 12
- 5 Guttman A. R-trees: A dynamic index structure for spatial searching. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1984, 47~57
- 6 Roussopoulos N, Kelley S, Vincent F. Nearest neighbor queries. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1995, 71~79
- 7 Hjaltason G R, Samet H. Distance browsing in spatial databases. ACM Transactions on Database System, 1999, 24 (2): 265~318
- 8 Papadopoulos A, Manolopoulos Y. Performance of Nearest Neighbor Queries in R-Trees. In: Proceedings of the 6th International Conference on Database Theory, 1997, 394~408
- 9 Katayama N, Satoh S. The SR-tree: An index structure for high-dimensional nearest neighbor queries. In: Proceedings of the International Conference on Management of Data, 1997, 369~380
- 10 White D A, Jain R. Similarity indexing with the SS-tree. In: Proceedings of the International Conference on Data Engineering, 1996, 516~523
- 11 Beckmann N, Kriegel H P, Schneider R, et al. The R*-tree: An efficient and robust access method for points and rectangles. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1990, 322~331
- 12 Korn F, Sidiropoulos N, Faloutsos C, et al. Fast nearest neighbor search in medical image databases. In: Proceedings of the VLDB Conference, 1996, 215~226
- 13 Seidl T, Kriegel H P. Optimal multi-step k-nearest neighbor search. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1998, 154~165
- 14 Albers G, Guibas L J, Mitchell J S B, et al. Voronoi diagrams of moving points. Int J Comput Geom Appl, 1998, 8(3): 365~380
- 15 Preparata F P, Shamos M I. Computational geometry: An introduction (texts and monographs in computer science), 5th edn. Springer, Berlin, Heidelberg, New York, 1993
- 16 Kollios G, Gunopulos D, Tsotras V J. Nearest neighbor queries in a mobile environment. In: Proceedings of the International Workshop on Spatio-Temporal Database Management, 1999, 119~134
- 17 Raptopoulou K, Papadopoulos A, Manolopoulos Y. Fast nearest-neighbor query processing in moving-object databases. GeoInformatica, 2003, 7(2): 113~137
- 18 Tao Y, Papadias D. Spatial queries in dynamic environments. ACM Transactions on Database System, 2003, 28(2): 101~139
- 19 Song Z, Roussopoulos N. K-nearest neighbor search for moving query point. In: Proceedings of the International Symposium on Spatial and Temporal Databases, 2001, 79~96
- 20 altenis S, Jensen C S, Leutenegger S T, et al. Indexing the positions of continuously moving objects. In: Proceedings of the ACM SIGMOD, 2000, 331~342
- 21 Tao Y, Papadias D, Sun J. The TPR*-tree: an optimized spatio-temporal access method for predictive queries. In: Proceedings of the VLDB Conference, 2003, 790~801
- 22 Korn F, Muthukrishnan S. Influence sets based on reverse nearest neighbor queries. In: Proceedings of the ACM SIGMOD, 2000, 201~212
- 23 Stanoi I, Agrawal D, El Abbadi A. Reverse nearest neighbor queries for dynamic databases. In: Proceedings of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 2000, 44~53
- 24 Yang C, Lin K-Ip. An index structure for efficient reverse nearest neighbor queries. In: Proceedings of the International Conference on Data Engineering, 2001, 485~492
- 25 Smid M. Closest point problems in computational geometry. In: Sack J R, Urrutia J. (eds). Handbook on computational geometry. Elsevier Science Publishing, 1997, 877~935
- 26 Maheshwari A, Vahrenhold J, Zeh N. On reverse nearest neighbor queries. In: Proceedings of the Canadian Conference on Computational Geometry, 2002, 128~132
- 27 Singh A, Ferhatosmanoglu H, Tosun A. High dimensional reverse nearest neighbor queries. In: Proceedings of International Conference on Information and Knowledge Management, 2003, 91~98
- 28 Tao Y, Papadias D, Lian, X. Reverse kNN search in arbitrary dimensionality. In: Proceedings of the VLDB Conference, 2004, 744~755
- 29 Tao Y, Papadias D, Lian X, et al. Multi-dimensional Reverse kNN Search. To appear in Very Large Data Base Journal
- 30 Tao Y, Yiu M, Mamoulis N. Reverse Nearest Neighbor Search in Metric Spaces. IEEE Transactions on Knowledge and Data Engineering (TKDE), 2006, 18(9): 1239~1252
- 31 Brinkhoff T, Kriegel H P, Seeger B. Efficient Processing of Spatial Joins Using R-trees. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1993
- 32 Lo M L, Ravishankar C V. Spatial Joins Using Seeded Trees. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1994
- 33 Mamoulis N, Papadias D. Integration of Spatial Join Algorithms for Processing Multiple Inputs. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1999
- 34 Patel J M, DeWitt D J. Partition Based Spatial-Merge Join. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1996
- 35 Arge L, Procopiuc O, Ramaswamy S, et al. Scalable Sweeping-Based Spatial Join. VLDB, 1998
- 36 Rigaux P, Scholl M, Voisard A. Spatial Databases with Application to GIS. Morgan Kaufmann, 2002
- 37 Sun J, Tao Y, Papadia D, et al. Spatio-Temporal Join Selectivity. Information Systems (IS), 2006, 31: 793~813
- 38 Zhang J, Mamoulis N, Papadias D, et al. All-nearest-neighbors Queries in Spatial Databases. In: Proceedings of the 15th IEEE Conference on Scientific and Statistical Database Management (SSDBM), Santorini, Greece, June 2004, 297~306
- 39 Hjaltason G R, Samet H. Incremental Distance Join Algorithms for Spatial Databases. In: Proceeding ACM SIGMOD, 1998, 237~248
- 40 Hjaltason G R, Samet H. Ranking in spatial databases. In: Proceedings 4th SSD Conference, 1995, 83~95
- 41 Hjaltason G R, Samet H. Distance browsing in spatial databases. ACM Transactions on Database Systems, 1999, 24 (2): 265~318
- 42 Shin H, Moon B, Lee S. Adaptive multi-stage distance join processing. In: Proceedings ACM SIGMOD Conference, 2000, 343~354
- 43 Corral A, Manolopoulos Y, Theodoridis Y, et al. Closest Pair Queries in Spatial Databases. SIGMOD, 2000; 189~200
- 44 Corral A, Manolopoulos Y, Theodoridis Y, et al. Algorithms for Processing k Closest Pair Queries in Spatial Databases. Data and Knowledge Engineering, 2004, 49: 67~104
- 45 Weber R, Schek H, Blott S. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In: Proceedings of Very Large Data Base Conference (VLDB) (Aug). 1998, 194~205
- 46 Peng Wenchin, Chen Mingsyan. Query Processing in a Mobile Computing Environment; Exploiting the Features of Asymmetry. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(7): 982~996
- 47 Zheng B, Lee D. Semantic caching in location-dependent query processing. In: Proceedings of Symposium on Spatial and Temporal Databases (SSTD), July, 2001, 97~116
- 48 Zhang J, Manli Z, Papadias D, et al. Location-based spatial queries. In Proceedings of the ACM SIGMOD Conference (June). ACM, New York, 2003