

基于道路网络不确定移动对象的连续概率 Skyline 查询

付世昌 董一鸿 陈华辉 钱江波

(宁波大学计算机科学与技术研究所 宁波 315211)

摘 要 Skyline 查询是基于位置服务 LBS 的一项重要操作,其目的是发现数据集中不被其它点支配的点的集合。对道路网络环境下移动对象的连续概率 Skyline 查询进行了研究。在对道路网络和移动对象建模的基础上,定义了基于道路网络的数据间支配概率和 Skyline 概率的表示方式,提出了两类可能引起 p-Skyline 集合变动的 event 事件,并提出 4 条剪枝方案进行优化。在此基础上,设计了对网络受限的不确定移动对象进行连续概率 Skyline 查询的动态增量算法 U-CPSQRN。该算法通过对 event 的跟踪计算实现了对 p-Skyline 的连续更新操作,减少了算法的查找和计算开销。实验结果显示了算法的有效性。

关键词 概率 Skyline,支配概率,道路网络,不确定移动对象

中图法分类号 TP311 文献标识码 A

Continuous Probabilistic Skyline Queries Based on Road Network for Uncertain Moving Object

FU Shi-chang DONG Yi-hong CHEN Hua-hui QIAN Jiang-bo

(Institute of Computer Science and Technology, Ningbo University, Ningbo 315211, China)

Abstract Skyline queries are an important operator of LBS, which aim to find all data points that are not dominated by any others. Skyline queries for moving objects with uncertainty in road network were studied. After modeling road network and moving object, the dominant probability and skyline probability in road network environment were defined. Then, two types of event that may affect p-Skyline and four pruning rules were devised. The dynamic incremental algorithm U-CPSQRN is supposed based on the above definition. By tracking and calculating these events, the operation of continuous updated p-Skyline can be achieved, which reduces search steps and system overhead. The experiments having positive results show effectiveness of the proposed algorithm.

Keywords Probabilistic Skyline, Dominate probability, Road network, Uncertain moving object

1 引言

Skyline 查询^[1]操作提供了一种基于位置服务的重要功能,其目的是发现数据集中不被其它点支配的点的集合,在多目标决策、数据挖掘和数据库可视化等方面有着潜在的应用。

在移动对象数据库中,数据库服务器通过位置更新来获得移动对象当前的位置信息,但是由于时间的延迟和当前技术手段的限制,采集到的位置信息与真实位置存在一定偏差,即位置信息具有不确定性,导致各查询对象间的支配关系不稳定,从而影响 Skyline 操作。在现实生活中,移动对象的运动大多是受到一定条件限制的,如火车在铁轨运行,汽车沿公路行驶等。因此对移动对象的 Skyline 查询操作不仅要考虑移动对象位置的不确定性,还要把移动对象的运动环境考虑在内。

本文对道路网络环境下移动对象的概率 Skyline 查询进行了研究,主要贡献如下:

1) 分析了道路网络建模和该环境下不确定移动对象建模问题,定义了数据间支配概率和 Skyline 概率的表示方式。

2) 充分分析数据点的道路网络位置对支配关系的影响,定义了两类可能引起 p-Skyline 变动的 event 事件,提出了对不确定移动对象进行连续概率 Skyline 查询的动态增量算法 U-CPSQRN(Continuous Probabilistic Skyline Queries algorithm for Uncertain Moving Object on Road Network)。

3) 本文首次研究基于路网的不确定数据的连续概率 Skyline 查询(目前还没有类似的算法),提出一个静态算法 U-SPSQRN(Static Probabilistic Skyline Queries algorithm for Uncertain Moving Object on Road Network)并将其与 U-CPSQRN 进行对比实验,实验结果证明了 U-CPSQRN 算法的有效性。

2 相关工作

Borzsonyi 等^[2]首次将 Skyline 操作引入数据库系统,提

到稿日期:2010-08-24 返修日期:2010-11-12 本文受国家自然科学基金(60973047,60803021),浙江省自然科学基金(Y1080490),浙江省公用技术应用研究项目(2010C33149),宁波市自然科学基金(2010A610098),宁波大学研究生科研创新基金资助。

付世昌(1984—),男,硕士,主要研究方向为数据挖掘、Skyline 计算,E-mail:Fu_shichang2010@163.com;董一鸿(1969—),男,教授,主要研究方向为数据挖掘、人工智能和软计算;陈华辉(1964—),男,副教授,主要研究方向为数据库、数据挖掘和数据流处理;钱江波(1974—),男,副教授,主要研究方向为数据库、数据流及硬件技术与开发。

出了BNL和D&C两种计算方法。文献[2]将原始数据集编码成位图(bitmap),提出了一种基于比特位运算的改进方法,同时还提出了一种基于索引的方法。Kossmann等^[3]基于R-tree索引结构提出的NN算法能够接受外界交互信息,具有良好的用户友好性。Papadias^[4]分析了NN方法在I/O和存储方面存在的不足并对之进行了改进,提出了BBS算法。

近几年关于Skyline的研究主要体现在Skyline在特定应用场景中的处理方法上。Huang等^[5]提供了一种高效的对移动对象环境下Skyline进行动态维护的方法;Lee等^[6]提出了对具有多维动态属性的数据对象进行连续Skyline查询的操作算法;Jian Pei等^[7]提出了不确定数据的概率Skyline模型,将Skyline点定义为概率的形式;王亚琴^[8]提出了RNASQ算法,其将Skyline连续查询转化为对查询路径的顶点和查询对象与查询点距离的交叉点的有限个独立查询,可以有效地计算Skyline连续查询的连续分段;Huang等^[9]提出TraverseSQ, BestSQ和GeneralSQ 3种算法,解决了基于道路网络环境的移动对象连续Skyline查询问题。

3 数据模型

3.1 道路网络模型

由于道路网络结构的复杂性,需要建立数学模型来简化对道路网络的描述,即对其包含的道路进行抽象化,并建立它们之间的逻辑关系。本文给出道路网络模型的定义如下:

定义1 道路网络结构表示为一个无向带权图 $G=(V, E, W)$,其中 V 是顶点(结点)的集合, E 是路径的集合, W 为正实数集合($W(v_i, v_j): E \rightarrow R^+$,表示顶点 v_i 到 v_j 的距离),表示路径对应的权值。网络中各数据点作为查询对象位于路径 $e(e \in E)$ 上,数据点在网络中的位置用三元组 (v_i, v_j, pos) 表示,其中 v_i, v_j 是数据点所在路径 e 的两个端点, $pos \in [0, W(v_i, v_j)]$ 是数据点到结点 v_i 的距离。

3.2 移动对象不确定数据模型

根据移动对象在道路网络上的移动特性,本文采用了概率性非确定段数据模型^[10]来表示:

定义2 移动对象 M 在 t 时刻的取值范围是一个非确定段 $[L(t), U(t)]$,记作 $UI(t)$ 。其中 $L(t)$ 和 $U(t)$ 是关于时间戳 t 的实值函数,且满足 $L(t) < U(t)$, $M \in [L(t), U(t)]$ 在 t 时刻恒成立。 M 在非确定段 $UI(t)$ 内的概率密度分布函数记作 $pdf(\vec{x}, t)$, $\vec{x} \in UI(t)$ 。如果在 t 时刻 $M \notin [L(t), U(t)]$,则 $pdf(\vec{x}, t) = 0$ 。 $\vec{v}_M(t)$ 是 M 在 t 时刻的移动速度。

移动对象的概率性非确定段模型如图1所示,其中桔黄色线段表示的是移动对象 M 的非确定段 $UI(t)$,白色圆圈表示 M 在数据库中实际记录的位置 $\vec{x}_0$,箭头指示的是 M 的运动方向 $\vec{v}_M(t)$ 。 M 可能出现在 $UI(t)$ 中的任何位置。

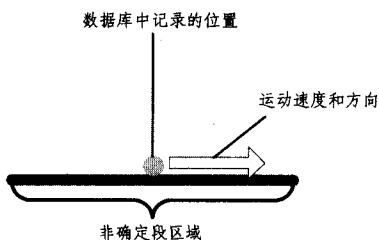


图1 非确定段模型实例

4 概率Skyline连续更新机制

假设 $S = \{\vec{s}_1, \vec{s}_2, \dots, \vec{s}_N\}$ 是 N 个被监控对象的集合,每个对象当前状态表示为 $\langle id, x_1, x_2, \dots, x_d \rangle$,其中 id 为对象的唯一标识, $x_i (i=1, 2, \dots, d)$ 是描述对象的 d 个属性,组成了一个 d 维空间。

定义3 设两个对象 $\vec{s}_1, \vec{s}_2$ 的属性分别为 $(x_1, x_2, \dots, x_d)$ 和 $(y_1, y_2, \dots, y_d)$,若 $\forall i, x_i \leq y_i$ 都成立,且 $\exists j$ 满足 $x_j < y_j (1 \leq i, j \leq d)$,则称 $\vec{s}_1$ 支配 $\vec{s}_2$,表示为 $\vec{s}_1 < \vec{s}_2$ 。

定义4 所有不被任何其它点支配的点的集合称为Skyline集合。其中每个点称为Skyline点。

对象的各维属性值,有的随移动对象的移动而不断变化,而有的随移动对象的移动并不发生变化。因此将 d -维空间分为 k 维静态属性和 m 维动态属性,其中 $d=k+m$ 。对于移动环境下的任意目标对象 $\vec{s}_i$,其静态维属性记为矢量 $\vec{s}_i^S = (x_{i1}^S, x_{i2}^S, \dots, x_{ik}^S)^T$,动态维属性记为矢量 $\vec{s}_i^D = (x_{i1}^D, x_{i2}^D, \dots, x_{im}^D)^T$ 。为了简化描述,本文假设仅有距离一项动态属性,不确定移动对象 M 的不断移动导致数据集中的点 $s_i (i=1, 2, \dots, N)$ 与 M 之间的距离不断变化。

4.1 支配概率

定义5 $\vec{s}_i, \vec{s}_j$ 是数据集 S 中两数据点, $\vec{s}_i^S$ 和 $\vec{s}_j^D$ 分别是 $\vec{s}_i$ 的静态维属性矢量和动态维属性矢量, $\vec{s}_j^S$ 和 $\vec{s}_j^D$ 分别是 $\vec{s}_j$ 的静态维属性矢量和动态维属性矢量, $\vec{s}_i$ 支配 $\vec{s}_j$ 的概率表示为:

$$\Pr(\vec{s}_i < \vec{s}_j) = \int_{UI(t)} pdf(\vec{x}) \cdot \begin{cases} 1, & \text{if } \vec{s}_i^S < \vec{s}_j^S \text{ and } |\vec{s}_j^D - \vec{x}| < |\vec{s}_i^D - \vec{x}| \\ 0, & \text{otherwise} \end{cases} d\vec{x} \quad (1)$$

式中, $|\vec{s}_i^D - \vec{x}|$ 表示数据点 $\vec{s}_i$ 到非确定段 $UI(t)$ 中位置点 $\vec{x}$ 的最短路径长度。

4.2 Skyline概率

定义6 $\vec{s}_i$ 是 S 中一数据点,则 $\vec{s}_i$ 不被集合 S 中任何其它数据点所支配的概率即为 $\vec{s}_i$ 点的Skyline概率。公式化表示为:

$$\Pr(\vec{s}_i) = \Pr(\bigwedge_{\vec{s}_j \in S} \vec{s}_j \not< \vec{s}_i) = \int_{UI(t)} pdf(\vec{x}) \cdot \left(\prod_{\vec{s}_j \in S} \begin{cases} 0, & \text{if } \vec{s}_j^S < \vec{s}_i^S \text{ and } |\vec{s}_j^D - \vec{x}| < |\vec{s}_i^D - \vec{x}| \\ 1, & \text{otherwise} \end{cases} \right) d\vec{x} \quad (2)$$

式中, $\vec{s}_j \not< \vec{s}_i$ 表示 $\vec{s}_i$ 不被 $\vec{s}_j$ 所支配, $\vec{s}_i$ 的Skyline概率的有效支配区域即 $\vec{s}_i$ 不被其它数据点支配的有效支配区域交集。

定义7 给定阈值 $p \in [0 \dots 1]$,对集合 S 中的任意数据点 $\vec{s}_i$,如果 $\Pr(\vec{s}_i) \geq p$,则 $\vec{s}_i$ 属于概率Skyline集合,即:

$$sky(p) = \{\vec{s}_i \in S | \Pr(\vec{s}_i) \geq p\} \quad (3)$$

4.3 影响p-Skyline的event事件

移动对象沿道路网络的移动,造成各数据对象间支配关系不断变化。举一简单例子,不确定移动对象 M 沿路径 (v_i, v_j) 向前移动, L 是 $|PQ| (|PQ| = |PM| + |QM|)$, $|PM|$ 、 $|QM|$ 分别是数据点 $\vec{P}, \vec{Q}$ 到 M 的最短路径的垂直平分线。假设 $\vec{P}^S < \vec{Q}^S$ 成立,在 t_1 时刻, M 的不确定段 $UI(t)$ 边界恰好与 L 相切(如图2(a)所示),当前 $\Pr(\vec{P} < \vec{Q}) = 1$,记 M 当前在

路径中的位置为 $(v_i, v_j, begin)$; 到 t_2 时刻 M 位于 L 上 (如图 2(b) 所示), L 将 $UI(t)$ 分为大小相等的两部分, $\Pr(\vec{P} < \vec{Q}) = 0.5$; 当 $UI(t)$ 边界恰好越过 L 时 (如图 2(c) 所示), $\Pr(\vec{P} < \vec{Q}) = 0$, 记 M 此时在路径中的位置为 (v_i, v_j, end) 。由此不难看出, M 沿路径 (v_i, v_j) 移动时, $\Pr(\vec{P} < \vec{Q})$ 只有在区域 $[begin, end]$ 才发生变化, 而 M 在其它位置点 $pos \notin [begin, end]$ 移动时, 并没有带来 $\Pr(\vec{P} < \vec{Q})$ 的变化, 因此我们将 $[begin, end]$ 记为 $\vec{P}, \vec{Q}$ 的有效支配区域。

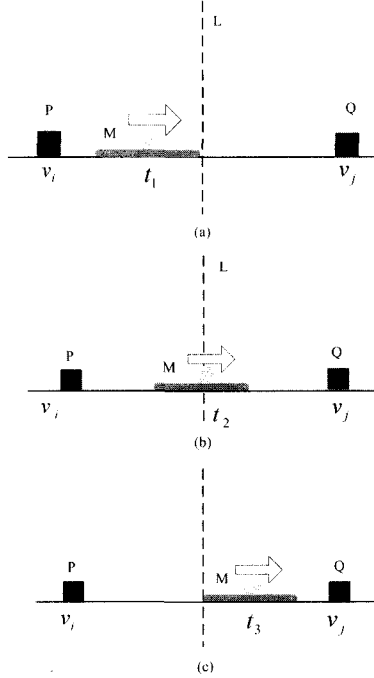


图2 支配关系变化示意

由式(2)知, 支配概率的变化可能引起 Skyline 概率的变化, 通过上述分析可以得知, 对不确定移动对象进行连续概率 Skyline 查询的过程可以归结为对各数据点间有效支配区域进行连续更新计算的过程。因此我们将移动对象经过数据点间有效支配区域的过程定义为一个 event 触发事件, Skyline 查询的操作也就等价于对各 event 进行的连续更新计算。本文定义了如下两种 event 事件:

定义 8 道路网络中任意两位置点 $pos_1, pos_2 \in [begin, end]$, 假设 $pos_1 < pos_2$, 对于集合 S 中满足 $\vec{x}_i^s < \vec{x}_j^s$ 的数据点 $\vec{P}, \vec{Q}$, 若 $\Pr(\vec{Q} < \vec{P})|_{pos_1} > \Pr(\vec{Q} < \vec{P})|_{pos_2}$ 成立, 则 $\vec{P}, \vec{Q}$ 产生一个 Up event, 表示为 $\langle \vec{P}, \vec{Q}, begin, end, UP \rangle$ 。其中 $[begin, end]$ 是 $\Pr(\vec{Q} < \vec{P})$ 的有效支配区域。

定义 9 道路网络中任意两位置点 $pos_1, pos_2 \in [begin, end]$, 假设 $pos_1 < pos_2$, 对于集合 S 中满足 $\vec{x}_i^s < \vec{x}_j^s$ 的数据点 $\vec{P}, \vec{Q}$, 若 $\Pr(\vec{Q} < \vec{P})|_{pos_1} > \Pr(\vec{Q} < \vec{P})|_{pos_2}$ 成立, 则 $\vec{P}, \vec{Q}$ 产生一个 Down event, 表示为 $\langle \vec{P}, \vec{Q}, begin, end, Down \rangle$ 。其中 $[begin, end]$ 是 $\Pr(\vec{Q} < \vec{P})$ 的有效支配区域。

4.4 剪枝策略

由 event 定义可知, 每一数据点都要与对其存在支配可能性的数据点计算可能产生的 event, 但是有些 event 对 p-Skyline 集合并不产生任何影响, 因此提出如下几条剪枝规则, 具体描述如下:

剪枝规则 1 数据点 $\vec{s}_i, \vec{s}_j$ 产生 Down event $(\vec{s}_i, \vec{s}_j)$ 且 $\Pr(\vec{s}_j < \vec{s}_i)|_{end} = 1$, 则 $\forall event(\vec{s}_i, \vec{s}_k) (k \neq i, j \text{ 且 } 0 < k < N)$, 如果

$event(\vec{s}_i, \vec{s}_k).begin \geq event(\vec{s}_i, \vec{s}_j).end$, 则 $event(\vec{s}_i, \vec{s}_k)$ 属于无效 event。

证明: 对于任意满足 $\Pr(\vec{s}_j < \vec{s}_i)|_{end} = 1$ 的 Down event $(\vec{s}_i, \vec{s}_j)$, 在移动对象完全经过其有效支配区域 $[begin, end]$ 之后, $\Pr(\vec{s}_j < \vec{s}_i) = 1$ 恒成立。由式(2)得 $\Pr(\vec{s}_i) = 0$ 恒成立, 因此 event $(\vec{s}_i, \vec{s}_k)$ 的更新计算对 $\Pr(\vec{s}_i)$ 无任何影响, 属于无效 event。

剪枝规则 2 对于任意满足 $\Pr(\vec{s}_j < \vec{s}_i)|_{begin} = 1$ 的 Up event $(\vec{s}_i, \vec{s}_j)$, 如果 $\exists event(\vec{s}_i, \vec{s}_k) (0 < k < N \text{ 且 } k \neq i)$ 且 $event(\vec{s}_i, \vec{s}_k).begin < event(\vec{s}_i, \vec{s}_j).begin$, 则在 $event(\vec{s}_i, \vec{s}_j).begin$ 之前对 $event(\vec{s}_i, \vec{s}_k)$ 的计算属于无效计算; 如果 $event(\vec{s}_i, \vec{s}_k).end \leq event(\vec{s}_i, \vec{s}_j).begin$, 则 $event(\vec{s}_i, \vec{s}_k)$ 属于无效 event。

证明: 根据式(2), 由于 $\Pr(\vec{s}_j < \vec{s}_i)|_{begin} = 1$, 则得出在 event $(\vec{s}_i, \vec{s}_j).begin$ 时刻之前 $\Pr(\vec{s}_i) = 0$ 恒成立。在 $event(\vec{s}_i, \vec{s}_j).begin$ 之前对 $event(\vec{s}_i, \vec{s}_k)$ 进行更新计算, 根据式(2)可以得出 $\Pr(\vec{s}_k < \vec{s}_i)$ 的变化对 $\Pr(\vec{s}_i)$ 无任何影响, 因此其计算属于无效计算; 同理, 如果 $event(\vec{s}_i, \vec{s}_k)$ 在 $event(\vec{s}_i, \vec{s}_j).begin$ 之前结束, 此 event 对 $\Pr(\vec{s}_i)$ 无任何影响, 因此属于无效 event。

剪枝规则 3 假设 Up event $(\vec{s}_i, \vec{s}_m)$ 是所有与数据点 $\vec{s}_i$ 相关且 $\Pr(\vec{s}_m < \vec{s}_i)|_{begin} = 1 (1 \leq m \leq N)$ 的 Up event 中 begin 最大的 event, 记其 begin 为 start; 假设 Down event $(\vec{s}_i, \vec{s}_n)$ 是所有与数据点 $\vec{s}_i$ 相关且 $\Pr(\vec{s}_n < \vec{s}_i)|_{end} = 1$ 的 Down event 中 end 最小的 event, 记其 end 为 finish; 如果 $finish \leq start$, 则所有与 $\vec{s}_i$ 相关的 event 都是无效 event。

证明: 由剪枝规则 2 得, 在 start 之前对 $\forall event$ 的更新计算都是无效计算; 由剪枝规则 1 得在 finish 之后的 $\forall event$ 都是无效 event, 对其更新计算也属于无效计算, 当 $finish \leq start$ 时, 很显然对于与 $\vec{s}_i$ 相关的 event 的更新计算都属于无效计算, 即这些 event 都是无效的。

剪枝规则 4 在移动对象沿任意路径 (v_k, v_h) 移动的过程中, 如果存在 $event(\vec{s}_i, \vec{s}_j)$, 其有效支配区域 $[begin, end] \cap (v_k, v_h) = \emptyset$ 且 $\Pr(\vec{s}_j < \vec{s}_i) = 1$ 成立, 则在此移动过程中, 所有与 $\vec{s}_i$ 相关的 $event(\vec{s}_i, \vec{s}_k) (k \neq i \text{ 且 } 0 < k < N)$ 都是无效 event。

证明: 对 $event(\vec{s}_i, \vec{s}_j)$, 由于 $[begin, end] \cap (v_k, v_h) = \emptyset$, 则在移动对象沿路径 (v_k, v_h) 移动过程中 $(\vec{s}_j < \vec{s}_i) = 1$ 恒成立, 根据式(2)得 $\Pr(\vec{s}_i) = 0$ 恒成立, 因此 $\vec{s}_i$ 在此移动过程中始终是非 Skyline 点, 则 $\forall event(\vec{s}_i, \vec{s}_k) (k \neq i \text{ 且 } 0 < k < N)$ 都是无效的。

5 算法实现策略

在算法中利用全局双向链表 L_{sp} 存储当前全部的 p-Skyline 点, Q_c 存储可能影响 p-Skyline 集合的 event, Q_{bundle} 用于存储连续跟踪计算的 event。

5.1 静态算法 U-SPSCRN

本文首次采用不确定模型对道路网络环境下移动对象的连续 Skyline 查询操作进行研究, 因此提出一个周期性的静态查询操作算法 U-SPSCRN 来与本文研究工作进行比较。

U-SPSCRN 主要思想: 移动对象沿道路网络移动的过程中, 每移动一段距离 dis , 就在移动对象的当前位置对整个数据集进行 p-Skyline 操作一次, 直到移动对象离开所要考察的路径为止。U-SPSCRN 的核心框架包括两部分:

1) 计算初始状态 p-Skyline 集合, 该阶段需要对算法进行数据初始化操作, 包括建立道路网络的无向图, 计算数据

点间静态支配关系、支配概率及初始 p-Skyline 等。

2) 周期性地查询移动对象移动过程中的 Skyline。该阶段需要根据当前移动对象的非确定段来更新计算数据点间的支配概率, 及各数据点的 Skyline 概率, 从而更新 p-Skyline 集合。

5.2 算法 U-CPSCRN

U-CPSCRN 算法的主要思想是对移动对象经过的每一子路径, 计算可能引起 p-Skyline 集合发生变动的 event, 通过对这些 event 的跟踪计算实现对 p-Skyline 的不断更新, 从而避免在移动对象的每一更新时刻遍历整个数据集。

算法 1 给出了算法 U-CPSCRN 的主程序。对初始路径要进行算法的初始化操作, 具体内容与静态算法类似; 对一非初始路径, 要更新路网中各数据点到移动对象的最短路径; 算法 CreateEvent 用于计算可能发生的 event 事件, HandleEvent 用于更新计算 event 并不断更新 p-Skyline 集合。

Algorithm 1 U-CPSCRN()

Input: M is the uncertain moving object

Output: the initial p-Skyline set

1. for each path during the movement of M
2. if the path is the initial path
3. Initial(); // 计算初始 p-Skyline
4. else
5. update the Shortestpath for every point to M ;
6. CreatEvent();
7. HandleEvent();

CreateEvent 算法用于计算移动对象沿某一路径移动过程中可能产生的 event 事件。对于存在静态支配的任意两数据点 $\vec{P}, \vec{Q}$, 首先计算移动对象移动过程中, 造成 $\Pr(\vec{P} < \vec{Q})$ 发生变化的有效支配区域 $[begin, end]$ 。如果 $[begin, end]$ 与当前移动对象所位于的路径无交集, 即 $\vec{P}, \vec{Q}$ 的有效支配区域在路径区域外, 则 $\vec{P}$ 与 $\vec{Q}$ 在此移动过程中不产生 event 事件; 否则要判定 event 类型并根据剪枝规则判定 event 的有效性, 然后将有效的 event 插入到 Q_e 中。

Algorithm 2 CreateEvent()

1. for two points $\vec{P}, \vec{Q}$ in S
2. if ($\vec{P}^s < \vec{Q}^s$)
3. compute the $[begin, end]$ of $\vec{P}^s < \vec{Q}^s$
4. if ($[begin, end]$ is valid)
5. compute event $(\vec{Q}, \vec{P})$;
6. insert $(\vec{Q}, \vec{P}, begin, end, type)$ into Q_e ;

HandleEvent 算法对可能影响 p-Skyline 的 event 进行更新计算。将 $e.begin < dis$ 的 event 从 Q_e 中移除, 将其插入 Q_{handle} 。Update 函数对 Q_{handle} 中的任意 $e_{(self, peer)}$ 更新计算 $Pr(peer < self)$ 和 $Pr(self)$ 。若 $Pr(self) \geq p$ 且 $self \notin L_{sp}$, 将其插入 L_{sp} ; 若 $Pr(self) < p$ 且 $self \in L_{sp}$ 将其从 L_{sp} 删除; 如果当前时刻 dis 大于 $e.end$, 将 event 从 Q_{handle} 中删除。递归运行上述操作, 直到 Q_e 和 Q_{handle} 都为空为止。

Algorithm 3 HandleEvent()

1. while (Queue(Q_e) != NULL)
2. $e = DeQueue(Q_e)$;
3. while ($e.begin \leq dis$)
4. EnQueue(Q_{handle}, e);
5. $e = DeQueue(Q_e)$;
6. Update(Q_{handle});

7. length = $e.begin$;

6 实验结果分析

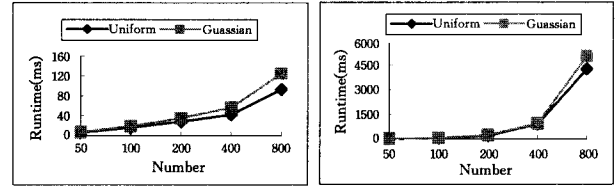
实验在 Windows XP 平台, Intel(R) Core(TM) 2 Duo, 2.93GHz CPU, 3GB RAM 的 PC 机上进行。

测试数据是通过随机数算法, 在浙江省慈溪市真实路网数据上随机生成的。实验中采用了两种非确定段的概率分布函数: 均匀分布和高斯分布, 其中实验采用的高斯分布函数为 $pdf(x) = \frac{1}{2\pi\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$, 本文假设 $\mu=0, \sigma=1$ 。p-Skyline 概率阈值 $p=0.3$ 。

6.1 数据规模对算法的影响

固定静态维空间维度 $d=2$, 非确定段长度 $Length=12$, 移动对象遍历路径 $Path=3$ 。

图 3 显示了随着数据规模 N 的增大, 在两种概率分布模型中, 两种算法的平均处理时间都不断增加, 这主要是因为数据规模的扩大导致数据间静态支配关系及支配概率计算次数的增加; 由于高斯模型的不规则程度高, 其算法运算时间要大于均匀分布模型; 算法 U-SPSCRN 需要在每一更新时刻, 重新遍历计算整个数据集, 而 U-CPSCRN 算法只有在移动对象经过一新路径时, 才会重新遍历整个数据集, 其它运动时刻, 仅对与 event 相关的数据点进行更新操作, 减少了遍历与计算的时间开销, 因此算法 U-CPSCRN 的平均运算时间远远少于 U-SPSCRN。



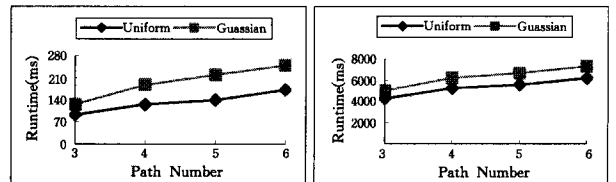
(a) U-CPSCRN 算法运算时间 (b) U-SPSCRN 算法运算时间

图 3 数据规模对算法的影响

6.2 遍历路径对算法的影响

固定数据规模 $N=800$, 静态维空间维度 $d=2$, 非确定段长度 $Length=12$ 。

由于道路网络的复杂性, 移动对象每移动到一个新路径时, 可能带来其它数据点到移动对象的最短路径的变化, 从而造成支配关系发生改变。因此当移动对象遍历至新路径时, 算法 U-CPSCRN 需要重新计算 event 事件, 遍历路径数量增加, 计算次数增大, 算法的平均处理时间将增大; 而 U-SPSCRN 算法需要在移动对象的每一运动时刻, 更新计算各数据点间的支配概率和 Skyline 概率, 路径数量的增加使更新计算次数和算法平均处理时间都将增大, 图 4 验证了上述分析。由于高斯分布模型的复杂性, 均匀分布模型在运算时间上的优势与之相比较依然存在。



(a) U-CPSCRN 规模 (b) U-CPSCRN 算法运算时间

图 4 遍历路径对算法的影响

6.3 静态维度

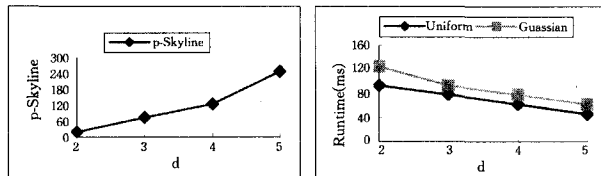
固定数据规模 $N=800$, 非确定段长度 $Length=12$, 移动对象遍历路径 $Path=3$ 。

对任意数据点 $\vec{P}, \vec{Q}$, 当 d 增大时, $\exists$ 两静态维 $i, j (1 \leq i, j \leq k \& \& i \neq j)$ 使 $x_i^P < x_j^Q$ 和 $x_j^P > x_i^Q$ 成立的概率增大, 导致数据点间存在静态支配的概率减少, 产生 event 的概率降低, 表 1 验证了上述分析, 剪枝后的 event 数量与剪枝前相比大大减少, 体现了剪枝规则的有效性。

表 1 剪枝前后 event 数量比较

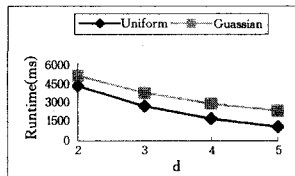
静态维度	event 规模	
	剪枝前	剪枝后
2	2084	39
3	1357	50
4	861	75
5	388	53

同理, 对任意数据点 $\vec{P}$ 来讲, 维度 d 的增大导致数据点 $\vec{Q}$ 使 $x_i^Q < x_i^P$ 成立的概率降低, 则 $\vec{P}$ 成为 p-Skyline 点的概率增大, p-Skyline 的规模随 d 的增大而不断增大, 图 5(a) 验证了上述分析。产生 event 的概率降低, 算法的平均处理时间减少, 图 5(b)、图 5(c) 验证了上述分析。



(a) p-Skyline 规模

(b) U-CPSCRN 算法运算时间



(c) U-SPSCRN 算法运算时间

图 5 静态维度对算法的影响

结束语 针对道路网络环境下不确定移动对象的连续概率 Skyline 计算问题进行了研究。首先选择路段模式作为路

网的建模方式; 然后提出了各数据点间支配概率和 Skyline 概率的表示方式, 并定义了两类可能引起 p-Skyline 集合变动的 event 事件, 提出了对不确定移动对象进行连续概率 Skyline 计算的动态增量算法框架 U-CPSCRN 和对比的静态算法 U-SPSCRN; 最后利用大量的实验分析, 从数据规模、路径数量、静态维度等几个方面论证了 U-CPSCRN 算法的有效性。

参考文献

- [1] Borzsonyi S, Kossmann D, Stocker K. The Skyline operator [C] // Proceedings of the Int'l. Conf. on Data Engineering. Heidelberg, Germany, 2001: 421-430
- [2] Tan K, Eng P, Ooi B. Efficient progressive skyline computation [C] // Proceedings of the Int'l. Conf. on Very Large Data Bases. Roma, Italy, 2001: 301-310
- [3] Kossmann D, Ramsak F, Rost S. Shooting stars in the sky: An online algorithm for skyline queries [C] // Proceedings of the Int'l. Conf. on Very Large Databases. Hong Kong, China, 2002: 275-286
- [4] Papadias D, Tao Y. Progressive skyline computation in database systems [J]. ACM Transactions on Database Systems, 2005, 30 (1): 41-82
- [5] Huang Zhi-yong, Lu Hua, Ooi B, et al. Continuous skyline queries for moving objects [J]. IEEE Transactions on Knowledge and Data Engineering, 2006, 18(12): 1645-1658
- [6] Lee W, Hwang S. Continuous skyline on volatile moving data [C] // Proceedings of the 25th IEEE Int'l. Conf. on Data Engineering. Shanghai, China, 2009: 1568-1575
- [7] Pei Jian, Jiang Bin, Lin Xue-ming, et al. Probabilistic skylines on uncertain data [C] // Proceedings of the 33th Int'l. Conf. on Very Large Databases. Trondheim, Norway, 2005: 253-264
- [8] 王亚琴. 道路交通流数据挖掘研究 [D]. 上海: 复旦大学, 2007
- [9] Huang Xue-gang, Christian S. In-Route skyline querying for location-based services [M]. Web and Wireless Geographical Information Systems, 2005: 120-135
- [10] 丁晓峰. 移动计算环境下非确定数据的索引与查询方法研究 [D]. 武汉: 华中科技大学, 2007

(上接第 147 页)

- [3] Wadler P. Comprehending Monads [J]. Math. Struct. in Comp. Sci., 1992, 2: 461-493
- [4] Wadler P. Monads for Functional Programming [C] // Jeuring J, Meijer E, eds. Advanced Functional Programming. Springer Lecture Notes of Computer Science (925). 1995
- [5] 袁华强, 孙永强. 基于 Monad 的纯函数式语言通道系统设计 [J]. 计算机科学, 2004, 31(3): 167-169
- [6] Kieburtz R B. Codata and Comonads in Haskell [R]. Unpublished Manuscript, 1999
- [7] Lewis J R, Shields M B, Meijer E, et al. Implicit Parameters; Dynamic Scoping with Static Types [C] // Proc. of 27th ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages (PoPL'00). ACM Press, 2000: 108-118

- [8] Uustalu T, Vene V. Comonadic Notions of Computation [J]. Electronic Notes in Theoretical Computer Science, 2008, 203: 263-284
- [9] Uustalu T, Vene V. Signals and Comonads [J]. Journal of Universal Computer Science, 2005, 22(7): 1310-1326
- [10] Brookes S, Geva S. Computational Comonads and Intensional Semantics [C] // Fourman M P, Johnstone P T, Pitts A M, eds. Applications of Categories in Computer Science. London Math. Society Lecture Note Series 177. Cambridge Univ. Press, 1992: 1-44
- [11] Uustalu T, Vene V. Comonadic Functional Attribute Evaluation [C] // van Eekelen M, ed. Trends in Functional Programming 6. Intellect, 2005: 145-162