

## 半限制空间内的 RFID 可能性 $k$ -近邻查询技术\*

谷 峪<sup>1,2</sup>, 于 戈<sup>1,2+</sup>, 李传文<sup>1</sup>

<sup>1</sup>(东北大学 信息科学与工程学院, 辽宁 沈阳 110004)

<sup>2</sup>(医学影像计算教育部重点实验室(东北大学), 辽宁 沈阳 110004)

### RFID Probable $k$ -NN Query Techniques in the Semi-Constraint Space

GU Yu<sup>1,2</sup>, YU Ge<sup>1,2+</sup>, LI Chuan-Wen<sup>1</sup>

<sup>1</sup>(School of Information Science and Engineering, Northeastern University, Shenyang 110004, China)

<sup>2</sup>(Key Laboratory of Medical Image Computing of Ministry of Education (Northeastern University), Shenyang 110004, China)

+ Corresponding author: E-mail: yuge@ise.neu.edu.cn

Gu Y, Yu G, Li CW. RFID probable  $k$ -NN query techniques in the semi-constraint space. *Journal of Software*, 2012, 23(3): 565-581. <http://www.jos.org.cn/1000-9825/4060.htm>

**Abstract:** As a promising technology for monitoring and tracing the vehicle flows and human activities, radio frequency identification (RFID) has received much attention in database community.  $k$ -nearest neighbor ( $k$ -NN) query over RFID monitored objects is one of the most important spatio-temporal queries used to support valuable information analysis. Different from the constraint-free space and constraint-based space, however, RFID monitoring scenario is usually merged into a semi-constraint space, which desires new data storage and distance evaluation strategies. Furthermore, the uncertainty of the monitored object locations challenges the query semantics and processing methods. In this paper, the concept of semi-constraint space is proposed, and the RFID-based semi-constraint space model is analyzed. Based on the semi-constraint space, three models and algorithms are proposed to estimate the probable  $k$ -NN results given a dynamic query point. Some special indexing techniques are adopted to speed the query. The experiment evaluates the efficiency and accuracy of the proposed algorithms and proves the effectiveness of relative methods.

**Key words:** RFID;  $k$ -NN; moving object; semi-constraint space; uncertain; continuous query

**摘 要:** 作为一种监控与跟踪车流和人类活动等的潜在技术,RFID(radio frequency identification)已经在数据库领域得到了很大关注.RFID 监控对象上的  $k$ -近邻查询是一种最重要的时空查询,能够用来支持有价值的高层信息分析.但是,不同于没有限制的空间和基于限制的空间,RFID 监控场景通常被设置在一种半限制的空间内,需要新的存储和距离计算策略.此外,监控对象位置的不确定性对查询语义和处理方法提出了挑战.提出了半限制空间的概念,并且分析了基于 RFID 的半限制空间的模型.基于半限制空间,在给定一个动态查询点的基础上,提出了 3 种模型和算法以有效地估计可能性  $k$ -近邻的查询结果,并采用一些特殊的索引技术加快查询的速度.实验对提出算法的效率和准确性进行了评估,表明了相关方法的有效性.

---

\* 基金项目: 国家自然科学基金(61003058, 60933001); 国家重点基础研究发展计划(973)(2012CB316201); 中央高校基本科研业务费专项资金(N110404006)

收稿时间: 2010-07-28; 定稿时间: 2011-05-18

关键词: RFID; $k$ -近邻;移动对象;半限制空间;不确定;连续查询

中图法分类号: TP311

文献标识码: A

无线射频识别(RFID)技术近些年经历了快速的发展,在我们的日常生活和生产中,越来越多的对象被 RFID 标签所标识,使得监控对象的时空信息能够被快速地获取.比起传统的识别方法例如条形码,RFID 技术具有自动高效的特点;而相比 GPRS 等定位设备,RFID 系统的价格又相对低廉,并且可以应用于有遮挡的空间场景.RFID 应用主要由阅读器和标签构成,标签被附着在监控对象上,当标签与阅读器的距离达到感知范围时,就能够主动发送或者被动响应唯一的标识码.基于 RFID 技术,物理世界的对象可以被有效地识别、分类和跟踪,并映射到数字世界进行处理.在供应链、智能交通、人类活动跟踪和控制等许多领域,RFID 都有着广泛的应用前景.

在 RFID 跟踪应用中,数目巨大的原始 RFID 读数实时地构成了多个数据流.这些流数据很好地反映了 RFID 监控对象的时空信息和变化过程,针对这些移动监控对象的时空信息管理也是非常必要的.常见的移动对象管理包括范围查询、 $k$ -近邻查询和轨迹查询等.在 RFID 应用场景中,这些查询能够返回有价值的查询结果,对于支持上层的分析和决策有着重要的意义.图 1 给出了一个基于 RFID 的智能博物馆场景,在这个场景中,展览的艺术品被放置于不同展厅的各个展位上,展厅之间由通道相连,每个展品都有一个 RFID 的阅读器负责监控,携带标签的游客穿梭于各个展品之间.当一个游客进入某个阅读器的感知区域时,包含监控对象 id、位置区域和感知时间的流数据就会连续地生成.在此基础上进行有效的时空查询,对于分析参观者的兴趣、优化场馆布置、提供个性化服务及应付异常情况,都有重要的意义.但是,比起较为精确的定位技术(例如 GPRS),RFID 技术具有典型的空间位置不确定性.具体的,基于 RFID 技术的空间查询,需要解决以下的挑战.因为这些挑战,获得的不是周期性的等间隔的点,而是非周期性的不等间隔的区域信息:

- 不准确性:当一个对象在某个感知区域里面,仅仅能够得知哪个阅读器感知到了这个对象,而无法获得监控对象在这个区域内的准确位置;
- 不连续性:由于成本和应用需要的考虑,阅读器不可能也不必要以全覆盖的方式进行部署.当监控对象从一个感知区域到另一个感知区域的过程中,监控对象的位置信息是无法获得的.

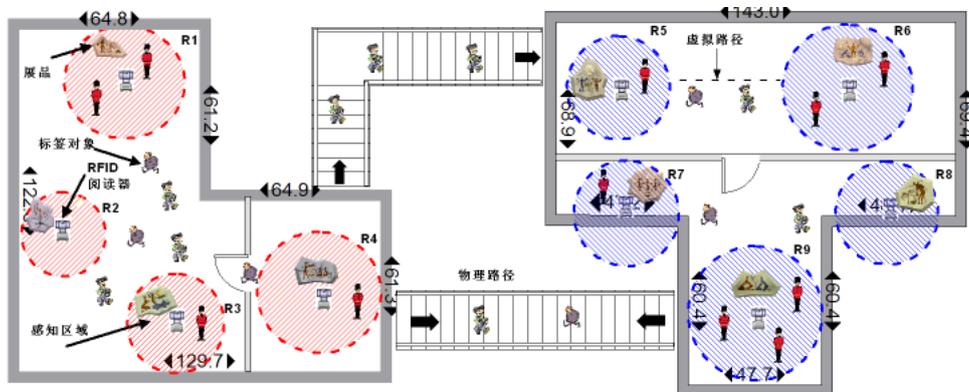


Fig.1 An RFID-enabled smart museum scenario

图 1 基于 RFID 的智能博物馆场景

特别地,不同于空间范围查询, $k$ -近邻查询是另一种移动数据管理中最重要和最复杂的查询类型,在基于 RFID 的监控场景中,有效的标签对象  $k$ -近邻查询对于支持复杂的统计分析和决策有着重要的意义.例如:在 RFID 智能博物馆中,返回某个损坏的或者丢失的展品最近的  $k$  个游客来分析相关的嫌疑人;再如,在 RFID 道路监控场景中,某个出租车查询离它最近的  $k$  个打车的行人来选择行驶路线.同样地,与范围查询类似,基于 RFID

的监控对象  $k$ -近邻查询也会受到位置不确定性的影响.由于范围查询的查询半径可以认为是确定的,只需考虑不确定的位置是否在确定的范围之内.而对于  $k$ -近邻查询,查询结果是由不同监控对象的位置关系决定的,因此位置不确定性将会对查询处理的过程带来非常大的挑战.特别对于 RFID 的位置不确定性,既存在连续的概率分布,又存在离散的概率分布,因此只能给出一个估计的可能成为  $k$ -近邻查询结果的监控对象,而无法给出具体的概率值.这就是所谓的基于 RFID 的可能性  $k$ -近邻查询.该查询作为真实  $k$ -近邻的候选集合,同样可以有效地支持上层的分析和决策.

另一方面,从空间模型上,基于 RFID 的空间查询通常存在于一种特别的空间内,对于  $k$ -近邻查询,由于监控对象的位置关系非常重要,因此必须采用更加符合查询语义的距离评估策略.现有的移动对象管理空间模型可以分成无限制(constraint free)的空间<sup>[1,2]</sup>和基于限制(constraint based)的空间<sup>[3,4]</sup>.在无限制的空间内,移动对象可以在空间的内部自由地运动,并且任意两点  $a$  和  $b$  之间的距离可以用他们的欧式距离 ED(euclidean distance)来表示.对于基于限制的空间,最典型的场景是道路交通网(road network).监控对象需要沿着某个空间的网路行进,任意两点  $a$  和  $b$  之间的距离需要基于网路距离来评估.现有的移动对象管理或者是针对无限制空间或者是针对基于限制的空间,但在真实的监控应用中,这两种空间可能会共同地合成一种半限制 SC(semi-constraint)空间.例如,图 2 说明了一个最简单的半限制空间场景,两个房间之间由一个走廊联通,某个房间对应一个无限制的空间,移动对象可以在其中自由走动.但是两个房间之间构成了一个单跳的网路,在房间外面,移动对象只能沿着走廊行进,这种情况就如同在一个基于限制的空间中移动一样.因此,一个基本的 SC 空间就被构建了.SC 空间在现实的监控场景,特别是基于 RFID 的监控场景中是非常常见的,例如对于一个室外的应用,可能存在很多大型的广场和一些停车场,它们之间通过多条道路相联通.显然地,对于单纯的道路交通网模型,点对点的单路连接不能表示这种区域与区域之间的多路连接.实际上,基于限制的空间或者无限制的空间都是 SC 空间的一种近似,都不能准确地表示 SC 场景的特点.特别对于  $k$ -近邻查询这种对于查询距离要求非常严格的查询类型,SC 空间可以提供更加合理的距离评估模型.另一方面,半限制空间对现有的评估模型和处理方法提出了挑战,特别针对移动对象进行管理并且位置存在不确定性时,需要设计高效的查询处理和优化策略.

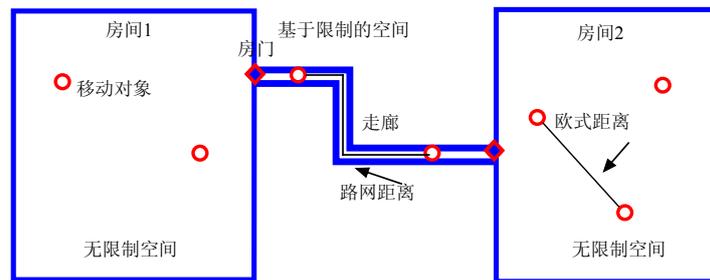


Fig.2 A simple scenario for semi-constraint space

图 2 一个简单的半限制空间场景示意图

本文首次提出了基于 RFID 应用的半限制空间模型和相应的距离评估策略,提出了对可能性  $k$ -近邻查询的 3 种由浅入深的估计方法,并提出了针对 RFID  $k$ -近邻查询的索引查询优化技术.

本文第 1 节介绍相关工作.第 2 节对半限制空间进行建模.第 3 节对基于 RFID 的半限制空间进行分析.第 4 节给出针对  $k$ -近邻查询的 3 种不同的处理策略.第 5 节通过实验对相关的模型和算法进行评估.最后总结全文并对未来工作进行展望.

## 1 相关工作

现有的 RFID 数据管理技术主要关注于数据清洗方法和复杂事件处理技术.特别地,对于 Cascadia 系统<sup>[5]</sup>和 Laha 系统<sup>[6]</sup>,从不确定性数据管理的角度,讨论了如何从低级概率的 RFID 物理读数信息推演高级的语义丰

富的复杂事件,而没有考虑到基于监控对象位置不确定性的时空查询问题.此外,文献[7]提出了在一个特定的RFID 监控仓库场景中概率定位和数据清洗的问题,但与本文的场景模型不同,阅读器是手持的和动态的,而附着标签的商品被认为是相对静止的.

对于一般的时空数据管理, $k$ -近邻查询受到了广泛的关注.但是,现有的  $k$ -近邻查询或者基于无限制的空间<sup>[8,9]</sup>或者是基于道路交通网<sup>[10,11]</sup>.特别地,基于单源最短路径(source shortest SSSP)算法<sup>[12]</sup>和基于 Diagram(例如 Vonorio-图)的方法<sup>[13]</sup>被广泛地使用以加快查询的速度,特别是针对道路交通网的场景.对于无限制空间内的移动对象的  $k$ -近邻查询,一些技术被广泛地研究.例如,基于概念分割技术,Wu 等人<sup>[1]</sup>提出了仅仅更新必要的与查询范围有相交的小方格(cell)来提高  $k$ -近邻查询的效率.但是,这些方法都没有考虑移动对象位置的不确定性和 SC 空间的特点.此外,Cheng 等人<sup>[2]</sup>首次提出了移动对象上的概率最近邻查询(1-近邻)的概念和处理方法.给定监控对象的不确定区域和查询点  $q$ ,查询返回所有具有非零概率与  $q$  距离最近的对象.为了减少最近邻查询的计算代价,Kriegel 等人<sup>[3]</sup>提出了基于蒙特卡罗(Monte Carlo)方法,用一个采样点集合表示每个对象,并在采样点集合中计算可能的最近邻居.此外,Cheng 等人<sup>[4]</sup>讨论了限制的概率最近邻查询,并采用了特殊的概率校验器(verifier).在限制的概率查询中,概率不是某个具体的值,而是通过上限和下限来约束的.但同样地,对象的位置要求服从同一种概率的分布.此外,Huang 等人<sup>[14]</sup>提出,如果在移动对象的速度和方向不确定时,基于  $TPR(s,d)$  树来有效地处理  $k$ -近邻查询.但是,这个模型是为那些准确的位置更新可以随时获取的场景来设计的.Zhang 等人<sup>[15]</sup>提出了使用一个  $b$ -树的变体结构  $b^x$ -树进行移动对象的索引,并能够服务于概率的  $k$ -近邻查询.但是,该方法针对的位置不确定性不是由于探测设备本身造成的,而是由于为了节省通信和网络资源,在两个等间隔的准确位置报告之间进行预测造成的.此外, $k$ -近邻查询可以看做是一种以距离为衡量的 top- $k$  查询,现有的设计不确定 top- $k$  查询的工作<sup>[16,17]</sup>一般是基于可能世界模型,其中,元组或者属性以离散的概率值形式在数据表给出,很少关注连续的概率分布,特别是二维空间上的情况.对于 RFID 应用,由于位置不确定性的特殊性,既存在感知区域内的连续概率分布,又存在潜在路径上的离散概率值.因此,现有的针对不确定性最近邻和  $k$ -近邻索引方法和处理技术并不适合 RFID 的监控场景.特别地,在本文提出的 SC 空间内,还需要深入地研究有效的距离评估和查询策略.

此外,Yang 等人针对室内移动对象的数据管理技术进行了研究<sup>[18-20]</sup>,可以应用于特定的 RFID 监控场景.通过将移动对象的状态分为激活状态和非激活状态,并定义关键设备进行有效的对象轨迹跟踪和范围查询.需要说明的是,与本文的场景不同,室内空间的监控需要通过假设在房门位置出现成对设备,并监控对象经过设备对的顺序,将对象分割在较粗粒度的单元区域;而本文则对各种监控场景进行了半限制空间的建模,并通过应用的特点引入虚拟路径等概念,来减小对象可能的位置范围,因此采用了不同的模型和方法.

## 2 半限制空间模型

本节将对半限制空间的一些关键问题进行建模和分析.首先给出半限制空间一些基本元素的定义.

**定义 1(块(block)).** 一个块定义为 SC 中一个基于无限制空间的组成部分,其中,任意两个对象的距离可以根据它们之间的欧式距离 ED 近似计算.本文将  $Id=m$  的块标记为  $B_m$ ,  $B_m(t)$  表示  $t$  时刻在  $B_m$  内的移动对象集合.

**定义 2(物理路径(physical path)).** 物理路径是两个块之间的受某个特定物理形状限制的通道.一个物理路径和一个块之间的交叉点,定义为成一个连接点 Joint,  $J_a$  表示  $id=a$  的 Joint,  $B(J_a)$  表示  $J_a$  所在的 block,  $pp(J_a, J_b)$  表示连接  $J_a$  和  $J_b$  的物理路径.其中,  $B(J_a) \neq B(J_b)$ .

例如,对于一个博物馆监控场景,块和连接点代表展厅和门,而物理路径代表走廊.图 3 给出了一个 SC 空间的说明.其中,每个块可能有几个连接点,并且在两个块之间可能存在不同的物理路径.在这种空间内,任意两点之间的距离需要被重新定义.不考虑 SC 空间内移动对象的动态性,SC 空间信息可以被建模成一个带权重的网络图,称为 SCN(SC space network graph).

为了构建一个有效的 SCN,首先,在同一个块中连接任意两个连接点构成一条边,并标记对应的欧式距离 ED;如果位于不同块内的两个连接点存在物理路径,连接它们构成一条边,并根据物理路径的形状来标记对应

的网络距离.按照这种方式,所有的连接点构成了 SCN 中的结点,边上的权重表示了任意两点之间的最短距离.例如,图 3 中所示的 SC 空间对应的 SCN 在图 4 中给出.

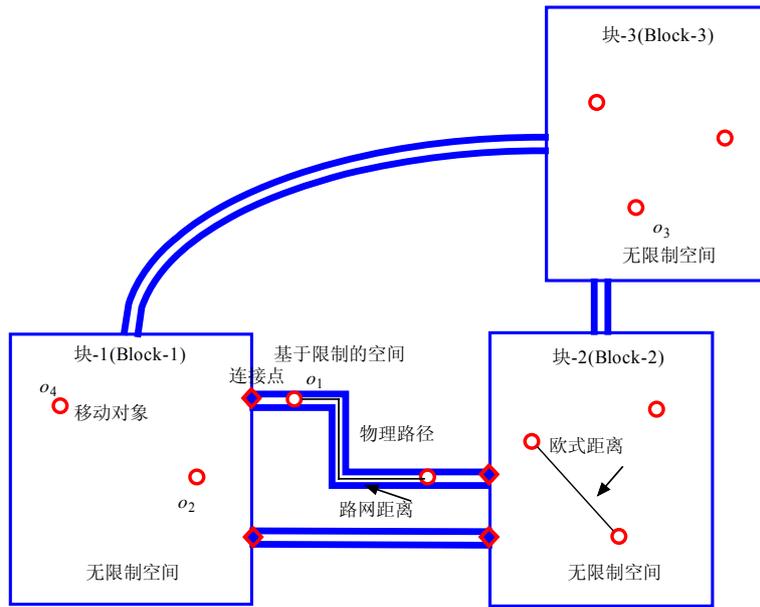


Fig.3 Illustration for a multi-path SC space  
图 3 多物理路径 SC 空间示意图

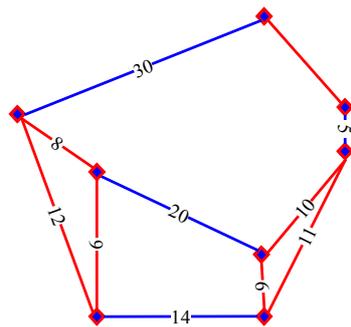


Fig.4 Illustration for a multi-path SCN  
图 4 多物理路径 SCN 示意图

因为 SC 空间中的块、连接点和物理路径等信息都是静态的,因此对于一个指定的 SC 空间场景,可以预先构建一个 SCN.但对于移动数据的管理,需要实时地获取某个时刻任意两个移动对象之间的距离.因为移动对象的位置是动态变化的,需要动态地将对象的位置合并到一个 SCN 中.本文将这种合并移动对象的 SCN 称作 MSCN(moving object merged SCN).给定一个对象的位置  $p(o_i)$ ,为了构建 MSCN,如果  $p(o_i)$ 在某条物理路径上,将对应的边分成两段,并分别记录分割的长度.如果  $p(o_i)$ 在某个块中,仅仅需要将  $p(o_i)$ 与该块中所有的连接点相连接,并标记对应的欧式距离.特别地,为了避免边的数量过于庞大,不能连接同一块中不同移动对象的位置.图 5(a)说明了一个 MSCN 的例子,该例子是基于图 3 所示的 SC 空间,其中,  $o_1, o_2$  和  $o_3$  被合并到对应的 SCN 中.

对于一个给定的 MSCN,任意两个对象  $o_i$  和  $o_j$  之间的距离  $d(o_i, o_j)$  被定义为公式(1),实际上表示了它们在 SC 空间中的最短距离,其中,  $d_{mscn}$  是对  $o_i$  与  $o_j$  之间任意一条 MSCN 网络距离的标记.

$$d(o_i, o_j) = \begin{cases} \sqrt{(o_i.x - o_j.x)^2 + (o_i.y - o_j.y)^2}, & \text{if } B^{now}(o_i) = B^{now}(o_j) \\ \min\{d_{mscn}(o_i, o_j)\}, & \text{otherwise} \end{cases} \quad (1)$$

对于  $\min\{d_{mscn}(o_i, o_j)\}$  的计算,可以归结为一个单源最短路径(SSSP)问题.最著名的 SSSP 算法是 Dijkstra 算法<sup>[21]</sup>.在 Dijkstra 算法中,由一个优先级队列来存储所有当前没有被最终确定的最短路径(SP),算法重复地选择队列中带有最短 SP 的结点,输出结果,并更新队列中受该 SP 影响的所有 SP.给定一个网络,通过优化的 Dijkstra 算法也最少需要  $O(\log V(V+E))$  的平均复杂度来计算一个单源最短路径,其中,  $V$  表示网络图中结点的数目,  $E$  表示边的数目.此外,由于计算是增量完成的,因此某个给定结点的  $k$ -近邻结点也可以用  $O(\log V(V+E))$  复杂度完成.但是,对于一个 MSCN,结点是由所有的监控对象和连接点共同构成的,即  $V=N_{obj}+N_{joint}$ .因为  $N_{obj}$  的数目非常的大,因此计算一个单源最短路径距离是比较耗时的.由于 RFID 应用背景中我们进行以感知区域为中心的可能性查询,将会很好地缩减对应网络空间的规模,从而一定程度上解决 SSSP 算法的效率问题.

根据 SC 空间距离模型,图 5(b)示意了对  $d(o_2, o_3)$  和  $d(o_2, o_4)$  的计算方法. $d(o_2, o_4)$  可以直接按照欧式距离计算得到,而  $d(o_2, o_3)$  需要基于 MSCN 采用 SSSP 算法来获取.此外需要强调的是,如图 3 所示,在 SC 空间内,两个不相邻块内的两个对象距离可能比同一个块内的两个对象的距离还要小,而且某两个块之间可能有多条物理路径相联通.因此,移动对象之间的距离很难简单地依靠块之间的距离关系来确定.

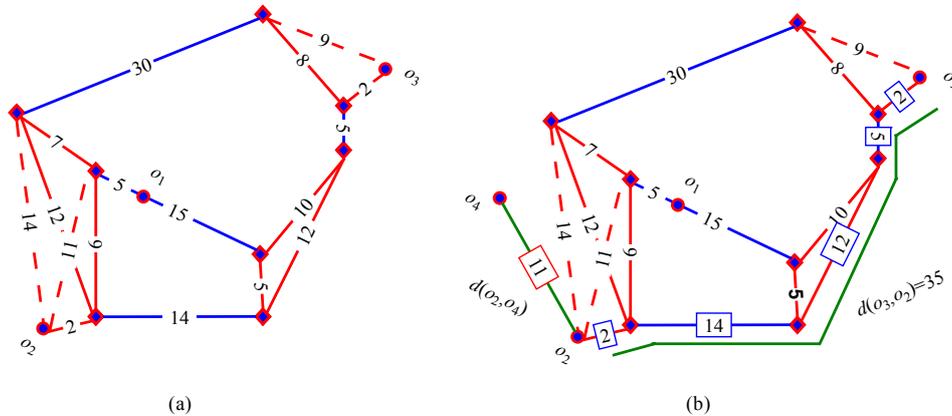


Fig.5 MSCN model and distance evaluation

图 5 MSCN 模型和对应的距离评估

需要说明的是,现实生活中的 SC 空间由于受到障碍物的影响,可能会更加地复杂.例如,在之前给出的 SC 空间里面,没有解释两个块直接通过 joint 相联和块是不规则形状的问题(例如图 1 给出的监控场景).首先,对于两个块之间直接通过 joint 相联而没有物理路径的情况(例如两个由墙相隔的房间),只需要假设两个块之间有一条长度为 0 的物理路径即可,之前提出的 SC 模型和距离评估策略也完全适用.对于块是不规则形状的问题,同一个块内的两个对象,它们之间的距离需要按照拐点的位置进行适当的修正,但由于本章重点针对基于 RFID 的可能性  $k$ -近邻查询,位于不同块之间的对象距离一般不受块形状的影响.因此,在同一块内的对象由于采用 ED 而造成的距离近似,一般不会对可能性  $k$ -近邻查询的结果造成影响.因此,本文为了说明 SC 空间的关键问题,基于如图 3 所示的基本模型进行讨论,所提出的模型和方法通过适当的修改是可以适用于任何复杂的 SC 空间场景的.

### 3 基于 RFID 的半限制空间模型

在本节中,将对基于 RFID 的半限制空间模型进行说明.

在阅读器固定、标签对象移动的场景中,一个阅读器  $r$  周期性地感知来自某个标签对象  $o$  的读数,前提是  $o$

在  $r$  的感知范围内,阅读器的感知周期为  $t_c$ ,在不同的阅读器之间可以认为是同步的.假设  $R, O$  分别代表阅读器的 ID 集合和标签对象的 ID 集合,一个 RFID 的读数可以建模为三元组  $p_i = \langle i \in O, \kappa \in R, t \rangle$ ,各自域的模式为  $\langle f_o, f_r, f_t \rangle$ ,表示某个对象  $o_i$  在时刻  $t$  被阅读器  $r_\kappa$  探测到.为了清楚地描述主要的问题,RFID 阅读器被假设是独立的,就是说没有两个探测区域彼此覆盖,这种假设在一般的监控场景中是合理的.

**定义 3(感知区域(aware region)).** 如果某个阅读器  $r_\kappa$  的天线能够探测到位于某个区域任何位置内的标签对象,该区域就被定义为感知区域  $Y_\kappa$ ,它在物理的二维空间里可以被近似地看作一个圆形.  $(Y_\kappa, c_x, Y_\kappa, c_y)$  表示  $Y_\kappa$  的圆心坐标,  $Y_\kappa, c_r$  表示  $Y_\kappa$  的半径.

同样地,没有被任何 RFID 阅读器覆盖的空间被称作空白区域(vacant region),用  $\tilde{Y}$  来表示.本文将  $t$  时刻在  $Y_\kappa$  内探测到的对象集合记做  $Y_\kappa(t) = \{o_i | \exists p = \langle i, \kappa, t \rangle\}$ ;同理,  $t$  时刻在空白区域的对象记做  $\tilde{Y}(t)$ .进一步,采用函数  $R^t(o_i)$  来表示某个监控对象  $o_i$  属于的区域,比如  $R^t(o_i) = \kappa$  iff  $o_i \in Y_\kappa(t)$ .为了有效地跟踪监控对象的情况,需要构建一个数组  $A'_o$ ,数组的索引是所有监控对象的 id,每个入口包含两个域  $\langle f_r, f_t \rangle$ .具体地,  $A'_o$  定义如下:

$$A'_o \triangleq \begin{cases} A'_o[i].f_t = \tau \Leftrightarrow \tau \leq t \wedge o_i \notin \tilde{Y}(\tau) \wedge \forall t \geq \bar{\tau} > \tau, o_i \in \tilde{Y}(\bar{\tau}) \\ A'_o[i].f_r = \kappa \Leftrightarrow o_i \in Y_\kappa(A'_o[i].f_t) \end{cases} \quad (2)$$

基于图 1 中给出的 RFID 智能博物馆的应用场景,进一步分析 RFID 时空监控的特点.感知区域通常对应于一些特定的活动或者过程,因此本身具有一定的逻辑语义.而空白区域提供了在不同的感知区域之间来往的空间.例如在图 1 中,1~9 号展品的展位对应了 1~9 号的 RFID 感知区域.特别地,不同阅读器的感知半径  $c_r$  可能是不同的,但是因为 RFID 阅读器的位置是固定的,当阅读器的位置和类型在某个场景中布置完毕后,感知区域的形状可以被近似地建模.在没有物理路径联通的阅读器之间,根据应用和场地的特点,可以构建一个理想的虚拟路径.例如对于智能博物馆应用,游客的目的是逐次参观美妙的艺术品,所以最合理的虚拟路径模型应该是两个阅读器之间的最短路径.对于图 1 中的区域 5 和区域 6,虚拟路径可以建模为连接圆心的直线.将物理路径和虚拟路径相结合,下面给出潜在路径的概念.

**定义 4(潜在路径(potential path)).** 感知区域  $Y_{\kappa_1}$  和  $Y_{\kappa_2}$  之间的潜在路径标记为  $\rho(Y_{\kappa_1}, Y_{\kappa_2})$ ,如果  $Y_{\kappa_1}$  与  $Y_{\kappa_2}$  之间存在物理路径,它表示这条物理路径;否则,它代表  $Y_{\kappa_1}$  与  $Y_{\kappa_2}$  之间的虚拟路径.潜在路径可以被建模成一个有向的直线或者折线.

具体地,本文使用一个邻接表  $AL_R$  来表示感知区域之间的单跳连通性.  $A_R$  是一个数组,第  $\kappa$  个入口存储了感知区域  $Y_\kappa$  的  $(c_x, c_y)$  和  $c_r$ ,以及邻接链表  $AL_\kappa$  的入口.  $AL_\kappa$  中的节点存储了所有与  $Y_\kappa$  具有单跳连通性的感知区域的指针,  $f_t$  表示对应的感知区域 id 域.如果存在房间和拐点的信息,将会被同时存储.

此外,为了支持有效的可能性  $k$ -近邻查询,需要在每个连接点布置一个阅读器,用来监控移动对象的流动,这类阅读器对应的感知区域被称作 JR(joint region).而其他的感知区域被称作 NJR(non-joint region).图 6 给出了一个简单的基于 SC 空间的智能 RFID 博物馆场景,标签对象在该空间内沿虚拟路径和物理路径行进, JR 对应的圆心位于某个连接点,而 NJR 的圆心对应某个展品.

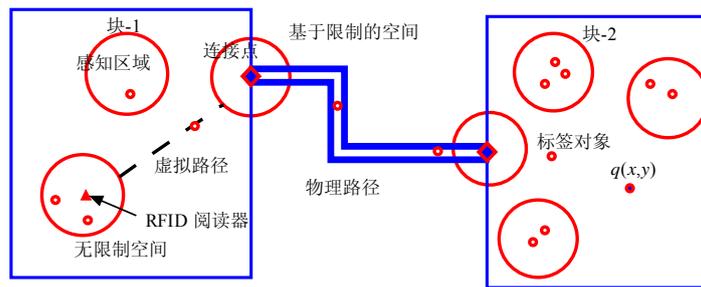


Fig.6 Illustration for a RFID-based SC space scenario  
图 6 基于 RFID 的 SC 空间示意图

根据对历史数据的增量维护,可以估计出对象行进的速度  $v_i$ ,从而估计出对象在潜在路径上的位置或偏移量.但是,从一个感知区域出发,可能有多个路径可供选择,此时需要一个附加的数组来存储每个对象的历史记录,并根据应用的隐含信息进行推演.例如, $t$ 时刻对象  $o_i$  的历史数组被标记为  $H_i^t$ ,具体地,

$$H_i^t[\kappa] = \begin{cases} 1, & \text{if } \exists \tau, o_i \in Y_\kappa(\tau) \wedge \tau \leq t \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

对于博物馆监控场景,应用的信息包括参观者不会重复地参观同一展台.基于  $AL_R$  和  $H_i^t$ ,可以给出算法 1 来对可能位置集  $PS^{t_{now}}(o_i, Y_\kappa)$  进行估计.此外,文献[22]也给出了针对供应链场景的路径概率估算方法.

**算法 1.** 可能的路径对象位置计算方法.

输入:  $o_i, Y_\kappa, H_i^{t_{now}}, AL_R, A_o^{t_{now}}$ ;

输出:  $PS^{t_{now}}(o_i, Y_\kappa)$ .

1. for each item  $e_m$  in  $AL_\kappa$
2. if  $H_i^t[e_m.f_l] = 0$  then  $flag = 1$ ; break;
3. for each item  $e_m$  in  $AL_\kappa$
4. if  $H_i^t[e_m.f_l] = 0 \vee flag = 0$  then
5. if  $Y_{e_m.f_l}$  is JR  $\wedge Y_\kappa$  is JR then
6.  $p.flag = 1$ ;  $p = v_i(t_{now} - Y_o^{t_{now}}[i].f_t)$ ;
7. if  $p < d(Y_\kappa, Y_{e_m.f_l})$  then  $p \rightarrow PS^{t_{now}}(o_i, Y_\kappa)$ ;
8. else
9.  $p.flag = 2$ ;
10.  $x = \frac{Y_\kappa.r + v_i^{t_{now}}(t_{now} - A_o^{t_{now}}[i].f_t)}{\sqrt{(Y_\kappa.c.x - Y_{e_m.f_l}.c.x)^2 + (Y_\kappa.c.y - Y_{e_m.f_l}.c.y)^2}}(Y_\kappa.c.x - Y_{e_m.f_l}.c.x) + Y_\kappa.c.x$ ;
11.  $y = \frac{Y_\kappa.r + v_i^{t_{now}}(t_{now} - A_o^{t_{now}}[i].f_t)}{\sqrt{(Y_\kappa.c.x - Y_{e_m.f_l}.c.x)^2 + (Y_\kappa.c.y - Y_{e_m.f_l}.c.y)^2}}(Y_\kappa.c.y - Y_{e_m.f_l}.c.y) + Y_\kappa.c.y$ ;
12. if  $d(Y_\kappa, (x, y)) < d(Y_\kappa, Y_{e_m.f_l})$  then
13.  $p = (x, y) \rightarrow PS^{t_{now}}(o_i, Y_\kappa)$ ;

为了以感知区域为中心统计监控对象的情况和数目,需要建立并维护以感知区域为中心的数据结构  $Y_\kappa^t(t_{now})$ ,存储当前在感知区域  $Y_\kappa$  中的监控对象和在某个以  $Y_\kappa$  为出发地的潜在路径上的监控对象,即

$$o_i \in Y_\kappa^t(t_{now}) \Leftrightarrow A_o^{t_{now}}[i] = \kappa.$$

基于  $A_o^{t_{now}}$  的维护,  $Y_\kappa^t(t_{now})$  可以被增量地维护.下一节会基于  $Y_\kappa^t(t_{now})$  讨论不同的可能性  $k$ -近邻查询的估计方法.

## 4 可能性 $k$ -NN 查询技术

本节将给出可能性  $k$ -NN 查询的定义,逐步提出 3 种基于 RFID 的可能性  $k$ -NN 查询的估计策略.针对不同的策略,采用了一些查询优化方法来提高查询处理的效率.

### 4.1 PRkNN:基于感知区域的计算

由于探测的不准确性和不连续,RFID 监控对象的位置具有不确定性.比起准确的定位方法,给定一个查询点  $q(x, y)$  和一个监控对象  $o_i, d(q, o_i)$  无法准确地获得,因此,查询点  $q$  的准确  $k$ -近邻查询结果无法获得,仅仅能够提供提供一个可能的结果集.

**定义 5(可能性  $k$ -近邻查询(probable  $k$ -NN query, Pk-NN query)).** 给定一个查询点  $q(x, y)$  和近邻数目  $k$ ,假设

$q(x,y)$ 的真实  $k$ -近邻结果集为  $RS_{KNN}$ ,可能性  $k$ -近邻查询返回所有满足  $prob(o_i \in RS_{KNN}) > 0$  的监控对象  $o_i$ ,该结果集被标记为  $RS_{PKNN}$ .

如果  $C(S)$ 表示某个集合  $S$  的势,明显地, $C(RS_{KNN})=k$  而  $C(RS_{PKNN}) \geq k$ .需要说明的是,对于某个监控对象的位置不确定性,因为感知区域内的不准确性和潜在路径上的不连续性会分别导致连续的和离散的概率分布,计算准确的概率  $prob(o_i \in RS_{KNN})$  或者在  $RS_{PKNN}$  内具有 top- $k$  最大概率的监控对象,都是 NP 难问题.特别地,对于基于 RFID 的应用,准确地给出  $RS_{PKNN}$  是十分复杂的,而且会造成非常大的处理开销,无法满足 RFID 应用的实时性.本文提出采用不同的思想来估计  $RS_{PKNN}$ ,从而在不确定的条件下提供有价值的候选信息.

首先,假设没有任何潜在在路径上的对象,则该问题被简化为一个 PRkNN 查询(probable region  $k$ -NN query),该查询会返回在感知区域内的可能性  $k$ -近邻.给定一个查询点,对于任意两个感知区域  $\mathcal{Y}_1$  和  $\mathcal{Y}_2$ ,如果  $d(q, \mathcal{Y}_1.c) < d(q, \mathcal{Y}_2.c)$ ,根据不同的  $\mathcal{Y}_1.c_r$  和  $\mathcal{Y}_2.c_r$ ,可以得出 4 种  $\mathcal{Y}_1$  和  $\mathcal{Y}_2$  的位置关系:

- $\mathcal{Y}_1 < \mathcal{Y}_2$ :需要满足的条件为

$$d(q, \mathcal{Y}_1.c) + \mathcal{Y}_1.c_r < d(q, \mathcal{Y}_2.c) - \mathcal{Y}_2.c_r.$$

该关系如图 7(a)所示;

- $\mathcal{Y}_1 < \mathcal{Y}_2$ :需要满足的条件为

$$d(q, \mathcal{Y}_1.c) - \mathcal{Y}_1.c_r < d(q, \mathcal{Y}_2.c) - \mathcal{Y}_2.c_r \wedge d(q, \mathcal{Y}_2.c) > d(q, \mathcal{Y}_1.c) + \mathcal{Y}_1.c_r > d(q, \mathcal{Y}_2.c) - \mathcal{Y}_2.c_r.$$

该关系如图 7(b)所示;

- $\mathcal{Y}_1 \subset \mathcal{Y}_2$ :需要满足的条件为

$$d(q, \mathcal{Y}_1.c) - \mathcal{Y}_1.c_r > d(q, \mathcal{Y}_2.c) - \mathcal{Y}_2.c_r \wedge d(q, \mathcal{Y}_1.c) + \mathcal{Y}_1.c_r < d(q, \mathcal{Y}_2.c) + \mathcal{Y}_2.c_r.$$

该关系如图 7(c)所示;

- $\mathcal{Y}_1 \supset \mathcal{Y}_2$ :需要满足的条件为

$$d(q, \mathcal{Y}_1.c) - \mathcal{Y}_1.c_r < d(q, \mathcal{Y}_2.c) - \mathcal{Y}_2.c_r \wedge d(q, \mathcal{Y}_1.c) + \mathcal{Y}_1.c_r > d(q, \mathcal{Y}_2.c) + \mathcal{Y}_2.c_r.$$

该关系如图 7(d)所示.

特别地,从图 7(b)~图 7(d)中可以分析出,对于  $\mathcal{Y}_1 < \mathcal{Y}_2$ ,  $\mathcal{Y}_1 \subset \mathcal{Y}_2$  和  $\mathcal{Y}_1 \supset \mathcal{Y}_2$  这 3 种情况,因为  $\mathcal{Y}_1$  和  $\mathcal{Y}_2$  之间从距离  $q$  的角度来说存在着重复的区域,因此,  $o_1 \in \mathcal{Y}_1(t_{now})$ ,  $o_2 \in \mathcal{Y}_2(t_{now})$  并不能得出  $d(q, o_1) < d(q, o_2)$ .为了描述方便,用  $\mathcal{Y}_1 \bowtie \mathcal{Y}_2$  来统一表示这 3 种情况.

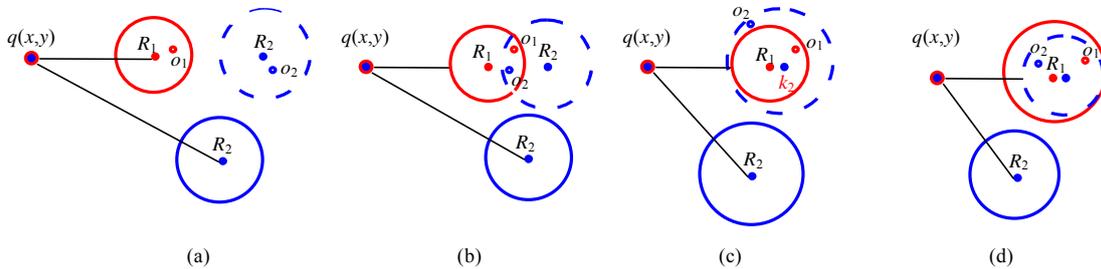


Fig. 7 Illustration for the relationship between two aware regions

图 7 两个感知区域之间的位置关系示意图

进一步,设计一个包含所有感知区域的排序数组  $L_q$ ,并根据  $d(q, \mathcal{Y}_k.c)$ 值升序地排列.假设  $L_q$  包含的感知区域数目为  $N_R$ ,如果  $\sum_{1 \leq i \leq n-1} C(L_q[i](t_{now})) < k \wedge \sum_{1 \leq i \leq n} C(L_q[i](t_{now})) \geq k$ ,根据上面的分析可得:  $\forall o_j \in L_q[i](t_{now}), 1 \leq i \leq n-1 \Rightarrow o_j \in RS_{PKNN}$ ;此外,进一步可以得到  $\forall o_j \in L_q[n](t_{now}) \vee (o_j \in L_q[i], n < i \leq N_R \wedge L_q[n] \bowtie L_q[i]) \Rightarrow o_j \in RS_{PKNN}$ .然而另一方面,  $\exists n < i', L_q[n] < L_q[i']$  不能推出  $\forall i' < i, L_q[n] < L_q[i']$ .例如在图 8 中,  $L_q[n] < L_q[n+1]$ ,但是  $L_q[n] \bowtie L_q[n+2]$ .因此,如果基于  $L_q$ ,给定任意大小的  $k$  值,都需要从头到尾遍历所有的  $N_R$  个感知区域.但是如果根据  $d'(q, \mathcal{Y}_k.c) = d(q, \mathcal{Y}_k.c) - \mathcal{Y}_k.c_r$  来排序,并且得到对应的排序数组  $L'_q$ ,就得出  $\exists n < i', L'_q[n] < L'_q[i'] \Rightarrow \forall i' < i, L'_q[n] < L'_q[i']$ ;在  $L'_q$  的基础上,仅需要确定某个  $L'_q[i']$ ,并满足条件  $n \leq i', L'_q[n] \bowtie L'_q[i'] \wedge L'_q[n] < L'_q[i'+1]$ .作为 PKNN 查询的起始和终止边界,本文将

$L'_q[n]$  和  $L'_q[i']$  分别称作 s-bound 和 e-bound.  $N_b=i'$  被称作边界号,  $B_r = d(q, L'_q[i']) + L'_q[i'].c_r$  被定义为边界范围.

基于以感知区域为中心的索引  $Y'_\kappa(t_{now})$  和对象当前状态索引  $A'_\kappa(t_{now})$ , 对于任意的  $\kappa$ , 都可以方便地得到  $Y'_\kappa(t_{now})$ . 因此, 关键问题是如何得到  $L'_q$ . 根据在图 4 说明的 SCN 模型, 当把所有的连接点区域 JR 的圆心作为 SCN 的结点, 可以得到一个 SCN 图. 进一步, 与 MSCN 的构建方法类似, 将同一块内所有区域的中心与 JR 的中心相连接, 就能够得到一个 RSCN(RFID-based SC network)图. 给定  $q$ , 一个基本的方法是基于 RSCN 来计算所有满足  $0 < \kappa \leq N_R$  的  $d(q, Y'_\kappa)$ . 与 MSCN 类似, 可以采用 Dijkstra 算法来计算不同块之间的距离. 然后, 根据  $d(q, Y'_\kappa) - Y'_\kappa.c_r$  的值对所有的感知区域进行排序, 这样就可以构建  $L'_q$ . 基于  $L'_q$ , 能够增量地获得  $RS_{PRkNN}$ . 但因为  $q(x, y)$  是动态的而且是不可预测的, Dijkstra 算法需要至少  $O(N_R \log(N_R))$  的计算复杂度. 需要说明的是, 基于 Voronoi 图的方法并不适用这种场景, 因为要求的是距离  $q$  最近的  $N_b$  个感知区域, 不同于预先给定的  $k, N_b$  是无法预先给定并且是动态变化的. 如果采用 Voronoi 图, 无法给定一个数目进行有效的划分.

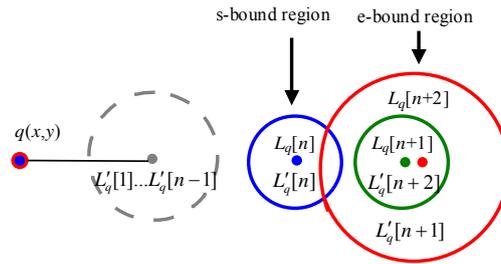


Fig.8 Illustration for the comparison between  $L_q$  and  $L'_q$

图 8  $L_q$  和  $L'_q$  比较示意图

针对查询优化问题, 本文提出一种基于 solution-based indexing 思想的方法来预先存储一些可以计算的结果. 因为 RSCN 的复杂度很高, 只能预先计算一些必要的共享结果, 而不是存储所有可能的解. 本文将该优化方法称作 PSI (partially solution indexing). 按照  $L'_q$  的定义, 如果  $Y'_\kappa$  是一个 JR, 根据已经构建的 RSCN 图, 算法会预先计算  $L'_{Y'_\kappa.c}$ , 并建立 PSI, 使得该索引的第  $\kappa$  项  $PSI[\kappa] = L'_{Y'_\kappa.c}$ . 特别地, 当给定了  $q(x, y)$ , 算法先计算一个特殊的排序链表  $PSI[0]$ , 用来存储所有  $q(x, y)$  所在块  $B(q(x, y))$  的感知区域. 例如所有的  $Y'_\kappa \in B(q(x, y))$ , 并根据距离  $d'(q, Y'_\kappa.c) = d(q, Y'_\kappa.c) - Y'_\kappa.c_r$  来排序. 基于 SC 空间距离的计算方法,  $d(q, Y'_\kappa.c)$  可以直接根据欧式距离获得, 所以  $PSI[0]$  的确定只需要很小的时间代价. 进一步, 根据  $PSI[0]$  和所有满足  $B(Y'_\kappa) = B(q(x, y))$  的  $PSI[k]$ , 通过移动一个指向当前状态的比较指针, 能够增量地比较得到当前所有对应值中最小的  $d'$  值. 图 9 给出了一个例子来说明 PSI 遍历的过程和  $L'_q$  增量维护的方式. 例如, 对于如图 9 所示的 PSI 索引数值, 给定查询点  $q$ , 可以快速地构建  $PSI[0]$ , 之后可以根据  $k$  值增量地得输出结果. 例如, 当感知区域的监控对象数目之和不满  $k$  值时, 可以依次获得的感知区域编号为  $\{1, 2, 10, 5, 3, 6, 14\}$ ; 特别地, 对于结果集, 如果某个结果代表的感知区域已经在之前的增量维护过程中被找到 (例如例子中的编号  $R_6(12)$ ), 那么该结果必须删除. 当已经确定了 e-bound, 遍历就可以终止了. 基于 PSI, 仅仅需要  $O(N_b)$  的平均运行时计算复杂度. 基于 SC 空间的 PRkNN 算法的关键步骤在算法 2 中给出.

**算法 2.** 基于 PSI 索引的 PRkNN 查询.

输入:  $PSI, q(x, y), Y'_\kappa(t_{now})$ ;

输出:  $RS_{PRkNN}, s-bound, e-bound$ .

1.  $m=B(q(x, y)); n=1$ ;
2. for each  $Y'_\kappa$
3. if  $B(Y'_\kappa)=B(q)$  then
4.  $PSI[0].add(\kappa, d'(q, Y'_\kappa))$ ;
5. if  $Y'_\kappa$  is JR then

6.  $L[n].RID=\kappa; L[n++].list.append(PSI[\kappa]);$
7.  $sort\ PSI[0]$  in ascending order;  $L[0].list.append(PSI[0]);$
8. for each  $0 \leq i < n$   $ptr_i \rightarrow L[i].list.head;$
9. while true do
10.  $minD=(*ptr_0).f_d; minR=(*ptr_0).f_R; minI=0;$
11. for each  $1 \leq i < n$
12. if  $(*ptr_i).f_d+d(q, Y_{L[i].RID}) < minD$  then
13.  $minD=(*ptr_i).f_d+d(q, Y_{L[i].RID}); minR=(*ptr_i).f_R; minI=i;$
14. if  $Y_{minR} \notin RS_R$  then
15. If  $s-bound=null$  then
16. if  $N < k$  then
17.  $RS_R.add(Y_{minR}); RS_{PRkNN}.add(o_i, o_i \in Y_{minR}(t_{now})); N=N+=C(Y_{minR}(t_{now}));$
18. else  $s-bound=Y_{minR};$
19. else if  $s-bound \setminus Y_{minR}$  then
20.  $RS_R.add(Y_{minR}); RS_{PRkNN}.add(o_i, o_i \in Y_{minR}(t_{now})); e-bound=Y_{minR};$
21. /\*  $pKNN^+$  is added in this place \*/
22. else break;
23.  $ptr_{minI}^{++};$

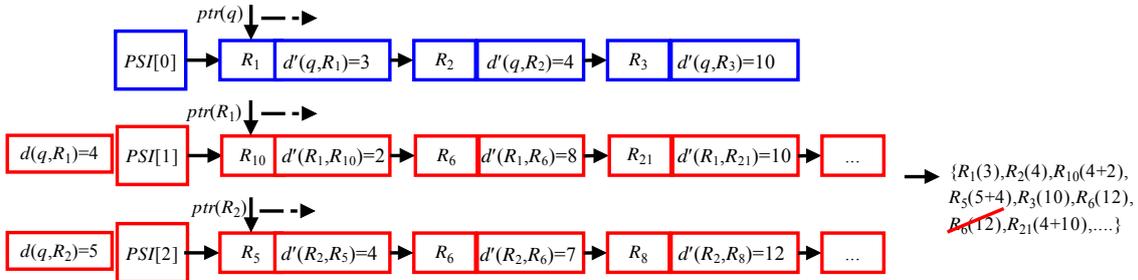


Fig.9 Partial solution indexing traversal

图9 PSI 索引遍历

4.2  $pKNN^+$ :基于路径对象的约简

$PRkNN$  查询仅仅计算了那些被 RFID 阅读器探测到的对象.本节提出一种对  $pKNN$  查询进一步估计的方法,称为  $pKNN^+$ . $pKNN^+$ 不仅考虑了感知区域内的对象,也考虑了潜在路径上的对象.如果某个对象所有的可能路径都被包含在增量确定的  $e-bound$  范围内,该路径对象显然应该被包含在查询结果中.从感知区域角度出发,如果某个潜在路径两端的感知区域都被包含在确定的边界范围中,该路径上的对象也应该属于查询的结果.特别地,边界范围是增量确定的,因此这种路径对象也可以增量地获得.按照这种思路, $N_b$  的大小可以被约简.概括地说, $pKNN^+$ 的基本思想是,每当遍历  $L'_q$  中的一个新的感知区域,判断是否从该感知区域出发的所有路径对象的可能目的感知区域已经被遍历了,执行该过程直至根据  $k$  的大小确定了  $s-bound$ .但是,如果直接按照这种思想去执行,显然会导致很高的时间代价,需要进行查询优化.

针对  $pKNN^+$ 的处理,提出一种基于块索引的方法 BI(block indexing),并融入了基于签名的技术.结合 RSCN 的特点,该方法能有效地支持  $pKNN^+$ 查询.首先,根据  $B(Y_q)$  的值来构建 RSCN,此后将所有存在物理路径的 JR 相连.特别地,对于感知区域和每个块,都设计了一个签名位,如果在  $L'_q$  中找到了某个感知区域,该感知区域对应的签名位就被置 1.一般的情况下,因为在同一块内的任意两个区域都是互相联通的,因此,只有当该块中所有的感

知区域都在  $L'_q$  中找到,从该块中所有 NJR 出发的监控对象才能被插入到  $RS_{PkNN^+}$  中.但是对于某个感知区域  $Y_k$ ,如果  $Y_k$  是一个 JR,与它相邻的另外块中的 JR 的签名对从  $Y_k$  中出发的监控对象也会造成影响.例如,该邻居 JR 如果在之前被找到,那么当某个块整体被找到后,从  $Y_k$  出发的对象才可以被插入到  $RS_{PkNN^+}$  中.具体地,在算法 3 中详细说明了  $PkNN^+$  的感知区域和块的签名规则.需要说明的是,算法 3 是在算法 2 的基础上给出的,是对算法 2 的一个补充.进一步,图 10 给出了一个基于 BI 方法进行查询的例子,假设根据  $L'_q$  中的增量获得的感知区域为  $\{6,5,2,8,1,3,4\}$ ,并且此时找到了 s-bound.如图所示,在该时刻,除了  $Y_5$  外(JR,编码不是 2), $B_1$ (编码是 2)所有的感知区域内出发的对象都会被输出;相反,从  $B_2$ (编码不是 2)内所有的感知区域出发的对象都不会输出.因此,根据 BI 和编码方法,可以快速地增量地输出感知区域  $\{1,2,3,4\}$ ,进一步根据  $Y'_k(t_{now})$  来输出对应的监控对象,从而大大提高了查询处理的速度.BI 索引的维护代价很小,因此基于 BI 的  $PkNN^+$  也只需要  $O(N_b)$  的复杂度,比起  $PRkNN, N_b$  值的减小使得  $PkNN^+$  的查询效率甚至更高.

**算法 3.** 基于块索引和签名的  $PkNN^+$  查询算法.

输入:  $BI, Si, q(x,y), Y_k(t_{now})$ ;

输出:  $RS_{PkNN^+}, s-bound, e-bound$ .

说明:该算法是基于算法 2 的,将以下过程插入算法 2 的注释部分/\*  $PkNN^+$  is added in this place \*/,并用  $PkNN^+$  取代  $PRkNN$ .

1.  $BI[B(Y_{minR})].count++; BI[B(Y_{minR})].Y_{minR}.sig+=1; BI[B(Y_{minR})].sig=1;$
2. if  $BI[B(Y_{minR})].count=BI[B(Y_{minR})].num$  then
3.      $BI[B(Y_{minR})].sig=1;$
4.     for each  $\delta \in BI[B(Y_{minR})]$
5.         if  $\delta$  is JR  $\wedge \delta.sig=2 \wedge \delta$  is NJR then
6.             for each  $o_i \in \delta(t_{now})$
7.                 if  $A_o^{t_{now}}[i].f_t < t_{now}$  then  $RS_{PkNN^+}.add(o_i); N+=1;$
8. if  $Y_{minR}$  is JR then
9.      $\beta=NR(Y_{minR}); BI[B(\beta)].\beta.sig+=1;$
10. if  $BI[B(\beta)].\beta.sig=2 \wedge BI[B(\beta)].sig=2$  then
11.      $RS_{PkNN^+}.add(o_i, o_i \in \beta'(t_{now})); N+=C(\beta'(t_{now}));$

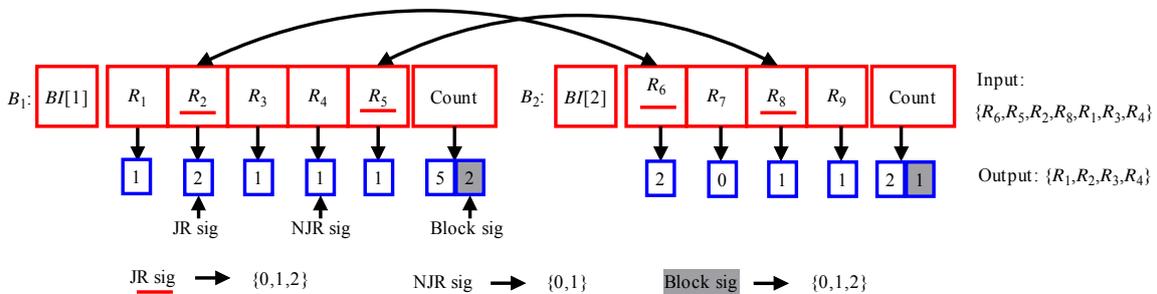


Fig.10 Block indexing and signature  
图 10 块索引和签名

### 4.3 $PkNN^+$ :小概率事件的 $k$ -NN补充

$PkNN^+$ 可以快速地确定一定被  $B_R$  包含的路径对象;反过来,这些路径对象会对  $PRkNN$  方法确定的  $N_b$  和 s-bound 造成影响.但是,目的潜在路径其中的一部分或者是某个潜在路径的部分路径也可能包含在  $B_R$  确定的范围中,从而导致了这些路径上的监控对象也可能被包含在  $B_R$  内.对于这些可能被包含在  $B_R$  内的路径对象,根

据  $PkNN$  的定义,尽管一般不会影响  $s$ -bound,但通过计算删选得到的结果也应该被插入到  $RS_{PkNN}$  中.例如在图 11 中,从  $R'$  和  $e$ -bound 出发的对象  $o_1$  和  $o_2$  实际上在  $B_R$  的范围内,但并没有被  $PkNN^+$  考虑.但是,对于同样从  $e$ -bound 和  $R'$  出发的对象  $o_3$  和  $o_4$ ,它们的位置在  $B_R$  之外.因此,为了更好地估计  $PkNN$  查询,给定  $B_R$ ,定义  $PkNN^*$  查询来补充所有可能在  $B_R$  范围之内,但没有被  $PkNN^+$  查询考虑的监控对象.进一步可以推断,对于某个块,除非该块内所有的感知区域都不在  $B_R$  中,或者  $e$ -bound 是该块内的某个  $JR$ ,同时该块内没有其他的感知区域在  $B_R$  中,否则,从该块内所有的感知区域出发的监控对象都需要根据算法 1 进行位置计算和过滤,来选择可能在  $B_R$  范围内的路径监控对象,这就是  $PkNN^*$  的基本思想.同样地,基于  $BI$  和签名技术,这些候选区域也可以被快速地确定.基于算法 1,根据  $o_i$  与  $q$  之间的  $SC$  距离,可以获得那些在  $PkNN$  查询结果中的对象.假设候选区域出发的对象数目为  $k_c$ ,每个区域联通的平均路径数目为  $k_p$ ,则计算这些候选对象的位置需要  $O(k_c k_p)$  的复杂度.

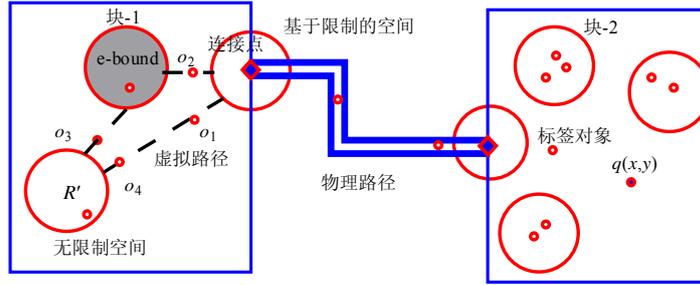


Fig.11 Illustration for possible path objects within  $B_R$

图 11  $B_R$  范围内的可能路径对象说明

特别地,对于  $d(o_i, q)$  的计算,可以进一步使用  $PSI$  技术的结果,而不是采用时间复杂度很高的 Dijkstra 算法.根据  $o_i$  估计位置的不同,式(4)是  $d(o_i, q)$  的计算公式,其中,所有的距离都可以直接使用欧式距离,或者通过算法 1 结合  $PSI$  技术得到.综上,在算法 4 中给出了  $PkNN^*$  算法的关键步骤.

$$d(o_i, q) = \begin{cases} \sqrt{(o_i.x - q.x)^2 + (o_i.y - q.y)^2}, & \text{if } B^{t_{now}}(q) = B^{t_{now}}(o_i) \\ \min\{\forall \kappa, d(q, \mathcal{Y}_\kappa) + d(o_i, \mathcal{Y}_\kappa)\}, & \text{if } B^{t_{now}}(o_i) = B(\mathcal{Y}_\kappa) \neq B^{t_{now}}(q) \wedge \mathcal{Y}_\kappa \text{ is JR} \\ \min\{d(q, \mathcal{Y}_{\kappa_1}) + d(o_i, \mathcal{Y}_{\kappa_1}), d(q, \mathcal{Y}_{\kappa_2}) + d(o_i, \mathcal{Y}_{\kappa_2})\}, & \text{if } \exists \kappa_1, \kappa_2, o_i \in pp(\mathcal{Y}_{\kappa_1}, \mathcal{Y}_{\kappa_2}) \end{cases} \quad (4)$$

**算法 4.** 基于块索引和签名的  $PkNN^*$  查询算法.

输入:  $e$ -bound,  $BI$ ;

输出:  $RS_{PkNN^*}$ .

1.  $RS_{PkNN^*}.add(RS_{PkNN^+})$ ;
2. for each entry in  $i \in BI$
3. if  $BI[i].sig=1 \wedge (e\text{-bound} \in BI[i] \wedge BI[i].count=1)$  then
4. for each  $\delta \in BI[i]$
5. if  $(\delta \text{ is JR} \wedge \delta.sig \neq 2) \vee (\delta \text{ is NJR} \wedge \delta.sig=0)$  then
6. for each  $o_i \in \delta'(t_{now})$
7. if  $A_o^{t_{now}}[i].f_i < t_{now}$  then
8. evaluate  $PS^{t_{now}}(o_i, \delta)$  according to algorithm1;
9. for each  $p' \in PS^{t_{now}}(o_i, \delta)$
10. if  $d(p'(x,y), q) < d(q, e\text{-bound}) + e\text{-bound}.r$  then
11.  $RS_{PkNN^*}.add(o_i)$ ; break;

#### 4.4 不同方法的比较

前文已经分别定义了  $PkNN$ ,  $PRkNN$ ,  $PkNN^+$  和  $PkNN^*$ , 本节中将对它们之间的关系进行进一步的比较和分析. 首先,  $PkNN^*$  与  $PkNN$  并不是完全等价的, 因为根据算法 4 计算得到的路径对象可能有 100% 的概率在  $B_R$  范围内, 当这类对象到达了一定的数目,  $N_b$  将会受到影响, 尽管这种情况发生的概率较小. 对于  $PkNN^+$ , 根据算法 3, 可以得到  $RS_{PkNN^+} \subseteq RS_{PkNN^*}$ . 因为  $PkNN^+$  仅考虑了那些一定被包含在  $B_R$  内的路径对象, 因此会错失  $RS_{PkNN}$  中的一些对象, 尽管这些对象在  $RS_{kNN}$  的概率相对较小; 而  $PkNN^*$  弥补了  $PkNN^+$  的不足, 通过上面的分析, 可以认为是  $PkNN$  的一个很好的近似. 至于  $PRkNN$ , 根据算法 2, 可以得出  $RS_{PRkNN} \cap RS_{PkNN^+} \neq \emptyset$ . 明显地,  $PRkNN$  过多地计算了  $N_b$  值, 很多的结果不会被  $RS_{PkNN}$  包含; 特别地,  $RS_{PkNN}$  中所有的路径对象都错失了, 因此准确性较差. 但  $PRkNN$  可以有效地针对那些只关心感知区域内对象的范围查询语义.

### 5 实验评估

#### 5.1 实验设置

在本节中, 将通过模拟实验对提出模型和方法的有效性和准确性进行评估. 我们模拟了基于 SC 空间的 RFID 支持的智能博物馆场景. 表 1 给出了默认的参数设置. 移动对象被设定沿着潜在路径从一个感知区域到另一个感知区域行进, 对应的 RFID 读数因此连续的生成. 特别地, 采用热点技术, 可以有效地仿真移动对象行进的过程和分布. 实验的硬件环境为: PC 机, 1.86GHz Intel Core 2 的 CPU 和 1G 的内存. 算法是用 C++ 实现的.

Table 1 Default parameters settings for the experiments

表 1 实验默认参数设置

参数	默认值	含义
$k$	1 000	$k$ -NN 查询的参数
$N_{obj}$	10 000	对象数目
$N_R$	1 200	感知区域数目
$N_B$	200	房间(块)数目
$N_j$	3	每个块的平均连接点数目
$v_{obj}$ (m/s)	0.4	对象的平均速度
$R_r$ (m)	3	感知区域的平均半径
$L_{path}$ (m)	20	物理路径的平均长度
$L_{block}$ (m)	30	块平均对角线长度

#### 5.2 效率评估

给定一个查询点, 我们将比较使用不同算法进行查询的响应时间, 以此来说明查询算法的效率. 首先, 如果能够获取所有移动对象的准确位置, 在默认参数下使用 Dijkstra 算法, 需要大约 10s 的响应时间. 但是在基于 RFID 的监控应用中, 使以感知区域为中心的索引  $\gamma'_k(t_{now})$ , 移动对象可以根据  $N_R$  的值以聚集的方式进行处理, 而不需要考虑  $N_{obj}$  的大小. 在实验中, 首先比较  $PRkNN$  查询的响应时间, 采用的方法分别是基于 Dijkstra 算法和本文提出的 PSI 方法. 从图 12(a) 和图 12(b) 的查询结果中可以看出,  $PRkNN$ (Dijkstra) 的代价要远远高于  $PRkNN$ (PSI) 的代价. 特别地, 随着  $k$  的增加, 排序的代价也会随之增加, 因此总代价会增加. 另一方面, 随着  $N_R$  的增大, 基于 Dijkstra 算法来确定  $q$  与其他感知区域之间的最短路径代价也会随之增加. 进一步, 在图 12(c) 和图 12(d) 中, 我们又比较了  $PRkNN$ ,  $PkNN^+$  和  $PkNN^*$  的响应时间. 其中, 获取  $RS_{PkNN}$  的代价包括 PSI 遍历代价以确定  $N_b$  的大小、删除重复的区域和判断某个对象在  $\gamma'_k(t_{now})$  中的对象是否在  $\gamma_k(t_{now})$  中. 随着  $k$  和  $N_R$  的增加,  $N_b$  的大小也会增加, 因此导致了更长的响应时间. 对于  $RS_{PkNN^+}$ , 直接确定所有包含路径的方法会造成很大的时间开销, 而采用本文提出的块索引和签名技术, 可以大大改进查询处理的效率. 特别地, 因为路径对象在  $RS_{PkNN^+}$  中得到了体现, 采用块索引技术,  $N_b$  的减小导致  $PkNN^+$ (BI) 比起  $PRkNN$  有着更高的查询效率. 对于  $PkNN^*$ , 因为需要计算大量可能的候选对象的位置和与  $q$  之间的距离, 需要相对较长的响应时间.

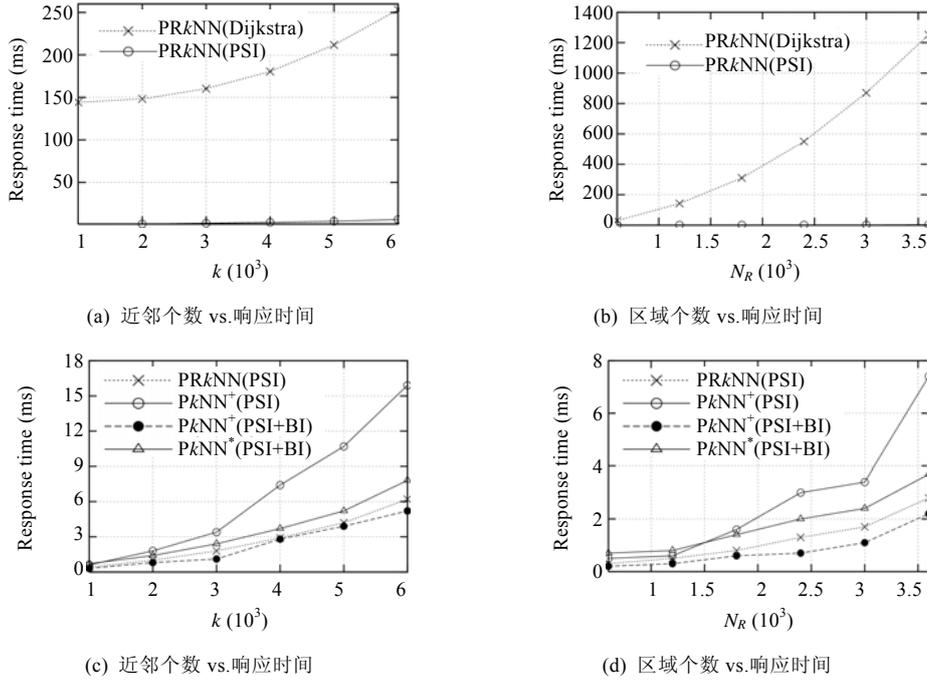


Fig.12 Response time comparison of different algorithms

图 12 不同算法的响应时间比较

5.3 准确性评估

进一步地,我们将从 recall 和 precision 的角度来比较算法的准确性.

对于一个  $PkNN$  查询,  $recall = \frac{C(RS_{PkNN} \cap RS_{kNN})}{C(RS_{kNN})}$ , 而  $precision = \frac{C(RS_{PkNN} \cap RS_{kNN})}{C(RS_{PkNN})}$ . 通过分析不难看出, 对于

一个准确的  $PkNN$  查询, 可以得到一个 100% 的 recall 和一个较高的 precision 结果. 作为对  $PkNN$  查询的估计, 我们分别比较  $PRkNN$ ,  $PkNN^+$  和  $PkNN^*$  的准确性.

给定查询点  $q$  和对应的参数, 能够求得  $RS_{kNN}$ ,  $RS_{PRkNN}$ ,  $RS_{PkNN^+}$  和  $RS_{PkNN^*}$ , 对应的 recall 和 precision 在图 13 所示的查询结果中给出.

从图 13(a) 中能够发现, 由于考虑了所有可能的路径对象,  $PkNN^*$  能够获得非常理想的 recall. 但是如图 13(b) 所示, 由于那些有较小概率成为  $RS_{kNN}$  结果的路径对象也被加入到了  $PkNN^*$  中, 因此  $PkNN^*$  的 precision 比  $PkNN^+$  要低. 对于  $PRkNN$ , 因为仅仅感知区域内的对象被考虑了, recall 和 precision 相对来讲都比较低. 尤其随着  $k$  的增加, 导致了  $N_b$  的增加, 因此可能有更多的候选对象在路径上移动, 所以 recall 和 precision 都会降低.

对于图 13(c) 和图 13(d), 通过调整  $N_{obj}$  的大小也可以得到类似的对比关系. 该实验可以进一步说明,  $PkNN^*$  是  $PkNN$  查询的一个非常好的近似.

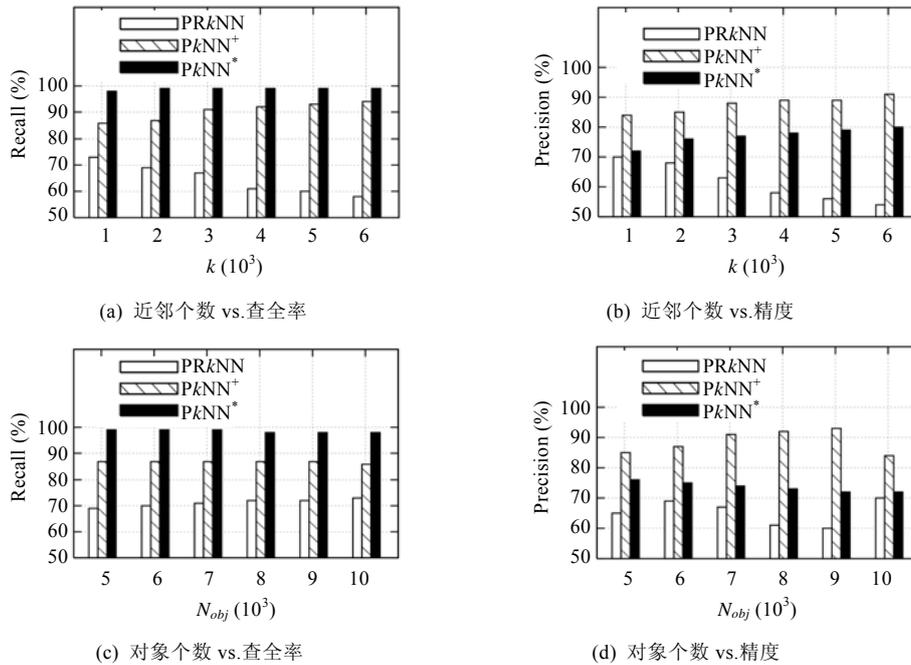


Fig.13 Recall and precision comparison of different algorithms

图 13 不同算法的 recall 和 precision 比较

## 6 结 论

本文提出了基于 RFID 应用的半限制空间模型,讨论了有效的空间建模和距离表示方法.针对 RFID 应用的位置信息的不确定性,提出采用 PkNN 查询来探测某个查询点的可能的  $k$ -近邻.此外,由于 RFID 应用的复杂性, PkNN 查询很难进行准确的评估,因此本文提出 3 种估计,并设计了一些查询优化方法来提高各种方法的查询效率.最后,通过实验验证了提出的查询模型和优化方法的有效性,并且得出结论:基于 RFID 和 SC 空间的 PkNN\* 查询是对 PkNN 查询的一种合理并且高效的估计模型.

### References:

- [1] Wu W, Tan KL. iSEE: Efficient continuous  $k$ -nearest-neighbor monitoring over moving objects. In: Proc. of the 19th Int'l Conf. on Scientific and Statistical Database Management. Banff: IEEE Computer Society, 2007. 36. [doi: 10.1109/SSDBM.2007.37]
- [2] Cheng R, Kalashnikov DV, Prabhakar S. Evaluating probabilistic queries over imprecise data. In: Halevy AY, *et al.*, eds. Proc. of the 2003 ACM SIGMOD Int'l Conf. on Management of Data. San Diego: ACM Press, 2003. 551–562. [doi: 10.1145/872757.872823]
- [3] Kriegel HP, Kunath P, Renz M. Probabilistic nearest-neighbor query on uncertain objects. In: Proc. of the 12th Int'l Conf. on Database Systems for Advanced Applications. LNCS 4443, Bangkok: Springer-Verlag, 2007. 337–348. [doi: 10.1007/978-3-540-71703-4\_30]
- [4] Cheng R, Chen JC, Mokbel M, Chow CY. Probabilistic verifiers: Evaluating constrained nearest-neighbor queries over uncertain data. In: Proc. of the 24th Int'l Conf. on Data Engineering. Cancun: IEEE Computer Society, 2008. 973–982. [doi: 10.1109/ICDE.2008.4497506]
- [5] Welbourne E, Khossainova N, Letchner J, Li Y, Balazinska M, Borriello G, Suci D. Cascadia: A system for specifying, detecting, and managing RFID events. In: Grunwald D, *et al.*, eds. Proc. of the Mobile Systems, Applications, and Services. Breckenridge: ACM Press, 2008. 281–294. [doi: 10.1145/1378600.1378631]
- [6] Ré C, Letchner J, Balazinska M, Suci D. Event queries on correlated probabilistic streams. In: Wang JTL, ed. Proc. of the 2008 ACM SIGMOD Int'l Conf. on Management of Data. Vancouver: ACM Press, 2008. 715–728. [doi: 10.1145/1376616.1376688]

- [7] Tran T, Sutton C, Cocci R, Nie YM, Diao YL, Shenoy P. Probabilistic inference over RFID streams in mobile environments. In: Proc. of the 25th Int'l Conf. on Data Engineering. Shanghai: IEEE Computer Society, 1096–1107. [doi: 10.1109/ICDE.2009.33]
- [8] Tao YF, Papadias D, Shen QM. Continuous nearest neighbor search. In: Proc. of the 28th Int'l Conf. on Very Large Data Bases. Hong Kong: Morgan Kaufmann Publishers, 2002. 287–298.
- [9] Nutanong S, Zhang R, Tanin E, Kulik L. The V\*Diagram: A query-dependent approach to moving  $k$ NN queries. PVLDB, 2008, 1(1):1095–1106. [doi: 10.1145/1453856.1453973]
- [10] Cho HJ, Chung CW. An efficient and scalable approach to CNN queries in a road network. In: Bohm K, *et al.*, eds. Proc. of the 31st Int'l Conf. on Very Large Data Bases. Trondheim: ACM Press, 2005. 865–876.
- [11] Kolahdouzan M, Shahabi C. Voronoi-Based  $k$  nearest neighbor search for spatial network databases. In: Nascimento MA, *et al.*, eds. Proc. of the 30th Int'l Conf. on Very Large Data Bases. Toronto: Morgan Kaufmann Publishers, 2004. 840–851.
- [12] Jensen CS, Kolářvr J, Redersen TB, Timko I. Nearest neighbor queries in road networks. In: Proc. of the 11th ACM Int'l Symp. on Advances in Geographic Information Systems. New Orleans: ACM Press, 2003. 1–8. [doi: 10.1145/956676.956677]
- [13] Kolahdouzan MR, Shahabi C. Continuous  $k$ -nearest neighbor queries in spatial network databases. In: Sander J, *et al.*, eds. Proc. of the 2nd Int'l Workshop of the Spatio-Temporal Database Management, 2004. 33–40.
- [14] Huang YK, Liao SJ, Lee C. Efficient continuous  $k$ -nearest neighbor query processing over moving objects with uncertain speed and direction. In: Ludäscher B, *et al.*, eds. Proc. of the 20th Int'l Conf. on Scientific and Statistical Database Management (SSDBM). LNCS 5069, Hong Kong: Springer-Verlag, 2008. 549–557. [doi: 10.1007/978-3-540-69497-7\_37]
- [15] Zhang MH, Chen S, Jensen CS, Ooi BC, Zhang ZJ. Effectively indexing uncertain moving objects for predictive queries. PVLDB, 2009,2(1): 1198–1209.
- [16] Hua M, Pei J, Zhang WJ, Liu XM. Ranking queries on uncertain data: A probabilistic threshold approach. In: Wang JTL, ed. Proc. of the 2008 ACM SIGMOD Int'l Conf. on Management of Data. Vancouver: ACM Press, 2008. 673–686. [doi: 10.1145/1376616.1376685]
- [17] Soliman MA, Ilyas IF, Chang KC. Top- $k$  query processing in uncertain databases. In: Proc. of the 23rd Int'l Conf. on Data Engineering. Istanbul: IEEE Computer Society, 2007. 896–905. [doi: 10.1109/ICDE.2007.367935]
- [18] Jensen CS, Lu H, Yang B. Indexing the trajectories of moving objects in symbolic indoor space. In: Mamoulis N, *et al.*, eds. Proc. of the 11th Int'l Symp. of Advances in Spatial and Temporal Databases, LNCS 5644, Aalborg: Springer-verlag, 2009. 208–227. [doi: 10.1007/978-3-642-02982-0\_15]
- [19] Yang B, Lu H, Jensen CS. Scalable continuous range monitoring of moving objects in symbolic indoor space. In: Cheung DW, *et al.*, eds. Proc. of the 18th ACM Int'l Conf. on Information and Knowledge Management. Hong Kong: ACM Press, 2009. 671–680. [doi: 10.1145/1645953.1646039]
- [20] Yang B, Lu H, Jensen CS. Probabilistic threshold  $k$  nearest neighbor queries over moving objects in symbolic indoor space. In: Manolescu I, *et al.*, eds. Proc. of the 13th Int'l Conf. on Extending Database Technology. Lausanne: ACM Press, 2010. 335–346. [doi: 10.1145/1739041.1739083]
- [21] Dijkstra EW. A note on two problems in connexion with graphs. *Numeriche Mathematik*, 1959,1(1):269–271. [doi: 10.1007/bf01386390]
- [22] Gonzalez H, Han JW, Li XL. Mining compressed commodity workflows from massive RFID data sets. In: Yu PS, *et al.*, eds. Proc. of the 15th ACM Int'l Conf. on Information and Knowledge Management. Arlington: ACM Press, 2006. 162–171. [doi: 10.1145/1183614.1183641]



谷峪(1981—),男,辽宁鞍山人,博士,副教授,CCF 会员,主要研究领域为 RFID 数据管理,空间数据管理。



李传文(1982—),男,博士生,CCF 学生会会员,主要研究领域为空间数据管理。



于戈(1962—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为数据库理论与技术。