

移动 k -支配最近邻查询验证研究

崔宁宁 杨晓春 王 斌 朱怀杰

(东北大学计算机科学与工程学院 沈阳 110819)

摘 要 现有的基于位置服务(Location-Based Services, LBS)查询结果都是直接基于 LBS 服务商返回的结果. 但 LBS 服务提供商是易被勾结的和不受信任的, 所以给用户提供一种可信查询服务是非常必要的. 因此, 研究可信环境下的空间数据库查询具有重要意义. 该文关注在移动过程中的 k -支配最近邻查询和验证, 该查询本质上是 k -最近邻(k -Nearest Neighbor, k NN)和轮廓(Skyline)查询的结合. 其目标是对于一个给定的查询 q , 返回在空间属性和非空间属性上不受支配的且距离最接近查询点 q 的 k 个点. 为了有效解决移动 k -支配最近邻查询验证问题, 该文提出了一种新的安全区域(Safe Region, SR)和验证数据结构(Authenticated Data Structure, ADS) Merkle Verifiable Voronoi R-tree(MV²R-tree)用于产生查询结果和验证对象(Verification Object, VO). 在此基础上, 该文基于比较分析提出了 Rectangle-based 验证策略用于验证查询结果和安全区域. 最后, 通过大量的实验验证了提出的方法的有效性.

关键词 基于位置服务; 可信查询; k -支配; 安全区域; 验证对象; 验证数据结构

中图法分类号 TP311 **DOI号** 10.11897/SP.J.1016.2018.01780

Research on Authentication of Moving k -Dominant NN Queries

CUI Ning-Ning YANG Xiao-Chun WANG Bin ZHU Huai-Jie

(College of Computer Science and Engineering, Northeastern University, Shenyang 110819)

Abstract With the rapid development of smart mobile terminal and mobile network, the process of Location-based services (LBS) has been promoted in a large extent and extracted a lot of researchers' attention. In Location-based services, various spatial queries have been proposed, e. g., k NN queries and Skyline queries, but these queries do not take spatial and non-spatial attributes into consideration simultaneously. In addition, existing query results in Location-based services are returned from Location-based services provider directly. But Location-based services provider is colluded easily and is not trusted which may collude with attackers or merchants for some interests and returns the garbled or incorrect results to users, it is necessary to provide users with trusted query services and supporting authentication of spatial queries in LBS has become more and more significant for LBS applications to provide high quality and trusted services. Hence, it has important significance to study spatial query under trusted environment. In this paper, in order to make users trust the query results, we mainly focus on the problem of authentication of moving k -dominant nearest neighbor queries, which is a natural combination of k NN queries and Skyline queries. For a given query q , k -dominant NN queries returns k points which are not dominated by spatial and non-spatial attributes and are nearest between all objective points and query q . In order to solve the problem of authentication of moving k -dominant nearest neighbor queries effectively, we first adopt the approach of safe region to address the problem of

收稿日期: 2017-01-27; 在线出版日期: 2017-07-11. 本课题得到国家自然科学基金重点项目(61532021)、国家自然科学基金(61272178, 61572122)、国家优秀青年科学基金(61322208)、中央高校基本科研业务专项资金(N161606002)资助. 崔宁宁, 男, 1988年生, 博士研究生, 主要研究方向为空间数据库、隐私保护. E-mail: willber1988@163.com. 杨晓春(通信作者), 女, 1973年生, 博士, 教授, 博士生导师, 主要研究领域为数据库管理与隐私保护. E-mail: yangxc@mail.neu.edu.cn. 王 斌, 男, 1972年生, 博士, 副教授, 主要研究方向为大数
据管理. 朱怀杰, 男, 1988年生, 博士研究生, 主要研究方向为空间数据库和隐私保护.

continuously updating query. Safe region guarantees that the results of moving k -dominant NN query are unchanged as long as the query locates in safe region. Within the safe region, the client does not need to communicate to services provider frequently, thus it can save calculation cost and communication cost. Then we propose a definition about node impact region and design a novel authenticated data structure (ADS) MV^2 R-tree for implementing k -dominant NN query motivated by MR-tree and VoR-tree. With authenticated data structure (ADS), services provider not only implements k -dominant NN query, but also generates verification object (VO) with verification region. Moreover, to construct the verification regions of the query results and safe region, we propose two methods, denoted as circle-based and rectangle-based, to authenticate the query results and safe region. We have analyzed that rectangle-based method has a better performance than circle-based method. In addition, we also specify the updating process of locations of the client. We propose an updating strategy which can reduce communication cost obviously when the user move out the safe region. Through this way it can reduce communication cost since the client does not need to communicate with services provider frequently. In order to reduce communication cost further, we utilize the method of reusing the previous verification object to solve this problem. At last, we conduct a comprehensive performance evaluation using real dataset and synthetic database to validate our ideas and the proposed algorithms, and the experiment results show that the proposed method can efficiently implement k -dominant nearest neighbor query and verify the query results.

Keywords location-based services; trusted queries; k -dominant; safe region; verification object; authenticated data structure

1 引 言

随着智能移动终端和移动通信网络的快速发展,在很大程度上推动了基于位置服务的发展进程.与此同时,基于位置服务吸引了大量研究学者的关注.在基于位置服务中,研究者提出了大量的查询应用,例如,范围查询^[1]、 k 最近邻查询^[2]和轮廓查询^[3].但是这些查询没有同时考虑目标物体的空间属性(例如,距离)和非空间属性(例如,价格).此外,已有的工作还研究了动态轮廓查询^[3-6]和 top- k 查询^[7-9].虽然这些查询都同时考虑了目标物体的空间属性和非空间属性.但是,动态轮廓查询返回了一些冗余点(距查询点较远的点,即实际中用户并不关注的点).而 top- k 查询需要定义不同的评价函数,且不同的评价函数可能会造成返回的查询结果中包含距查询点较远的点或者包含非空间属性被其他点支配的点.本文研究中,主要关注了一种新的查询方式 k -支配最近邻查询,该查询本质上是结合了 k 最近邻查询和轮廓查询且同时考虑了目标物体的空间属性和非空间属性.一个典型的 k -支配最近邻查询例子就是“查找 k 个附近的加油站或酒店,且这些加油

站或酒店拥有较低的价格和较高的排名^①”.其中,空间属性指的是查询点与目标点间的距离,非空间属性指的是目标点的价格和排名.如图 1 所示, q 为用户查询点, $a \sim h$ 为目标查询点且包含了价格和排名等非空间属性.对于 2-最近邻查询结果为 $\{f, g\}$ (如图 1 中虚线圆所示),但是 f 和 g 在价格和排名等非空间属性上受 b 和 c 支配.因此, f 和 g 不能满足用户查询需求.对于轮廓查询, LBS 返回 $\{a, b, c, d\}$.但是, a 和 d 与查询点 q 之间的距离较远,也不能完全满足需求.而 2-支配最近邻返回结果为 $\{c, g\}$, 可以看出 g 和 c 是满足用户查询需求的目标点,因为 g 和 c 在空间和非空间属性上不受其他点支配且距离查询点最近.

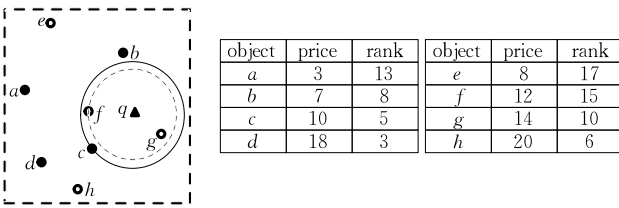


图 1 不同查询的示例

① 本文中,我们假设非空间属性值越小越好,例如排名越小,排名越靠前.

早期的 LBS 服务主要应用于军事和一些民用领域,而如今的 LBS 涵盖了所有的社交和商业等领域.瑞典市场研究公司 Berg Insight 于 2015 年发布预测报告,指出全球 LBS 市场规模将以 22.5% 复合年增长率(CAGR)从 2014 年的 103 亿欧元,增加至 2020 年的 348 亿欧元.面对如此巨大的 LBS 服务需求,如何保障服务的可靠性和安全性成为了首要条件.在基于位置服务中,服务提供商 (Services Provider, SP) 是不被信任的,他们为了更多利益可能与攻击者或者商家相互勾结,从而将篡改的结果或不正确的结果返回用户.因此,在基于位置服务中,给用户提供可信的空间查询变得越来越重要.这样不仅可以验证结果的正确性,同时保证了服务的质量和安全性.空间查询验证已经历经了很多年的研究发展^[10-13].为了向用户提供可信查询服务,本文研究了移动 k -支配最近邻查询验证问题.服务外包数据库验证模型主要包括:(1) 数据所有者(Data Owner, DO), (2) 服务提供者 (Service Provider, SP) 和 (3) 客户端(Client).如图 2 所示,数据所有者进行原始数据的处理,建立数据的索引结构,并利用数字签名技术提供对数据的真实性和可靠性的验证.服务提供商在收到客户端的查询请求后,在数据所有者签名的数据上进行查询,将查询结果和验证对象 VO 一起发给客户端.其中,VO 中包括数据所有者的签名和用于验证的必要信息.客户端向服务提供商发起查询服务请求,并根据认证中心(Certificate Authority, CA)的公钥密匙和服务提供商返回的 VO 对查询结果进行可靠性的验证.

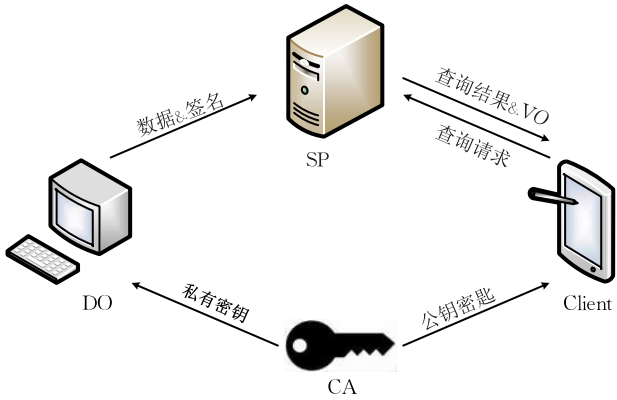


图 2 服务外包数据库的验证模型

然而,已经存在的查询验证方法不能直接用于移动 k -支配最近邻查询.其主要表现为针对不同的查询方式,验证数据结构(ADS)的设计可能不同.例如,对 k NN 查询而言,其验证方法不能实现对数据

间支配关系的验证;而对轮廓查询而言,其验证方式不能实现对数据间空间距离关系的验证.进行移动 k -支配最近邻查询验证面临的主要挑战如下:

(1) 如何设计一个简洁的验证数据索引结构.因为服务提供商 SP 在该索引结构上进行查询的过程中,通过验证数据索引结构产生查询结果和验证对象 VO.

(2) 合适的验证区域.合适的验证区域保证了较小的 VO 大小以及较低的传输代价.

针对上述的问题和挑战,本文设计了一种新的验证数据结构 MV^2 R-tree.在此基础上,提出了移动 k -支配最近邻查询的安全区域.并通过提出的验证策略 Rectangle-based,用于验证查询结果和安全区域.归结起来,本文的贡献如下:

(1) 提出了一种新的查询方式 k -支配最近邻查询和验证索引结构 MV^2 R-tree.首先,该索引结构结合了 MR-tree^[11] 和 VoR-tree^[14],不仅可以执行 k -支配最近邻查询,同时还可以高效地产生 k -支配最近邻查询的安全区域.据所查文献,本文工作是最先解决 k -支配最近邻查询和验证问题.

(2) 在验证数据索引结构和产生的安全区域的基础上,首先提出一种 Circle-based 的验证策略,并在此基础上提出了优化的 Rectangle-based 验证策略,用于快速地构建查询结果和安全区域的验证区域,进而验证查询结果和安全区域的正确性.

(3) 设计了详细的性能评价实验,实验结果表明本文所提出的算法能够很好的实现 k -支配最近邻查询验证问题.

本文第 2 节概述研究问题的相关工作;第 3 节详述研究问题的预备知识;第 4 节介绍验证数据结构;第 5 节讲述 k -支配最近邻查询验证的方法;第 6 节对所提方法的有效性进行实验分析;第 7 节总结全文.

2 相关工作

查询验证随着查询方法和索引技术的发展,已经经历了很多年的研究发展,也受到了越来越多的研究学者的关注.目前,这些研究工作主要分为两类:(1) 基于快照的查询验证;(2) 基于移动的查询验证.

2.1 快照查询验证

Merkle 哈希树 (Merkle Hash Tree, MHT)^[12] 是空间查询验证中应用最广泛的框架,MHT 与其他索引结构相结合用于构建验证数据结构(ADS).

对于一维数据查询验证, Li 等人^[13]提出了 MB-tree, 通过在原始 B^+ -tree 的基础上, 每个节点上都存储了一个 hash 值, hash 值的计算与 MHT 相似. 为了进一步减小 VO 的大小, Li 等人^[13]又提出了 MB-tree 的变形 EMB-tree, 即在 B^+ -tree 的每个节点中嵌入一棵 MB-tree.

而针对多维数据查询验证, Pang 等人^[10]将签名链与 R-tree 结合, 根据每个点的横坐标, 将其存储在 R-tree 的叶子节点中. 此外, VR-tree 给叶子节点中每三个点序列创建一个签名, 并且将这个签名存储在相应的点中. Yang 等人^[11]在 MHT 和 R^* -tree 的基础上, 提出了 MR-tree, 一种高效的空间索引验证结构, 实现了对任意空间查询验证的功

能. 相较于 VR-tree, MR-tree 在索引构建时间、存储空间消耗、查询处理和查询验证等性能方面都有了较大的提升. MR-tree 结构如图 3 所示, 其中包含了 8 条数据记录 $a \sim h$, MR-tree 节点的摘要计算如下: (1) 如果是叶子节点 entry, 其摘要为 entry 的哈希值. 例如, 点 a 的摘要为 $h(a)$, 其中 $h(\cdot)$ 为哈希函数; (2) 如果节点是叶子节点, 其摘要为所包含 entry 的摘要连接后的哈希值. 例如, 对于叶子节点 N_3 , 包含两条记录 b 和 e , 其摘要为 $h(h(b) | h(e))$, 其中 ‘|’ 为连接符; (3) 如果是非叶子节点, 其摘要为所包含节点摘要连接后的哈希值. 例如节点 N_1 , 包含两个孩子节点 N_3 和 N_4 , 其摘要为 $h((h(b) | h(e)) | (h(f) | h(a)))$.

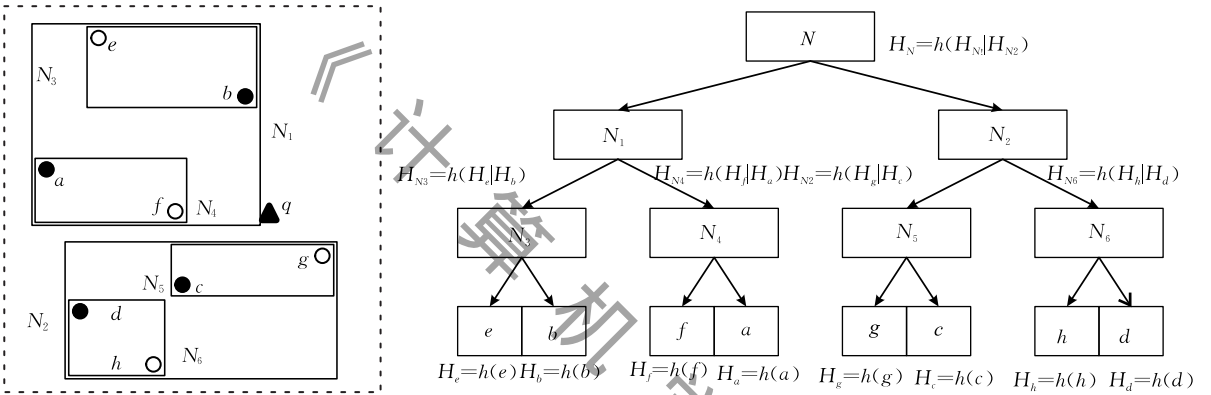


图 3 MR-tree 的示例

与此同时, Mouratidis 等人^[15]提出了一种部分实体化摘要的策略 (Partially Materialized Digest scheme, PMD), 该方法将用于查询的数据索引 (Main Index, MI) 和用于验证的摘要索引 (Digest Index, DI) 分离, 构建了 PMKD-tree, 实现了高效的性能. 该方法可以与多种索引验证结构结合, 既可以应用于一维的数据查询验证, 同时也适用于多维空间查询验证. 在 MR-tree 的基础上, 许多的空间查询得到了验证. 文献[4-5]等基于每个目标点的轮廓范围构建了 MR-Sky-tree 用于验证空间轮廓查询. 并且在此基础上, 提出了任意子空间的轮廓查询验证并通过提出的局部 S4-tree 实现对任意子空间的轮廓查询验证. 文献[16]等利用同态加密技术解决了在线整合服务中的范围查询验证问题且同时不泄露用户的位置隐私. 文献[7]等通过提出 MT-tree, 实现了利用树-森林索引解决 top- k 关键字查询和验证的问题. 并且文献[16-17]等基于加密构建块, 提出了 PPB 和 PLB 等策略, 解决了在进行 top- k 查询验证时的隐私保护问题. 此外, 文献[18-19]

等实现了子图和子图相似性的查询验证.

2.2 移动查询验证

移动查询是指用户在移动的过程中连续返回查询结果. 而在已经存在的移动查询验证技术中, 主要是基于安全区域 (SR) 的方法实现移动查询验证. 文献[20]等提出了基于弧度 (Arc-Based) 的方式实现移动范围查询验证, 且同时满足最小化传输消耗. 文献[21-22]等利用 k 阶 voronoi 多边形构建安全区域, 并利用此安全区域实现了移动 k -最近邻查询验证. 具体地, 首先提出了 Vertex-based 的方法, 主要利用了 k 阶 voronoi 多边形顶点与 k NN 结果间的关系. 然后提出了一种实体化的方法, 通过实体化验证数据结构中的部分 voronoi 单元, 减少服务器端和客户端的计算消耗. 此外, 文献[9]通过结合 MR-tree 和 IR-tree, 构建了一种验证数据结构 MIR-tree 和 MIR*-tree. 在此基础上, 通过构建查询结果的安全区域, 实现了移动 top- k 关键字查询验证. 文献[6]等解决了移动 Skyline 查询验证问题, 主要是通过提出的 MR-Sky-tree 验证数据结构, 并

且结合了三种方式:有效范围、可视化区域和增量VO,有效地减少了计算和传输代价.本文工作与以上相关工作相比的创新之处在于结合了 k NN 查询验证^[21]和轮廓查询验证^[6]的特点,既考虑了数据之间的空间位置关系,同时也考虑了数据间的非空间属性的支配关系.针对新的 k -支配最近邻查询,通过设计高效简洁的验证结构,有效地解决了移动 k -支配最近邻查询验证的问题.

3 预备知识

所有空间目标物体的集合记作 O , o_i 表示集合中的第 i 个物体.每个空间物体 $o \in O$ 都有 d 维属性,其中, $d-1$ 维是非空间属性(例如,价格和排名);最后一维属性是空间属性(例如,距离).将查询点 q 和物体 o_i 之间的距离记作 $dist(q, o_i)$,文中采用欧式距离作为距离的评价准则.为了方便描述,表 1 中给出了本文主要的符号定义.

表 1 符号表示	
符号	符号定义
$o_i <_s o_j$	物体 o_i 静态支配物体 o_j
$o_i <_q o_j$	物体 o_i 完全支配物体 o_j
$o_i <_{\sim q} o_j$	物体 o_i 非完全支配物体 o_j
$IR(o_i)$	物体 o_i 的影响区域
$\perp(p, p')$	平面内距离 p 比 p' 近的区域
$G(o_i, O)$	产生 $IR(o_i)$ 的物体集合
$SSR(o)$	物体 o 的安全区域
$V_k(S, R)$	关于 R 的 k 阶 voronoi 多边形
$KSSR(S, R)$	k -支配结果 S 的安全区域

文中 k -支配最近邻查询中物体间的支配关系与轮廓查询中物体间的支配关系相似,定义如下.

定义 1(静态支配). 对于两个空间物体 o_i 和 o_j ,如果对于每一维非空间属性, o_i 的属性值都不比 o_j 的属性值差,且 o_i 至少有一维属性优于 o_j ,称作 o_i 静态支配 o_j ,记作 $o_i <_s o_j$.

定义 2(完全支配). 对于两个空间物体 o_i, o_j 和给定的查询点 q ,如果 $o_i <_s o_j$,且 $dist(q, o_i) < dist(q, o_j)$,称作 o_i 完全支配 o_j ,记作 $o_i <_q o_j$.

定义 3(非完全支配). 对于两个空间物体 o_i, o_j 和给定的查询点 q ,如果 $o_i <_s o_j$,且 $dist(q, o_j) < dist(q, o_i)$,称作 o_i 非完全支配 o_j ,记作 $o_i <_{\sim q} o_j$.

例 1. 如图 4 所示,位于实线上的物体不被其他物体静态支配(例如,物体 a, b, c, d).与此相反,位于虚线上的物体被此虚线包围的物体静态支配(例如,物体 f 被物体 a, b, c 静态支配).

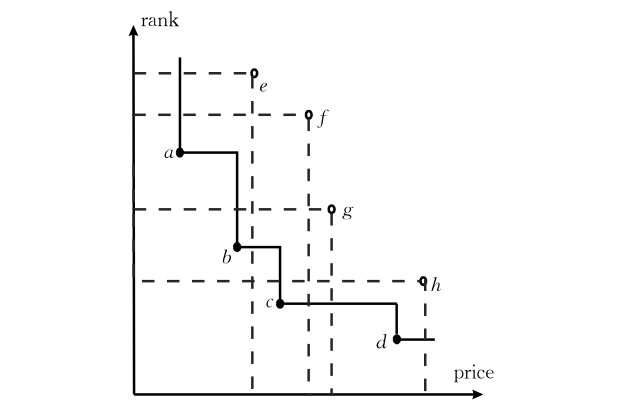


图 4 静态支配示例

对于一个空间物体 o_i ,如果对于任意空间物体 $o \in O$ 且 $o \neq o_i$, o_i 不被 o 静态支配,将物体 o_i 称作一个支配点,所有支配点的集合记作 D (例如, $D = \{a, b, c, d\}$).如果存在空间物体 $o \in O$,且满足 $o <_s o_i$,称物体 o_i 为一个被支配点,所有静态支配 o_i 的物体集合记作 $Dom(o_i)$ (例如, $Dom(e) = \{a, b\}$).考虑上述情景,“查找 k 个附近的加油站或酒店,且这些加油站或酒店拥有较低的价格和较高的排名”,在定义 2 和定义 3 的基础上,定义了 k -支配最近邻查询.

定义 4(k -支配最近邻查询). 给定一个查询点 q , k -支配最近邻查询返回物体集合 S , S 满足如下条件:

- (1) 对任意物体 $o \in S$ 和 $o' \in O$ 且 $o \neq o'$, o 不被 o' 完全支配.
- (2) 所有不被完全支配的物体集合 C ,对任意物体 $o \in S$ 和 $o' \in C - S$, $dist(o, S) < dist(o', C - S)$ 且 $|S| = k$.

例 2. 如图 1 所示,对于给定的查询点 q ,所有不被完全支配的物体集合 N 为 $\{a, b, c, d, g\}$.若 $k=2$,则 2-支配最近邻查询返回物体集合 S 为 $\{g, c\}$.

移动 k -支配最近邻查询连续的返回不被完全支配的前 k 个最近邻物体.本文中,我们采用安全区域的方法解决移动 k -支配最近邻查询的问题.安全区域确保了当用户在安全区域内移动时, k -支配最近邻查询结果不发生改变.在安全区域内,用户不需要频繁地与服务提供商通信,这种方式可以有效地减少计算和传输消耗.在介绍 k -支配最近邻查询的安全区域前,首先给出影响区域和 k 阶 voronoi 图的概念.

定义 5(影响区域)^[3]. 给定一个物体 o_i ,如果区域内的所有点到物体 o_i 的距离比到所有静态支配 o_i 的物体的距离近,称这个区域为 o_i 的影响区域,记

作 $IR(o_i)$.

从定义 5 可以看出,如果用户查询点 q 位于物体 o_i 的影响区域 $IR(o_i)$ 内,那么意味着 $dist(q,o_i) < dist(q,Dom(o_i))$,即物体 o_i 不被 $Dom(o_i)$ 完全支配.如果物体 o_j 是一个支配点,那么 o_j 的影响区域 $IR(o_j)$ 为整个空间区域.

如图 5 所示,物体 f 的支配点集为 $Dom(f) = \{a,b,c\}$,且物体 f 的影响区域 $IR(f)$ 为多边形 AMN. 我们可以看出,一个物体的影响区域是由该物体和物体支配点集形成的 voronoi 多边形,记作 $V(o_i,G(o_i,O))$,其中, $G(o_i,O)$ 为产生 $IR(o_i)$ 的点集.例如, $G(f,O) = \{a,b,c\}$.此外值得注意的是, $G(o,O)$ 是 $Dom(o)$ 的一个子集.

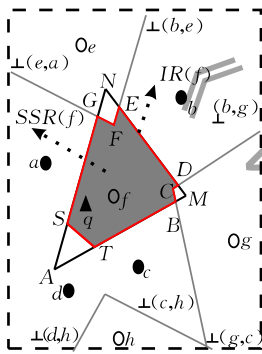


图 5 物体 f 的影响区域

基于物体的影响区域,其安全区域定义为 $SSR(o_i) = \bigcap_{o_j \in R} IR(o_i) \cap \bigcap_{o_j \in O-R} (IR(o_j) \cap \perp(o_i,o_j))$ (1) 其中, R 为所有支配结果集.从图 5 中可以看出,物体 f 的安全区域 $SSR(f)$ 为多边形 BCDEFGST.安全区域保证了当用户在该物体的安全区域内移动并连续发起查询请求时,1-支配最近邻始终为该物体.基于已经存在的方法^[19],为了验证移动 k -支配最近邻查询,我们给出了 k 阶 voronoi 多边形的定义.

定义 6(k 阶 voronoi 多边形). 给定一个子集 $S \subseteq O$ 且 $|S| = k$,集合 S 关于集合 O 的 k 阶 voronoi 多边形为

$$V_k(S,O) = \bigcap_{p \in S} \left(\bigcap_{p' \in O-S} \perp(p,p') \right) \quad (2)$$

如果集合 S 为集合 O 中关于查询点 q 的 k -支配最近邻查询结果集, k 阶 voronoi 多边形保证了查询结果始终为 S .如图 6 所示,结果集 S 为 $\{a,f\}$,关于 O 的 k 阶 voronoi 多边形为图中灰色区域.

但是计算 $V_k(S,O)$ 的复杂度为 $O(|S| \times |O|)$,即 $O(k \times n)$.当数据集特别大时, $V_k(S,O)$ 的计算量非常大,影响算法的执行效率.于是,我们提出了关于 R 的 k 阶 voronoi 多边形,其中, R 为所有支配结

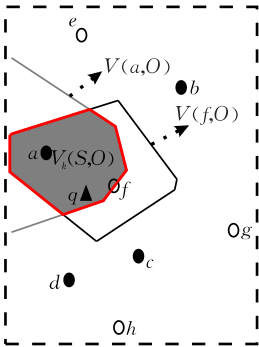


图 6 关于 O 的 k 阶 voronoi 多边形

果集.由于 $|R|$ 的大小可以认定为常数 c ,所以计算 $V_k(S,R)$ 的复杂度为 $O(|S| \times |R|)$,即 $O(k \times c)$.可以看出,关于 R 的 k 阶 voronoi 多边形的计算效率明显高于前者.如图 7 所示,结果集 S 为 $\{a,f\}$,关于 R 的 k 阶 voronoi 多边形为图中灰色区域.

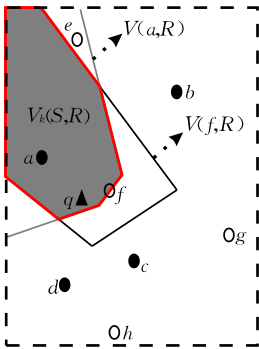


图 7 关于 R 的 k 阶 voronoi 多边形

引理 1. $V_k(S,O) \subseteq V_k(S,R)$.

证明. 我们知道,集合 R 为集合 O 的子集,即 $R \subseteq O$. (1) 若 $R = O$,则 $V_k(S,O)$ 等于 $V_k(S,R)$; (2) 若 $R \subset O$,则 O 中至少存在物体 o ,使得 $o \notin R$.当物体 o 对 k 阶 voronoi 多边形 $V_k(S,O)$ 的形成没有影响,即 $V_k(S,O)$ 等于 $V_k(S,R)$.而当物体 o 对 k 阶 voronoi 多边形 $V_k(S,O)$ 的形成有影响,那么必然存在 $o' \in S$,使得 $\perp(o',o) \cap V_k(S,R) \neq \emptyset$.进而所有对 k 阶 voronoi 多边形 $V_k(S,O)$ 的形成有影响的 o ,使得 $\Sigma \perp(o',o) \cap V_k(S,R) = V_k(S,O)$,即 $V_k(S,R)$ 的区域包含 $V_k(S,O)$ 的区域.故引理 1 成立.

证毕.

定义 7(k -支配安全区域). 给定一个查询点 q ,

(1) 如果查询点 q 位于物体 o 的安全区域 $SSR(o)$ 内,则 k -支配安全区域为

$$KSSR(S,R) = SSR(o) \cap V_k(S,R) \quad (3)$$

(2) 如果查询点 q 位于物体 o 与其他物体 o' 的影响区域的相交区域内,则 k -支配安全区域为

$$KSSR(S,R) = (IR(o) \cap (\bigcap IR(o'))) \cap V_k(S,R) \quad (4)$$

例 3. 如图 8 所示,查询点 q 位于物体 f 的安全区域内,结果集 S 为 $\{a, f\}$,支配点集 R 为 $\{a, b, c, d\}$. k -支配安全区域 $KSSR(S, R)$ 为不规则多边形 $OIJFG$. 其中预设 $k=2$. 它是由图 5 中的 $SSR(f)$ 和图 7 中的 $V_k(S, R)$ 的相交区域组成.

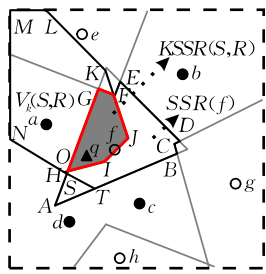


图 8 查询点 q 在 $SSR(o)$ 内的 $KSSR(S, R)$

引理 2. 基于 $V_k(S, O)$ 和 $V_k(S, R)$ 产生的 k -支配安全区域 $KSSR$ 是一致的.

证明. 假设查询点 q 包含在被支配点 o 的安全区域内. 从定义 7 可知, $V_k(S, O)$ 由区域 V_1 和 V_2 组成. 其中, V_1 表示在 $V_k(S, O)$ 区域内, 到点 o 的距离小于到 S 中其他支配点的距离的区域, 且 $V_1 \cap SSR(o) = V_1$. 对于 $V_k(S, R)$ 也有相应两部分 V'_1 和 V'_2 组成, 且 V'_1 中可能包括了其他被支配点的影响区域与之相交的区域. 但是, 由式(1)和引理 1 可知, $V'_1 \cap SSR(o) = V_1$. 故引理 2 成立. 证毕.

4 验证数据结构

在本节中,我们提出了一种新的验证数据结构 (ADS). 基于验证数据结构,服务提供商不仅可以执行 k -支配最近邻查询,同时在查询的过程中,可以利用提出的验证区域产生验证对象(验证区域详见第 5 节).

4.1 构建验证数据结构

为了高效的验证查询结果,验证数据结构需要满足较低的存储开销、计算消耗和传输开销. 其中,存储开销是指索引结构的大小;计算消耗是指查询和产生验证对象的消耗;传输开销是指验证对象的大小. 所以,如何设计一种高效的和简洁的验证数据结构至关重要. 此外,借助验证对象,客户端需要去验证查询结果的完整性和健壮性. 具体来说,完整性是指服务提供商返回了所有 k -支配最近邻查询结果且没有遗漏;健壮性是指服务提供商返回的所有 k -支配最近邻查询结果都是真实的数据且没有被篡

改. 在已有的验证数据结构 MR-tree 的基础上,我们提出了一种新的验证数据结构 MV^2R -tree. 在介绍验证数据结构 MV^2R -tree 之前,先介绍节点影响区域的定义.

定义 8(节点影响区域). 对于 R-tree 上的一个节点 N , (1) 如果 N 是叶子节点,其影响区域为覆盖节点 N 中所包含所有被支配点的 voronoi 单元 (IR) 的最小矩形; (2) 如果 N 是非叶子节点,其影响区域为覆盖所包含节点影响区域的最小矩形. 记作 $NIR(N)$.

例 4. 如图 9 所示,节点 N_4 包含了两个物体,其中物体 a 为支配点,物体 f 为被支配点. 所以节点 N_4 的影响区域为覆盖 $IR(f)$ 的最小矩形. 物体 f 的影响区域 $IR(f)$ 为三角形 ABC , 节点 N_4 的影响区域 $NIR(N_4)$ 为矩形 $ADEF$.

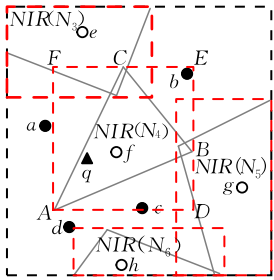


图 9 节点 NIR 示例

MV^2R -tree 的结构如图 10 所示,其构建的过程与 R-tree 的构建过程相似. 在 MV^2R -tree 中,每个非叶子节点由指向孩子节点的指针、最小边界矩形 (Minimum Bounding Rectangle, MBR)、节点摘要和节点影响区域 (NIR) 组成. 其中,每个非叶子节点的摘要由其节点 MBR、节点 NIR 和孩子节点摘要共同组成. 例如,节点 N_1 的摘要计算为 $D(N_1) = h(N_1.MBR | N_1.NIR | D(N_3) | D(N_4))$; 每个叶子节点包含了指向实际物体的指针、最小边界矩形 (MBR)、节点摘要和节点影响区域 (NIR). 其中,每个叶子节点的摘要由其节点 MBR、节点 NIR 和所有叶子 entry^① 的哈希值共同组成. 例如,节点 N_3 的摘要计算为 $D(N_3) = h(N_3.MBR | N_3.NIR | h(e) | h(b))$; 而每个叶子 entry 则由它的影响区域 (IR) 的顶点和相交支配点集合 (记作 T) 组成. 叶子节点 entry 的相交支配点集合 (T) 是指在由所有支配点组成的 voronoi 图中 (例如,支配点 $d_i \in D$, 其

① 叶子 entry 是指叶子节点所包含的物体元素.

voronoi 单元为 $V(d_i, D)$), 与叶子节点 entry 的影响区域 (IR) 相交的支配点的集合^①. 此外, MV^2R -tree 还包含了一个辅助链表结构, 如图 10 的右部分所示. 该辅助链表由两部分组成, 其中, 左部分为所有支配点; 右部分为产生每个支配点 voronoi 单元的集合 $G(o, D)$. 通过 MV^2R -tree, 可以快速查询出 k -支配最近邻结果, 并且产生验证对象 (VO).

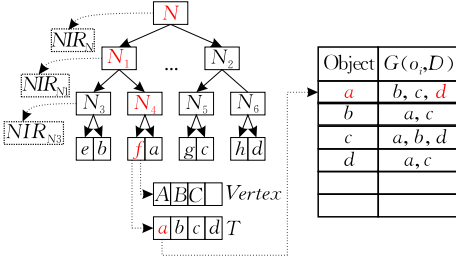


图 10 MV^2R -tree

4.2 k -支配最近邻查询

在介绍 k -支配最近邻查询过程之前, 我们首先介绍一个定理.

定理 1. 给定一个查询点 q 和物体 o , 如果查询点 $q \in IR(o)$, 那么对于支配点 $d_i \in D$, 与查询点 q 之间具有最短距离的支配点 $d_i \in T$.

证明. 我们从以下两个方面证明定理 1.

(1) $IR(o) \subseteq V(d_i, D)$. 基于 $q \in IR(o)$ 和 $IR(o) \subseteq V(d_i, D)$, 可以推断出 $q \in V(d_i, D)$. 进而可以推断出对于任意点 $p \in V(d_i, D)$, 点 p 与查询点 q 间的距离 $dist(p, q)$ 比 $V(d_i, D)$ 的产生点集 $G(d_i, D)$ 与查询点 q 间的距离 $dist(G(d_i, D), q)$ 近 (对任意 $d' \in G(d_i, D)$, 其 voronoi 单元 $V(d', D)$ 与 $IR(o)$ 不相交); (2) $IR(o) \not\subseteq V(d_i, D)$. 这意味着 $IR(o)$ 与多个支配点的 voronoi 单元 $V(d, D)$ 相交. 由于查询点 $q \in IR(o)$, 因此, q 一定位于这些相交的 voronoi 单元中的某一个支配点 d_i 的 voronoi 单元内. 进而可以推断出查询点 q 与支配点间的最短距离为 $dist(d_i, q)$. 定理 1 得证. 证毕.

注意到每个叶子节点 entry 存储了它的相交支配点集合 (T), 例如, 物体 f 的相交支配点集合为 $\{a, b, c, d\}$. 基于定理 1, 可以依次获得 k -支配最近邻查询结果. 但是, 我们观察到查询点 q 可能位于多个物体的影响区域的相交区域内 (例如, $q \in V(f, G(f, O)) \cap V(g, G(g, O))$), 造成结果集可能包含多个被支配点. 一种基本方法是直接比较所有可能的支配结果. 但是, 由于这种方法需要重复计算和检验查询点 q 与冗余支配点间的距离, 造成系统的计算代价比较大. 于是, 我们提出了一种优化的解决

方法. 首先, 将所有影响区域包含查询点 q 的被支配点加入到一个距离优先级队列, 并且按距离由小到大排列. 然后, 依次取队首元素并判断, 如果该元素是被支配点, 则将其加入结果集中并将其相交支配点集合 T 加入优先级队列; 如果该元素是支配点, 则将其加入结果集并将产生其 voronoi 单元的点 ($G(d, D)$) 加入优先级队列. 重复上述过程, 直至结果集的大小为 k 时停止算法. 我们知道一个被支配点的相交支配点集合 T 的平均大小是一个常数^[8]. 所以, 优化算法的比较次数要远远小于基本方法.

根据上述的处理, 实现在 MV^2R -tree 的 k -支配最近邻查询. 算法 1 描述了查询过程的具体细节.

算法 1. k -支配最近邻查询 k -dominantFind(q , $root, k$).

输入: 查询点 q , MV^2R -tree, k

输出: k -支配最近邻结果集 S

1. 初始化: 结果集 $S = \emptyset$; 栈 $U = \emptyset$; 优先级队列 $Q = \emptyset$;
2. 将 MV^2R -tree 的根节点 $root$ 加入栈 U 中;
3. WHILE (U 不为空)
4. 将栈首元素 e 出栈;
5. IF (e 是非叶子节点)
6. IF ($e.NIR$ 包含查询点 q)
7. 将 e 的孩子节点加入栈 U 中;
8. ENDIF
9. ELSEIF (e 是叶子节点)
10. IF ($e.NIR$ 包含查询点 q)
11. 将 e 的 entry 加入栈 U 中;
12. ENDIF
13. ELSEIF (e 是叶子 entry)
14. IF ($e.T$ 包含查询点 q)
15. 将 e 加入优先级队列 Q 中; // 优化算法
16. ENDIF
17. ENDIF
18. ENDWHILE
19. WHILE (Q 不为空 & $|S| < k$)
20. 将队首元素 e 出队;
21. IF (e 是被支配点)
22. 将 $e.T$ 加入 Q 中; // 定理 1
23. 将 e 加入 S 中;
24. ELSEIF (e 是支配点)
25. 将 $e.G(e, D)$ 加入 Q 中; // 定理 1
26. 将 e 加入 S 中;
27. ENDIF
28. ENDWHILE

① 如果一个支配点的 voronoi 单元包含该叶子节点的 IR 或 IR 包含此支配点的 voronoi 单元, 都将该支配点加入相交支配点集合 T 中.

根据算法 1, 首先利用栈 U 遍历 MV^2R -tree, 并将其影响区域包含查询点 q 的物体加入优先级队列 Q 中. 定理 1 说明了, 对于被支配点, 其支配点与查询点 q 最近距离必定在其相交集合 T 中 (Line 22); 同理, 也说明了对于支配点, 其下一个距离查询点 q 最近的支配点必定在其 voronoi 邻居点集中 (Line 25). 然后利用优化的方法将 k -支配最近邻查询结果加入结果集 S 中. 根据图 9 和图 10 所示, 例 5 详述了算法 1 的执行过程, 其中 $k=3$.

例 5.

- (1) $U: \{\emptyset\}; Q: \{\emptyset\}; S: \{\emptyset\}$
- (2) $U: \{N\}; Q: \{\emptyset\}; S: \{\emptyset\}$
- (3) $U: \{N_1, N_2\}; Q: \{\emptyset\}; S: \{\emptyset\}$
- (4) $U: \{N_1, N_2\}; Q: \{\emptyset\}; S: \{\emptyset\}$
- (5) $U: \{f, N_2\}; Q: \{\emptyset\}; S: \{\emptyset\}$
- (6) $U: \{N_2\}; Q: \{f\}; S: \{\emptyset\}$
- (7) $U: \{\emptyset\}; Q: \{f\}; S: \{\emptyset\}$
- (8) $U: \{\emptyset\}; Q: \{a, b, c, d\}; S: \{f\}$
- (9) $U: \{\emptyset\}; Q: \{b, c, d\}; S: \{f, a\}$
- (10) $U: \{\emptyset\}; Q: \{b, c\}; S: \{f, a, d\}$

4.3 算法分析

在本小节中, 分析了本文提出的 k -支配最近邻查询算法 (记为 OPT) 和 4.2 节中没有优化的算法 (记为 Naïve) 的计算代价, 并给出了两种算法的时间复杂度和空间复杂度.

所有空间物体集合 O , 空间所有支配点集合 D . 为了方便计算和不失一般性, MV^2R -tree 的每个叶子节点和非叶子节点的扇出为 t . 假设查询点 q 位于单个物体 o 的安全区域 $SSR(o)$ 内. 此外, 生成每个被支配物体 o (或支配物体 d) 的 voronoi 单元所需点集 $G(o, O)$ (或 $G(d, D)$) 的大小为 c . 可以知道, MV^2R -tree 的叶子节点的数目为 $|O|/t$, 且 MV^2R -tree 的树高 h 为 $\log_t |O|$. Naïve 算法和 OPT 算法遍历 MV^2R -tree 的过程是相同的且遍历的时间代价为 $O(\log_t |O|)$. 在进一步计算 k -支配最近邻的过程中, Naïve 算法需要在所有非完全支配点集 ($|D|+1$) 中计算到查询点 q 的距离前 k 近邻. 这部分计算代价为 $O((|D|+1)^2)$. 于是, Naïve 算法的时间复杂度为 $O(\log_t |O| + (|D|+1)^2)$. 而 OPT 算法只需利用优先级队列判断 $k-1$ 次且最大比较次数为 $c \times (k-1)$. 故 OPT 算法的时间复杂度为 $O(\log_t |O| + c \times (k-1))$. 对于空间复杂度, 在遍历 MV^2R -tree 时, 需要维护 $\log_t |O|$ 个数据元组. 而在进一步计算 k -支配最近邻的过程中, Naïve 算

法需要维护 $|D|+1$ 个数据元组, 空间复杂度为 $O(\log_t |O| + |D|+1)$. 而 OPT 算法只需要维护 $c \times (k-1)$ 个数据元组, 故空间复杂度为 $O(\log_t |O| + c \times (k-1))$.

5 移动 k -支配最近邻查询验证

5.1 构建验证区域

当服务提供商接收到客户端请求的查询 q 时, 首先根据算法 1, 服务提供商在 MV^2R -tree 上计算 k -支配最近邻查询结果 S . 然后, 服务提供商按照定义 7 的规则计算 k -支配安全区域 $KSSR(S, R)$. 如果查询点 q 位于一个物体 o 的安全区域内, 则服务提供商需要计算物体的安全区域 $SSR(o)$ 和 k 阶 voronoi 单元 $V_k(S, R)$. 如果查询点 q 位于多个物体的影响区域的相交区域内, 则服务提供商需要计算该相交区域和 k 阶 voronoi 单元 $V_k(S, R)$. 为了向用户提供可靠的查询服务, 服务提供商在返回查询结果的同时, 需要向用户提供必要的信息, 使得用户可以根据这些信息验证查询结果和安全区域的正确性. 为此, 本文设计了查询结果和安全区域的验证区域 (Verification Region, VR), 分别记作 VR-S 和 VR-KSSR. 此外, 本文还提出了两种产生验证区域的方法, 分别为 Circle-based 和 Rectangle-based.

(1) Circle-based

在 Circle-based 的方式下, 查询结果的验证区域 VR-S 是一个圆形区域, 记作 $\odot(q, r)$. 其中, r 是查询点 q 到结果集 S 中点的最远距离. 而安全区域 VR-KSSR 的验证区域为一系列圆形区域的并集区域, 记作 $\bigcup \odot(\phi, \maxdist(\phi, S))$. 其中, ϕ 表示安全区域 $KSSR(S, R)$ 的顶点集合. $\maxdist(\phi, S)$ 表示安全区域 $KSSR(S, R)$ 的顶点集合中的点距结果集 S 中点的最远距离. 如图 11 所示, 查询结果的验证区域 VR-S 为实心圆; 而安全区域 VR-KSSR 的验证区域为所有虚线圆所覆盖区域.

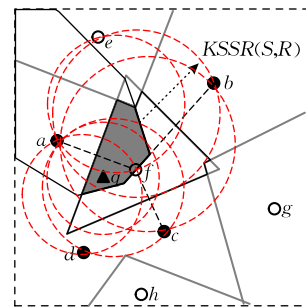


图 11 Circle-based 验证区域

可以明显观察到,安全区域 VR-KSSR 的验证区域为一个不规则的区域. 所以,基于此方法验证安全区域时,会使计算过程复杂化和显著地增加计算量. 此外, Circle-based 的方式在验证安全区域 VR-KSSR 时需要逐个圆验证是否存在点 p 影响安全区域 KSSR(S, R). 针对上述原因,我们提出了一种优化的 Rectangle-based 的方式生成查询结果和安全区域的验证区域.

(2) Rectangle-based

服务提供商在生成验证对象时,需要将必要的信息加入验证对象中以便客户端能够依据这些信息验证查询结果和安全区域. 于是,将安全区域的验证区域 VR-KSSR 分为两个部分: (a) $V_k(S, R)$. 为了验证这个部分,验证对象 VO 中需要包含 $G(V_k(S, R))$ 的信息. 其中, $G(V_k(S, R))$ 为生成 k 阶 voronoi 单元 $V_k(S, R)$ 的点集合; (b) $SSR(o)$. 为了验证这个部分,需要将精炼 $SSR(o)$ 区域的点集合 $G(SSR(o))$ 加入验证对象 VO 中. 值得注意的是,对于支配点,需要直接被安全区域的验证区域 VR-KSSR 包围; 而对于被支配点,只需要满足安全区域的验证区域 VR-KSSR 与被支配点所在的节点影响区域 NIR 相交. 所以,安全区域的验证区域 VR-KSSR 需要包围 $SSR(o)$. 基于上述两个条件,我们分别定义了安全区域和查询点的验证区域的最大和最小边界点. 对于安全区域,其分别为集合 $\{G(V_k(S, R)), G(o, O), SSR(o)\}$ 中坐标最大值和最小值对应的点,分别记作 $kssr_{\max}$ 和 $kssr_{\min}$. 而对于查询结果,其分别为集合 $\{S\}$ 中坐标最大值和最小值对应的点,分别记作 $r_{\max}$ 和 $r_{\min}$. 例 6 详细阐述了 Rectangle-based 方式的验证区域的生成过程.

例 6. 图 12 为 Rectangle-based 方式生成的验证区域. 如图 12 所示($k=2$),对于条件(a),安全区域的验证区域 VR-KSSR 应该包含 $G(V_k(S, R))$, 故将点集 $\{b, c, d\}$ 加入 VO 中. 因为该点集精炼了 $V_k(S, R)$. 而对于条件(b), VR-KSSR 应该包含 $G(f, O)$, 于是将点集 $\{a, b, c\}$ 加入 VO 中. 并且由于被支配点集 $\{e, g\}$ 精炼了 $SSR(f)$, 故 VR-KSSR 应

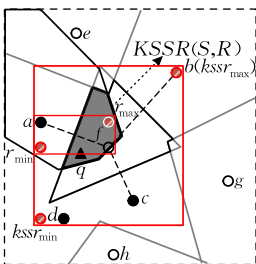


图 12 Rectangle-based 验证区域

该与被支配点集 $\{e, g\}$ 所在的节点影响区域 NIR 相交. 最后形成的安全区域的验证区域 VR-KSSR 为图中虚线矩形且上界点($kssr_{\max}$)和下界点($kssr_{\min}$)分别为矩形的右上点(b)和左下点.

定理 2. 查询结果的验证区域 $VR-S \subseteq$ 安全区域的验证区域 VR-KSSR.

证明. 我们从以下两个方面证明定理 2. (1) 对于 Circle-based 方式. 由于 $q \in IR(o) \Rightarrow q \in SSR(o)$, 且 $q \in V_k(S, R)$, 又 $KSSR(S, R) = SSR(o) \cap V_k(S, R) \Rightarrow q \in KSSR(S, R)$. 根据 Circle-based 方式下验证区域的定义可知, q 到结果集 S 的最远距离必然小于安全区域 KSSR(S, R) 内某一顶点到结果集 S 的最远距离, 即安全区域的验证区域 VR-KSSR 包含查询结果的验证区域 VR-S, 故定理 2 成立; (2) 对于 Rectangle-based 方式. 很容易得到结果集 $S \subseteq \{G(V_k(S, R)), G(o, O), SSR(o)\}$, 即安全区域的验证区域 VR-KSSR 包含查询结果的验证区域 VR-S, 故定理 2 成立. 证毕.

定理 2 说明了安全区域的验证区域与查询结果的验证区域之间的从属关系. 如果攻击者改变了它们之间的这种从属关系, 我们可以判定安全区域或查询结果被篡改. 下面算法 2 描述了查询结果和安全区域的验证区域的构建过程.

算法 2. 构建验证区域 VR_Construction(q , $root, k$).

输入: 查询点 q , MV^2 R-tree, k

输出: 验证区域 VR-S 和 VR-KSSR

1. $S \leftarrow k\text{-dominantFind}(q, root, k)$; // k -支配最近邻查询
2. $R \leftarrow DU(S - D)$;
3. $SSR(o_i) = (\cap IR(o_i)) \cap (\cap IR(o_j))$, 其中, $o_i \in R$, $o_j \in O - R$; // 式(1)
4. $V_k(S, R) = \cap (\cap \perp(p, p'))$, 其中, $p \in S, p' \in R - S$; // 式(2)
5. $KSSR(S, R) = SSR(o) \cap V_k(S, R)$; // 式(3)
6. $VR-KSSR \leftarrow \square(kssr_{\min}, kssr_{\max})$;
7. $VR-S \leftarrow \square(r_{\min}, r_{\max})$;
8. $VO \leftarrow VO_Construction(root, VR-KSSR)$;

在算法 2 中, 需要首先计算 k -支配最近邻查询结果和所有未被完全支配物体集合 (Lines 1~2). 第 3 行至第 5 行计算物体和查询结果的安全区域. 然后按照 Rectangle-based 方式构建验证区域 (Lines 6~7). 最后, 将得到的验证区域 VR-KSSR 代入验证对象的构建函数中进一步计算验证对象 (验证对象的计算见 5.2 节).

5.2 构建验证对象

验证对象的构建算法如算法 3 所示.

算法 3. 构建验证对象 VO_Construction($root$, VR-KSSR).

输入: MV^2R -tree, VR-KSSR

输出: 验证对象 VO

1. 初始化栈 $U = \emptyset$;
2. 将 MV^2R -tree 的根节点 $root$ 加入栈 U 中;
3. WHILE(U 不为空)
4. 将栈首元素 e 出栈;
5. IF(e 是非叶子节点)
6. IF($e.NIR$ 与 VR-KSSR 相交)
7. 将 e 的孩子节点加入栈 U 中;
8. ELSE
9. $(e.MBR, e.digest, e.NIR) \rightarrow VO$
10. ENDIF
11. ENDIF
12. IF(e 是叶子节点)
13. IF($e.NIR$ 与 VR-KSSR 相交)
14. 将 e 中所有物体加入 VO 中;
15. ELSE
16. $(e.MBR, e.digest, e.NIR) \rightarrow VO$
17. ENDIF
18. ENDIF
19. ENDWHILE
20. RETURN VO;

在构建验证对象的过程中,通过深度优先遍历 MV^2R -tree,将所有与安全区域的验证区域 VR-KSSR 不相交的的非叶子节点的 MBR、摘要和 NIR 加入验证对象 VO 中.此外,对于叶子节点,如果与 VR-KSSR 相交,则将节点中所有物体加入 VO 中;反之,则将节点的 MBR、摘要和 NIR 加入 VO 中.

5.3 验证 k -支配最近邻查询

当客户端接收到服务提供商发送的验证对象之后,客户端利用验证对象包含的信息验证 k -支配最近邻查询结果和安全区域的正确性.算法 4 描述了客户端的验证过程.

算法 4. 验证 k -支配最近邻查询.

输入: VO, $Sig(root)$

输出: 验证结果

1. $h(root) \leftarrow$ 利用 VO 重新构建 MV^2R -tree 根节点的签名;
2. IF($h(root) = Sig(root)$)
3. $D' \leftarrow$ 从 VO 中解析出所有的物体;
4. $N \leftarrow$ 从 VO 中解析出所有的非叶子节点;
5. $S' \leftarrow$ 从 VO 中解析出 k -支配最近邻查询结果;
6. FOR(S' 中每个物体 e)
7. IF($e.NIR$ 包含查询点 q)
8. 物体 e 是真实的支配结果;

9. ELSE
10. 物体 e 是非真实的支配结果;
11. ENDIF
12. FOR(D' 中每个非支配点 t)
13. IF($t.NIR$ 不包含查询点 q)
14. 没有遗漏非支配点;;
15. ELSE
16. t 是遗漏的非支配点结果;
17. ENDIF
18. ENDFOR
19. $VR-S \leftarrow \square(r_{min}, r_{max})$;
20. $VR-KSSR \leftarrow \square(kssr_{min}, kssr_{max})$;
21. FOR(N 中的每个非叶子节点 n)
22. IF($n.NIR$ 不包含查询点 q)
23. 没有遗漏支配点;
24. ENDIF
25. IF($n.NIR \cap VR-S = \emptyset$)
26. S' 是真实的 k -支配最近邻查询结果;
27. IF($n.NIR \cap VR-KSSR = \emptyset$)
28. 安全区域 VR-KSSR 是真实的;
29. ENDIF
30. ENDIF
31. ENDFOR
32. ELSE
33. RETURN Failure;
34. ENDIF

在验证的过程中,客户端首先按照自底向上的方式,利用验证对象中包含的节点摘要信息和物体信息计算 MV^2R -tree 根节点的签名(Line 1).如果重新计算的签名与数据所有者的签名匹配,则说明所有数据信息都是源于数据所有者且满足健壮性.反之,则说明一些数据被遗漏或被攻击者篡改.然后,客户端从验证对象中解析出物体信息、非叶子节点和查询结果(Lines 3~5).如果对于结果集中每个物体的影响区域都包含查询点 q ,则说明每个物体都是真实的支配结果.接着,算法第 13 行和 14 行表示根据查询结果和安全区域计算相应的验证区域.值得注意的是,在此过程中,需要首先利用 VO 重构安全区域.我们知道,安全区域分为两个部分: k 阶 voronoi 多边形和被支配点的安全区域.其中,从 VO 中解析出的所有物体集合 D' 中,可以确定计算 k 阶 voronoi 多边形和被支配点的安全区域所需的物体集合.此外,通过判断查询结果和安全区域的验证区域是否与非叶子节点的节点影响区域相交,如果不相交,则说明查询结果和安全区域都是真实的且满足完整性.反之,则说明验证结果不正确.

5.4 用户位置更新

通过定义 7 可知,查询用户不将向服务提供商发起新的查询请求当且仅当该用户没有离开安全区域 $KSSR_{old}$. 在这种方式下,用户不需要频繁地与服务提供商通信和交互. 因此,这样可以在一定程度上减少传输消耗. 但是,为了进一步减少传输消耗,我们充分利用了先前生成的验证对象 VO_{old} 解决这个问题.

具体来说,当用户离开安全区域并发起新的查询 q_{new} 时,用户首先检查新的查询点 q_{new} 是否与验证对象 VO_{old} 中的节点 NIR 相交. 如果相交,这意味着相交节点中的某些物体可能为潜在的查询结果. 在这种情况下,由于用户不能获取这些节点的物体信息,所以用户不能直接借助验证对象 VO_{old} 去计算新的查询结果,并且需要向服务提供商发送新的查询请求 q_{new} . 服务提供商接收到新的查询请求后,重新计算验证对象 VO_{new} ,并返回给用户;如果不相交,这意味着查询结果仍然在先前的验证对象 VO_{old} 中. 于是,用户遍历先前的验证对象 VO_{old} 中的每个物体,并检查该物体的影响区域是否包含新的查询点 q_{new} . 如果包含,则说明该物体可能为一个可能的查询结果,并加入到一个距离优先级队列 Q 中;反之,则说明该物体不可能为查询结果,并直接过滤掉. 在完成整个遍历之后,依次取出 Q 的队首元素,并加入新结果集 S' 中,直至 $|S'|$ 为 k . 基于优先级队列 Q 和结果集 S' ,用户可以重新计算新的查询结果 S' 和安全区域 $KSSR_{new}$ 的验证区域. 下面例 7 描述了当用户离开安全区域 $KSSR_{old}$ 时,验证对象 VO_{old} 的更新过程和重新计算新的查询结果和安全区域的验证区域的过程.

例 7. 如图 13 所示,当前查询点 q 所在安全区域 $KSSR_{old}$ 为多边形 $ABCDE$,此时验证对象 VO_{old} 中包含了 $\{[[[[[e], [b]], [[f], [a]]], [[N_5.MBR, N_5.digest, N_5.NIR], [N_6.MBR, N_6.digest, N_6.NIR]]]\}$. 而当用户在移动的过程中,离开安全区域 $KSSR_{old}$

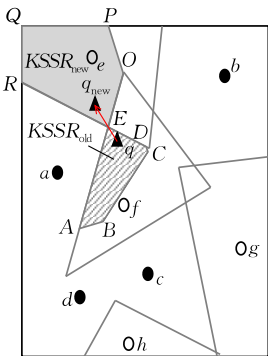


图 13 用户位置更新

时,首先判断新查询点 q_{new} 与 VO_{old} 中的非叶子节点的影响区域 $NIR(N_5.NIR$ 和 $N_6.NIR)$ 不相交. 所以,新的查询结果仍然在 VO_{old} 所包含的物体中. 然后,依次遍历 VO_{old} 中的物体 $\{e, b, f, a\}$,并将物体影响区域包含新查询点 q_{new} 的加入优先级队列 Q 中 ($Q = \{a, b, e\}$). 于是,新的查询结果集 S' 为 $\{e, a\}$. 按照式(1)~(3),可以进一步计算出新的安全区域 $KSSR_{new}$ 为多边形 $OPQRE$ 且新的验证对象 $VO_{new} = VO_{old}$.

6 实验分析

6.1 实验设置

在本小节中,实验的硬件环境为 Intel Core2 Duo CPU@2.93 GHz、8 GB DDR3 内存、1 TB 硬盘和 Windows7 操作系统;实验所有程序使用 JAVA 程序语言实现.

在实验中,本文采用了真实的数据集 CAR^① 和合成数据集 SYN(均匀分布). 其中,CAR 为加利福尼亚道路分割数据,其共包含 2 096 702 个数据点. 对于文中数据索引结构 MV^2 R-tree,其每个页面的大小为 4 KB,且每个节点摘要的大小为 32 Bytes. 在计算 MV^2 R-tree 的摘要和根节点签名时采用 SHA-256 和 RSA-1024 作为哈希函数和签名策略. 此外,实验中,我们将平面空间区域归一化到一个 $100\,000 \times 100\,000$ 的区域内且每个单位长度表示 1 m. 假设用户的平均移动速度为 5 m/s,并按照设置的取样间隔(s)更新用户的位置. 实验中,我们利用文献[23]的轨迹生成方法,模拟了用户的移动轨迹. 此外,还为数据集中的每一个物体合成在区间 $[0, 100\,000]$ 变化且满足均匀分布的非空间属性值. 实验中的其他参数设置见表 2.

表 2 系统参数设置

参数名称	默认值	参数范围
数据集基数	100 000	$[100\,000, 1\,000\,000]$
数据集	CAR, SYN	CAR, SYN
非空间属性维度	2	$[2, 8]$
k 值	10	$[10, 100]$
取样间隔	100	$[100, 1000]$

实验中,主要针对如下 4 个方面进行测试:(1) 数据基数的影响;(2) k 值的影响;(3) 取样间隔的影响;(4) 非空间属性维度的影响. 实验中,选取 CAR 和 SYN 为默认数据集,并且对比了两个数据集对

① <http://snap.stanford.edu/data/roadNet-CA.html>

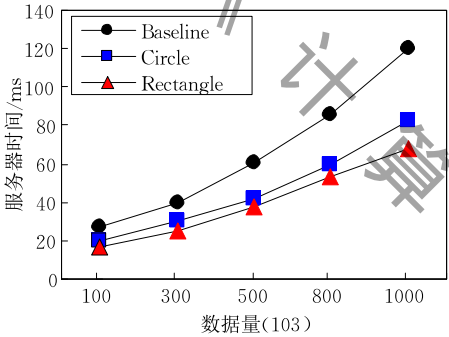
实验的影响. 此外, 实验中, 我们将文中提出的算法 Circle-based 和 Rectangle-based 与一种 Baseline 方法在上述 4 个方面应用不同数据集进行详细地对比. 其中, Baseline 方法为查询中没有优化且验证区域基于 Circle-based 和验证中没有更新策略的方法; Circle-based 为优化了查询且验证区域基于 Circle-based 和验证中应用了更新策略的方法; 而 Rectangle-based 为优化了查询且验证区域基于 Rectangle-based 和验证中应用了更新策略的方法.

6.2 数据基数的影响

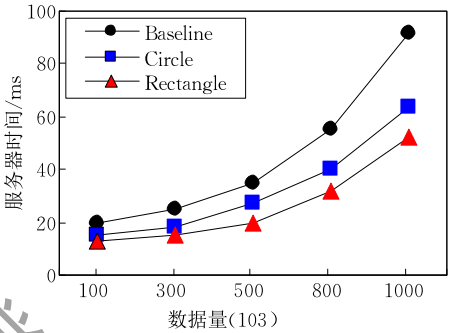
在本小节中, 主要测试了不同数据集大小对性能的影响. 实验中, 我们从数据集 CAR 中随机抽取 100 000、300 000、500 000、800 000 和 1 000 000 的数据, 并生成了均匀分布的 SYN-100 000、SYN-300 000、SYN-500 000、SYN-800 000 和 SYN-1 000 000 用于

测试.

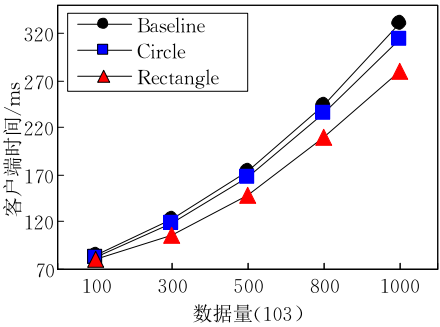
如图 14 所示, Baseline、Circle 和 Rectangle 在服务器时间、VO 大小和客户端时间等方面的变化曲线随着数据量的增大而增长. 这主要是因为当数据量增大时, 会造成更多物体的影响区域相交, 从而导致物体的安全区域的面积缩减且用户移动出安全区域的概率增加. 这样, 用户可能会频繁地与服务提供商通信, 从而相应的传输代价增加. 此外, 更大的数据量会产生更大的索引结构, 在遍历树和过滤不必要的结果时, 会消耗更多的计算时间. 从图 14 的 (a)、(c) 和 (e) 可以明显看出, 在真实数据集 CAR 上, Rectangle 的方法要优于 Circle 和 Baseline, 这是由于 Rectangle 产生的验证区域比较小和简洁, 能够更快的遍历树, 产生验证对象. 而 Baseline 和 Circle 方法都是基于复杂的验证区域, 而在遍历树



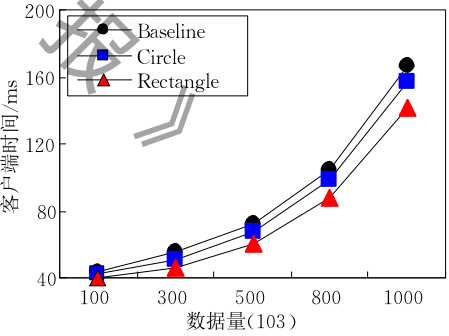
(a) 服务器时间(CAR)



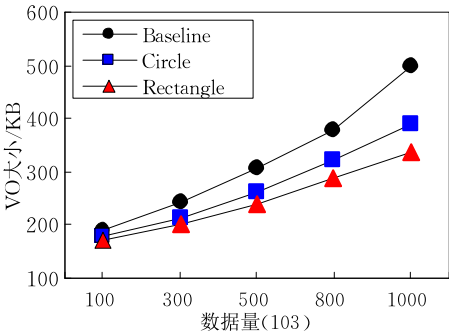
(b) 服务器时间(SYN)



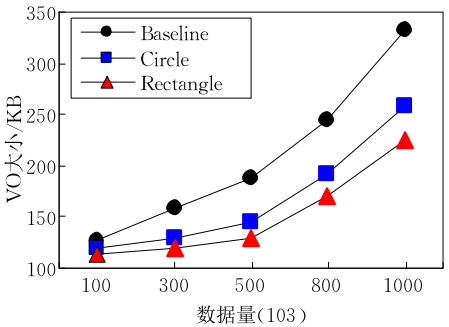
(c) 客户端时间(CAR)



(d) 客户端时间(SYN)



(e) VO大小(CAR)



(f) VO大小(SYN)

图 14 数据基数大小的影响

时,较大的验证区域会造成验证对象 VO 中包含更多的对象,使得传输代价变大.

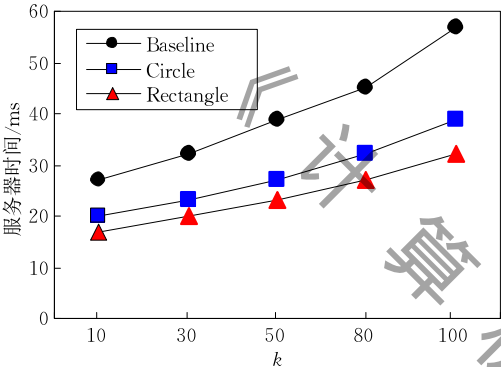
图 14(b)、(d)和(f)所示为在 SYN 数据集上测试的数据基数的影响. 可以看出其结果与在 CAR 数据集上测试的结果曲线整体变化趋势相似. 但是,我们可以注意到图(b)、(d)和(f)中,当数据集大小较小(小于 500 000)时,曲线的涨幅比较缓慢;而大于 500 000 时,曲线增长较快. 这主要由于 SYN 中的数据点是均匀分布的. 相较于 CAR,SYN 会产生一个较大的安全区域,从而用户移动出安全区域的概率减小并且相应的计算代价降低. 而在数据集 CAR 中,数据分布较密集,即使在数据量较小时,也会产生一个较小的安全区域,从而用户移动出安全

区域的概率增大,使得曲线的涨幅较快.

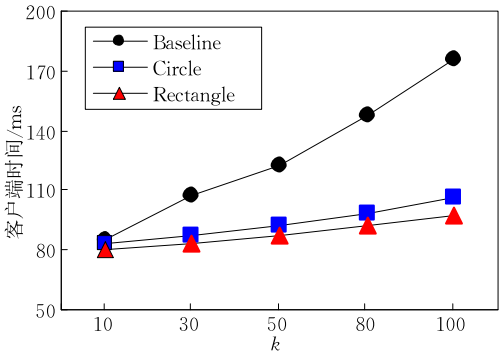
6.3 k 值的影响

本小节中,我们主要测试了 k 值的变化对服务器时间、VO 大小和客户端时间等方面的影响. 实验中,本文对每个 k 值随机生成了 1000 个查询点,记录的是 1000 次查询的平均值.

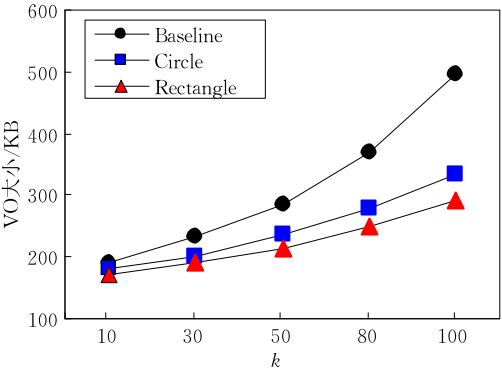
图 15(a)所示为当 k 值变化时,相应的服务器时间的变化. 我们知道,服务器时间主要包括查询时间和产生验证对象 VO 的时间. 从图中可以明显看出 Circle-based 和 Rectangle-based 的方法明显优于 Baseline 方法. 主要是由于前者在查找结果时,优化了比较次数,节省了时间. 而随着 k 值增大,优势越明显. Rectangle-based 优于 Circle-based,主要是



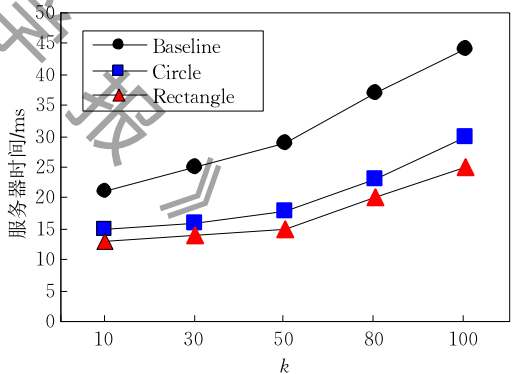
(a) 服务器时间(CAR)



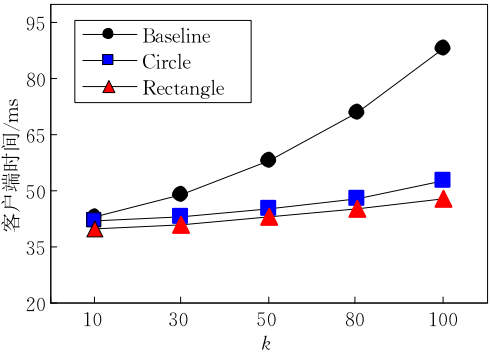
(b) 客户端时间(CAR)



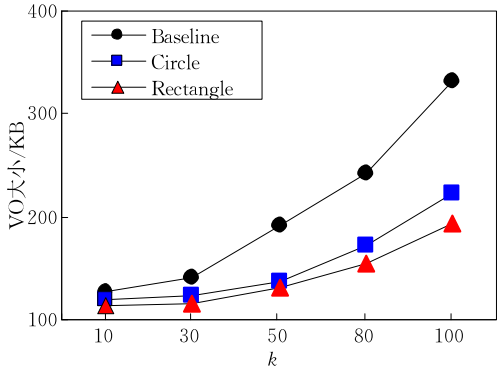
(c) VO大小(CAR)



(d) 服务器时间(SYN)



(e) 客户端时间(SYN)



(f) VO大小(SYN)

图 15 k 值的影响

因为后者在产生验证对象时,需要生成一个复杂的验证区域,增大了计算量.图 15(b)和(c)所示为当 k 值变化时,相应的客户端时间和 VO 大小的变化.我们可以知道,随着 k 值的变大,安全区域的面积会逐渐缩小.而由于 Baseline 没有应用提出的 VO 更新策略,所以当用户移动出安全区域后,需要重新计算验证对象,故客户端时间消耗会比较大.但是,随着 k 值的变大,验证区域会相应增大,从而 VO 中所包含的物体数目也会随之增加,故 VO 的大小会随之变大.

图 15(d)、(e)和(f)所示为在 SYN 数据集上测试的 k 值的影响.可以看出其与在 CAR 数据集上测试的结果曲线整体变化趋势相差不大.但是,可以明显看出,服务器时间、客户端时间和 VO 大小等要

优于 CAR.在最好的情况下,SYN 仅占 CAR 的 50%.这主要是因为数据分布对索引结构、安全区域等造成的影响.

6.4 非空间属性维度的影响

本小节中,我们主要测试了非空间属性维度的变化对服务器时间、VO 大小和客户端时间等方面的影响.实验中,我们分别设置维度为 2,4,6 和 8.此外,我们从数据集 CAR 中随机抽取 100 000 数据和生成 SYN-100 000 用于测试.我们对每个维度随机生成了 1000 个查询点,记录的是 1000 次查询的平均值.

如图 16 所示,随着非空间属性维度的增加,Baseline、Circle 和 Rectangle 在服务器时间、VO 大

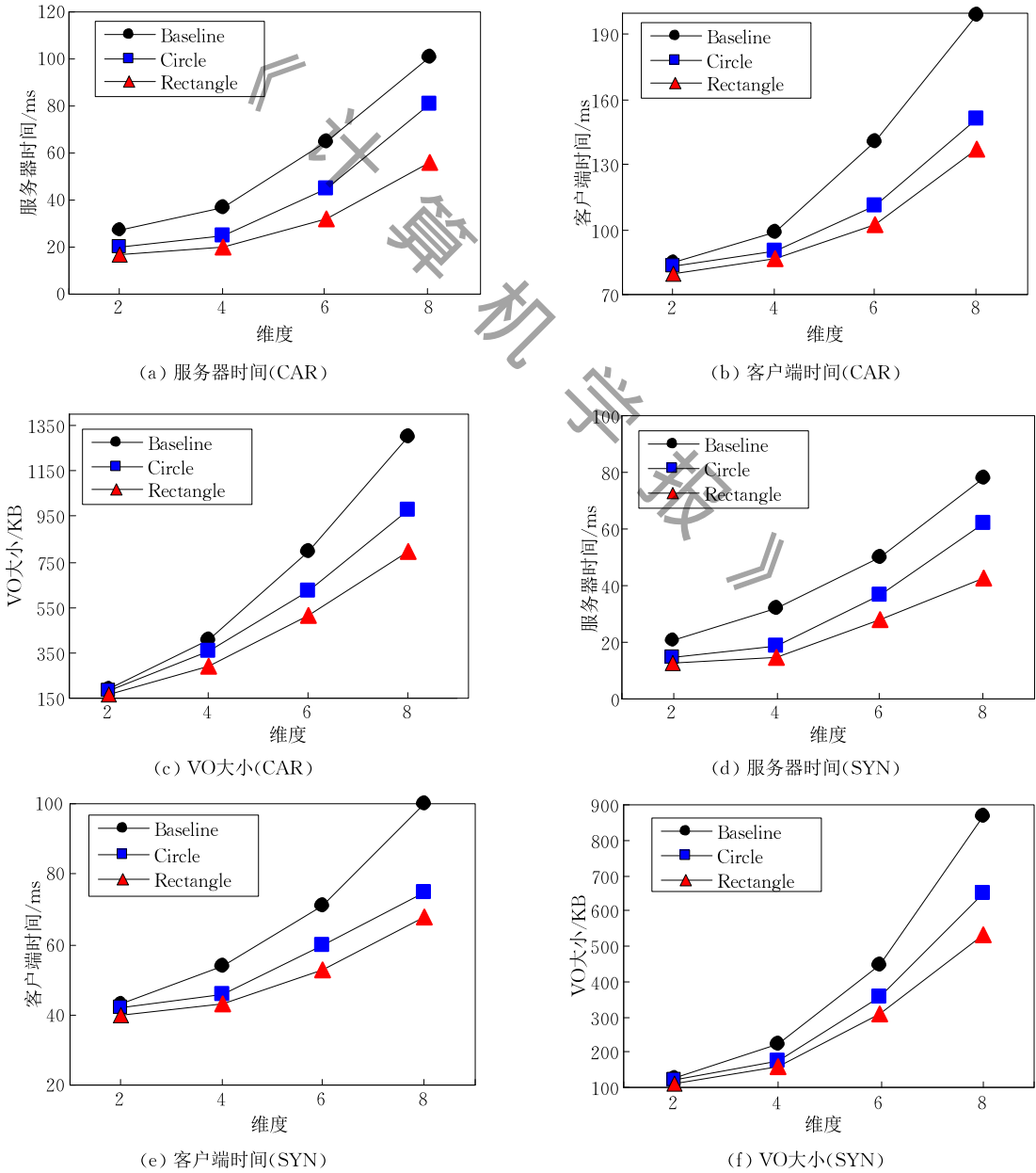


图 16 非空间属性维度的影响

小和客户端时间等方面都相应增长且曲线增长趋势基本相同。还可以看出,当维度较低时,服务器时间、VO大小和客户端时间等曲线增长缓慢;而当维度较大时,曲线增长明显变快。此外,Baseline 方法的曲线增长明显快于后者,主要是由于随着数据非空间属性维度的增加,物体间的支配关系会发生改变。原来有支配关系的两个物体间,可能不存在支配关系。这样会造成总支配点的数目增加,而 Baseline 方法需要从所有支配点及其 voronoi 单元中查询结果,从而导致查询和验证的时间变长。

此外,从图 16(d)、(e)和(f)中可以看出,其与 CAR 数据集上测试结果曲线的整体变化趋势相近。但是,SYN 在服务器时间、VO大小和客户端时间等方面的表现优于 CAR,主要是因为随着维度增

加,物体间支配关系变复杂,而 SYN 会产生一个较大的安全区域,相较于 CAR,用户不用频繁与服务器通信,减少了这种复杂支配关系的比较。

6.5 取样间隔的影响

本小节中,主要测试了取样间隔的影响。分别设置取样间隔为 100,200,300,400 和 500(单位:s)。我们知道,如果用户移动离开安全区域,需要发起新的查询请求并重新计算查询结果和安全区域。图 17 所示为取样间隔变化时,服务器时间、VO大小和客户端时间等曲线随之变化的结果。图 17 可以观察到,当取样间隔比较小且小于 300 时,服务器时间、VO大小和客户端时间等曲线变化不明显;而当取样间隔变大且大于 300 时,曲线都发生明显地增长。这是由于当取样间隔比较小时,用户在这个时间间

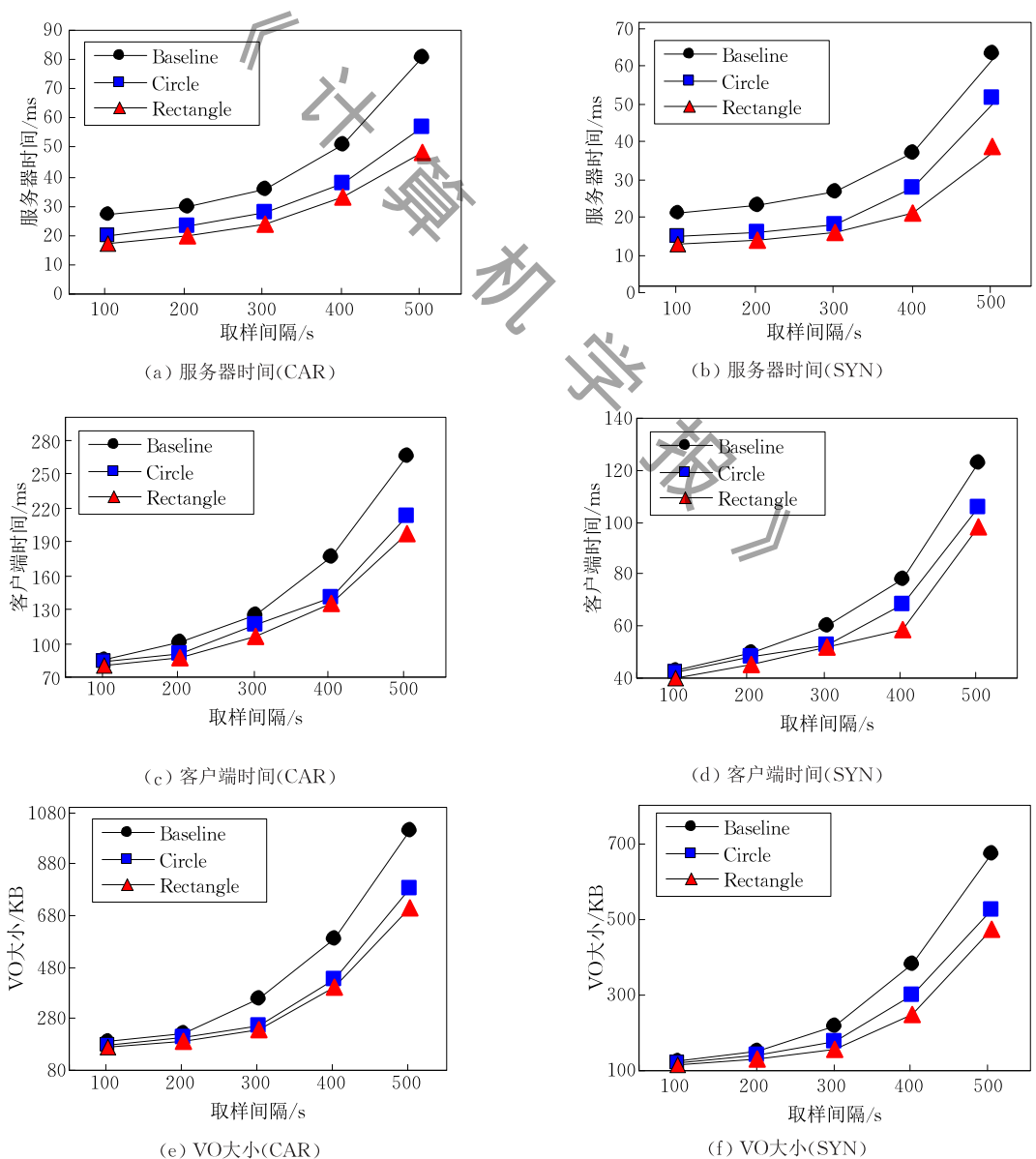


图 17 取样间隔的影响

隔内移动的区域也相对较小,即移动出安全区域的概率较小.反之,当取样间隔比较大时,用户移动出安全区域的概率较大.此外,我们还可以看出,当取样间隔大于 300 时,Baseline 方法的性能明显低于 Circle 和 Rectangle 方法.这是因为 Baseline 没有应用 VO 更新策略,所以,每当用户离开安全区域,都需要重新计算查询结果和安全区域,造成频繁地与服务提供商通信,使得性能下降.而 Rectangle 方法与 Circle 方法的性能相似,主要是由于产生了相同的安全区域.

从图 17(b)、(d)和(f)可以看出,随着取样间隔变大,SYN 曲线变化比较平缓.而当取样间隔大于 400 时,曲线出现明显增长.主要是由于 SYN 中均匀分布的数据产生比较大的安全区域,使得用户更新位置的概率降低.而大于 400 时,用户移动出安全区域的概率变大,故曲线出现明显变化.

7 总 结

由于 LBS 提供商是不受信任的,为了提供可靠的查询服务,LBS 服务商应以一种可验证的方式提供查询服务,即客户可以对查询结果进行可靠性验证.在大数据的背景下,如何快速的响应查询服务请求,以较小的数据传输代价,给客户提供更可靠的查询结果变得至关重要.本文针对移动 k -支配最近邻查询验证的问题,提出了一种新的验证数据结构 MV^2R -tree.通过 MV^2R -tree 可以高效地执行移动 k -支配最近邻查询和验证.同时,在验证的过程中,提出了两种产生验证区域的方法,分别为 Circle-based 和 Rectangle-based.此外,还提出了一种验证对象的更新策略,可以在一定程度上有效地减少传输消耗.最后,通过大量的对比实验证明了本文算法的正确性和有效性.

在今后的工作中将继续研究在移动和连续查询时的可信查询问题,以及针对其他复杂的查询,如反向查询等在执行查询时,如何保证查询可靠性方面的研究.

参 考 文 献

- [1] Wu K L, Chen S K, Yu P S. Incremental processing of continual range queries over moving objects. *IEEE Transactions on Knowledge and Data Engineering*, 2006, 18(11): 1560-1575
- [2] Nutanong S, Zhang R, Tanin E, et al. The V^* -diagram: A query-dependent approach to moving k NN queries. *Very Large Data Bases Journal*, 2008, 1(1): 1095-1106
- [3] Cheema M A, Lin Xuemin, Zhang Wenjie, et al. A safe zone based approach for monitoring moving skyline queries// *Proceedings of the International Conference on Extending Database Technology*, Genoa, Italy, 2012: 275-286
- [4] Lin X, Xu J, Hu H. Authentication of location-based skyline queries//*Proceedings of the CIKM International Conference on Information and Knowledge Management*. New York, USA, 2011: 1583-1588
- [5] Lin X, Xu J, Hu H, et al. Authenticating location-based skyline queries in arbitrary subspaces. *IEEE Transactions on Knowledge and Data Engineering*, 2014, 26(6): 1479-1493
- [6] Lin X, Xu J, Gu J. Continuous skyline queries with integrity assurance in outsourced spatial databases//*Proceedings of the Web-Age Information Management*. Harbin, China, 2012: 114-126
- [7] Su S, Yan H, Cheng X, et al. Authentication of top- k spatial keyword queries in outsourced databases//*Proceedings of the International Conference on Database Systems for Advanced Applications*. Hanoi, Vietnam, 2015: 567-588
- [8] Chen Q, Hu H, Xu J. Authenticating top- k queries in location based services with confidentiality//*Proceedings of the Very Large Data Bases*. Hawai'i, USA, 2013: 49-60
- [9] Wu D, Choi B, Xu J, et al. Authentication of moving top- k spatial keyword queries. *IEEE Transactions on Knowledge and Data Engineering*, 2015, 27(4): 922-935
- [10] Pang H H, Jain A, Ramamritham K, et al. Verifying completeness of relational query results in data publishing// *Proceedings of the International Conference on Management of Data*. Baltimore, USA, 2005: 407-418
- [11] Yang Y, Papadopoulos S, Papadias D, et al. Authenticated indexing for outsourced spatial databases. *Very Large Data Bases Journal*, 2009, 18(3): 631-648
- [12] Merkle R C. A certified digital signature//*Proceedings of the International Cryptology Conference Advances in Cryptology—CRYPTO*. California, USA, 1989: 20-24
- [13] Li F, Hadjieleftheriou M, Kollios G, et al. Dynamic authenticated index structures for outsourced databases//*Proceedings of the International Conference on Management of Data*. New York, USA, 2006: 121-132
- [14] Sharifzadeh M, Shahabi C. VoR-tree: R-trees with voronoi diagrams for efficient processing of spatial nearest neighbor queries. *Very Large Data Bases Journal*, 2010, 3(1): 1231-1242
- [15] Mouratidis K, Sacharidis D, Pang H H. Partially materialized digest scheme: An efficient verification method for outsourced databases. *Very Large Data Bases Journal*, 2009, 18(1): 363-381
- [16] Chen Q, Hu H, Xu J. Authenticated online data integration services//*Proceedings of the International Conference on Management of Data*. Melbourne, Australia, 2015: 167-181
- [17] Hu H, Xu J, Chen Q, et al. Authenticating location-based services without compromising location privacy//*Proceedings of the International Conference on Management of Data*. Scottsdale, USA, 2012: 301-312

[18] Fan Z, Peng Y, Choi B, et al. Towards efficient authenticated subgraph query service in outsourced graph databases. *IEEE Transactions on Services Computing*, 2014, 7(4): 696-713

[19] Peng Y, Fan Z, Choi B, et al. Authenticated subgraph similarity search in outsourced graph databases. *IEEE Transactions on Knowledge and Data Engineering*, 2015, 27(7): 1838-1860

[20] Yung D, Lo E, Yiu M L. Authentication of moving range queries//*Proceedings of the CIKM International Conference on Information and Knowledge Management*. Maui, USA, 2012; 1372-1381

[21] Yiu M L, Lo E, Yung D. Authentication of moving k NN queries//*Proceedings of the 24th International Conference on Data Engineering*. Hannover, Germany, 2011; 565-576

[22] Yung D, Li Y, Eric L, et al. Efficient authentication of continuously moving k NN queries. *Transactions on Mobile Computing*, 2015, 14(9): 1806-1819

[23] Brinkhoff T. A framework for generating network-based moving objects. *Geoinformatica*, 2000, 6(2): 153-180



CUI Ning-Ning, born in 1988, Ph. D. candidate. His research interests include spatial databases and privacy preserving.

YANG Xiao-Chun, born in 1973, Ph. D. , professor, Ph. D. supervisor. Her research interests include database management and privacy preserving.

WANG Bin, born in 1972, Ph. D. , associate professor. His research interest is big data management.

ZHU Huai-Jie, born in 1988, Ph. D. candidate. His research interests include spatial databases and privacy preserving.

Background

The popularity and rapid development of mobile internet-work, such as 4G network, is propelling the evolution of Location-based services in a large extent. While the data owner is in the face of more and more complicated query services, it cannot satisfy the requests. In order to solve this problem, outsourcing the spatial databases to the LBS provider is a certain choice. However, when the LBS provider who is untrusted provides more advanced query services to the client, the LBS provider should make its location-based services in an authenticatable manner for reliable query services. So that the correctness of the service results can be verified by the client. In the background of big data, how to response the service request quickly with low transmission cost and return more reliable results is the focus and research direction of researchers.

As an important issue, various spatial queries and moving queries are authenticated in location-based services. But these queries do not take spatial and non-spatial attributes

into consideration simultaneously.

In order to solve above problems, in this paper, we study the authentication of k -dominant nearest neighbor query which is a natural combination of k NN query and skyline query. We design an efficient authenticate data structure which can implement k -dominant NN query and authentication. In the process of verification, we design two methods to construct verification region. Moreover, we also propose a verification object updating strategy. Lastly, we evaluate the performance of proposed algorithms.

This work is partially supported by the National Science Foun for Distinguished Young Scholars under Grant No.61322208, the National Natural Science Foundation of China under Grant Nos. 61272178, 61572122, the National Natural Science Foundation of China for Key Program under Grant No. 61532021, and the Fundamental Research Funds for the Central Universities No. N161606002.