doi:10.11959/j.issn.1000-436x.2015109

基于 Voronoi 图预划分的 LBS 位置隐私保护方法

马春光，周长利，杨松涛，赵蕴龙
（哈尔滨工程大学 计算机科学与技术学院，黑龙江 哈尔滨 150001）

摘 要：为了解决服务器面临大量用户请求时匿名效率下降的问题，分别提出适用于静态用户和动态用户的协作匿名方法。首先基于 Voronoi 图划分全局区域，再由中心服务器组织本区域内用户实现协作匿名，由于服务器无需为每个用户单独构造匿名区，降低了服务端的负担；针对查询过程中用户提供真实位置信息带来位置隐私泄露的问题，提出了逆向增量近邻查询算法。用户以固定锚点代替真实位置，向位置服务器逐步获取兴趣点候选集并计算出想要的结果，避免位置隐私直接泄漏的同时获取精准查询结果。该算法同时解决了锚点与用户过近而带来的位置隐私被推断问题。实验表明本方法在有效保护用户位置隐私的同时，具有良好的工作效率。
关键词：位置隐私；协作匿名；锚点；逆向增量查询
中图分类号：TP311 **文献标识码**：A

Location privacy-preserving method in LBS based on Voronoi division

MA Chun-guang, ZHOU Chang-li, YANG Song-tao, ZHAO Yun-long
(School of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China)

Abstract: In view of low efficiency when the anonymous server faces a large number of users, a cooperative anonymous method is proposed for static and dynamic users respectively. Based on the Voronoi division of the global area, a central server organizes the users in its region to achieve cooperative anonymity, the server needn't construct anonymous region alone for each user, and the burden of the server is reduced. In view of providing actual location when users query from a LBS server, a decrease nearest neighbor query algorithm is proposed. A user's actual location is replaced with a stationary anchor location and gets the points of interest candidate set from the LBS server gradually. By running the algorithm, precise results can be got and avoiding exposure to a user's location privacy. The algorithm can also help to reduce the possibility of location inference when the anchor chooses closely to the user. Experiments show that our method can guarantee the user's location privacy, and provide a good working efficiency.
Key words: location privacy; cooperative anonymity; anchor; decrease nearest neighbor query

1 引言

近年来，以移动通信技术和 GPS 等定位技术为代表的现代无线通信业得到快速发展，基于位置的服务(LBS, location based service)得到广泛应用。其中，用户基于自身位置的兴趣点查询服务是 LBS 中使用频率最高的服务类型之一。用户对兴趣点(PoI, points of interest)的查询，按照查询方式可以分为快照查询(snapshot query)和连续查询(continuous query)[1]，典型查询语言分别为“距我最近的商场”和“持续报告距我最近的加油站”。按照查询结果可以分为近邻查询(nearest neighbor query)和范围查

收稿日期：2014-03-11；**修回日期**：2014-07-15
基金项目：国家自然科学基金资助项目(61170241, 61073042)；中央高校基础科研业务费重大专项基金资助项目(HEUCFZ1105)；高等学校博士学科点专项科研基金资助项目(20132304110017)；黑龙江省自然科学基金资助项目(F201229)
Foundation Items: The National Natural Science Foundation of China (61170241, 61073042); The Fundamental Research Funds for the Central Universities (HEUCFZ1105); Specialized Research Fund for the Doctoral Program of Higher Education (20132304110017); The Natural Science Foundation of Heilongjiang Province (F201229)

询(range query)，典型查询语言分别为“距我最近的 K 个餐馆”和“距我 R km 范围内的所有餐馆”。用户在获取这些服务时需要先提供自身的位置数据，而由于位置信息与用户身份、习惯爱好及家庭住址等隐私数据的时空关联特性，使位置数据的泄漏会造成用户更为深入的隐私信息泄露，位置隐私必须在服务过程中得到有效保护。

LBS 中需要保护的隐私可分为身份隐私[2]、位置隐私[3]和查询隐私[4]3 类。如用户发出“距离我家最近的传染疾病医院”，在这条查询中用户首先不想让人知道谁发出了这样的查询，其次不想让人知道自己当前可能所在的位置及查询内容。而攻击者以推断出用户身份与隐私数据之间的映射关系为最终目标，任何脱离具体身份的隐私数据对于攻击者来说都是没有价值的。所以，所有隐私保护方法都围绕用户身份与隐私数据分离来进行的。

LBS 服务中的位置隐私保护过程按照先后顺序可以分为位置匿名及查询中的位置隐私保护 2 个过程。在位置匿名研究过程中，如何有效隐匿(cloaking)用户真实位置信息进而不被唯一锁定得到了广泛研究[5~8]，这些以匿名服务器为中心的模式用来实现某个用户与其他 k–1 个用户不可区分(即 k 匿名)，达到模糊某个用户真实位置的目的。但是这种中心架构模式需引入匿名服务器作为假设可信第三方，以用户提供自身真实位置为基础构造匿名区，如图 1 所示，这种基于中心架构的单独匿名方式需要为每个用户构造不同的匿名区，在面临大量用户服务请求时，易造成服务器负担重、匿名时延长、成功率低及等问题。

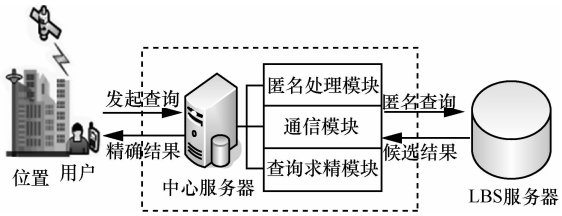


图 1 中心服务架构

针对有中心架构的缺陷，Chow 等[9]提出了无中心服务器的 P2P 用户协作匿名方式，使匿名区和 k 匿名首次实现了分离，此类方法由头节点发起匿名，用户之间协作构成匿名组共同实现 k 匿名而无需构造匿名区，具有架构简易、匿名效率好等优势。但这种用户协作匿名方式依然存在着用户端负担重，用户间可信度难于鉴别等缺陷。

另一方面，查询过程中的隐私保护方法也得到了广泛的研究，如采用假位置点[10,11]替代真实位置的查询方法，但是这类方法存在结果不精确或额外开销过大等缺陷。Yiu 等[12]提出的 SpaceTwist 方法以增量查询的方式，较为准确地实现了 K 邻近查询(KNN)，但是该方法依然存在没有实现 k 匿名的缺陷。针对这个缺陷 Gong[13]及孟小峰等[14]分别提出 KAWCR 及 Coprivacy，弥补了 SpaceTwist 的不足。

针对用户在获取 LBS 服务的匿名和查询 2 个过程中存在的问题，本文的研究内容及贡献如下。

1) 针对匿名过程中匿名服务器为每个用户构造匿名区效率低的问题，提出基于 Voronoi 图全局预划分的协作匿名方法。由中心服务器承担类似 P2P 方法中头节点的功能，组织同一匿名区的用户构造协作匿名组实现 k 匿名，而无需单独构造匿名区。该方法针对静态用户和动态用户分别提出了相应的协作匿名方法。

2) 针对查询过程中用户以真实位置发起查询服务请求可能带来的位置隐私泄露问题，提出无需用户提供真实位置信息的查询算法 DNN，用户利用锚点替代真实位置向 LBS 服务器发起兴趣点查询，在返回候选集中再结合自身真实位置计算出精确查询结果，避免 LBS 服务器不可信带来的位置隐私直接泄露。

3) 针对查询过程中选取的锚点位置与用户位置过近带来的位置隐私泄露等问题，提出查询算法 DNN，算法以固定锚点为圆心逐步收缩查询范围的方式，使攻击者很难锁定用户所在范围，并能够获取用户所需的邻近兴趣点。

2 相关工作

为了更好地从数据生成端保护用户位置隐私，2003 年 Gruteser 等[15]提出了位置 k 匿名模型。通常的位置 k 匿名是指对用户位置进行空间上的扩展，降低位置分辨率，增加攻击者锁定用户准确位置的难度。而时空匿名是指在空间模糊的基础上，增加时间轴的延迟，解决用户稀疏环境下匿名成功率低的问题。2005 年 Gedik 等[5]提出了矩形匿名区 Clique Cloak，该方法生成以用户位置为中心的矩形框。2006 年 Mokbel[16]首次提出了中心服务结构，利用匿名服务器为用户构造匿名区，用户需要向匿名服务器提供自己的准确位置。同年，Mokbel[6]又提出了 Casper 空间匿名模型，该模型对匿名服务器覆盖区域进行网格划分，以单元格为基本单位实现 k 匿

名，本文借鉴了这种空间划分实现 k 匿名思想。2007 年，Kalnis 和 Ghinita 等[9]提出了 Center Cloak 模型，该模型构造以用户为中心的圆形匿名区域，但是用户位置易被攻击者锁定。同年，2 位作者又提出了 Hilbert Cloak[8]方法，该方法利用 Hilbert 曲线填充构造区域并以此为依据检索 k−1 个用户。2009 年 Chow 等[17]提出了改进的 Casper*，该模型首次提出查询处理也是位置隐私保护中需要考虑的重要内容。2011 年杨晓春等[3]提出了路网环境的匿名区域构造方法。2012 年，Li 等[18]提出了对匿名区域进一步划分的思想，通过划分碎片区域解决了稀疏用户条件下匿名成功率下降的问题，这种划分思想也为本文研究提供了参考。2014 年，马春光等[19]提出基于伪随机置换的位置隐私保护方法，实现了基于位置的盲查询和完美匿名。现有研究多采用中心匿名服务器构造匿名区来实现 k 匿名，但这种架构存在系统负担过重、易成为攻击热点及强假设可信等的问题。Chow 等[9]提出了无中心的 P2P 匿名方法，用户在发起请求前自行寻找满足需求的 k 个邻近用户，但该方法存在用户计算资源消耗多的缺陷。

在查询过程中的位置隐私保护研究方面，2004 年，Hong 等[10]采用标志对象的方法，向 LBS 服务器发送替代真实位置的路标 (significant object)位置，该方法存在查询结果不够精确的缺陷。2005 年，Kido[11]提出了假位置(dummy)的方法，该方法同时发送多个假位置给 LBS 服务器，其中只有一个是用户真实位置，该方法虽然可以保证查询结果精确，但是会成倍增加服务器额外负担。2008 年 Yiu 等[12]提出了客户端运行的 SpaceTwist 查询算法，该方法结合路标位置的思想，基于增量近邻查询方法(INN, increasing nearest neighbor)所设计，如图 2 所示，用实心“•”表示真实的用户坐标，用“×”表示锚点(anchor)位置。增量查询开始时 LBS 服务器以锚点为中心寻找其附近的 PoI，供应空间(supply space)逐步扩大，当供应空间完全覆盖需求空间(demand space)时表示用户找到了满足其需求的 PoI 集合，查询结束。该文同时探讨了锚点位置的选取原则，指出了锚点与用户位置过近会造成查询过程中供应空间迅速覆盖需求空间使查询结束，无法找到 K 个邻近兴趣点。SpaceTwist 方法依然存在未实现 k 匿名等问题。针对这个问题，Gong 等[13]提出基于中心匿名服务器的 KAWCR 方法，实现了 k 匿名。孟小峰等[14]提出了 Coprivacy 方法，该方法采用 P2P 用户协作的思想，用户在发起查询前构造包含 k 个用户的匿名组，并利用匿名组的密度中心作为锚点逐步获取查询结果。

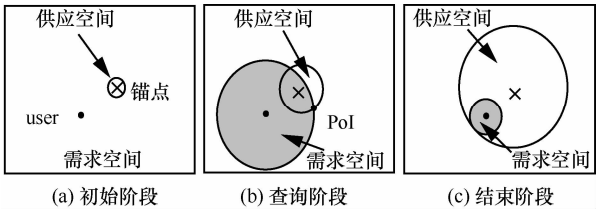


图 2　SpaceTwist 增量查询

根据上述文献，可以总结出 LBS 中位置隐私保护方法设计必须考虑的 2 个原则条件。

1) 必须实现用户位置 k 匿名。

2) 不轻易提供自身真实位置给其他任何实体。

满足 k 匿名，可以使用户在提出服务请求时，与其他 k−1 个用户不可区分，防止被唯一锁定。条件 2)是指用户在匿名或查询过程中，将真实位置信息提供给任何其他实体，都存在被暴露的可能。尽量少释放位置信息可以从根本上防止位置隐私泄露。

3　预备知识

本节主要介绍系统结构、协作匿名概念及基于 Voronoi 图的路网划分 3 个内容。

3.1　系统结构

本文提出的协作匿名位置隐私保护方法的系统架构由 3 部分组成，如图 3 所示，移动用户端包含慢速移动用户(如行人等)和快速移动用户(如车辆等)以 u_k 表示，配备智能通信终端，终端包含匿名模块、定位/通信模块和增量查询模块。其中，匿名模块负责向自己附近某个适当的中心服务器 CS_i (CS, central service)登记加入匿名组；定位/通信模块负责从 GPS 等定位系统获取自身位置坐标，负责与其他实体通信；增量查询模块负责向 LBS 服务器提出查询请求，并根据查询候选集计算出想要的精确结果。

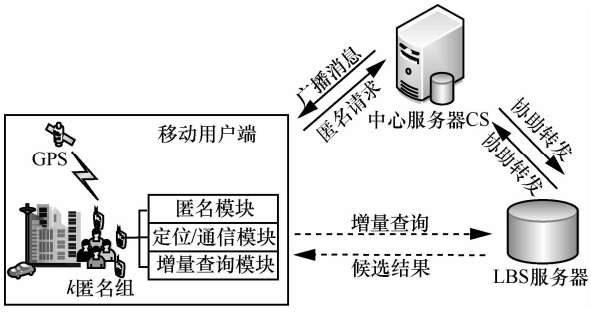


图 3　系统结构

CS 持续更新并广播用户登记情况等信息，表明在其附近有若干用户以便有匿名需求的用户判断是否加入匿名组，在组织用户实现协作匿名后转发用户查询请求。LBS 服务器根据用户查询请求，将用户查询结果候选集逐步发送给用户。

3.2 协作匿名概念

匿名方法可以分为单独匿名和协作匿名。单独匿名是指匿名处理器为每个用户单独构造匿名区。而协作匿名则是处于同一区域内的几个用户一起实现 k 匿名，无需单独为每个用户构造匿名区。

用户相互协作构成匿名组共享匿名区的方式可以解决单独匿名给服务器带来负担重的问题，但需要有一个通信实体组织其他用户共同形成 k 匿名组。现有协作匿名多以用户间协作(P2P)方式实现，这种方式难以保证所有参与者均可信，且被选举作为代理的簇头结点资源消耗很大，并不适合于资源受限的移动用户。基于此，引入能力较强的 CS 组织管理区域内全体用户协作实现 k 匿名。

在实际情况中，用户位置是时刻变化的，用户需要对所属区域进行判断，包括判断下一时段是否仍在该区域停留和是否即将进入另一个区域，以便 CS 能够获取本区域内的用户准确信息并组织匿名。匿名区域的划分方法是构造协作匿名组的基础。

3.3 基于 Voronoi 图的路网划分

Voronoi[20]图是一种广泛应用于空间分割的几何结构，它能清晰表现平面空间内实体间的邻近关系。平面内给定 n 个点，其中距离 CS_i 点比其他点都近区域的确定方法是将每个 $CS_i \in CS$ 均与周围 N 个邻近点连接并做连线的中垂线，N 条中垂线围成的多边形为所求区域。如图 4 所示，以 CS_1 为核心的多边形区域内任意点 A 与 CS_1 距离比其他 CS_i 点更近。

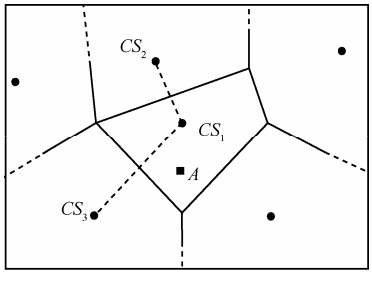


图 4 Voronoi 划分

类似地，由于 CS 多部署在人流密集及交通便利处，如十字路口、加油站等。当一定密度的 CS 部署成功后，利用 Voronoi 图将整个地理区域 Ω 划分成以 CS 为核心的多个子区域，作为实现协作匿名的匿名区。定义城市 Voronoi 划分图如下。

定义 1 设平面区域 Ω 内包含 n 均匀分布的离散中心服务器集合 $CS=\{CS_1,CS_2,\cdots,CS_n\}$，其中任意点 $CS_i \in CS$ 的 Voronoi 区域 $V(CS_i)$ 为 Ω 内所有到 CS_i 距离最小点的集合，称为 V 区，如图 4 所示，$V(CS_1)=\{A \mid dist(A,CS_1) \leqslant dist(A,CS_k), A \in \Omega, k \neq 1\}$，$dist(A,B)$ 表示点 A、点 B 间的欧几里德距离，CS_i 称为 Voronoi 图生成元。

以 CS_i 为生成元的整个城市 Voronoi 图可以表示为 $V(\Omega)=\{V(CS_1),V(CS_2),\cdots,V(CS_n)\}$。显然，相邻 V 区均有公共边，且相互不重合覆盖了整个路网区域。路网内的所有用户均属于某个 V 区，并且总能找到一个邻近的 CS_i 为其服务，聚集在同一 CS_i 周围的移动用户协作构成匿名组。

4 匿名及查询中的位置隐私保护方法

用户获取 LBS 查询服务分为 3 个阶段：初始化阶段、协作 k 匿名阶段和查询兴趣点阶段。初始化后有查询需求的用户首先进入 k 匿名阶段，向 CS_i 发起加入协作匿名组的请求，CS_i 组织本区域内用户构成匿名组成功后转发用户密文查询请求，否则等待下一时刻再次构建匿名组，直至该用户离开该区。有查询需求的用户必须先向某个 CS_i 登记，登记成功后周期向 CS_i 发送告知报文，直至离开该区或完成查询。在查询阶段，LBS 服务器以 CS_i 为圆心锚点，逐步收缩半径收集兴趣点并转发给用户，用户用自身真实位置自行计算出所需兴趣点集合。

4.1 初始化阶段

可信权威机构 TA (trust authority)选取合适的 CS 部署位置，通过控制邻近 CS 最小距离的方式确保生成 V 区总保持在合理大小范围内。CS 部署完成后，TA 获取所有 CS_i 坐标生成路网 V 图。

TA 对 CS、用户等各类合法实体进行注册并分配密钥材料。为了保证通信安全，可采用公钥基础设施。用户用密钥材料生成假名集用来向某个 CS_i 登记，CS_i 可以验证假名的合法性，文献[21]给出了保护隐私的假名认证方法，本文不做类似研究。

TA 将每个 CS_i 负责的 V 区范围 $V(CS_i)$ 通知相应 CS_i，CS_i 生成并初始化一个按登记先后时间排列的用户登记表 $List=(u_k,t_r,n_i)$，u_k 及 t_r 分别为用户登记假名和登记时间，n_i 为登记用户总数，同时以

一定频率定期广播公布当前登记用户情况报文 $B_{cs_i}=(CS_i,loc_{cs_i}n_i,t_i,r_{\min},r_{\max})$，其中，$loc_{cs_i}$ 表示 CS_i 坐标，t_i 表示广播时间，$r_{\max}$ 表示 CS_i 到 V 区多边形顶点距离，称为 V 区最大半径，$r_{\min}$ 表示 R_i 到 V 区多边形边的最小距离，称为 V 区最小半径。如图 5 所示，虚线圆半径分别为 $r_{\max}$ 和 $r_{\min}$。

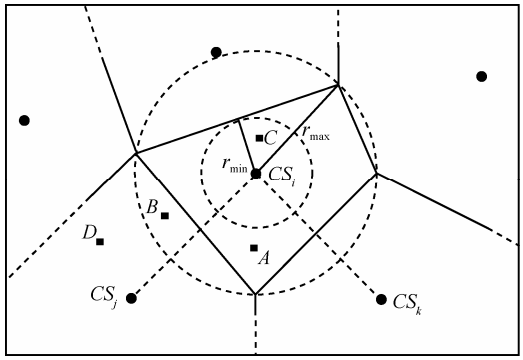


图 5　CS_i 所管辖的 V 区

定义 2　与 CS_i 距离 $dist(u_k,CS_i)<r_{\min}$ 的用户 u_k 位置所构成的区域称为确信区，距离 $dist(u_k,CS_i)>r_{\max}$ 的点构成排除区，距离 $r_{\min}\leqslant dist(u_k,CS_i)\leqslant r_{\max}$ 的点构成的环形区域称为非确信区。

对 V 区的再划分可以判断处于任意 CS_i 确信区的用户一定在包含在该 V 区的多边形中，如图 5 点 C；处于排除区的用户一定不在，如点 D；处于非确信区的用户 2 种情况都有可能，如点 A 和点 B。

4.2　协作 *k* 匿名阶段

在协作匿名阶段，用户会收到多个邻近 CS 转发来的用户登记情况报文 $B=\{B_{cs_1},B_{cs_2},\cdots,B_{cs_n}\}$。用户唯一处于某个 V 区内，需要判断自己所处的 V 区及即将进入的区域，以便向合适的 CS_i 登记。用户发起查询前需要预先判断所在 V 区广播报文是否满足 k 匿名需求，是则将密文查询内容一起发送给 CS_i，否则等待下一个时段的报文 B_{cs_i}。

具体说，如图 5 所示，用户首先判断自己所在的 V 区：u_k 收到若干个 B_{cs_i} 时，首先判断报文广播时间，丢弃旧报文。在剩余报文中计算自身位置与每个 CS_i 的距离 $dist(loc_{u_k},loc_{cs_i})$，当存在使 $dist(loc_{u_k},loc_{cs_i})<r_{\min}$ 成立的报文，由定义 2 可知此时用户一定处在 $V(CS_i)$内，如点 C，直接向 CS_i 发送登记消息；否则丢弃所有 $dist(loc_{u_k},loc_{cs_i})>r_{\max}$ 的报文，此时用户处于 CS_i 的排除区。当 $r_{\min}\leqslant dist(loc_{u_k},loc_{cs_i})\leqslant r_{\max}$ 成立时，用户无法准确判断自身是否处于 $V(CS_i)$内，如 B 点。此时用户计算其余每个 CS_i 最小距离 $\min\{dist(loc_{u_k},loc_{cs_i})\}$ 选出与自己最近的 CS，进行登记。但这意味着处于非确信区的用户可能没有收到所在 V 区的广播报文，而向其他最近的 CS 登记，由于 CS 为访问热点，周围协助转发报文用户较多，处在非确信区用户没收到所在 V 区广播报文概率较低或暂时性的，因此错误登记的概率较低。当然，一定存在用户一直没有收到所处 V 区的报文的情况，这说明 CS_i 周围能够转发广播报文的用户很少，这种情况可以认为即使收到广播报文也无法满足用户 k 匿名需求，不必进行登记。上述过程算法描述如下。

算法 1　静态/低速用户所在 V 区判断及登记算法

1) **Procedure**：用户 u_k 收到 n 个 CS 广播报文集合 $B=\{B_{cs_1},B_{cs_2},\cdots,B_{cs_n}\}$，集合 B 中每个元素表示成为 $B_{csi}=(CS_i,loc_{cs_i},n_i,t_i,r_{\min},r_{\max})$

2) 生成空数组 *Array* 和 *Register*

3) **for** 每个 $B_{cs_i}\in B$ **do**

4) **while** $t_{\text{now}}-t_i<\delta$ 且 $t_{\text{now}}-T_{\text{reg}}(CS_i)>\lambda$ **do** //t_{now} 为当前时间，δ、λ 为阈值，T_{reg} 为上次向 CS_i 登记时间，左侧表达式帮助丢弃所有旧报文，右侧表达式避免重复登记

5) 计算 $dist(loc_{u_k},loc_{cs_i})$

6) **if** $dist(loc_{u_k},loc_{cs_i})<r_{\min}$ **then**

7) 丢弃其他报文

8) **else if** $dist(loc_{u_k},loc_{cs_i})>r_{\max}$ **then**

9) 丢弃 B_{cs_i}

10) **else** $Array\leftarrow B_{cs_i}$

11) 计算 *Array* 中的最小距离 $\min\{dist(loc_{u_k},loc_{cs_i})\}$ //u_k 找到与自己最近的 CS_i 登记

12) **if** 存在 $\min(dist(loc_{u_k},loc_{cs_i}))$ **then**

13) 向 CS_i 发送登记报文

14) $Register\leftarrow T_{\text{reg}}(CS_i)$ //在时刻 T_{reg} 向 CS_i 注册

15) **End Procedure**

上述算法适用于静止或移动速度较慢的用户，如路边座椅上休息或散步的用户。当用户移动相对速度较快时，就存在错误登记的情况。如图 5 所示，当用户处于位置 A，并且运动方向大致指向图下方时，此时用户虽然处于 CS_i 管辖区，但是用户即将离开此区域，此时再向 CS_i 登记显然是不合理的。针对动态用户，修改静态算法，同时加入方向预估，

以确保移动用户向适当的 V 区 CS_i 登记。算法如下。

算法 2 动态用户 V 区的判断及登记算法

1) **Procedure**：用户 u_k 收到 n 个 CS 广播报文 $B=\{B_{cs_1},B_{cs_2},\cdots,B_{cs_n}\}$，其中每个元素表示成为 $B_{cs_i}=(CS_i,loc_{cs_i},n_i,t_i,r_{\min},r_{\max})$

2) 生成空数组 *Array* 、*Register*

3) **for** each $B_{cs_i}\in B$ **do**

4) **while** $t_{\text{now}}-t_i<\delta$ **do**

5) 计算 $dist(loc_{u_k},loc_{cs_i})$

6) **if** $dist(loc_{u_k},loc_{cs_i})<r_{\min}$ 且 $t_{\text{now}}-T_{\text{reg}}(CS_i)>\lambda$ **then**

7) 丢弃其他报文 // u_k 处在 CS_i 为圆心 $r_{\min}$ 为半径的圆形区域内，无需再判断其他报文

8) **else if** $dist(loc_{u_k},loc_{cs_i})>r_{\max}$ **then**

9) 丢弃 B_{cs_i}

10) **else** $Array\leftarrow B_{cs_i}$

11) **while** $Array=!\text{NULL}$ **do** // 数组非空

12) **for** 每个 $B_{cs_i}\in Array$ **do** // 进行方向筛选

13) 计算 $\theta=(\overrightarrow{V_{u_k}},\overrightarrow{u_kCS_i})$ // $\overrightarrow{V_{u_k}}$ 表示 u_k 行驶方向，$\overrightarrow{u_kCS_i}$ 为 u_k 指向 CS_i 向量，θ 为两向量夹角

14) **if** $\theta\geqslant 90^\circ$ **then**

15) 丢弃该报文

16) 计算 *Array* 中的最小距离 $\min\{dist(loc_{u_k},loc_{cs_i})\}$

17) **if** 存在 $\min\{dist(loc_{u_k},loc_{cs_i})\}$ **then**

18) 向 CS_i 发送登记报文

19) $Register\leftarrow T_{\text{reg}}(CS_i)$ // 在时刻 T_{reg} 向 CS_i 注册

20) **End Procedure**

算法 2 描述了当用户速度较快时，存在即将进入新区域的可能，用户不能轻易丢弃其他区域发来广播报文。用户在判断此时所在区域并预估即将进入下一区域可能时，情况相对复杂，分 3 种主要情况来说明算法 2 中登记区预估依据。

1) 处于确信区的用户无需预估即将进入的 V 区。如图 6(a)所示，当动态用户 u_k 处于确信区收到广播报文 B_{cs_i}，如位置 C 时，用户根据自身运动情况决定是否向 CS_i 登记。用户运动情况大致可以分为 2 种：a) u_k 与 CS_i 的连接向量 $\overrightarrow{u_kCS_i}$ 与 u_k 运动方向 $\overrightarrow{V_1}$ 夹角 $\theta_1<90^\circ$ 时，表明用户首次进入该区域并收到广播报文 CS_i，此时用户直接向 CS_i 登记，无需预估即将进入的 V 区；b) 当用户方向为 $\overrightarrow{V_2}$、夹角 $\theta_2>90^\circ$ 时，表示用户即将离开确信区，甚至即将离开 V 区，这表明用户早已进入 V 区甚至在确信区已行驶过一段时间，可以认为用户已经在 CS_i 登记过。此时在 C 处收到的广播 B_{cs_i} 为重复报文，可以丢弃，算法 2 第 6 行第 2 个表达式实现该功能。当然，会存在用户刚进入 V 区到达点 C 又迅速离开的可能(如遇到紧急情况立即返回)，但这种情况为小概率事件。综上，当用户处于确信区时，如果首次收到该区 B_{cs_i} 广播报文则会向其登记，而即将离开时收到的重复报文会被丢弃，无需计算方向角并预估即将进入的 V 区后再登记。因此，算法 2 并未将处于确信区的用户运动方向作为登记依据。

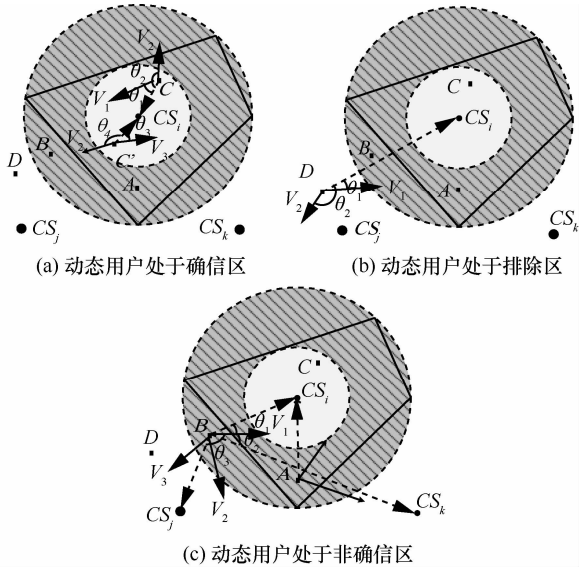


图 6 动态用户登记 V 区预估

2) 处于排除区的用户无需预估即将进入的 V 区。如图 6(b)所示，当动态用户处于排除区，如位置 D 时收到广播报文 B_{cs_i}，无需向该报文发出者 CS_i 登记。这是因为当用户行驶方向 $\overrightarrow{V_2}$ 与向量 $\overrightarrow{u_kCS_i}$ 夹角 $\theta_2>90^\circ$ 时，表示用户将远离 $V(CS_i)$，所以丢弃该报文。当夹角 $\theta_2<90^\circ$ 时，说明用户有进入 $V(CS_i)$ 的可能，可能会与 CS_i 更近且有更大概率再次收到该报文。综上，由于此时与该区距离稍大，总是丢弃该报文，如算法第 8、9 行。因此，算法 2 将算

法 1 第 4 行中限制重复登记的第 2 个语句移到第 6 行中，确保处于排除区的用户下一时刻进入 $V(CS_i)$ 能不丢弃该报文。

3) 处于非确信区的用户需要预估即将进入的 V 区。如图 6(c)所示，当动态用户处于非确信区，如位置 B 时可能收到多个 CS 的广播报文，如 CS_i 或 CS_j 甚至是 CS_k。处于这个边界区域的用户必须根据自身运行方向预估下一时刻所在的区域：处于 B 点的用户如果行驶方向为 $\vec{V}_1$ 时：a) 其与向量 $\overrightarrow{u_k CS_j}$ 夹角 $\theta_3 > 90^\circ$，表示用户正在远离 CS_j，此时虽然用户仍在 $V(CS_j)$中，仍然丢弃该报文，此处可以看到，虽然在算法 2 第 4 行中取消了对重复登记限制的语句，但算法 2 第 10 行~第 15 行仍然能够利用行驶方向丢弃重复报文，具有良好的健壮性；b) $\vec{V}_1$ 与向量 $\overrightarrow{u_k CS_i}$、$\overrightarrow{u_k CS_k}$ 的夹角 θ_1、θ_2 均小于 90° 且 $\theta_1 \approx \theta_2$，此时用户很难断定将进入哪个 V 区，需要进一步判断，在算法 2 的第 16 行~第 19 行中对夹角均小于 90° 的 CS 采用近距离选取原则，选取距离更近的 CS_i 作为登记目标；c) 用户运行方向为 V_2、V_3 及处于位置 A 时的情况类似，这里不再赘述。

综上，通过 V 区再划分只有处于非确信区的用户需要结合行驶方向预估下一时刻所在 V 区，说明 V 区再划分方法简化了用户登记方式。加入方向判断，可以帮助用户提高登记的准确率。当 CS_i 收到用户登记消息及密文查询请求后，组织段时间内积累的 $k_{\max}$ 个用户构成协作匿名组($k_{\max}$ 为用户组中最大 k 匿名需求)，去掉其假名后将 $k_{\max}$ 个用户密文查询请求同时发送给 LBS 服务器，实现 k 匿名。

4.3　逆向增量查询阶段

当用户所处区域满足其 k 匿名需求后，则向 LBS 服务器发起查询请求。用户利用锚点代替自己的真实位置向 LBS 服务器发送逆向增量查询请求 DNN，服务器以锚点为圆心逐渐缩小半径将兴趣点返回给用户，用户根据返回的候选结果结合自己的真实位置计算出自身周围的兴趣点。

这种近邻增量查询 INN 用户首先需要选择锚点，锚点位置多围绕用户真实位置选择，其选择方法有多种，比较之前的研究成果，本文选取用户所在区域 CS_i 的位置作为锚点位置，理由如下。

1) 邻近原因。由于用户时刻处于某个 V 区内，且与登记 CS_i 最近或即将进入该 V 区，易于围绕该 CS_i 进行查询。

2) 安全原因。用户自行计算选取的锚点，攻击者利用公开的锚点选取算法，能够将用户限制在某个区域内，多个限制区域的交集很容易确定用户实际位置。本文采用固定锚点无需计算，只能表明用户处于该 V 区内，避免用户真实位置因计算锚点而被推断。

3) 查询效率原因。固定锚点 CS_i 作为查询标志点提交给 LBS 服务器时，服务器首先查看自己缓存表中是否有以 CS_i 为锚点的同类查询，是则直接发送给用户。而计算锚点方式每次提交的锚点坐标不同，LBS 服务器需要以新坐标反复查询数据库，系统负担重且查询时间长。

但根据文献[12]的论述，用户选择距离自己较近的锚点会使查询很快结束，用户也会被限制在某个区域内，存在隐私泄漏的风险。根据 4.2 节中的论述，以 CS_i 为锚点的用户处于半径较小的确信区时，用户位置与锚点位置距离较近，如果仍采取 INN 查询，则存在上 τ 述问题。为此，设计了用户端运行的逆向增量查询算法(DNN, decreasing nearest neighbor)，解决上述问题，实现位于确信区用户 K 近邻查询，过程描述如下。

如图 7 所示，处于 CS_i 确信区的用户 u_k 实现 k 匿名后，开始提出 DNN 查询请求。用户自行定义查找范围为以 γ 为半径的需求空间，并以 CS_i 的位置为锚点、半径为的圆为供应空间，τ 初始值为 $\tau \geqslant dist(loc_{u_k}, loc_{cs_i}) + \gamma$ 的随机值。以 γ 赋值初始化小顶堆 W_k，用户来存储当前若干个邻近兴趣点距离。LBS 服务器以供应空间为基础，逐步缩小搜寻兴趣点 P_i，算法如下。

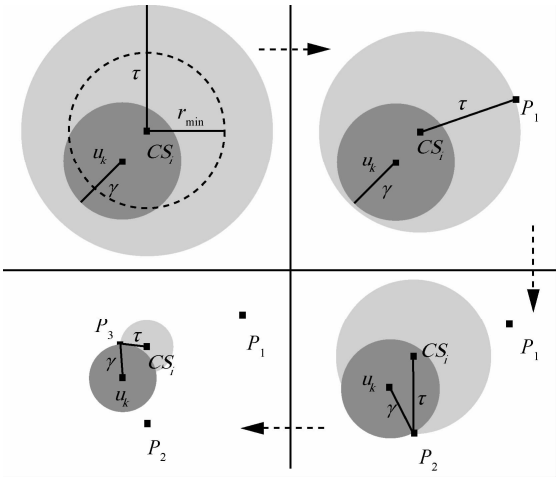


图 7　逆向增量查询

算法 3 客户端逆向增量 K 近邻查询

1) **Procedure:** 用户 u_k 在 CS_i 登记，获取 CS_i 位置 loc_{cs_i}，用户定义 k、需求空间半径 γ，$z \leftarrow \gamma$，服务器每次返回 β 个兴趣点 P_i，本算法置 $\beta = 6$，空数组 S

2) 生成小顶堆 W_K 并初始化，$W_K \leftarrow \gamma$

3) $\tau \leftarrow$ 生成满足 $\tau \geqslant dist(loc_{u_k}, loc_{cs_i}) + \gamma$ 的随机值 // 供应空间半径赋初值，覆盖需求空间

4) **if** $n_i \geqslant k$ **then**

5) 向 LBS 服务器发送 DNN 查询请求内容

6) **while** $dist(loc_{u_k}, loc_{cs_i}) < \tau + \gamma$ **do** // 供应空间与需求空间位置关系未相离

7) $S \leftarrow$ 从 LBS 服务器获取 β 个兴趣点 P_i

8) $\tau \leftarrow$ 获取 β 个兴趣点中与 CS_i 最小距离

9) **for** 每个 $P_i \in S$ **do**

10) **if** $dist(loc_{u_k}, P_i) < z$ **then** // z 无变化确保范围内的兴趣点无遗漏

11) $W_K \leftarrow P_i$，$\gamma \leftarrow dist(loc_{u_k}, P_i)$ // 用户获取自己邻近的兴趣点

12) **return** W_K

13) **End Procedure.**

图 7 描述了算法 3 的主要过程，未处于确信区的用户由于距锚点较远可以利用 INN 方法[17]查询。算法 3 第 3 行表示用户在选取 τ 值时需确保覆盖初始需求空间，τ 越大保护效果越好，但额外开销越大。用户运行 DNN 查询仅将锚点作为查询参照，算法 3 第 5 行用户通过 CS_i 将 k 匿名后的查询内容转发给 LBS 服务器，第 6 行~第 12 行用户利用返回候选集计算出想要的结果集 W_K，并在 W_K 中按从小到大的距离获取精确的 K 个兴趣点信息，无需提供自身真实位置，从数据生成端保护了自己的位置隐私。

4.4 性能分析

4.4.1 匿名过程分析

匿名成功率。对于全局进行 Voronoi 划分后，形成了类似于以通信基站为中心的蜂窝网络，任一用户均唯一属于某个 V 区。假设全局共有均匀分布的 U 个用户，被划分成了 n 个 V 区，每个区域内用户数平均值为 U/n 个，其匿名成功率取决于此时所在 V 域内的用户数 U/n，每个区域内的用户数在均值上下波动，从全局看并没有因为区域划分而带来某个 V 区人数减少，这类似于将全体用户预先划分成了 n 个匿名组，随时可以于所在 V 区内实现 k 匿名。因此 Voronoi 划分不会带来匿名率下降的问题，相反会帮助用户迅速实现 k 匿名。

服务端的负担。假设区域 $V(CS_i)$ 内有 U/n 个用户，每个用户向 CS_i 提出一次匿名请求，则 CS_i 需要执行 U/n 次匿名操作构造相应的匿名区，而 Voronoi 预划分，使得用户随时处于某个 V 区内，CS_i 无需再为每个用户构造单独的匿名区。在 P2P 协作匿名过程中，资源受限的用户头节点用户要查找附近 k–1 个用户，至少需要通信 k–1 次才能构建匿名组，本方法将头节点功能交给 V 区的 CS_i 来进行，降低头节点负担。因此本方法相对与中心匿名方式和 P2P 匿名方式都能够降低服务端的负担。

匿名实时性。对于运动速度较慢的用户，其离开所在 V 区的运动时间比匿名及查询所需的处理时间大数个量级，因此不会对匿名区用户的实时性造成实质性影响；对于速度相对较快的用户，假定其即将进入一个半径为 500 m 的 V 区，其行驶速度取二级公路限速常用值 70 km/h，经过该区域用时约为 $t = 0.5 \times 2 / 70 \times 3\,600 \approx 51$ s，如果其从行驶后半程才发起服务请求，使用移动通信带宽为 2 Mbit/s 的 3G 网络，由 5.2 节对比实验可知平均匿名时间不足 1 s，其离开该区的时间明显大于匿名服务时间，且算法 2 第 4 行使用户不会重复向 CS 发起匿名登记请求，即将要离开 V 区的用户已被 CS 排除在匿名组之外。不存在下一时刻由于用户离开 V 区带来的匿名人数不精确的问题。因此，本方法的匿名实时性较好。

匿名安全性。在用户分布均匀情况下，如果攻击者掌握用户数量为 n 的某个 $V(CS_i)$ 内所有用户的分布情况，用户每次向 CS_i 登记匿名被唯一锁定的概率 $p(x_i) = 1/n$，则单次匿名所产生的信息熵为

$$H(q) = \sum_{i=1}^{n} p(x_i) \log \frac{1}{p(x_i)} = \sum_{i=1}^{n} \frac{1}{n} \log n = \log n \quad (1)$$

此时单次匿名信息熵最大，而对于攻击者来说不确定性升高。用户进入下一区域发起匿名时更换假名，使 m 次匿名相互独立，其联合信息熵为

$$H(q_1 \cdots q_m) = \sum_{i=1}^{n} H(q_i) = m \log n \quad (2)$$

此时，多次匿名的联合信息熵最大。从信息熵角度说用户信息释放量越少，用户隐私泄露可能性越低，在协作匿名中，用户无需提供精确位置信息，

减少了位置信息的释放，增加了攻击者的不确定性。CS 组织构成匿名组后同时转发匿名组的 k 个查询请求，增加了 LBS 服务器端锁定用户的难度。

部署成本。由于 CS 多部署在用户流量较大及密集程度较高地区，因此用户在短时间内全部离开所在 V 区为小概率事件。考虑到以此方法构造的 V 区依赖 CS 的大量部署，带来成本问题，对比当前移动通信基站普遍采用 1 000 m 左右的小距离部署方式，可见部署成本在可接受范围内，而 CS 的协作匿名功能也可作为现有移动通信基站的一个工作模块运行。同时，本方法对 CS 的资源消耗较少，实现匿名功能的软硬件成本并不高。

匿名算法复杂度。由于用户无需构造匿名区，用户登记过程的时间复杂度由收到周围 CS 广播数据量 $|B|$ 决定，为 $O(|B|)$。CS 构造一次协作匿名组的时间复杂度取决于匿名组中用户最大匿名度 $k_{\max}$，为线性阶 $O(k_{\max})$。

4.4.2　查询过程分析

查询安全。由于用户无需提供真实位置数据，实现在数据生成端保护位置隐私，攻击者唯一能够获取的位置是锚点位置，只能表明用户所在的 V 区。此时锚点位置与用户虽然较近，但是供应空间以半径缩减的方式返回兴趣点使攻击者很难确定用户所在范围。因此，采用固定锚点是安全的。

查询精确度。由于采用了逆向增量查询算法，用户能够根据 LBS 服务器返回以锚点为中心的兴趣点候选集计算出自身周围最近的 K 个兴趣点。因此查询结果是精确的。

查询负担。客户端运行的查询算法复杂度取决于兴趣点比对个数，为 $O(|P_i|)$。由于用户需要根据候选集计算出精确结果，需要付出一定的计算代价，但可以不向任何实体提供真实位置信息。本方法采用固定锚点的方式，可以促进 LBS 服务端利用缓存提高查询效率，服务端的查询负担显著下降。

5　实验

本部分主要围绕用户获取 LBS 服务过程中匿名成功率、响应时间及数据通信量 3 个指标说明该方法的实际效果。上述 3 项指标的影响因素，主要考虑用户数量 U、匿名度 k 及查找兴趣点数 $|P_i|$。实验首先在不同数据集上验证方法的性能，然后分别对匿名过程和查询过程进行比对实验。

5.1　环境配置

实验在 Windows 7 平台上利用 C++语言实现，运行环境为 3.2 GHz Intel Core i5 处理器，4 GB 内存。实验设置了真实数据集 GDS [注1]和模拟数据集 TBS [注2]2 组对比。真实数据集来自美国地名委员会提供的地理数据集，包含约 160 000 个 PoI。模拟数据集采用广泛使用的 Thomas Brinkhoff 路网移动节点数据生成器，它以德国奥尔登堡市交通路网数据为基础，用户可以自定义数据集属性，城市区域面积为 24 km×27 km 的矩形区域。用户与 CS 的通信带宽为 2 Mbit/s，用户每次从 LBS 获得上限为 128 byte 的查询数据分组，除去 40 byte 分组头，每个 PoI 需要 8 byte，则每次返回数据分组约包含 $(128-40)/8=11$ 个 PoI。表 1 为参数配置情况。

表 1　　实验默认参数配置

参数名	取值范围	默认值
用户匿名需求 k	$5\leqslant k\leqslant 30$	10
全局移动用户总数 U	$25\,000\leqslant U\leqslant 200\,000$	100 000
用户定义查找范围 γ	$500\leqslant \gamma\leqslant 2\,500$	2 000 m
LBS 单次 PoI 最大负载 β	$1\leqslant \beta\leqslant 11$	6
用户查询 PoI 个数 $\|P_i\|$	$5\leqslant \|P_i\|\leqslant 30$	10
区域内 PoI 总数 m	—	160 000

5.2　2 类数据集对比实验

实验在真实数据集 GDS 和模拟数据集 TBS 上对本方法的匿名成功率、平均响应时间及平均数据通信量随用户数量增加的变化情况进行了对比分析。其中匿名成功率是系统中匿名成功用户数与全体用户数量的百分比。响应时间包括用户匿名时间和请求获取查询结果所需时间。数据通信量为响应时间内的数据分组发送数量。

如图 8 和图 9 所示，随着区域内用户总数的不断增长，用户的匿名成功率逐步升高，维持在 80%以上，用户能够获得良好的匿名成功率。平均响应时间逐步下降，并稳定在 0.3 s 附近，这主要是由于用户数量增加使得有更多用户参与协作匿名，提高匿名成功率的同时降低了用户匿名所需的响应时间。

注1 http://geonames.usgs.gov/index.html
注2 http://iapg.jade-hs.de/personen/brinkhoff/generator/

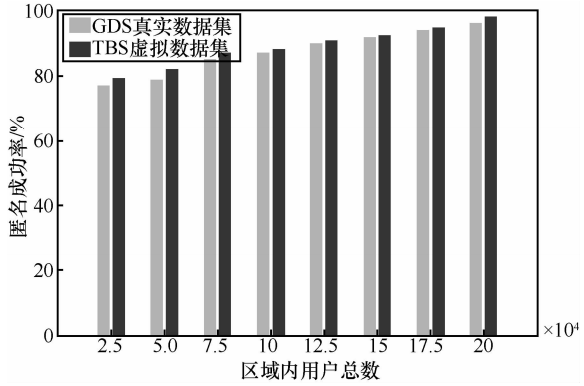


图 8 匿名成功率

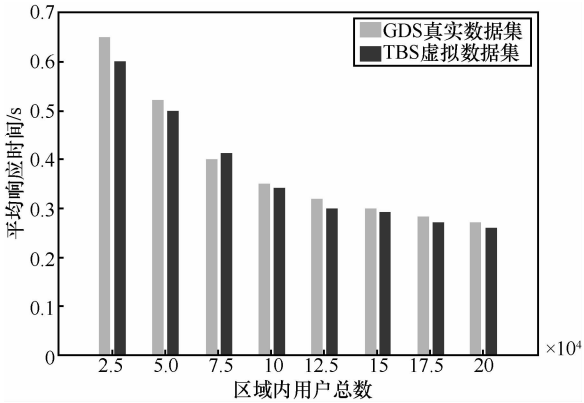


图 9 平均响应时间

如图 10 所示，用户在匿名过程中的通信量随区域内用户数 U 增加而递增，这是由于匿名需求量增多造成的，但由于 CS 无需为每个用户都构造匿名区，因此相对于其他方法通信量并没有因为用户数量的显著增长而大幅提高，与其他方法比较可以参见图 11 和图 12。由于虚拟数据集参数存在人为设置偏好，使其平均匿名通信量略低于实际数据集。如图 13 所示，由于用户数逐渐增加导致查询请求增多，DNN 查询过程的平均消息数量逐步递增。

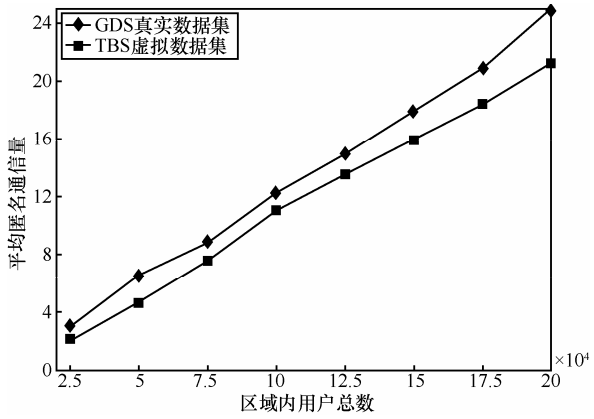


图 10 平均匿名消息数量

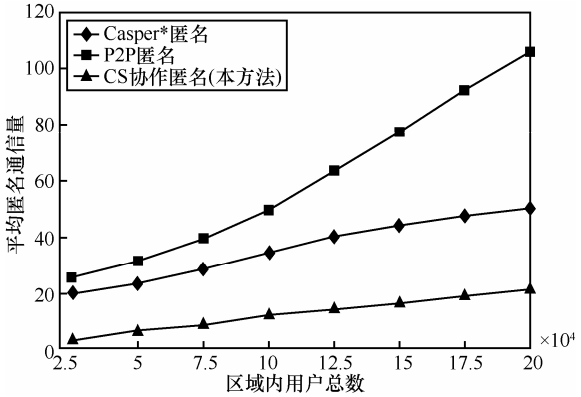


图 11 平均匿名通信量，U 增加，k 不变

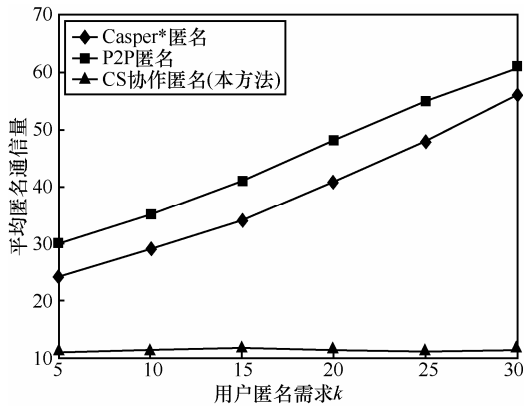


图 12 平均匿名通信量，k 增加，U 不变

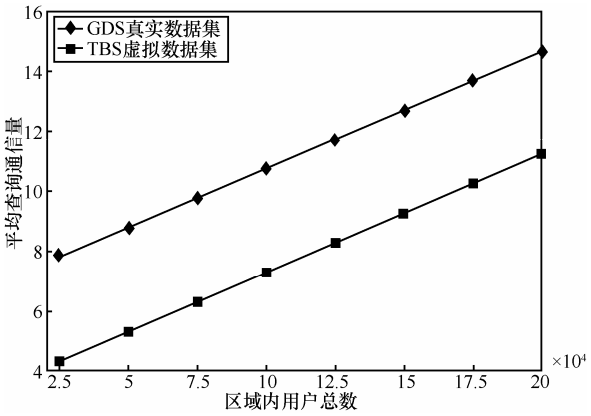


图 13 平均查询消息数量

5.3 匿名方法对比实验

以下对比实验均在模拟数据集 TBS 上进行，将本文提出的协作匿名方法与中心匿名方法 Casper*[17]、P2P 方式匿名方法[9]对比。如图 11 所示，在用户数量逐步增长过程中，P2P 匿名的通信量会显著增加，这是由于 P2P 匿名方式用户间的通信交互频繁造成的。采用中心服务器匿名方式的 Casper* 通信量也逐步增长，相对 P2P 方式较低。而本文提出的利用 CS 组织协作匿名方法由于无需计算构造

匿名区，用户之间除消息转发外，无需协作构造匿名组，因此通信量随用户数增加而小幅增长。

如图 12 所示，在用户匿名需求 k 增大而其他参数为默认值的过程中，对比方法的通信量保持类似增长变化趋势，而本方法由于采用 CS 组织协作匿名的方式，通信量受 k 的影响较小，通信量基本维持不变。需要说明的是，当 k 增大时匿名成功率会有所下降，这是由于部分用户匿名要求升高，短期内没有足够用户共同匿名造成的。

5.4　查询对比实验

在查询方面，选取 SpaceTwist[12]提出的查询算法和提出的 DNN 算法相比较。SpaceTwist 通过向 LBS 服务器提出近邻增量查询的方式逐步获取结果候选集并计算出符合自己要求精确结果。

图 14 显示其他参数为表 1 中的默认值，用户查询兴趣点从 5 增加到 30 的过程中 2 种方法通信量均增加，这是由于 2 种方法的查询方式类似。但是 DNN 查询方法却又明显的查询时间优势。如图 15 所示，由于采用固定锚点，LBS 能够迅速在缓存中找到曾经查询过的结果，并立即发送给用户，在降低 LBS 服务端查询负担的同时缩短了用户的查询时间。因此，采用固定锚点的方式相对其他非固定方式具有明显的优势。

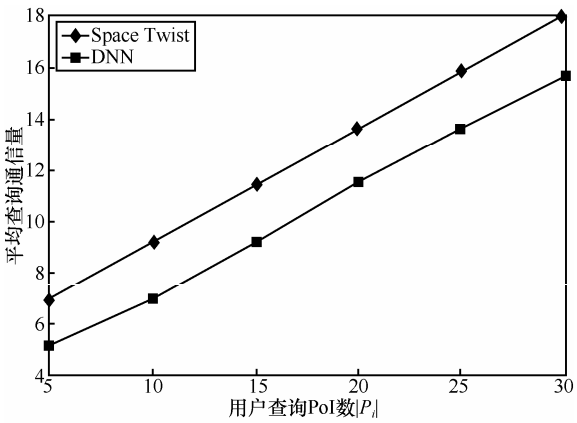


图 14　$|P_i|$ 增加时的平均匿名通信量

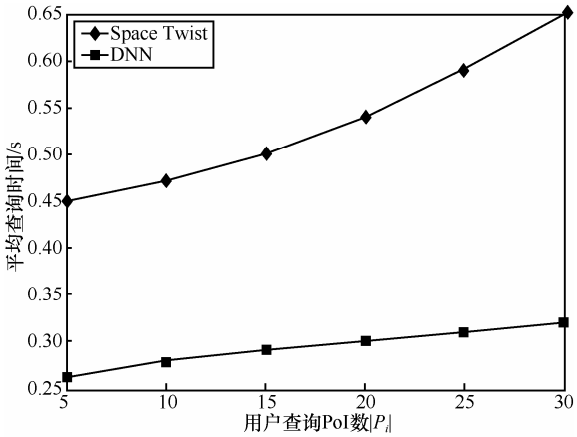


图 15　$|P_i|$ 增加时的平均查询时间

6　结束语

本文针对中心匿名结构在单独匿名过程中存在的匿名效率低、无中心 P2P 匿名结构在协作匿名过程中存在的头节点负担重等问题提出了基于 Voronoi 图预划分的协作匿名方法，该方法由 CS 负责组织用户构成协作匿名组，并将 k 个用户的查询请求同时转发给 LBS 服务器，实现 k 匿名的目标。本方法具有匿名成功率高、通信量低等优势。在此基础上提出了逆向增量查询方法 DNN，该方法无需用户提供真实位置即可获取精确的查询结果，同时解决了由于锚点位置选与用户位置过近带来的隐私泄漏问题。查询算法利用固定锚点向 LBS 发起查询请求，能够降低服务器查询负担缩短查询时间。本文在真实数据集和模拟数据集上验证了提出匿名方法和查询算法的优越性，并与其他匿名方法及查询算法进行了比对，实验表明本方法运行性能良好，运行效率较高。

同时，本方法也存在一些弱点，如基于欧式空间解决问题、用户运动状态预估还不够精细、存在查询内容关联的可能等问题。将在下一阶段的研究中进一步提出解决这些问题的方法。

参考文献：

[1] CHEEMA M A, ZHANG W, LIN X, *et al*. Efficiently processing snapshot and continuous reverse k nearest neighbors queries[J]. The VLDB Journal, 2012, 21(5): 703-728.

[2] WU D, WANG X, SUN L, *et al*. Identity privacy-based reliable routing method in VANETs[J]. Peer-to-Peer Networking and Applications, 2014, 7(3): 285-294.

[3] 薛姣, 刘向宇, 杨晓春, 王斌等. 一种面向公路网络的位置隐私保护方法[J]. 计算机学报, 2011, 34(5): 865-878.
XUE J, LIU X Y, YANG X C, *et al*. A location privacy preserving approach on road network[J]. Chinese Journal of Computers, 2011, 34(5): 865- 878.

[4] 潘晓, 郝兴, 孟小峰. 基于位置服务中的连续查询隐私保护研究[J]. 计算机研究与发展, 2010, 47(1): 121-129.
PAN X, HAO X, MENG X F. Privacy preserving towards continuous query in location-based services[J]. Journal of Computer Research and Development, 2010, 47(1): 121-129.

[5] GEDIK B, LIU L. Location privacy in mobile systems: a personalized anonymization model[A]. Distributed Computing Systems, ICDCS 2005, Proceedings of 25th IEEE International Conference[C]. 2005.

620-629.

[6] MOKBEL M F, CHOW C Y, AREF W G. The new Casper: query processing for location services without compromising privacy[A]. Proceedings of the 32nd International Conference on Very Large Data Bases[C]. VLDB Endowment, 2006. 763-774.

[7] KALNIS P, GHINITA G, MOURATIDIS K, *et al*. Preventing location-based identity inference in anonymous spatial queries[J]. Knowledge and Data Engineering, IEEE Transactions, 2007, 19(12): 1719-1733.

[8] GHINITA G, KALNIS P, SKIADOPOULOS S. PRIVE: anonymous location-based queries in distributed mobile systems[A]. Proceedings of the 16th International Conference on World Wide Web[C]. ACM, 2007. 371-380.

[9] CHOW C Y, MOKBEL M F, LIU X. A peer-to-peer spatial cloaking algorithm for anonymous location-based service[A]. Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems[C]. 2006. 171-178.

[10] HONG J I, LANDAY J A. An architecture for privacy-sensitive ubiquitous computing[A]. Proceedings of the 2nd International Conference on Mobile Systems, Applications, and Services[C]. 2004. 177-189.

[11] KIDO H, YANAGISAWA Y, SATOH T. An anonymous communication technique using dummies for location-based services[A]. Pervasive Services, Proceedings of International Conference[C]. 2005. 88-97.

[12] YIU M L, JENSEN C S, HUANG X G, LU H. SpaceTwist: managing the trade-offs among location privacy, query performance, and query accuracy in mobile services[A]. IEEE 24th International Conference on Data Engineering[C]. 2008.366-375.

[13] GONG Z, SUN G Z, XIE X. Protecting privacy in location-based services using *k*-anonymity without cloaked region[A]. Mobile Data Management (MDM), 2010 Eleventh International Conference[C]. 2010. 366-371.

[14] 黄毅, 霍峥, 孟小峰. CoPrivacy: 一种用户协作无匿名区域的位置隐私保护方法[J].计算机学报, 2011, 34(10): 1976-1985.
HANG Y, HUO Z, MENG X F. CoPrivacy: a collaborative location privacy-preserving method without cloaking region[J]. Chinese Journal of Computers, 2011, 34(10):1976-1985.

[15] GRUSTER M, GRUNWALD D. Anonymous usage of location-based services through spatial and temporal cloaking[A]. Proceedings of the 1st International Conference on Mobile Systems, Applications and Services[C]. 2003. 31-42.

[16] MOKBEL M F. Towards privacy-aware location-based database servers[A]. Data Engineering Workshops, Proceedings of 22nd International Conference[C]. 2006. 93.

[17] CHOW C Y, MOKBEL M F, AREF W G. Casper*: query processing for location services without compromising privacy[J]. ACM Transactions on Database Systems (TODS), 2009, 34(4): 24.

[18] LI T C, ZHU W T. Protecting user anonymity in location-based services with fragmented cloaking region[A]. Computer Science and Automation Engineering (CSAE), 2012 IEEE International Conference[C]. 2012. 227-231.

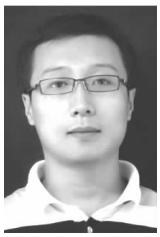
[19] 杨松涛, 马春光, 周长利. 面向 LBS 的隐私保护模型及方案[J]. 通信学报, 2014, 35(8): 116-124.
YANG S T, MA C G, ZHOU C L. LBS-oriented location privacy protection model and scheme[J]. Journal on Communications, 2014, 35(8): 116-124.

[20] SAYE R I, SETHIAN J A. Analysis and applications of the Voronoi implicit interface method[J]. Journal of Computational Physics, 2012, 231(18): 6051-6085.

[21] LIN X, LU R, LIANG X, *et al*. STAP: A social-tier-assisted packet forwarding protocol for achieving receiver-location privacy preservation in vanets[A]. INFOCOM, 2011 Proceedings IEEE[C]. 2011. 2147-2155.

作者简介：

马春光（1974-），男，黑龙江双鸭山人，哈尔滨工程大学教授、博士生导师，主要研究方向为密码学、网络与信息安全。



周长利（1985-），男，黑龙江哈尔滨人，哈尔滨工程大学博士生，主要研究方向为位置隐私保护、网络与信息安全。

杨松涛（1972-），男，黑龙江佳木斯人，哈尔滨工程大学博士生，主要研究方向为位置隐私保护。

赵蕴龙（1975-），男，黑龙江哈尔滨人，哈尔滨滨工程大学教授，主要研究方向为移动自组网、可信计算等。