ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.20, No.4, April 2009, pp.1058−1068
doi: 10.3724/SP.J.1001.2009.03428

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

LBS 中连续查询攻击算法及匿名性度量*

林 欣[1,2+], 李善平[1], 杨朝晖[1]

[1](浙江大学 计算机学院,浙江 杭州 310027)

[2](华东师范大学 信息科学与技术学院,上海 200241)

Attacking Algorithms Against Continuous Queries in LBS and Anonymity Measurement

LIN Xin[1,2+], LI Shan-Ping[1], YANG Zhao-Hui[1]

[1](College of Computer and Science, Zhejiang University, Hangzhou 310027, China)

[2](School of Information Science and Technology, East China Normal University, Shanghai 200241, China)

+ Corresponding author: E-mail: xlin@cs.zju.edu.cn

Lin X, Li SP, Yang ZH. Attacking algorithms against continuous queries in LBS and anonymity measurement. *Journal of Software*, 2009,20(4):1058−1068. http://www.jos.org.cn/1000-9825/3428.htm

Abstract: *k*-Anonymity is an important solution to protecting privacy of queries in LBS (location-based service). However, it is pointed out in literatures that *k*-anonymity cannot protect privacy of continuous queries effectively. A continuous query issuing model is proposed, which incorporates a query issuing interval model and a consecutive queries relationship model. Under this continuous query issuing model, two attacking algorithms are proposed for Clique Cloaking and Non-clique Cloaking respectively. Then this paper argues that the cardinality of anonymity-set is not a good anonymity measurement under such attack and an entropy-based anonymity measurement AD (anonymity degree) is proposed. Experimental results demonstrate that the attacking algorithms have high success rate in identifying query senders when the consecutive queries have strong relationship, and that AD is a better anonymity measurement than the cardinality of anonymity-set.

Key words: LBS (location-based service); *k*-anonymity; continuous query; attacking algorithm; anonymity measurement

摘 要: *k*-匿名机制是 LBS(location based service)中保证查询隐私性的重要手段.已有文献指出,现有的 *k*-匿名机制不能有效保护连续性查询的隐私性.提出一种连续查询发送模型,该模型融合了查询发送时间的间隔模型和连续性模型,针对此模型下的两种 *k*-匿名算法 Clique Cloaking 和 Non-clique Cloaking,分别提出了一种连续查询攻击算法.在此攻击算法下,匿名集的势不再适合作为查询匿名性的度量,因此提出一种基于熵理论的度量方式 AD(anonymity degree).实验结果表明,对连续性很强的查询,攻击算法重识别用户身份的成功率极高;AD 比匿名集的势更能反映查询的匿名性.

关键词: LBS(location-based service);*k*-匿名;连续查询;攻击算法;匿名性度量

* Supported by the National Natural Science Foundation of China under Grant Nos.60473052, 60773180 (国家自然科学基金); the Natural Science Foundation of Zhejiang Province of China under Grant No.Y106427 (浙江省自然科学基金); the Int'l Scientific Collaborate Foundation of Shanghai of China under Grant No.075107006 (上海市国际科技合作基金)

Received 2007-12-08; Accepted 2008-08-11

中图法分类号: TP309 文献标识码: A

移动通信技术和传感定位技术的迅速发展,使得基于位置信息的服务(location-based services,简称 LBS,典型的例子如 CyberGuide[1])成为未来计算环境中的重要组成部分,并无缝地融入人们的日常生活中[2,3],如场景:用户可以随时随地向 LBS 发出查询“找出离我现在的位置最近的诊所”.但是,LBS 在带给用户方便的同时,也带来了隐私泄漏的危险.在以上的场景中,用户必须将现在所处的位置发送给 LBS,而 LBS 有可能被恶意攻击者所控制,攻击者根据收集到的用户位置信息和查询内容可以挖掘出用户的私人信息(如生活习惯、健康状况等),即使系统用假名技术和加密技术掩盖了用户的真实身份,攻击者仍然可以根据位置信息的排他性重识别(re-identify)出用户身份(如,与某人的私人办公室相关联的很可能是办公室的主人).

针对 LBS 带来的隐私问题,众多学者提出借鉴数据库中的 k-匿名(k-anonymity)机制来解决[4−7].所谓 LBS 中的 k-匿名机制,是指查询中的位置信息必须包含 k 个不同的用户(包括发送者在内),这样,LBS 接收到的是这 k 个用户和查询组合成了一个快照(snapshot),恶意攻击者无法从这 k 个用户中分辨出真正的发送者,这 k 个用户组成的集合称为匿名集(anonymity-set),匿名集的势越大,快照的隐私性越好.具体的做法如图 1 所示.首先,每个基于位置信息的查询都由客户端加密发送给一个绝对可信的匿名器(anonymizer),它负责降低用户位置信息在时空二维的精度,直到位置信息包含其他 k–1 个用户为止,此过程称为 cloaking(模糊化).如将某查询“我在一号楼 605 室,找出我周围最近的饭店”模糊化为快照“我在一号楼,找出我周围最近的饭店”.匿名器将快照发送给 LBS,LBS 根据低精度的位置信息返回相应的结果集合给匿名器,由匿名器根据用户实际的位置过滤出真正的查询结果返回给用户.

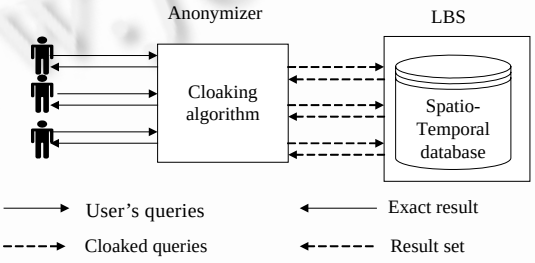


Fig.1 Architecture of k-anonymity in LBS

图 1 LBS 中 k-匿名机制实现架构

现有的 LBS 中 k-匿名机制可以分为两大类:Clique Cloaking 和 Non-Clique Cloaking,它们的不同点在于,Clique Cloaking 要求快照内的 k 个用户都使用快照提供的地理位置区域作为查询的模糊化位置信息[6,8],而且快照内的每个用户都发送过快照内的一个查询,而 Non-Clique Cloaking 没有这两项要求[3,9].上述表明,这两类算法能够较好地提高单快照查询(snapshot query)的隐私性.然而,除了单快照查询以外,还有一大类基于位置信息的查询属于连续查询(continuous query)[10−13],即由于查询的结果会随着用户的位置改变而动态变化,用户必须在一段时间内连续发出相同的查询以获得最新的查询结果,如某位正在驾驶的司机定期地向 LBS 查询离他最近的餐馆.现有的 Cloaking 算法并不能很好地保护连续查询中的用户隐私[14],这是因为所有的 Cloaking 算法都是基于这样的假设:对于恶意攻击者来说,匿名集中的 k 个用户发送查询的概率是相等的,即都等于 1/k.然而当恶意攻击者收集到一组连续快照时,可以将它们联系起来,估算出 k 个用户发送查询的概率.此时,每个用户的概率不再相等,攻击者挑选出概率最高的用户,很可能就是真正的查询发送者,隐私被破坏.本文正是基于以上思想展开,主要贡献可以归纳为 3 点:

(1) 为了对用户发送连续查询的规律加以建模,提出一种连续查询发送模型,该模型融合了查询发送时间的间隔模型和连续性模型;

(2) 面向经过 Clique Cloaking 和 Non-Clique Cloaking 保护的连续查询分别提出一种攻击算法,实验结果

表明,对连续性很强的查询,攻击算法重识别用户身份的成功率极高;

(3) 由于匿名集中的用户发送消息的概率不均等,因此快照中匿名集的势(即 k 的大小)不再能正确地反映算法的匿名性.本文提出一种基于熵理论的匿名度量 AD(anonymity degree)取代 Cloaking 算法中匿名集的势,实验结果表明,前者较后者更能体现查询的匿名性.

本文第 1 节提出一系列系统模型假设.第 2 节主要介绍面向 Clique Cloaking 和 Non-Clique Cloaking 的连续查询攻击算法.第 3 节提出一种基于熵理论的匿名度量.第 4 节介绍相关模拟实验.

1 系统模型假设

为了简化问题,我们提出一系列系统模型假设.其中,整个系统架构仍然沿用现有 k-匿名机制中的架构(如图 1 所示),分为用户、匿名器和 LBS 这 3 个部分.我们假设所有用户发送消息遵循第 1.1 节中介绍的连续查询发送模型.

1.1 连续查询发送模型

连续查询发送模型分为两个部分:查询发送时间间隔模型和连续性模型.现有的连续查询研究机构主要关注如何减少连续查询的通信数量[10−13],对连续查询都简单地假设为周期性定时发送的.本文采用一种更加通用的时间间隔模型:设现在时刻距离用户最近一次发送查询的时间间隔为 t,该时刻再次发送查询的概率密度为 $\varphi(t)$,t 时段内再次发送查询的概率分布函数是 $\phi(t)$,根据概率密度与概率分布的关系,可以得到 $\phi(t)=\int_0^t \varphi(x)\mathrm{d}x$. $\phi(t)$可以是任意的概率分布函数,因此,该模型也涵盖了定时发送的连续查询,即取概率分布函数:

$$\phi(t)=\begin{cases}1, & t\ge T\\ 0, & t<T\end{cases},$$

其中,T 为定时发送的周期.

用户查询的连续性与查询的内容有关,如查询 q_1:“我附近的路况”在整个行车过程中都是用户关心的,会不断重复发出相同的查询,而查询 q_2:“我附近的餐馆”只是在用餐前一段时间发送几次,我们认为 q_1 比 q_2 的连续性更强.连续性模型采用用户发送查询的内容 q 和前一次查询内容 q'相同的概率 ρ(0≤ρ<1)来描述查询的连续性,ρ 越大说明查询的连续性越强.作为特例,ρ=0 可以描述不具备连续性的查询.同时,我们假设用户可能发送的查询种类总数为 N,用户发送的查询和前一次查询不相同,则发送其他任何一个查询的概率是均等的,所以可以得出用户要发出和 q'不同的查询 q 的概率为(1−ρ)/(N−1).我们将函数 $\delta(q',q)=\begin{cases}\rho, & q'=q\\ (1-\rho)/(N-1), & q'\ne q\end{cases}$ 称为连续查询关联函数.

根据以上假设的模型,可以得到一组条件概率公式:令事件 A 为某用户在 0 时刻发送了一个查询 q_0,事件 B 为该用户在(t,t+Δt)时段发送第 2 个查询,事件 C 为该用户在 0 时刻到 t 时刻都没有发出任何查询,事件 D 为该用户在(t,t+Δt)时段发出第 2 个查询且查询为 q_t,则

$$P(B|A)=\phi(t+\Delta t)-\phi(t),$$

$$P(C|A)=1-\phi(t),$$

$$P(D\mid A)=P(B\mid A)\delta(q_0,q_t)=(\phi(t+\Delta t)-\phi(t))\cdot\begin{cases}\rho, & q_0=q_t\\ (1-\rho)/(N-1), & q_0\ne q_t\end{cases}.$$

1.2 匿名器假设

匿名器以 TUnit 为周期收集用户发出的查询,执行 Cloaking 算法,在每个周期末将该周期 Cloaking 得到的快照发送给 LBS,将无法成功 Cloaking 的查询(无法在指定的区域内找到足够多的用户作为掩护)丢弃,一个 TUnit 之内,任何用户最多只发出一个查询.匿名器只使用两种 Cloaking 算法中的一种.Clique Cloaking 算法产生的快照包含了 k 个用户,也包含了 k 个查询,表示为四元组〈U,t,r,Q〉,其中,U={u_1−u_k}表示匿名器为用户赋予的代号的集合,可以是用户真实的 id,也可以是假名,Q={q_1−q_k}表示此 k 个用户发出的 k 个查询,r 表示 k 个用户所共

享的模糊化区域,t 表示匿名器发出此快照的周期序号.这样,LBS 只知道快照内 k 个用户中的每个用户发送且只发送了此 k 个查询中的一个,但无法将某一个用户 $u_i(i=1,2,\ldots,k)$和任何一个查询 $q_j(j=1,2,\ldots,k)$对应起来.Non-Clique Cloaking 算法产生的快照记为四元组$\langle U,t,r,q\rangle$,表示查询 q 是由 U 中的一个用户发送的,其中,$U=\{u_1-u_k\}$表示匿名器为用户赋予的代号的集合,t 和 r 的意义与 Clique Cloaking 中相应的元素相同.由此,匿名器通过 Cloaking 算法,消除了用户 id 和查询之间的关联,实现了匿名性.

1.3 攻击者假设

恶意攻击者的目标是要将查询和用户 id 联系起来.假设恶意攻击者无法破解用户和匿名器之间的传输密钥,但可以截取由匿名器发出的所有快照并明文解析快照的内容,还能够根据长期的历史经验统计得到以下几种数据:查询种类总数 N、每一种查询的连续性参数ρ和用户发送查询的时间间隔参数λ.

2 连续查询攻击算法

2.1 面向Clique Cloaking的连续查询攻击算法

恶意攻击者将截获的所有快照按照匿名器发出的时间顺序排成一组列表,记作 CMList.对于指定的一个快照 cm:$\langle U,t,r,Q\rangle$,$cm.U$,$cm.t$,$cm.r$ 和 $cm.Q$ 分别代表快照 cm 中对应的元素.我们把“事件 u_i 在快照 cm 中发送了 q_j”记为 $u_i \xrightarrow{cm} q_j$,连续查询攻击算法的目标是通过计算概率 $P(u_i \xrightarrow{cm} q_j)$,找出最有可能发送 q_j 的用户(令 $P(u_i \xrightarrow{cm} q_j)$ 最大的 u_i),其中,$i,j=1,2,\ldots,k$.根据第 1 节中的介绍,攻击者可以由 cm 获知 u_1-u_k 中的每一个用户都发送且只发送了 q_1-q_k 中的一个查询,因此得到两个约束条件:

$$\sum_{i=1}^{k} P(u_i \xrightarrow{cm} q_j)=1 \tag{1}$$

$$\sum_{j=1}^{k} P(u_i \xrightarrow{cm} q_j)=1 \tag{2}$$

公式(1)表示所有用户发送查询 q_j 的概率和为 1,公式(2)表示用户 u_i 发送所有查询的概率总和为 1.

攻击算法分为两步:第 1 步,在未知约束条件(1)和约束条件(2)的情况下,用户 u_i 发送查询 q_j 的概率,我们记作 $P(u_i\rightarrow q_j)$;第 2 步,根据 $P(u_i\rightarrow q_j)$结合约束条件(1)和约束条件(2)求出 $P(u_i \xrightarrow{cm} q_j)$.

第 1 步,计算 $P(u_i\rightarrow q_j)$:首先,找出每个用户 u_i 的前继快照 PCM_{u_i}.

定义 1. 用户 u_i 对于指定快照 cm 的前继快照是指在 cm 之前,u_i 最后一次真正发送查询的快照.

由第 1 节介绍可知,Clique Cloaking 的特点是要求快照 $cm.U$ 内的每个用户都发了 $cm.Q$ 中的一个查询,因此,只要在 CMList 中从 cm 开始向前搜寻,第 1 个出现 u_i 的快照就是 u_i 对于指定快照 cm 的前继快照.此过程会出现两种结果:1) 找到 PCM_{u_i};2) 找不到 PCM_{u_i}.对于情况 2),可以认为用户 u_i 发送 cm 中任何一个查询的概率均等,即 $P(u_1\rightarrow q_j)=P(u_2\rightarrow q_j)=\ldots=P(u_i\rightarrow q_j)=\ldots=P(u_k\rightarrow q_j)$,下面着重介绍求情况 1)下的 $P(u_i\rightarrow q_j)$:

记查询 PQ_i 和 cm 接收到的时间间隔为$\Delta t=cm.t-PCM_{u_i}.t$,得到条件概率:

$$P(u_i\rightarrow q_j \mid u_i \xrightarrow{PCM_{u_i}} q'_x)=(\phi(\Delta t)-\phi(\Delta t-1))\delta(q'_x,q_j) \tag{3}$$

由于可以将 $u_i \xrightarrow{PCM_{u_i}} q'_x$ ($x=1,2,\ldots,k$)当作相互独立的事件,得到:

$$P(u_i\rightarrow q_j)=\sum_{x=1}^{k}(P(u_i\rightarrow q_j \mid u_i \xrightarrow{PCM_{u_i}} q'_x)P(u_i \xrightarrow{PCM_{u_i}} q'_x)) \tag{4}$$

为了简化算法复杂度,假设 u_i 在查询 PCM_{u_i} 中发送 $q'_1-q'_k$ 中任何一个查询的概率均等,即

$$P(u_i \xrightarrow{PCM_{u_i}} q'_x)=1/k\ (x=1,2,\ldots,k).$$

第 2 步,求 $P(u_i \xrightarrow{cm} q_j)$.

定义 2. $A(k)$表示自然数 1–k 的所有排列的集合,a 为 $A(k)$中任意一种排列,a_l 为排列 a 中第 l 个元素.

在得到 $P(u_i\rightarrow q_j)$之后,结合约束条件(1)和约束条件(2),则有公式(5):

$$P(u_i \xrightarrow{cm} q_j)=\frac{\sum_{\forall a=(a_1,a_2,\ldots,a_k)\in A(k)\ 且 a_i=j}\ \prod_{l=1,2,\ldots,k} P(u_l \to q_{a_l})}{\sum_{\forall a=(a_1,a_2,\ldots,a_k)\in A(k)}\ \prod_{l=1,2,\ldots,k} P(u_l \to q_{a_l})} \tag{5}$$

说明:若设“u_1–u_k分别发送q_1–q_k之一”为事件A,“u_i发送q_j”为事件B, $P(u_i \xrightarrow{cm} q_j)$本意上是指在事件$A$发生的条件下,发生事件$B$的条件概率,即$P(B|A)$.对于恶意节点来说,$cm$中用户和查询的对应关系有$P_k^k=k!$种,这些对应关系发生的总概率也就是事件$A$发生的概率$P(A)=\sum_{\forall a=(a_1,a_2,\ldots,a_k)\in A(k)}\ \prod_{l=1,2,\ldots,k} P(u_l \to q_{a_l})$,即公式(5)等号右边的分母,而公式(5)等号右边的分子因为加上约束条件“$a_l=j$”,因而代表事件A和事件B同时发生的概率,即$P(AB)$.

2.2 面向Non-Clique Cloaking的连续查询攻击算法

2.2.1 符号约定

为了方便叙述,本节约定以下符号:

- 对于快照$cm=\langle U,t,q,r\rangle$,$cm.U$,$cm.t$,$cm.q$和$cm.r$分别代表快照$cm$中对应的元素.
- CM_T表示匿名器在周期T发出的所有快照集合,即$CM_T=\{cm|cm.t=T\}$.
- CM_u表示所有包含用户u的快照集合,即$CM_u=\{cm|u\in cm.U\}$.
- $CM_{u,T}$表示匿名器发出时间等于T且包含用户u的快照集合,即$CM_{u,T}=\{cm|cm.t=T\wedge u\in cm.U\}$.
- 记所有查询的集合为Q,根据第1节假设,$|Q|=N$.周期T内与用户u相关的查询集合记为$Q_{u,T}$,即$Q_{u,T}=\{q|q=cm'.q,cm'\in CM_{u,T}\}$.
- 定义快照约束为:对于每个快照cm,有且只有$cm.U$中的一个用户发送了$cm.q$.
- 定义周期约束为:在周期T,用户u只能发送集合$Q_{u,T}$内的一个查询或者不发查询.
- 记在约束快照约束和周期约束都不存在的情况下,用户u在周期t发送查询q的概率为$V(u,q,t)$,记u在t时刻不发送查询的概率为$V(u,null,t)$.根据第1节假设,某一用户在一个周期内至多只发送一个查询,对于指定的用户u_i和周期t有
$$\sum_{q\in Q} V(u,q,t)+V(u,null,t)=1.$$
- 记在仅有周期约束而没有快照约束条件下,用户u周期t内发送查询q的条件概率为$W(u,q,t)$,用户u在t时刻没有发送查询的概率记为$W(u,null,t)$.周期约束可以理解为用户u发送$Q_{u,T}$内的查询的概率之和加上用户u不发送查询的概率为1,对于给定的u和t,有公式(6):
$$\sum_{q\in Q_{u,t}} W(u,q,t)+W(u,null,t)=1 \tag{6}$$

$W(u,q,t)$和$V(u,q,t)$的区别在于,对于任意$q\notin Q_{u,t}$,$W(u,q,t)=0$而$V(u,q,t)$可以不等于0.

- 记在快照约束和周期约束同时成立的条件下,用户u在快照cm内发送查询的条件概率为$P(u\xrightarrow{cm}cm.q)$,快照约束可以理解为:对于给定的cm,快照内所有用户的发送查询的概率总和为1,所以有公式(7):
$$\sum_{u_i\in cm.U} P(u\xrightarrow{cm}cm.q)=1 \tag{7}$$
- 记$PQ(u,q,j,t)$为事件“用户u在周期t–j(j>0)发送了查询q,且在t–j+1到t–1周期都不曾发送任何查询”,其概率为$P(PQ(u,q,j,t))$.
- 记$PT(u,j,t)=\sum_{q\in Q_{u,t-j}} P(PQ(u,q,j,t))$,表示用户$u$在$t$–$j$周期发送了查询,且在$t$–$j$+1到$t$–1周期都不曾发送任何查询的概率,规定对于所有用户u和周期t,都有$PT(u,0,t)=0$.

2.2.2 算法介绍

针对给定的快照cm,面向Non-Clique Cloaking的连续查询攻击算法的目标是通过计算任意一个$u_i\in cm.U$

的 $P(u_i \xrightarrow{cm} cm.q)$,找出最有可能发送 $cm.q$ 的用户(令 $P(u_i \xrightarrow{cm} cm.q)$ 最大的 u_i).算法分为 3 步:

第 1 步,计算 $V(u_i,cm.q,cm.t)$.

事件 $PQ(u,q,j,t)$可以分解为两个事件:“用户 u 且在 $t–j+1$ 到 $t–1$ 周期都不曾发送任何查询”(定义为事件 $u \xrightarrow{[t-j+1,t-1]} null$)和事件“u 在周期 $t–j$ 发送了 q”(定义为事件 $u \xrightarrow{t-j} q$).事件 $u \xrightarrow{[t-j+1,t-1]} null$ 发生的概率是 $1-\sum_{x=0}^{j-1} PT(u,x,t)$,根据 $W(u,q,t)$的定义,在事件 $u \xrightarrow{[t-j+1,t-1]} null$ 发生的条件下,事件 $u \xrightarrow{t-j} q$ 发生的概率为 $W(u,q,t–j)$.所以有

$$P(PQ(u,q,j,t))=\left(1-\sum_{x=0}^{j-1} PT(u,x,t)\right)W(u,q,t-j) \tag{8}$$

在以上事件 $u \xrightarrow{t-j} q$ 发生的条件下,用户 u_i 的下一个查询可能发生在 2 个时间段:① 周期 $t–j+1$ 到周期 $t–1$ (即事件 $u \xrightarrow{[t-j+1,t-1]} q_{Any}$);② 周期 t 到无穷远(即事件 $u \xrightarrow{[t,\infty)} q_{Any}$).在事件 $PQ(u,q,j,t)$发生的条件下,其包含的事件 $u \xrightarrow{[t-j+1,t-1]} null$ 排斥了事件 $u \xrightarrow{[t-j+1,t-1]} q_{Any}$ 发生的可能,因此,事件 $PQ(u,q,j,t)$等价于发生事件 $u \xrightarrow{t-j} q$ 且发生事件 $u \xrightarrow{[t,\infty)} q_{Any}$.定义事件“用户在周期 t 发送了查询”为事件 $u \xrightarrow{t} q_{Any}$,根据连续查询发送模型定义,

$$P(u \xrightarrow{t} q_{Any} \mid u \xrightarrow{t-j} q)=\phi(j\bullet TUnit)-\phi((j-1)TUnit),$$

$$P(u \xrightarrow{[t,\infty)} q_{Any} \mid u \xrightarrow{t-j} q)=1-\phi((j-1)TUnit),$$

$$\begin{aligned}P(u \xrightarrow{t} q_{Any} \mid PQ(u,q,j,t)) &= P(u \xrightarrow{t} q_{Any} \mid ((u \xrightarrow{t-j} q)\wedge(u \xrightarrow{[t,\infty)} q_{Any})))\\ &=\frac{P(u \xrightarrow{t} q_{Any} \mid u \xrightarrow{t-j} q)}{P(u \xrightarrow{[t,\infty)} q_{Any} \mid u \xrightarrow{t-j} q)}\\ &=\frac{\phi(j\bullet TUnit)-\phi((j-1)TUnit)}{1-\phi((j-1)TUnit)},\end{aligned}$$

$$\begin{aligned}P(u \xrightarrow{t} null \mid PQ(u,q,j,t)) &= P(u \xrightarrow{t} null \mid ((u \xrightarrow{t-j} q)\wedge(u \xrightarrow{[t,\infty)} q_{Any})))\\ &=\frac{P(u \xrightarrow{[t,\infty)} q_{Any} \mid u \xrightarrow{t-j} q)-P(u \xrightarrow{t} q_{Any} \mid u \xrightarrow{t-j} q)}{P(u \xrightarrow{[t,\infty)} q_{Any} \mid u \xrightarrow{t-j} q)}\\ &=\frac{1-\phi(j\bullet TUnit)}{1-\phi((j-1)TUnit)}.\end{aligned}$$

由上式得到:

$$V(u,q,t)=\sum_{j=1}^{t}\sum_{q'\in Q_{u,t-j}}\left(P(PQ(u,q',j,t))\frac{\phi(j\bullet TUnit)-\phi((j-1)TUnit)}{1-\phi((j-1)TUnit)}\delta(q',q)\right) \tag{9}$$

$$V(u,null,t)=\sum_{j=1}^{t}\sum_{q'\in Q_{u,t-j}}\left(P(PQ(u,q',j,t))\frac{1-\phi(j\bullet TUnit)}{1-\phi((j-1)TUnit)}\right) \tag{10}$$

在公式(9)、公式(10)中,当 t 不断增大时,q'数量也不断增大.为了控制计算开销,我们采用公式(11)、公式(12)来分别近似取代公式(9)、公式(10).

$$\begin{aligned}V(u,q,t)=&\sum_{j=1}^{M}\sum_{q'\in Q_{u,t-j}}\left(P(PQ(u,q',j,t))\frac{\phi(j\bullet TUnit)-\phi((j-1)TUnit)}{1-\phi((j-1)TUnit)}\delta(q',q)\right)+\\ &\left(1-\sum_{j=1}^{M}\sum_{q'\in Q_{u,t-j}}P(PQ(u,q',j,t))\right)\frac{\phi((M\bullet TUnit)-\phi((M-1)TUnit)}{1-\phi((M-1)TUnit)}\frac{1}{N}\end{aligned} \tag{11}$$

$$V(u,null,t)=\sum_{j=1}^{M}\sum_{q'\in Q_{u,t-j}}\left(P(PQ(u,q',j,t))\frac{1-\phi(j\bullet TUnit)}{1-\phi((j-1)TUnit)}\right)+\left(1-\sum_{j=1}^{M}\sum_{q'\in Q_{u,t-j}}P(PQ(u,q',j,t))\right)\frac{1-\phi((M\bullet TUnit)}{1-\phi((M-1)TUnit)} \tag{12}$$

在公式(11)、公式(12)中,只计算 $Q_{u,t-M}$ 到 $Q_{u,t-1}$ 中 q'的 $P(PQ(u,q',j,t))$,对于 $t-M$ 周期前的情况,我们近似看作 u 在 $t-M-1$ 周期发送了最后一个查询,此查询为 Q 中任意查询且概率均等,当参数 M 取足够大时,近似计算的误差很小.

第 2 步,计算 $W(u,q,t)$:根据 $W(u,q,t)$定义结合周期约束:

$$W(u,q,t)=\frac{V(u,q,t)}{\sum_{q'\in Q_{ui,cm.t}}V(u,q',t)+V(u,null,t)} \tag{13}$$

公式(13)中的分母是代表周期约束出现的概率总和.攻击算法将得到的 $W(u,q,t)$保存 M 时刻,为后面周期计算 $V(u,q,t)$所用.

第 3 步,计算 $P(u_i\xrightarrow{cm}cm.q)$,根据 $P(u_i\xrightarrow{cm}cm.q)$ 定义结合快照约束:

$$P(u_i\xrightarrow{cm}cm.q)=\frac{W(u_i,cm.q,cm.t)\prod_{u'\in cm.U\wedge u'\neq ui}W(u',cm.q,cm.t)}{\sum_{u\in cm.U}\left(W(u,cm.q,cm.t)\prod_{u'\in cm.U\wedge u'\neq u}W(u',cm.q,cm.t)\right)} \tag{14}$$

公式(14)中,分子表示快照 cm 内 u_i 发送查询而其他用户不发送查询的概率,分母表示快照约束出现的概率.

3 基于熵理论的匿名性度量

Cloaking 算法的根本目标是要让恶意攻击者无法将查询和真正发送者的 id 联系起来.因此,衡量算法匿名性最直接的度量指标是看对于查询的被识别率(identified rate,简称 IR),即,被恶意攻击者采用攻击算法成功识别出真正发送者的查询数量除以所有查询的数量.IR 需要统计得到,对任意快照 cm 中的查询 q,需要不经过统计而直接给出一个匿名性度量 $g(q)$,满足:$g(q)$值较大的一组查询相应的 IR 较小,反之亦然(统计意义上的).显然,可用的 $g(q)$不止一个,我们认为好的 $g(q)$应该满足:如果对于 $g(q)$值为 g_0 的查询集合 $Q=\{q|g(q)=g_0\}$,存在一个理论上的 IR 下限 $IR_{\min}(g_0)$,那么,实际的识别率 $IR(Q)$与 $IR_{\min}(g_0)$越接近越好.

现有的 Cloaking 算法都采用匿名集的势(k-匿名中的 k)作为匿名性度量,它的确满足 k 越大 IR 越低这一特性.对于具有度量值 k 的查询集合,其理论 IR 的下限为 $1/k$(即一个快照中,k 个用户是查询的发送者的概率均等).但是在实验中我们发现,实际的 IR 偏离这个理论下限较远.因此,我们借鉴信息论中描述不确定性的熵理论,提出一种新的查询匿名性度量来取代匿名集的势.

对于一个给定的查询 q,假设经过 Cloaking 保护之后,产生了 q 的匿名集 $cm.U=\{u_1,u_2,\ldots,u_k\}$,采用连续查询攻击算法求得用户 u_i 发送 q 的概率为 $P(u_i\xrightarrow{cm}q)$.根据香农公式[15],信息"查询 q 的真正发送者"的熵为

$$H(cm,q)=-\sum_{u_i\in cm.U}P(u_i\xrightarrow{cm}q)\log_2 P(u_i\xrightarrow{cm}q)\ .$$

利用指数函数的单调性,我们提出查询匿名性度量 AD(anonymity degree):

$$AD(q)=2^{H(q)}.$$

$AD(q)$符合以下两个性质:

① $AD(q)$越大,$H(q)$也越大,查询匿名性越高;

② 当 $AD(q)$为整数时,对于任意给定的 q, $P(u_i\xrightarrow{cm}q)$ $(u_i\in cm.U)$最大值大于等于 $1/AD(q)$,即快照 cm 中查询 q 的理论 IR 下限为 $1/AD(q)$(附录中将给出证明).

由下面的实验数据可以看出,在使用连续查询攻击算法重识别连续性很强的查询时,具有匿名度量 AD 的查询集的实际 IR 与理论下限比较接近.

4 模拟实验

4.1 实验配置及系统参数

实验采用 Network-Based Generator of Moving Objects[16]模拟器来模拟用户在市区交通网络内的运动轨迹,并采用德国奥登伯格市的市区交通网络图作为上述模拟程序的输入(如图 2 所示).用户总数为 10 000 个,用户的速度取 5~50km/h,平均速度为 15km/h.每个用户遵循λ=0.5 的连续查询发送模型,查询的种类数 N=20000.匿名器周期 $TUnit$=30s,平均每个用户出现的时间为 200$TUnit$,搜索匿名集的范围是边长为 2km 的正方形.分别采用文献[4,6]中的算法作为 Clique Cloaking 和 Non-Clique Cloaking 的算法原型.

本实验的主要内容是在采用不同的连续查询时间间隔模型、连续性参数ρ和 k 的条件下,使用连续查询攻击算法对 k-匿名保护的查询进行攻击,考察查询的实际被识别率 IR(即被恶意攻击者采用攻击算法成功识别出真正发送者的查询数量除以所有查询的数量,IR 越高,查询的匿名性被破坏的程度越严重).本实验中连续查询时间间隔模型分为两种,并分别做两组实验:1) 定时发送查询,即同一个用户发送的相邻两次查询的时间间隔总是相等,这是基于位置的连续查询中最常用也是最简单的时间间隔模型.实验中用户采用的周期取值范围是 1~3 个 $Tunit$;2) 查询间隔时间呈指数分布,即$\varphi(t)=\lambda e^{-\lambda t}$,该分布比较符合基于位置信息的连续查询场景,因为随着距离最后一次发送查询的时长的增加,查询结果的可用性(或称寿命)在减少,用户发送下一个查询的可能性也有所增加.

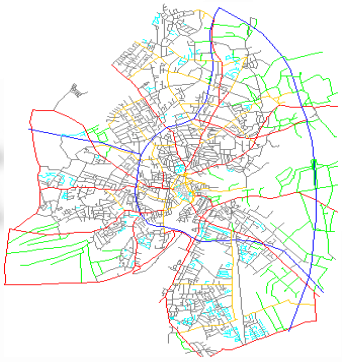


Fig.2 City map of Oldenburg

图 2 奥登伯格市区图

实验结果如图 3 和图 4 所示,对连续性参数(continuity argument)ρ分别取 0,0.1,0.3,0.5,0.7 和 0.9 进行实验,每组实验中匿名集的势 k 分别取 3,5 和 7,分别采用面向 Clique Cloaking(图中标志为 cc)和面向 Non-Clique Cloaking(图中标志为 ncc)连续查询攻击算法来重识别 200 000 个左右查询的发送者.从图 3 中可以看出,在ρ=0,即查询无连续性时,查询被识别率接近 20%(即 1/k),随着ρ的不断增大,查询的被识别率显著提高,当ρ取 0.9 时,查询的被识别率超过了 75%,匿名性被严重破坏.因此,可以考虑采用降低ρ的方式来抵抗连续查询攻击算法.另外,从图中可见,cc 栏在ρ等于 0.1~0.7 之间,被识别率比 ncc 栏要略高出一点,这是因为在攻击算法中,Clique Cloaking 的前继查询相对于 Non-Clique Cloaking 较为确定.随着 k 的增大,实际 IR 确实能够降低,但是在后面第 4.3 节的实验中将看到,用 k 对查询进行归类,其实际 IR 和理论 IR 的下限偏差较远.图 4 描述了当采用定时发送查询时的实验结果,可以看出,图 4 遵循与图 3 同样的规律,但图 4 的每一组实验得到的 IR 均略高于图 3 中相应的结果,这是因为使用指数分布作为时间间隔模型比定时查询更加具有不确定性.

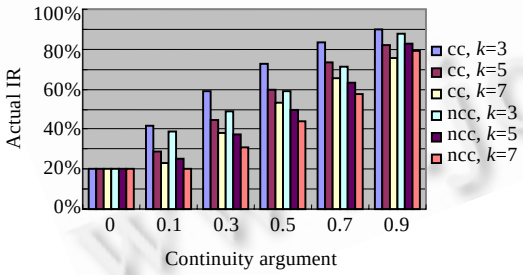


Fig.3 Effect of ρ to IR when query intervals follow exponential distribution

图 3 查询间隔时间呈指数分布时,连续性参数ρ对查询被识别率的影响

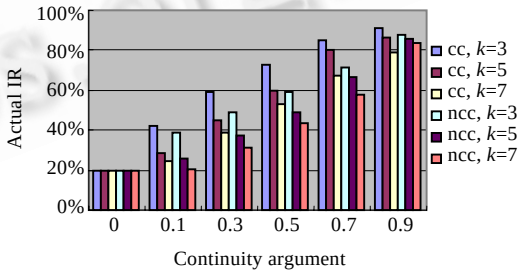


Fig.4 Effect of ρ to IR when query intervals are periodic

图 4 定时发送查询时,连续性参数ρ对查询被识别率的影响

4.2 AD与匿名集的势(*k*)的比较

第 3 节介绍了 AD 的概念,并得到性质:在 AD 取整数时,查询的理论 IR 下限为 1/*AD*.而在现有的 Cloaking 算法中,采用 *k* 作为匿名性度量,认为对于大样本统计得到查询的 IR 应该和 1/*k* 相吻合.因此可以看出,当 AD 和 *k* 取值为相等的整数时,两者理论 IR 下限相等.本实验分别取ρ=0.9 和ρ=0.5 作为连续性参数,向匿名器发出 1 000 000 个查询,匿名器从 1~7 中随机取一个整数,作为 *k* 对查询实行 Clique Cloaking 和 Non-Clique Cloaking 保护.恶意攻击者采用相应的连续查询攻击算法进行重识别,并将每个查询按照获得的 AD 和 *k* 进行分类,统计每一类的实际 IR.由于 AD 取值范围是一个连续的区间,无法大样本地收集到 AD 取整数的查询,我们将 AD 落在区间(n–0.05,n+0.05)近似作为 *AD*=*n*(n 为整数)的标准.图 5 和图 6 分别描述了当ρ=0.9 时,采用面向 Clique Cloaking 和 Non-Clique Cloaking 的连续查询攻击算法下,AD 和 *k* 分别取 1~7 之间的整数的比较.同时,在图上也增加了“理论 IR”一栏,表示理论 IR 下限作为比较的依据.从图 5(Clique Cloaking)中可以看出,使用 AD 分类的查询理论 IR 下限与实际 IR 比较接近,而使用 *k* 分类的查询理论 IR 与实际 IR 普遍偏差较大,即使 *k* 不断增大,也不能显著降低查询的实际 IR,而查询的理论 IR 下限与实际 IR 的偏差反而增大.图 6(Non-Clique Cloaking)中使用 AD 分类的查询理论 IR 与实际 IR 的差距比图 5 要大,但仍然远小于使用 *k* 分类的查询理论 IR 和实际 IR 的偏差.同样,图 7 和图 8 描述了当ρ=0.5 时,Clique Cloaking 和 Non-Clique Cloaking 中 AD 和 *k* 的对比效果,它们仍然满足上述规律.实验结果表明,在使用连续查询攻击算法重识别连续性较强或很强的查询时,AD 更适合作为查询匿名性的度量.

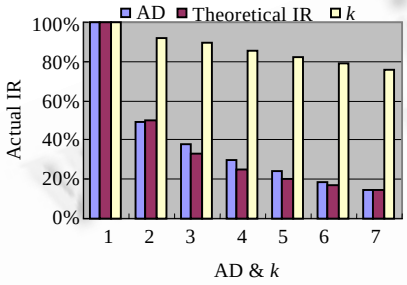


Fig.5 Comparison of AD and *k* in the continuous query attack algorithms against Clique Cloaking

图 5 面向 Clique Cloaking 的连续查询攻击算法中 AD 和 *k* 的比较

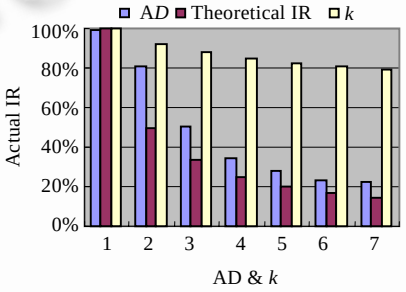


Fig.6 Comparison of AD and *k* in the continuous query attack algorithms against Non-Clique Cloaking

图 6 面向 Non-Clique Cloaking 的连续查询攻击算法中 AD 和 *k* 的比较

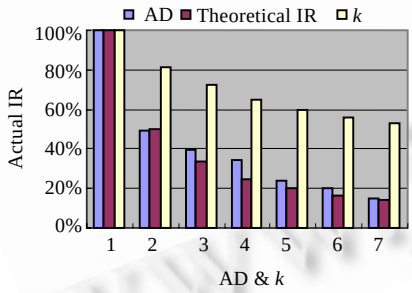


Fig.7 Comparison of AD and *k* in the continuous query attack algorithms against Clique Cloaking

图 7 面向 Clique Cloaking 的连续查询攻击算法中 AD 和 *k* 的比较

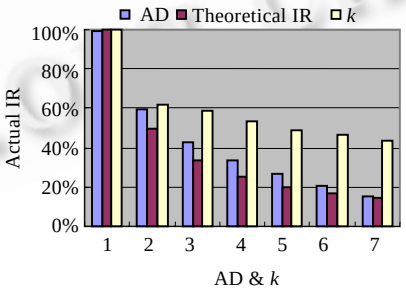


Fig.8 Comparison of AD and *k* in the continuous query attack algorithms against Non-Clique Cloaking

图 8 面向 Non-Clique Cloaking 的连续查询攻击算法中 AD 和 *k* 的比较

5 结论与展望

本文针对现有的 LBS 中 Clique Cloaking 和 Non-Clique Cloaking 分别提出连续查询攻击算法,并提出基于熵理论的查询匿名性度量方法 AD.实验结果表明,在使用攻击算法下,连续性较高的查询被识别率较高,且 AD 比匿名集的势更适合作为查询匿名性的度量.

下一步的工作将立足于本文的结论,改进现有的 Cloaking 算法,使之适用于连续性较高的查询.

References:

[1] Abowd G, Atkeson C, Hong J, Long S, Kooper R, Pinkerton M. Cyberguide: A mobile context-aware tour guide. ACM Wireless Networks, 1997,3(5):421−433.

[2] Bisdikian C, Christensen J, Davis J, Ebling M, Hunt G, Jerome W, Lei H, Maes S. Enabling location-based applications. In: Devarakonda M, *et al*., eds. Proc. of the 1st Workshop on Mobile Commerce. Roma: ACM, 2001. 38−42.

[3] Jose R, Davies N. Scalable and flexible location-based services for ubiquitous information access. In: Gellersen H, ed. Proc. of the 1st Int'l Symp. on Hand-Held and Ubiquitous Computing. LNCS 1707, Heidelberg: Springer-Verlag, 1999. 52−66.

[4] Gruteser M, Grunwald D. Anonymous usage of location-based services through spatial and temporal cloaking. In: Siewiorek D, *et al*., eds. Proc. of the USENIX MobiSys. San Francisco: ACM, 2003. 31−42.

[5] Machanavajjhala A, Gehrke J, Kifer D. l-Diversity: Privacy beyond *k*-anonymity. In: Jain R, *et al*., eds. Proc. of the Int'l Conf. on Data Engineering. Atlanta: IEEE, 2006. 24−24.

[6] Gedik B, Liu L. Location privacy in mobile systems: A personalized anonymization model. In: Lai H, ed. Proc. of the Int'l Conf. on Distributed Computing Systems. Columbus: IEEE, 2005. 620−629.

[7] Yang X, Liu X, Wang B, Yu G. *k*-Anonymization approaches for supporting multiple constraints. Journal of Software, 2006,17(5): 1222−1231 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/17/1222.htm

[8] Ghinita G, Kalnis P, Skiadopoulos S. PRIVE: Anonymous location-based queries in distributed mobile systems. In: Williamson C, *et al*., eds. Proc of the World Wide Web. Banff: ACM, 2007. 371−380.

[9] Mokbel M, Chow C, Aref W. The new casper: A privacy-aware location-based database server. In: Dogac A, *et al*., eds. Proc. of the Int'l Conf. on Data Engineering. Istanbul: IEEE, 2007. 763−774.

[10] Mokbel M, Xiong X, Aref W. SINA: Scalable incremental processing of continuous queries in spatio-temporal databases. In: Konig A, *et al*., eds. Proc. of the ACM SIGMOD. Paris: ACM, 2004. 623−634.

[11] Xiong X, Mokbel M, Aref W. SEA-CNN: Scalable processing of continuous *k*-nearest neighbor queries in spatio-temporal databases. In: Aberer K, *et al*., eds. Proc. of the Int'l Conf. on Data Engineering. Tokyo: IEEE, 2005. 643−654.

[12] Huang X, Jensen C. Towards a streams-based framework for defining location-based queries. In: Nascimento A, *et al*., eds. Proc. of the Int'l Workshop on Spatio-Temporal Database Management. Toronto: ACM, 2004. 73−80.

[13] Mokbel M, Aref W. SOLE: Scalable online execution of continuous queries on spatiotemporal data streams. The Int'l Journal on Very Large Data Bases, 2008,17(5):971−995.

[14] Chow C, Mokbel M. Enabling private continuous queries for revealed user locations. In: Kollios G, *et al*., eds. Proc. of the Int'l Symp. on Spatial and Temporal Databases. Boston: Springer-Verlag, 2007. 258−275.

[15] Shannon C. A mathematical theory of communication. Bell System Technical Journal, 1948,27(7):379−423; 623−656.

[16] Brinkhoff T. A framework for generating network-based moving objects. Geoinformatica, 2002,6(2):153−180.

附中文参考文献:

[7] 杨晓春,刘向宇,王斌,于戈.支持多约束的 *k*-匿名化方法.软件学报,2006,17(5):1222−1231. http://www.jos.org.cn/1000-9825/17/1222.htm

附录

定理 1. 令 $f(p_1,p_2)=-p_1\log_2 p_1-p_2\log_2 p_2$,当给定一个正数 r,规定 $p_1,p_2\in(0,r/2]$,若 $p_1<p_2$,则有 $f(p_1,r-p_1)<f(p_2,r-p_2)$.此定理可以理解为:当两数和相等时,两数差的绝对值越小,它们的 f 值越大.

证明:

$$\frac{\mathrm{d}f(p,r-p)}{\mathrm{d}p}=\frac{\mathrm{d}(-p\log_2 p-(r-p)\log_2(r-p))}{\mathrm{d}p}=\frac{-\mathrm{d}(p\ln p+(r-p)\ln(r-p))}{\mathrm{d}p}=\ln\frac{r-p}{p}=\ln\left(\frac{r}{p}-1\right).$$

当 $p\in(0,r/2)$时, $\ln\left(\frac{r}{p}-1\right)>0$.

所以函数 $f(p,r-p)$是增函数,定理得证. □

定理 2. 根据第 3 节定义的 $AD(q)$,当 $AD(q)$为整数时, $P(u_i \xrightarrow{cm} q)$ $(u_i\in cm.U)$最大值大于等于 $1/AD(q)$.

证明:我们采用反证法.

用 p_i 代表 $P(u_i \xrightarrow{cm} q)$,假设 p_i 的最大值比 $1/AD(q)$小,则所有 p_i 都比 $1/AD(q)$小,且 $k=|cm.U|>AD(q)$,将 p_i 从大到小排列,记为排列 PK_0,根据 $AD(q)$的定义,排列 PK_0 的熵 $H(PK_0)=\log_2 AD(q)$.

将排列 $\underbrace{1/AD(q),1/AD(q),...,1/AD(q)}_{AD(q)个}\underbrace{0,0,...,0}_{k-AD(q)个}$ 称为目标排列,记作 PK_{AD}, $H(PK_{AD})=\log_2 AD(q)$.

以下操作为将排列 PK_i 变为熵更小的排列 PK_{i+1},称为降熵演化:

(1) 在排列 PK_i 中从左到右找到第 1 个小于 $1/AD(q)$的数,记为 p_{left},从右到左找到第 1 个不等于 0 的数,记为 p_{right},所以 $PK_i=\left\{\underbrace{1/AD(q),1/AD(q),...,1/AD(q)}_{0个或多个}p_{left},...,p_{right},\underbrace{0,0,...,0}_{0个或多个}\right\}$.

(2) 若 $p_{left}+p_{right}\geq 1/AD(q)$, 令 $p'_{left}=1/AD(q)$, $p'_{right}=p_{right}-(1/AD(q)-p_{left})$; 否则,令 $p'_{left}=p_{left}+p_{right}$, $p'_{right}=0$.

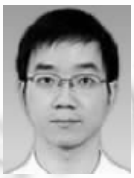
(3) 将 PK_i 中的 p_{left} 和 p_{right} 分别替换成 p'_{left} 和 p'_{right} ,得到排列:

$$PK_{i+1}=\left\{\underbrace{1/AD(q),1/AD(q),...,1/AD(q)}_{0个或多个}p'_{left},...,p'_{right},\underbrace{0,0,...,0}_{0个或多个}\right\}.$$

从上面的步骤中可以看出,每一步降熵演化中 $p'_{left}+p'_{right}=p_{left}+p_{right}$,且 $|p'_{left}-p'_{right}|>|p_{left}-p_{right}|$,根据定理 1, $f(p'_{left},p'_{right})<f(p_{left},p_{right})$,所以 $H(PK_{i+1})<H(PK_i)$.由于降熵演化每次要么在排列前端增加一个 $1/AD(q)$,要么在末端增加一个 0,所以总存在至多 k 步降熵演化: $PK_0\to PK_1\to\ldots\to PK_{AD}$,使得任意 PK_0 变成目标排列 PK_{AD},而且 $H(PK_0)>H(PK_{AD})=\log_2 AD(q)$,与假设中 $H(PK_0)$等于 $\log_2 AD(q)$矛盾,定理得证. □

林欣(1981－),男,福建福州人,博士,主要研究领域为普适计算,上下文感知计算,基于位置的服务.

李善平(1963－),男,博士,教授,博士生导师,主要研究领域为分布式计算,Linux 操作系统内核.



杨朝晖(1979－),男,博士,主要研究领域为普适计算,实时系统.