doi:10.11959/j.issn.1000-436x.2015286

移动对象不确定轨迹隐私保护算法研究

王爽，周福才，吴丽娜
（东北大学 软件学院，辽宁 沈阳 110169）

摘 要：随着移动设备和定位技术的发展，产生了大量的移动对象轨迹数据，相伴而来的是个人隐私泄露问题。现有的轨迹隐私保护研究均假设轨迹数据是准确无误的，但由于数据采集设备不精确、移动对象延迟更新等原因，轨迹数据不确定性普遍存在。提出了一种基于 *K*-匿名的不确定轨迹数据隐私保护方法，对发布的数据进行隐私处理，该方法首次将线性轨迹转化为不确定区域的思想引进轨迹数据的隐私处理。首先，使用概率统计的方法将轨迹泛化成一个更为真实的轨迹区域，然后将相似度高的轨迹域聚合成等价类进行数据的隐匿和发布，最后在真实的数据集上进行实验。

关键词：轨迹数据发布；隐私保护；不确定性轨迹；*K* 匿名；轨迹聚类
中图分类号：TP309.7 **文献标识码**：A

Uncertain trajectory privacy-preserving method of moving object

WANG Shuang, ZHOU Fu-cai, WU Li-na
(Software College,Northeastern University, Shenyang 110169, China)

Abstract: With the development of location based service(LBS) and location-aware devices, the amount of trajectories of moving objects collected by service providers was continuously increasing, meanwhile, it can cause great threaten for personal privacy. Most researches of trajectory privacy preserving were on deterministic data, however, trajectory's uncertainty was inherent due to the inaccuracy of data acquisition equipment, delayed update, and so on. A new method was prosed to protect the privacy of trajectory data in publishing. It is the first time to present the idea that transforming the trajectory to an uncertain area to cluster. First, a probability statistics method to model the trajectory to an uncertain area was proposed. Second, the similar uncertain area into a cluster was put and sanitized in an equivalence class. Finally, the performance of the proposal was compared with (K δ)-anonymity model in real datasets.

Key words: trajectory data publishing; privacy-preserving; uncertain trajectory; *K*-anonymity; trajectory clustering

1 引言

轨迹数据蕴含了大量的时空信息，这些信息可以应用在诸如交通监控、救援服务、军事指挥和移动计算等诸多领域。轨迹数据挖掘可以找出大量移动对象的活动特征，从而为决策提供技术支持。但是利用轨迹数据的攻击性推理也能够推测出用户的社会习惯、行为模式和兴趣爱好等隐私相关的信息[1,2]。因此轨迹隐私保护研究既要保证数据质量以供研究者进行挖掘，同时又要达到保护数据隐私的目的。

由于受到网络带宽、传感器电量、设备测量精度等技术条件的限制，轨迹数据具有不确定性[3,4]。轨迹不确定性的产生主要有如下 2 方面原因。1) 由于定位技术本身以及设备精度、网络延迟、环境干扰等因素会导致获得的位置信息带有不确定性。例如 GPS 设备读取的位置信息，本身就不是一个精确的地理位置，而是由一个采样位置点(经度、纬度)和一个误差范围（3~15 m）组成。另外，传送延时也可能引起定位位置的误差，例如 GPS 系统在返回数据的途中遭遇了信道受阻或者网络信号弱等不良状况，那么返回

收稿日期：2015-10-23
基金项目：国家自然科学基金资助项目（61440014,61300196）；中央高校基本科研业务费专项基金资助项目（130317003）
Foundation Items: The National Natural Science Foundation of China (61440014,61300196)；The Fundamental Research Funds for the Central Universities (130317003)

给服务器的数据就产生了延时，从而导致数据的精确度降低。2) 由于低采样导致轨迹不连续，使 2 个连续离散采样点中间的轨迹位置带有不确定性。

尽管现在针对轨迹隐私保护已经开展了一些工作[5~7]，但这些研究都是针对确定轨迹数据，没有考虑轨迹数据本身固有的这种不确定性特点，因此并不符合实际情况。现有方法主要有两方面不足。

1) 没有考虑采样点位置的不确定性，因此在计算轨迹相似度的时候，仅以采样点欧式距离的和作为度量标准。但在实际情况下，采样点位置是一个不确定的区域，轨迹相似度并不能使用简单的公式计算得到。

2) 认为采样点间移动对象的运动轨迹是直线运动，然后使用这些直线进行诸如聚类等一系列后续处理，但这种方法往往并不符合实际情况。以基于聚类的轨迹隐私保护处理为例，聚类处理是为了将相似的数据放在一起来做隐匿处理，从而减小信息损失而达到保护数据隐私的目的。假设有 2 条轨迹，第一条轨迹在 t_1 时刻的采样点是 A_1，在 t_2 时刻的采样点是 A_2；第二条轨迹在 t_1 时刻的采样点是 B_1，在 t_2 时刻的采样点是 B_2。2 条轨迹的真实情况如图 1(a)所示，使用传统的方法定义的轨迹如图 1(b)所示，使用传统的方法就会将这 2 条轨迹放在一个等价类中，但实际上这 2 条轨迹的差异是很大的。

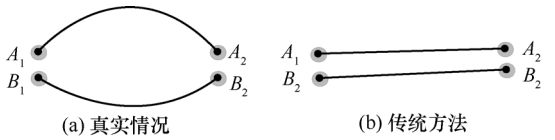

图 1　传统方法与真实情况对比

针对以上问题，本文首次考虑轨迹数据固有的不确定性特征，提出了不确定轨迹的隐私保护方法。相比之前的工作，本文提出的方法在既保证数据的高真实性和降低数据质量损失的同时，又能在用户满意的程度下达到保护隐私的目的。

2　相关工作

2.1　轨迹隐私保护

Gruteser[8]最早将 K-匿名[9]技术应用到轨迹数据上，并成为轨迹隐私保护最普遍采用技术之一。K-匿名方法是在保证客户需要的数据质量前提下，形成一些轨迹数据的空间匿名区域（等价类），每个等价类中至少包含 K 条轨迹。这样使在一个等价类中，即使攻击者定位到当前用户所在的空间匿名区域，其中，至少还存在 $K-1$ 个与其无法区分的轨迹数据，因此确定每个用户的概率不大于 $\frac{1}{K}$。

(K,δ)-匿名[10]方法首次提出了轨迹不确定性的概念，该模型不再将轨迹表示为一条确定的线段，而是用误差半径为 δ 的圆柱体表示不确定轨迹。该模型将任意时刻采样距离不超过 δ 的轨迹定义为相似轨迹，并同样采用构造空间匿名区域的方法对轨迹进行隐私保护。如图 2 所示，2 个误差半径为 δ 的轨迹 τ_1 和 τ_2，若任意时刻 2 个轨迹的采样点距离均不大于误差 δ，那么两者是相似轨迹，则匿名区 S 可以定义为以 $\frac{\delta}{2}$ 为半径，以 S 中所有采样点组成的最小边界圆的圆心为圆心的一组圆柱体。尽管(K,δ)-匿名方法考虑了轨迹的不确定性问题，但其存在 2 个不足。1) δ 是固定的，但在实际环境中，由于受到周围环境的影响，往往不同时刻测量的误差是不同的。例如 GPS 信号在空旷的或密闭的环境测量的精度是不一样的。2) 在隐私处理过程中，仍然是将轨迹当作精确的线段进行处理，并没有考虑真实的轨迹是一个不确定的区域，这与之前的方法并无不同。

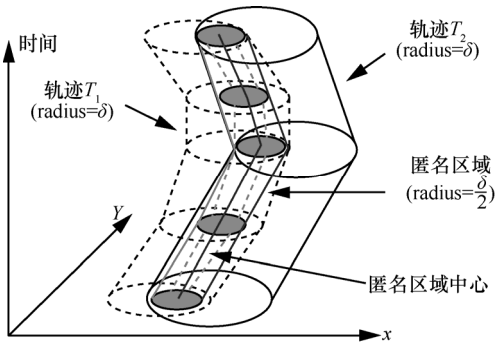

图 2　(K,δ)-匿名区域构造示意

2.2　不确定轨迹模型

目前表示轨迹的不确定模型主要有 4 种：cylinder[11]模型、beads[12,13]模型、grid 模型和 evolving[14]模型。1) cylinder 模型将一个二维空间中的轨迹数据表示为一个三维的圆柱体，其中，圆柱体的中心为轨迹的采样值，圆柱体每一个横切面的半径为最大误差值。该模型主要存在 2 个问题。①假设物体是按直线匀速行驶，很显然这不符合大部分物体的运动规律。②需要最大误差 ε 参数。2) beads 模型用连续的椭圆形链条来表示运动物体的轨迹。即给定 2 个连续的采样坐标 $P_1(x_1, y_1)$、$P_2(x_2, y_2)$(对应时间 t_1、t_2)和移动对象的最大运动速度 v_{max}，对于时间变量 $t_x(t_1<t_x<t_2)$)，该移动对象在 P_1 和 P_2 坐标点之间所有可能的运动轨

迹满足一个椭圆形覆盖区域。该模型的问题在于：①采样间隔很大时，不确定区域的范围很大；②需要参数 v_{max}；3) grid 模型将空间划分为若干不相交的网格，将移动轨迹表示为一组网格序列，尽管网格方法较前 2 种方法更加简单，处理效率更高，但是该模型的主要问题是，需要网格尺寸参数，该参数难以确定，而网格尺寸是反映位置不确定程度，影响算法运行效率的至关重要因素；4) evolving 模型改进了上述 3 种模型都需要确定静态参数的问题，提出了使用金融分析领域采用的GRACH模型动态计算不同时刻轨迹误差的方法，将 cylinder 模型改进为半径不同的圆台，但该模型与 cylinder 模型一样，同样存在假设物体是按直线匀速行驶的问题，并未考虑轨迹位置的时间相关性。

针对现有不确定轨迹模型存在的问题，本文对 evolving 模型进行了改进，构建了一种新的基于马尔可夫关联的不确定轨迹模型（EMUT, evolving Markov uncertain trajectory model），该模型既能够捕获轨迹随时间变化的动态特征，又考虑了位置间的时间关联性。基于该模型，设计了一种新的基于 K 匿名技术的适用于不确定轨迹数据的隐私保护算法。

3 马尔可夫关联的不确定轨迹模型

首先介绍不确定轨迹的定义。然后描述 EMUT 模型的构建与度量标准，最后介绍基于布朗运动的 EMUT 模型。

定义 1 不确定轨迹 T。不确定轨迹 T 是若干个不确定轨迹数据项集构成的有序序列，可表示为 $T=\{(<x_1, y_1>, v_1, t_1), (<x_2, y_2>, v_2, t_2), \cdots, (<x_n, y_n>, v_n, t_n)\}$, $(t_1<t_2<\cdots<t_n)$。其中，$<x_i,y_i>$是移动对象 O 在 t_i 时刻的采样位置信息，v_i 表示不确定区域半径。

图 3 所示为一条不确定轨迹，每个时刻的测量误差是动态变化，这里用 v_i 高斯分布的方差表示，v_i 可采用 Evolving 模型的方法计算得到。另外，相邻时刻的运动轨迹可以是任意路线，不仅仅限定为直线运动。

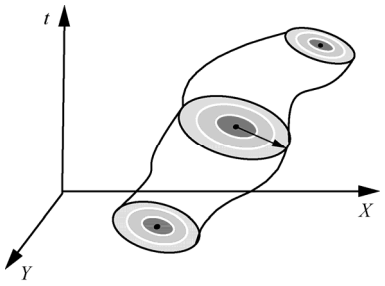

图 3 不确定轨迹

3.1 不确定轨迹相似度量函数

轨迹隐私保护的一个重要内容是将相似的轨迹聚在一起，构造成一个等价类，进而做隐匿处理。对于不确定轨迹数据的聚类处理也要划分等价类，那么就要建立不确定轨迹的相似度判定标准，然后根据轨迹的相似程度来判定 2 条轨迹是否能聚合在一起。下面介绍不确定轨迹的相似度量标准。

定义 2 时刻距离 S（T_i,T_j,t）。2 条轨迹 T_i 和 T_j 在 t 时刻的距离如式（1）所示。

$$S(T_i,T_j,t)=\begin{cases} area\left(T_i(t)\cap T_j(t)\right), & d<(\delta_i+\delta_j) \\ +\infty\quad , & d\geq(\delta_i+\delta i) \end{cases} \tag{1}$$

其中，2 条轨迹域 T_i 和 T_j 在 t 时刻的采样点分别是 (x_i,y_i,δ_i) 和 (x_j,y_j,δ_j)，2 个圆心的距离为 $d=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}$。当 $d<(\delta_i+\delta_j)$时，$S(T_1,T_2,t)$为 2 条轨迹相交部分的面积；当 $d\geq(\delta_i+\delta_j)$时，距离为$+\infty$。

从图 4 可以看出，时刻距离就是 2 个圆的相交部分面积。在任意时刻，2 个相交圆的面积可以转化成一个函数去计算，此函数自变量应当是两圆心位置与两半径长度。根据时刻距离的定义，可以定义时刻相似度的判定标准。

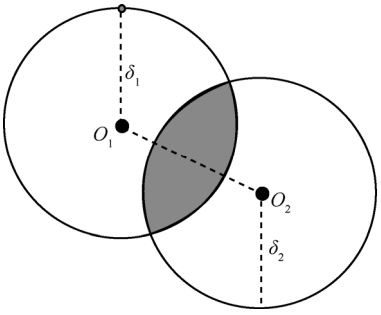

图 4 不确定轨迹时刻距离

定义 3 时刻相似度 $I\_sim(T_i,T_j,t)$。2 条轨迹在时刻 t 的相似度是 2 个不确定区域相交面积占其中较小一方面积的百分比，如式（2）所示，其中，$\pi(T_i,t)$ 表示轨迹 T_i 在 t 时刻的不确定区域面积。

$$I\_sim(T_i,T_j,t)=\frac{s(\tau_i,\tau_j,t)}{\min\left\{\pi(\tau_i,t)\pi(\tau_j,t)\right\}} \tag{2}$$

定义 4 时段距离 V（T_i,T_j,t_m）。2 条轨迹在相邻时间段$[t_m,t_{m+1}]$间的距离为

$$V(T_i,T_j,t_m)=\int_{t_m}^{t_{m+1}} S(T_i,T_j,t)\,\mathrm{d}t \tag{3}$$

一条轨迹有相邻的 2 个采样时刻 t_m 和 t_{m+1}，若 t_i 到 t_{i+1} 时刻距离不为 0，轨迹时段距离为轨迹域相交部分的体积，如图 5 所示，其数值可使用积分计算得出。

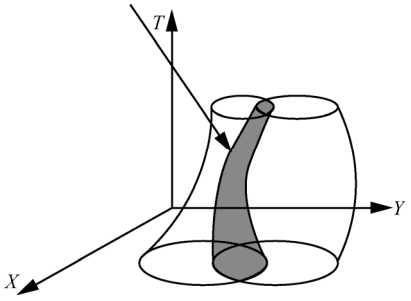

图 5　时段距离

定义 5　时段相似度 $S\_sim(T_i,T_j,t)$。在相邻时间段$[t_m,t_{m+1}]$，轨迹 T_i 和 T_j 的相似度为轨迹域相交体积与 2 个轨迹域体积较小的一方的体积的比值，如式（4）所示，其中 $vol(\tau_j)$ 是轨迹域的体积。

$$S\_sim(T_i,T_j,t_m)=\frac{s(\tau_i,\tau_j,t_m)}{\min\{vol(\tau_i)vol(\tau_j)\}} \tag{4}$$

定义 6　轨迹相似度 $T\_sim(T_i,T_j)$。轨迹 T_i 和 T_j 在整个时间段的相似度为每段相似度之和，如式（5）所示。

$$T\_sim(T_i,T_j)=\sum_{m=1}^{n-1} S\_sim(T_i,T_j,t_m) \tag{5}$$

因此，用户可以设定阈值 α 作为判定轨迹是否相似的度量标准，当 $T\_sim<\alpha$ 时，认为 2 条轨迹满足聚类条件，便将这 2 个轨迹放入一个等价类中。每当有 2 条轨迹被放入一个等价类时，为提高下一次聚类的效率，引入聚类中心的概念。

聚类中心是为了解决当等价类中存在 2 条或 2 条以上轨迹时，如何继续进行聚类的问题。聚类中心的存在可以把一个等价类看做是一个新的轨迹域，利用这个轨迹域可以与其他轨迹域进行聚类，最终会得到包含至少 K 条轨迹域的等价类。

定义 7　聚类中心。当等价类中只有一条轨迹时，这条轨迹被认为是这个等价类的聚类中心；否则，在每一时刻，取截面的相交弦中点 O 为圆心，线段 d 为直径所形成的圆为该时刻的聚类中心，在整个时间段上累积成的几何体为新的聚类中心，如图 6 所示。

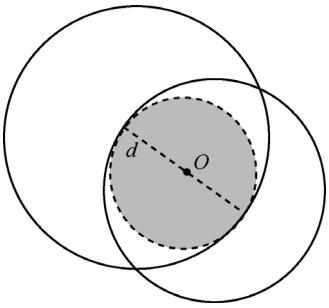

图 6　聚类中心在某一时刻的截面

3.2　不确定轨迹模型 EMUT

由定义 1 可知，不确定轨迹在任意时刻是以采样点值(x,y)为圆心，测量误差 v 为半径的不确定圆形区域。对于采样时刻，这些值是已知的，而对于非采样时刻或者采样值缺失的情况，这些值是未知的，需要计算得到。

针对非采样值的计算，目前研究工作均假设移动对象的运动轨迹随时间独立，并未考虑时间的关联性，这显然不符合实际情况。实际生活中很多对象的运动特征都符合马尔可夫特性，因此本文使用马尔可夫随机过程刻画轨迹数据的时间关联性。马尔可夫性质表示 t+1 时刻系统状态的概率分布只与 t 时刻的状态有关，与 t 时刻以前的状态无关。基于该性质，可以计算任意时刻不确定轨迹的位置和误差半径。

1) 计算圆心。圆心的计算本文采用简单的直线连接的方式计算。如计算$[t_i,t_{i+1}]$2 个连续采样时刻中间某时刻 t 的不确定区域的圆心，仅将 2 个采样点用直线连线，直线上 t 时刻的点即为圆心，如式（6）所示。

$$\begin{aligned} x_t &= x_i+(x_{i+1}-x_i)\frac{t-t_i}{t_{i+1}-t_i}\\ y_t &= y_i+(y_{i+1}-y_i)\frac{t-t_i}{t_{i+1}-t_i} \end{aligned} \tag{6}$$

2) 计算半径。已知 t_i 时刻的半径，如何计算 t_j 时刻的半径是本文的难点之一。这里采用一种特殊的马尔可夫过程—布朗运动来计算任意时刻不确定区域的半径。

下面简单地介绍布朗运动的一些性质[15,16]。布朗运动在 s 时刻服从高斯分布，概率密度函数如式(7)所示。

$$p(s,x)=\frac{1}{\sqrt{2\pi s}}\mathrm{e}^{-\frac{x^2}{2s}} \tag{7}$$

当已知 s 时刻的概率密度函数, t 时刻的概率密

度函数如式（8）所示。布朗运动在 t 时刻仍然满足高斯分布，期望 $\mu=x$，方差 $\sigma=\sqrt{t-s}$。

$$p(s,x,t,y)=\frac{1}{\sqrt{2\pi(t-s)}}\mathrm{e}^{-\frac{(y-x)^2}{2(t-s)}} \tag{8}$$

根据高斯分布的 3σ 定理可知，x 落在$[\mu-3\sigma, \mu+3\sigma]$ 区间的概率是 99.7%的，这个值通常能够满足用户的需求。因此，当已知轨迹在采样点 s 时刻的半径 x，则 t 时刻的半径应在$[x-\sqrt{t-s}, x+\sqrt{t-s}]$范围之内。因此，本模型使用布朗运动之后，时刻距离 $S(T_i,T_j,t)$ 变成了一个环形，环的内圆半径为 $x-\sqrt{t-s},$，外圆半径为 $x+\sqrt{t-s}$。

使用布朗运动后，时刻距离 $S(T_i,T_j,t)$ 的值从 2 个圆相交部分面积转换到 2 个环形相交部分的面积，如图 7 所示，但这并不影响 3.1 节介绍的计算方法，其面积仍然是与两圆心坐标面积和内外半径相关的函数。

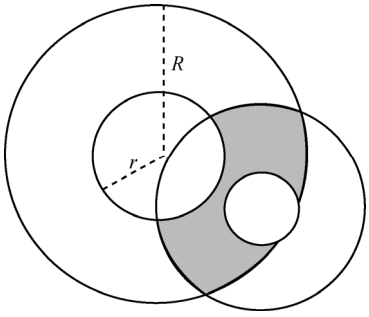

图 7 应用于布朗运动的时刻距离

通过上述分析，已经得到了计算轨迹时刻距离 $S(T_i,T_j,t)$ 需要的 2 要素（圆心和半径）的计算方法，它们都是关于时间 t 的函数，因此 $S(T_i,T_j,t)$ 也是一个关于时间 t 的函数，为了描述方便，该函数简化用 $f(t)$表示。

为了进行轨迹聚类，最终要计算轨迹的相似度，即是要根据轨迹段的相交部分体积 $V(T_i,T_j,t_m)$ 来计算轨迹的相似度。因为 $S(T_i,T_j,t)$ 是关于时间 t 的一重积分函数，那么相交部分体积 $V(T_i,T_j,t_m)$ 是关于时间 t 的二重积分函数，这个二重积分问题可以转化为一个在时间区间$[T_i,T_j]$的定积分问题，根据辛普森法则，可以求得定积分的近似值，如式（9）所示，利用式（9）就可以计算出轨迹相似度的值。

$$\int_{T_i}^{T_j} f(t)\,\mathrm{d}t \approx \frac{T_j-T_i}{6}\left[f(T_i)+4f\left(\frac{T_i+T_j}{2}\right)+f(T_j)\right] \tag{9}$$

4 基于 *K*-匿名的不确定轨迹隐私保护算法

本文提出的轨迹隐私保护算法包括 UT-Tree 的构建和 UT-Tree 的剪枝。

1) UT-Tree 的创建。受到 PrivateCheckIn 算法[2]中前缀树思想的启发，本文采用前缀树结构来表示不确定轨迹，本文将该树命名为 UT-Tree(uncertain trajectory tree)，可以达到节省存储空间的目的。在轨迹聚类的处理上，UT-Tree 的每一层代表一个采样时刻，每个节点(*cp*)表示一个采样时刻的位置区域。只要某轨迹在 t 时刻采样点的位置在树中出现过或其满足相似性判定标准，那么就将该采样位置归并到这个节点之中；否则再创建一个新节点，以此类推，直到所有的轨迹处理完毕。

算法 1 构建 UT-Tree

输入：原始轨迹数据集 D

输出：UT-Tree

1) create *root* for UT-Tree;
2) FOR each trajectory Tr_i in D DO
3) call *insert_trajectory*(Tr_i,*root*);
4) Proc *insert_trajectory*(*Tr*,*N*)
5) FOR each point sp_i in *Tr* DO
6) IF($I\_sim(sp_i,N,t_i)<\delta$)
9) add sp_i to N ; N.count++;
10) ELSE
11) compute the cluster center of path(*root* ,*N*);
12) IF $T\_sim(Tr', path(root,N)<\delta$
13) add sp_i to N_i of current path;
14) N_i.count++;
15) ELSE
create a newNode, count =1, it's parent as N

算法 1 描述了 UT-Tree 构建的过程。首先创建树的根节点(算法第 1)行)，然后调用 *insert_trajectory* (Tr_i, *root*)函数依次插入每条轨迹(算法第 2)、第 3)行)。该函数将轨迹 *Tr* 中的采样点 sp_i 与当前路径上的节点 N 比较，若满足时刻相似条件，则将该采样点加入到 N 节点中，N 的计数增 1(第 4)~第 9)行)；若不满足，计算从根节点到当前节点 N 的聚类中心(第 10)~第 11)行)，选取轨迹 *Tr* 中的$\{sp_1,\cdots,sp_i\}$的采样点构成 Tr'，计算 Tr' 与当前路径的聚类中心的轨迹相似度，若满足相似条件(第 12)行)，则可以将 Tr' 中

的点分别加入当前路径的对应节点集合中，对应点计数增 1(第 13)、第 14)行)；若还不满足，以节点 N 为父亲节点，创建新的节点，计数初始化为 1(第 15)行)。

表 1 是一个含有 5 条轨迹的数据库，其中，L 表示采样点，每行代表一条轨迹，下标相同的采样点表示它们的位置是相似的。采用算法构造 UT-Tree 结果如图 8 所示。

表 1　　不确定轨迹数据集

ID	t_1	t_2	t_3
1	L_1	L_2	L_1
2	L_1	L_2	—
3	L_1	L_2	L_4
4	L_1	L_3	L_5
5	L_1	L_3	L_5

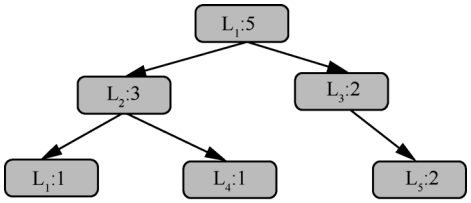

图 8　UT-Tree 构建

2) UT-Tree 剪枝。最后可以对 UT-Tree 进行剪枝处理，即去掉树中一些节点使其能够满足聚类条件。如果树的分支中包含的轨迹数量不小于 K 且轨迹长度不小于 m，那么就可以认为前缀树分支所包含的轨迹满足聚类条件并生成一个等价类，具体如算法 2 所示。

算法 2　UT-Tree 剪枝

输入：UT-Tree UT，聚类阈值（支持度）K，树深度阈值 m

输出：剪枝后的 UT-Tree UT^*

1)　$C_{\text{list}} \leftarrow \varnothing$;

2) IF n_i.count<K and $depth$<m

3)　　IF n_i is a leaf node and　$depth \le 2$

4)　　$UT^* \leftarrow UT - n_i$;

5)　　ELSE

6)　　FOR each n_j in Path(n_i,UT)

7)　　n_j.−=n_j.count−n_i.count;

8)　　C_{list}=Set of sequences in Path(n_i,UT);

9)　　$UT^* \leftarrow UT$ -*subTree* included by n_i;

10) ELSE

11)　FOR each $n_c \in n_i$.children

12)　　$C_{\text{list}} \cup UT^*$ (n_i,UT,K);

13) RETURN(UT^* and C_{list});

算法 2 由根节点开始向下遍历前缀树中所有节点，如果某一个节点的支持度是小于 K 的，则对该节点进行剪枝处理。由于节点的类型存在差异性，剪枝处理分为 2 种情况：如果节点是深度大于 m 的叶子节点，因为其不影响父节点的支持度，所以可以直接将该节点删除(第 1)~第 4)行)；如果节点是深度不大于 m 或者是非叶子节点，那么就需要剪除掉该节点所在的整条路径，然后所有以该节点为终点路径上的节点的支持度减去该节点的支持度(第 5)~第 9)行)。被剪除的序列全部存储到 C_{list} 中。除了单个叶子节点，同时对该节点的子节点进行的相同剪枝操作(第 10)~第 12)行)，最后返回进行剪枝处理后的树和剪除序列集合 C_{list}(第 13)行)。

如图 8 的树状结构，假设需要得到满足 K=2，m=2 的等价类，需要对得到的 UT-Tree 进行剪枝处理，如果去掉第 3 层的 L_1 和 L_4 就能得到{$L_1 \rightarrow L_2$}的等价类，同时还可以得到{$L_1 \rightarrow L_3 \rightarrow L_5$}的等价类。

5　实验结果

据本文所知，目前尚没有将轨迹表示成不确定区域的序列，并采用不确定区域相似度度量标准，将其进行隐私化处理的算法。目前，与本文比较相关的算法仅有(K,δ)-匿名算法，因此本节描述了本文提出的算法 *EMUT* 与(K,δ)-匿名提出的算法 W4M 在真实数据集中测试的实验结果。

实验结果从运行时间、聚类成功率和信息损失度 3 个方面进行分析。聚类成功率是指可发布数据中等价类的个数，或者说是不能构成等价类所抛弃的轨迹数据情况；信息损失度是指数据进行聚类和匿名化后，原始数据与可发布数据的差异程度，在计算方法上采用计算每个采样点间的欧几里得距离作为差异度的衡量标准。

5.1　实验环境

开发环境。本文采用 C++语言编程，硬件环境采用 Intel Core i7 3.4 GHz 处理器，8 GB 内存；软件环境为 Windows 2007 操作系统，使用 Visual Studio 2010 开发平台。

实验数据。数据集使用社交网络 Brightkite 从 2008 年 10 月到 2010 年 10 月的真实数据，该数据表示用户在不同时间签到位置的信息，因此能够表示用户在一定范围内的轨迹。因为该数据不具备数

据的不确定性，本文人工地为数据添加一个不确定半径的属性，数据值在一定范围内随机添加。默认情况，数据规模为 2 000 条轨迹（约 15 000 个采样点），K=15，α=0.6。

数据预处理。每条轨迹可能因为采样技术的原因而造成采样时间的延时或缺失，从而造成同一时间抓取的信息的采样时间是不同的。在这样的情况下，2 条轨迹的采样点的采样时间会出现 2 种情况，一种是 2 个采样点的采样时间不会相差很大，本文的做法是将其同一对待，视为同一时间；另一种则是在某一采样时间内没有采样信息，本文的做法是取前一时刻和后一时刻坐标的平均值作为这一时刻的坐标，不确定半径的计算方法则按照 3.2 节介绍的方法计算。

本文使用的数据预处理方法是：首先扫描数据进行统计，根据大多数轨迹的首采样点和尾采样点规定起始时间和终止时间，再次扫描数据，仅保留轨迹中在起始时间和终止时间之内的采样点；然后根据第一次扫描数据得到的采样时间频度对不同轨迹采样点做时间填补法，并且用序列号代替原始数据的采样时间，这样能简化编码阶段处理过程，提高程序效率。

5.2 实验结果

1) 参数 K 对结果的影响。参数 K 是基于 K-匿名聚类算法的重要参数，其规定了等价类的大小，直接决定了数据等价类存在的数量和数据匿名化程度，间接地影响了数据损失度。从理论上讲，参数 K 越大则等价类中所容纳的轨迹数目也就越多，越不容易构成等价类并且聚类时各个度量标准的计算也就越复杂，因此对聚类成功率和运行效率影响也就越大。参数 K 对实验结果的影响如图 9~图 11 所示。

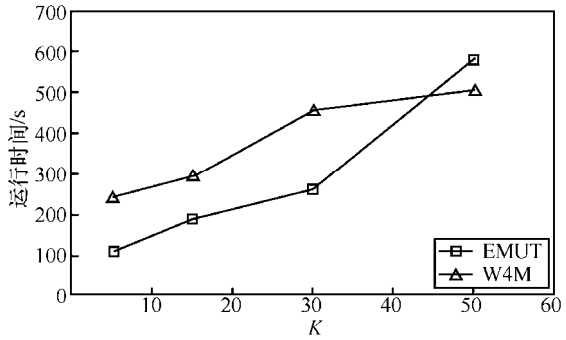

图 9 参数 K 对运行的时间影响

从图 9 可以看出，算法运行时间随着 K 的增大而增大，因为在聚类过程中，等价类要求的轨迹数量越多，每次进行相似度度量和更新聚类中心的开销就会越大，这是导致运行时间增大的主要原因。W4M 所呈现出的趋势与 EMUT 总体相同，当 K 逐渐增大时，W4M 的运行时间是高于 EMUT 的。

从图 10 可以看出，整体上随着 K 的增大聚类成功率是随之减小的，尤其是在 K 取 50 时，聚类成功率低达 61%，这说明 EMUT 在聚类成功率方面不适合 K 值过大。当 K 值较小时，等价类形成的选择性也就越多，因此更加容易形成等价类，随之被抑制的轨迹数据也就较少。与 W4M 相比，EMUT 的聚类成功率要略高于 W4M。但是当 K 值逐渐增大时，EMUT 的聚类成功率急剧下降，与 W4M 相同。

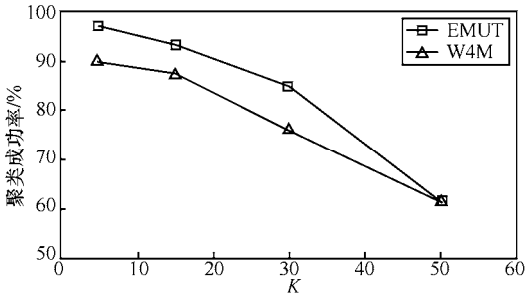

图 10 参数 K 对聚类成功率的影响

信息损失度度量标准规定原始采样点与发布采样点的欧几里得距离偏差不超过 800 m 记为在允许损失范围，信息损失度就是计算在允许损失范围内与所有采样点数量的比例。

从图 11 中可以看出，随 K 值的增大，信息损失度是随之增大的。当 K 值增大到 50 时，信息损失度高达 51%，这说明 EMUT 在信息损失度方面不适合 K 值过大。当 K 值变大时，大多数等价类中轨迹数目变大，轨迹间的差异也相应变大，在之后的隐匿过程中与原始数据的差异也就越大。在 K 值较小时，EMUT 与 W4M 的信息损失度差异不大，但是 K 逐渐增大时，由于本文提出的不确定聚类的条件更加严格，W4M 的信息损失度低于 EMUT。

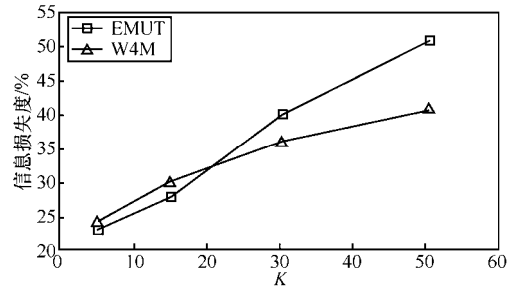

图 11 参数 K 对信息损失度的影响

2) 参数 α 对结果的影响。参数 α 是基于 K-匿名聚类的重要参数，其决定了轨迹相似度的判定标

准，也就是说对轨迹相似度的要求越高，聚类的成功率也就越低，从而造成被抑制的点增多，信息损失度会有所增加。参数 α 对实验结果的影响如图 12~图 14 所示。

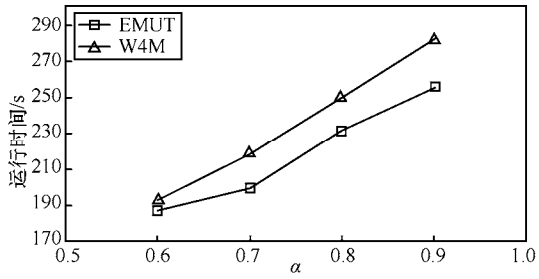

图 12　参数 α 对运行时间的影响

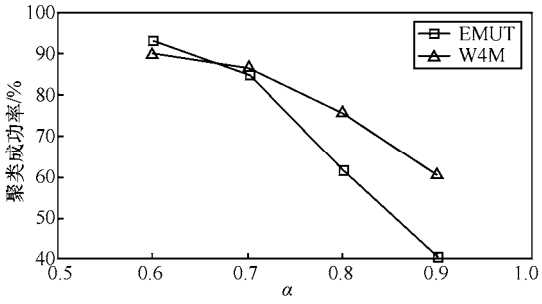

图 13　参数 α 对聚类成功率的影响

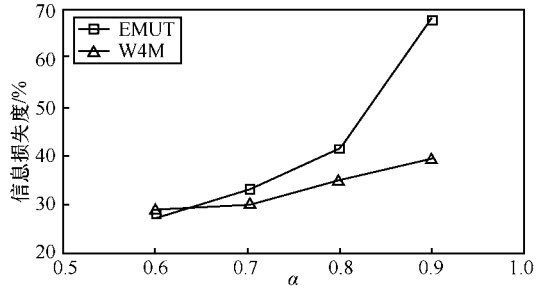

图 14　参数 α 对信息损失度的影响

从图 12 可以看出，运行时间随着 α 的增大而增大，但是波动不大，比较平缓，这说明 α 的值对运行时间的影响不是很大。在聚类过程中，轨迹相似度的规定阈值越大，每次判定的不成功率增大，相应的判定次数会增多，这也就是运行时间会增大的原因，但从实验结果来看对其影响并不是很大。

从图 13 可以看出，整体上随着 α 的增大，聚类成功率是随之急剧减小的，尤其是在 α 取 0.9 时，聚类成功率低达 41%，这说明 EMUT 在聚类成功率方面不适合 α 值过大。当 α 值过大时，能够满足聚类条件的轨迹也就越少，自然轨迹难以聚成一类。

EMUT 相较于 W4M，当 α 相同时，运行效率相差不大。当 α 较小时，二者的聚类成功率相差不大，但是当 α 变大时，EMUT 的聚类成功率明显下降，说明 EMUT 不适合聚类要求很高的需求。

3) 数据规模对结果的影响。数据规模是 EMUT 处理数据量的大小，通常数据规模都是影响运行时间的一个重要因素，而聚类成功率和信息损失度受算法的好坏和其设定参数影响，数据规模对其影响不大。实验中分别采用 1 000、2 000 和 5 000 条轨迹数据，数据规模对实验结果的影响如图 15~图 17 所示。

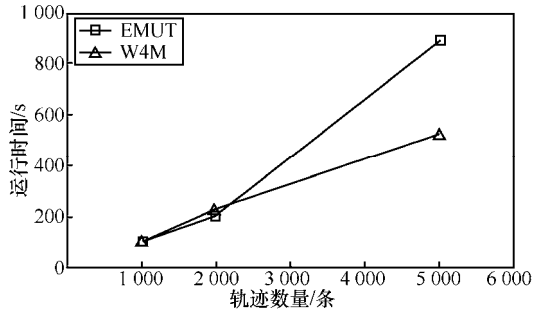

图 15　数据规模对运行时间影响

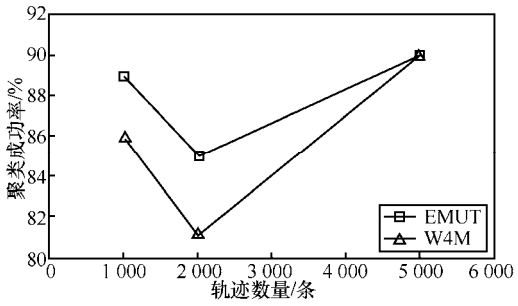

图 16　数据规模对聚类成功率的影响

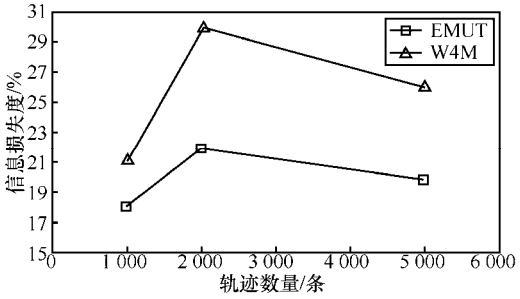

图 17　数据规模对信息损失度的影响

从图 15 中可以看出，随着数据规模的增大，算法运行时间增大，当数据量只有 1 000 和 2 000 时，EMUT 处理数据的效率还算正常，但是数据量增大到 5 000 时，运行时间急剧增加到达 896 s。随着数据量的增大，聚类时每次寻找可并入当前等价类的轨迹的候选集增大，程序需要花费更多的开销计算 候选集中轨迹的相似度，因此造成了运行时间的大幅度增加。当数据规模不大时，EMUT 和 W4M 的运行效率

相差不大，但是数据规模急剧增大时，EMUT 所花费的运行效率要远远高出 W4M。因为 EMUT 的轨迹模型更加复杂，因此处理开销也要增加。

从图 16 中可以看出，无论数据量大小，聚类成功率表现较为稳定，说明 EMUT 面对数据规模的变化，是能够保证聚类成功率的。从变化趋势上看，当数据量由 1 000 增大到 2 000 时聚类成功率略有降低的，从 2 000 增大到 5 000 时聚类成功率反而升高。这说明当数据规模急剧增大时，等价类的构成在候选集中提升了可选择性，能够构成等价类的机会也就增加了，这就造成了面对大量数据，聚类成功率反而会增加的现象。EMUT 和 W4M 在数据规模增大的情况下二者聚类成功率相差不大，说明二者对于数据规模的变化而言都是稳定的。

从图 17 中可以看出，在参数 K 取 15，α 取 0.7 时，信息损失度较低且比较稳定。选择良好的参数，使得等价类中轨迹的相似度较高，故在匿名化的阶段的信息损失较少。对于不同的数据规模，EMUT 的信息损失度总体上的总是低于 W4M，说明随着数据规模的变化，EMUT 总能保证较高的数据质量。

6 结束语

针对(K,δ)-匿名技术和进化式模型的不足，本文提出了一种新的具有时间相关性的不确定轨迹模型 EMUT，并详尽地介绍了 EMUT 模型的构建方法和不确定轨迹相似度度量标准，并采用了一种特殊的、具有马尔可夫性质的随机过程—布朗运动，应用到 EMUT 模型上。之后，提出了一种基于 EMUT 模型的不确定轨迹隐私保护算法，该算法按照 EMUT 模型的聚类标准构造等价类，之后对每个等价类按照 K 匿名的思想进行匿名处理。最后通过实验数据，分析了本文提出的算法在不同条件影响下，在运行时间、聚类成功率和信息损失度 3 个方面的性能。

参考文献：

[1] 霍峥，孟小峰. 轨迹隐私保护技术研究[J]. 计算机学报，2011, 34(10): 1820-1830.
HUO Z, MENG X F. A survey of trajectory privacy-preserving techniques[J]. Chinese Journal of Computers, 2011, 34(10): 1820-1830.

[2] 霍峥，孟小峰，黄毅. PrivateCheckIn：一种移动社交网络中的轨迹隐私保护方法[J]. 计算机学报, 2013, 36(4):716-726.
HUO Z, MENG X F, HUANG Y. PrivateCheckIn: trajectory privacy-preserving for check-in services in MSNS[J]. Chinese Journal of Computers, 2013, 36(4):716-726.

[3] EMRICH T, KRIEGEL H-P, MAMOULIS N, *et al*. Querying uncertain spatio-temporal data[A]. Proc of the 2012 IEEE 28th International Conference on Data Engineering[C]. 2012. 354-365.

[4] CHUNYANG M, HUA L, LI D S, *et al*. KSQ: top-*k* similarity query on uncertain trajectories [J]. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(9): 2049-2062.

[5] SHENG G, JIAN F M, WEI S S, *et al*. a trajectory privacy-preserving framework for participatory sensing [J]. IEEE Transactions on Information Forensics and Security, 2013, 8(6): 874-887.

[6] GHASEMZADEH M, FUNG BCM, CHEN R, *et al*. Anonymizing trajectory data for passenger flow analysis [J]. Transportation Research Part C: Emerging Technologies, 2014, 39(2): 63-79.

[7] CHEN R, FUNG BCM, MOHAMMED N, *et al*. Privacy-preserving trajectory data publishing by local suppression[J]. Information Sciences, 2013, 231(9): 83-97.

[8] MARCO GRUTESER, Dirk GRUNWALD. Anonymous usage of location-based services through spatial and temporal cloaking[A]. Proc of the First International Conference on Mobile Systems, Applications, and Services. San Francisco[C].USA, 2003. 277-286.

[9] WEENEY S L. *K*-anonymity: a model for protecting privacy [J]. International Journal of Uncertainty on Fuzziness and Knowledge-based System, 2002, 10(5): 557-570

[10] ABUL O, BONCHI F, NANNI M. Anonymization of moving objects databases by clustering and perturbation [J]. Information Systems, 2010, 35(8):884-910.

[11] FRENTZOS E, GRATSIAS K, THEODORIDIS Y. On the effect of location uncertainty in spatial querying [J]. IEEE Transactions on Knowledge and Data Engineering, 2009, 21(3): 366-383.

[12] KUIJPERS B, OTHMAN W. Trajectory databases: data models, uncertainty and complete query languages[J]. Journal of Computer and System Sciences, 2010, 76(7): 538-560.

[13] LIU H, SCHNEIDER M. Querying moving objects with uncertainty in spatio-temporal databases [A]. Proc of the 16th Database Systems for Advanced Applications[C]. 2011. 357-371

[14] JEUNG H, LU H, SATHE S, et al. Managing evolving uncertainty in trajectory databases[J]. IEEE Transactions on Knowledge and Data Engineering (TKDE), 2014, 26(7): 1692-1705.

[15] FAROOKH K H, ELIZABETH C, THARAM S D. Markov model for modelling and managing dynamic trust[A]. Proc of the 3rd IEEE International Conference on Industrial Informatics[C]. India, 2005. 725-733.

[16] CHEN C, YAN L. Remarks on the intersection local time of fractional Brownian motions [J]. Statistics & Probability Letters, 2011, 81(5): 1003-1012.

作者简介：

王爽（1980-），女，辽宁沈阳人，博士，东北大学讲师，主要研究方向为隐私保护、数据挖掘，不确定数据管理。

周福才（1964-），男，辽宁沈阳人，博士，东北大学教授、博士生导师，主要研究方向为可信计算、网络安全。

吴丽娜（1991-），女，辽宁阜新人，东北大学硕士生，主要研究方向为隐私保护、数据挖掘。