PAPER Special Section on Trust, Security and Privacy in Computing and Communication Systems

A Clustering K-Anonymity Scheme for Location Privacy Preservation

Lin YAO[†], **Guowei WU**[†], **Jia WANG**[†], **Feng XIA**[†a)], **Chi LIN**[†], ***Nonmembers***, ***and*** **Guojun WANG**[††], ***Member***

SUMMARY The continuous advances in sensing and positioning technologies have resulted in a dramatic increase in popularity of Location-Based Services (LBS). Nevertheless, the LBS can lead to user privacy breach due to sharing location information with potentially malicious services. A high degree of location privacy preservation for LBS is extremely required. In this paper, a clustering K-anonymity scheme for location privacy preservation (namely CK) is proposed. The CK scheme does not rely on a trusted third party to anonymize the location information of users. In CK scheme, the whole area that all the users reside is divided into clusters recursively in order to get cloaked area. The exact location information of the user is replaced by the cloaked spatial temporal boundary (*STB*) including K users. The user can adjust the resolution of location information with spatial or temporal constraints to meet his personalized privacy requirement. The experimental results show that CK can provide stringent privacy guarantees, strong robustness and high QoS (Quality of Service).

key words: *K-anonymity, spatial-temporal constraints, location-based services, location privacy, clustering*

1. Introduction

The explosive growth of location-detection devices and wireless communication has resulted in the wide application of Location-Based Services (LBS). The main aim of LBS is to provide services to mobile users based on the knowledge of their locations as well as augmenting many existing services with location information [1]. Examples of LBS are location-based tourist information, live traffic reports, food and drink finder, etc [2].

The location information of the user is required in LBS, and it is essential in delivering a mobile service, but it may pose a threat on a user's privacy. A person's preference, employ status, or health condition may be inferred based on the location information. Various distressing privacy violations caused by sharing sensitive location information with potentially malicious services have highlighted the importance of location privacy preservation in LBS [2].

Most of the solutions proposed [3]–[12] to preserve the location privacy are based on Trusted Third Parties (TTP) entity, which is used to blur the exact location information of the user before sending the request to LBS provider. Although this approach is widely accepted, there are some limitations. First, the TTP could become the attack critical point. Once attackers succeed in invading the TTP, it poses great risk on user privacy. The recent reports related to the disclosure of personal data by this kind of trusted entities [2] has proved it. Second, if the TTP is unreliable, the location information may be abused and the users may face undesired advertisements, e-coupons, etc. Thus, the users would prefer to trust nobody, which leads to TTP-free schemes [2], [13]–[20]. Instead of trusting a third party, users collaborate to protect their privacy.

In this paper, we propose a clustering K-anonymity scheme for location privacy preservation (namely CK) is proposed. CK scheme does not rely on a trusted third party to anonymize the location information of users. In CK scheme, the whole area that all the users reside is divided into clusters recursively in order to get a cloaked spatial temporal boundary (*STB*). The exact location information of the user is replaced by *STB* including K users. The user can adjust the resolution of location information with spatial or temporal constraints to meet his personalized privacy requirement.

The rest of this paper is organized as follows. The related work is summarized in Sect. 2, The system model is described in Sect. 3. The clustering K-anonymity algorithm is presented in Sect. 4. The experiment and result analysis are given in Sect. 5. Finally, we conclude the paper in Sect. 6.

2. Related Work

Many recent research efforts have been done on preserving a users location privacy while interacting with a LBS provider. These studies can be categorized into two different groups: TTP-based approach and TTP-free approach.

In order to protect the location information of mobile users in the context of LBS, Gruteser and Grunwald [4] firstly employed K-anonymity, which is a TTP-based approach. TTP is used to blur the location information of the user. A subject is considered as K-anonymity with respect to location information, if and only if the location information sent from one mobile user is indistinguishable from the location information of at least K-1 other mobile users. Two representative approaches to location anonymization are Cliquecloak algorithm [14] and the Casper system [5].

The Cliquecloak algorithm adopts a customizable K-anonymity model instead of a uniform K. Every user can specify a different K-anonymity value based on his minimum anonymity level and his preferred spatial and temporal

Manuscript received March 18, 2011.
Manuscript revised June 30, 2011.
[†]The authors are with School of Software, Dalian University of Technology, Dalian 116620, China.
[††]The author is with School of Information Science and Engineering, Central South University, Changsha 410083, China.
a) E-mail: f.xia@ieee.org (Corresponding author)
DOI: 10.1587/transinf.E95.D.134

tolerance level in order to maintain the personalized variable privacy requirements. This model can avoid the drawback of a large K-anonymity spatial region, which is an area that encloses the mobile user querying to a LBS server. However, due to the computation overhead of the clique graph, this approach is only able to meet the small K-anonymity requirements of mobile users.

The Casper approach performs the location anonymization using the quadtree-based pyramid data structure, allowing fast cloaking. However, due to the coarse resolution of the pyramid structure and lack of mechanisms to ensure QoS and constrain the size of the cloaking region, the cloaking areas in Casper are much larger than necessary, leading to poor QoS. Other related work includes anonymization of high dimensional relations [15] and extending the concept of K-anonymization via l-diversity [5], t-closeness [6]and m-invariance [9]. We have proposed a TTP-based location privacy protection scheme for pervasive computing in [10], which can achieve personalized K-anonymity.

TTP-based approaches have some limitations: (1) The system relies on a TTP between the mobile users and the LBS providers. (2) TTP is vulnerable to Denial of Service (DoS) attacks because TTP easily becomes the bottleneck. (3) Furthermore, if the TTP is unreliable, the location information may be abused and the users privacy is disclosed.

Due to the shortcomings of the TTP-based schemes, other methods that do not rely on TTP have been proposed. In [13], the first collaborative TTP-free algorithm for location privacy in LBS is proposed. The user aims to select K-1 neighbors to form a centroid including K users, and send to the LBS provider the K perturbed locations including his own. This method does not achieve K-anonymity because the centroid is only used by a single user to identify himself. In addition, due to the noise cancellation, users cannot use this method several times without changing their locations [2]. In [21], a similar peer-to-peer scheme for location privacy is presented. Its main idea is to generate a cloaking area including K users. When the other K-1 users are selected, the mobile user must exchange his real location with others. If one user among the K-1 users is not trusted, the location privacy may be disclosed. In [14], a method based on Gaussian noise addition to compute a fake location is proposed. K-anonymity is adopted too, and the LBS provider is unable to distinguish one user from the rest according to the fake area. Based on the work in [14], the method is extended to support non-centralised communications in [2]. Users have to trust each other because they share their locations. A centroid is computed as the fake location. But one advantage of this methord is that the users real location could be deduced. In [2], Agusti also proposed a TTP-free scheme, it computes the centroid amongst user and other K-1 companions to achieve K-anonymity. The scheme can achieve robust against the collusion of a modular user and a LBS provider, however, it just get spatial anonymity.

Although the existing schemes play important roles to preserve location privacy, location privacy preservation design is still a challenging area in LBS. The major difference between our work and the aforementioned approaches includes the following aspects:

1. In order to guarantee the QoS in LBS, personalization privacy profile is adopted. Every user can specify his temporal and spatial constraint to meet personalization privacy preservation requirement.
2. To overcome the problems brought by TTP, no TTP entity is used in CK scheme. The user acts as an anonymity server and communicates with LBS providers directly.
3. The selection methods of the cluster center can prevent "center-of-cloaked-area" privacy attack, thus, high privacy preservation can achieve.

3. System Model

In this section, we describe the architecture of our location privacy protection system in Fig. 1. Mobile users communicate with LBS providers directly. The user acts as an anonymity server. Cluster algorithms run in the mobile device and the exact location information of a mobile user can be blurred into a cloaked *STB* by the clustering algorithms. The *STB* composed of K users is sent to the LBS provider. Due to lack of the mobile user's exact location information, the LBS provider may send back a list of results to the user. Lastly, the user will select the most optimal result based on his exact location information. The value of K can vary with the anonymity level of each mobile user, and the personalized K-anonymity is achieved. The following six steps in Fig. 1 describe the whole process.

1. Request Composer module of every mobile user sends a message consisting of his temporal tolerances, K, and a LBS request.
2. Location Provider module provides Privacy Protection Engine (PPE) with the exact location information of the user.
3. PPE performs cluster algorithms. The exact location information is replaced by a cloaked area of the cluster where the mobile user locates.
4. The cloaked area is returned to the Request Composer module.
5. The cloaked area and the temporal tolerance as a *STB*

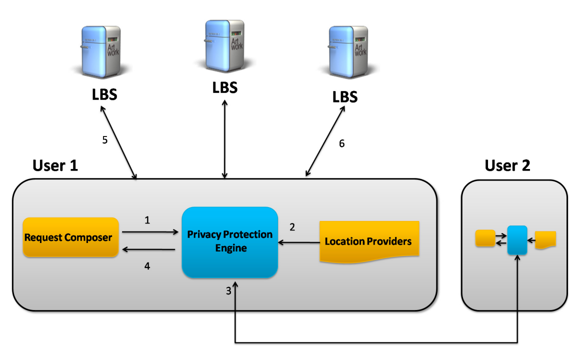

Fig. 1 System model.

is sent to the appropriate LBS providers by the user

6. LBS providers return a list of results to the user during the temporal tolerance. The mobile user will select the optimal result.

3.1 Privacy Threat Model

We consider a model in which users query LBS server directly. We assume that the LBS users are trusted. However, we do not any assumption about the trustworthiness of the location-based service providers.

3.2 User Privacy Personalization Profile

Each user can specify her privacy requirements in a privacy personalization profile. The profile can support temporal, spatial, the required anonymity level and the required cloaked area size.

When a mobile user requests LBS, the PPE will generate a user profile. A user profile is a message defined as follows: $m_{si} \in S$: $\{u_{id}, n_{id}, (x, y), K, C_t\}$. The payload content in m_{si} is omitted. In order to achieve the K-anonymity, the PPE module must find other K-1 users. Thus the cluster algorithm is run to divide the whole area into several clusters. The exact location information in m_{si} is replaced by *STB* of the users cluster so as to achieve K-anonymity. Consequently the user sends a message m_{ti} to LBS. Let $\varnothing(t, s) = [t - s, t + s]$, which extends a numerical value t to a range by amount s. m_{ti} is defined as follows.

$$m_{ti} \in T : \left\{ u_{id}, n_{id}, X : \varnothing\left(cx, \frac{1}{2}W_{STB}\right), Y : \varnothing\left(cy, \frac{1}{2}H_{STB}\right), C_t \right\}$$

3.3 K-1 Companions

In order to achieve the K-anonymity, every user must find K-1 companions for hiding their location from LBS providers. Depending on the number of users into their cover range, we can face the situations as follows [2]:

- `There are no users`: In this case the users cannot proceed with the next steps of the method because they cannot find the required amount of companions.
- `There are less than K users`: In this case the user must extend the cover range repeatedly until the required number of users K is found. If the procedure ends without the required number of companions, the whole process is stopped.
- `There are K users or more`: In this case the K anonymity level is reached because the needed K companies are easily found. In this case in which there are more than K users, say K′, the procedure continues with a number of companions between K and K′.

In our scheme, the clustering algorithm is used to set up and adjust the *STB* which can guarantee include K users.

4. Clustering K-Anonymity Algorithm

The CK algorithm runs on the mobile device of the user to blur his exact information. The notions used in CK algorithm are listed in Table 1.

The whole process of CK algorithm is depicted in Fig. 2. The process is divided into four stages: initialization, cluster construct, cluster adjustment and cluster cloaking finish.

In initialization phase, CK algorithm selects the initial cluster center. The choice of initial center has strong relations with the complexity of clusters construction. In this paper, we adopt the same methods as those in [10]. i.e.,

Table 1 List of notations.

Notation	Description
S	A message set the source sends
T	A message set LBS sends
m_{si}	A message in set S
m_{ti}	A message in set T
u_{id}	User ID
n_{id}	Message ID
K	Anonymity level
cx, cy	Coordinate of center of every cluster
x, y	Coordinate of a user
t, dt	Beginning time and temporal tolerance
X, Y	Coordinate range of *STB*
H_{STB}	Height of *STB*
W_{STB}	Width of *STB*
$B_{STB}(m_{si})$	*STB* of msi
C_t	Content of message
c_i	The i-th cluster
P_{need}	The probability of rebuilding a cluster when a mobile user moves.
N_{ex}	The number of extra nodes without which the cluster can still keep robust.

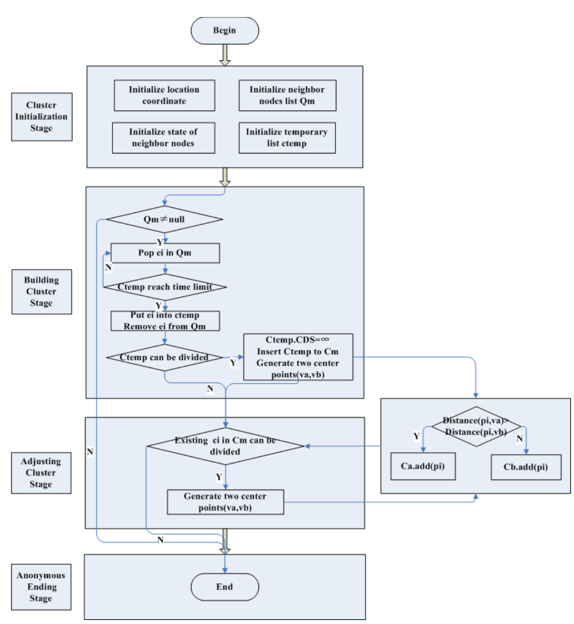

Fig. 2 The process of clustering algorithm.

Algorithm 1 Constructing Cluster.

1 % Q_m collects the messages sent from the mobile clients in the order of their start time.
2 **Repeat**
3 Create a temporary List c_{temp}
4 $e_1 \leftarrow$ Pop the first item in Q_m
5 Add e_1 into c_{temp} with $(e_1.t)$
6 Remove the message e_1 from Q_m
7 **For each** $e_i \in Qm$
8 **If** after puting e_i into c_{temp}, it can reach $min\{c_{temp}.(t + d_t)\} \geq max\{c_{temp}.t\}$
9 **Then** put e_i into c_{temp}
10 Remove the message e_i from Q_m
11 **If** $c_{temp}.divided = true$
12 **Then**
13 $c_{temp}.CDS = \infty$
14 Insert c_{temp} into C_m.
15 Generate two initial center points: v_a, v_b
16 *BinaryCluster*(c_j, v_a, v_b).
17 **While**(true)
18 **If** $\forall c_i \in C_m, \exists c_j.divided \neq false$ **Then**
19 **If** $c_j.divided = true$ **Then**
20 Generate two initial center points: v_a, v_b.
21 *BinaryCluster*(c_j, v_a, v_b).
22 **Else** break
23 **End**
24 **Until** Qm=null

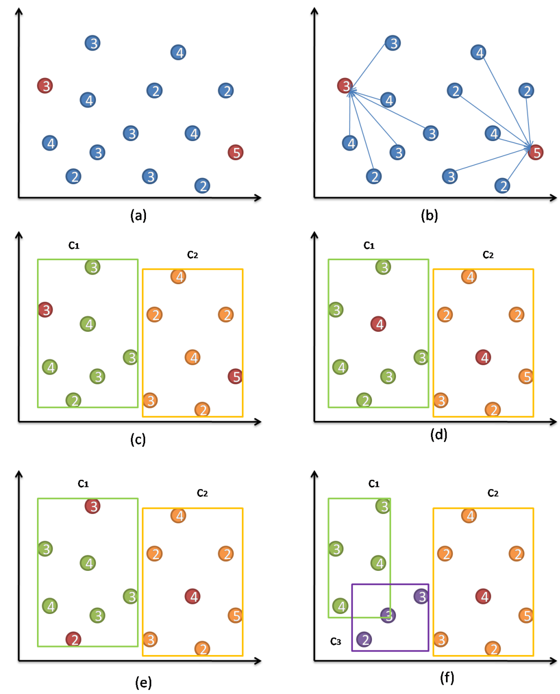

Fig. 3 The illustration of constructing cluster.

After initialization, CK algorithm begins to construct clusters and adjust clusters such as the division and mergence of clusters.

4.1 Constructing Cluster

The user who requests LBS services constructs a cluster including K users. To achieve the K anonymity, after selecting the cluster center, each point is assigned to the nearest cluster according to the distance from it to the center. Then the new center will be calculated and each point is assigned to the nearest cluster again. The above process will repeat until the sum distance between every point and cluster center (*CDS*) converges to a certain range.

The process of constructing clusters is illustrated in Algorithm 1. The initialization phase is in lines 2-4. C_m is defined as a structure which records the cluster identifier, the nodes identifier, the cluster center, the cluster size, $CDS, STB, P_{need}, N_{ex}$, and a variable *divided*. The local variable *divided* represents if the cluster needs division, depending on the value of P_{need} and N_{ex}. When P_{need} is equal to zero and N_{ex} is more than one, *divided* is true.

C_m changes as a new cluster is created or an old cluster is merged. Initially, C_m only contains the initial cluster c_0, if an old cluster c_j can be divided into two new clusters c_a and c_b, then c_j is deleted and c_a and c_b are inserted into C_m. When a user leaves the old cluster and joins another one, both centers will be adjusted.

Figure 3 illustrates the process of constructing cluster. In Fig. 3, the circle represents a user, and the number of the circles indicates the K-anonymity level. First, the initial center of the cluster is selected. The distances from other nodes to the center are computed as shown in (a) and (b). Then, every node is assigned to its nearest cluster and the cluster center is re-computed as shown in (c) and (d) until the center does not change. The procedure is repeated and the cluster C_1 is divided into two clusters as shown in (e) and (f) until any cluster cannot be divided any longer.

4.2 Cluster Adjustment

A mobile user is roaming from one domain to another domain, so clusters may be adjusted when some users join or leave a cluster. Firstly, a cluster does not need adjusting if a user roams in his original cluster. Secondly, a user will be assigned to the nearest cluster if he leaves his home cluster. If his home cluster cannot meet the K-anonymity level, it should be merged with its nearest cluster.

4.2.1 A User's Joining

We denote $k_1, k_2, \ldots, k_m$ as anonymity levels of m users, where $k_1, k_2, \ldots, k_m$ are arranged in the ascending order. When one or multi-users join a new cluster, it should divide the cluster into two clusters. But if either cluster cannot meet the requirement of K-anonymity, cluster adjustment is not successful. Only P_{need} and N_{ex} are re-calculated and users can obtain higher privacy levels because the cluster size is larger than k_m. The process is shown in the Algorithm 2 (Joining part).

4.2.2 A User' s Leaving

When a user leaves the home cluster, four scenarios may occur.

Firstly, if m is bigger than k_m, the cluster can still keep robust. When one user leaves, it holds that $m-1 \geq k_m$, which means the anonymity levels of the rest users can still be met. Therefore, P_{need} and N_{ex} are re-calculated and the cluster does not need rebuilding.

Secondly, if the equations of $m = k_m$ and $k_m > k_{m-1}$ are true, a user whose anonymity level is k_m leaves the cluster. The cluster size will become $m-1$, and $k_1 \leq k_2 \leq \ldots \leq k_{m-1} \leq m-1$ is true. The anonymity levels of the rest users can still be met. Therefore the cluster does not need rebuilding.

Thirdly, if the equation of $m = k_m$ is true, a user whose anonymity level is k_i leaves ($k_i \neq k_m$). Therefore, the cluster size will become $m-1$ and k_m is bigger than $m-1$. At this time, K-anonymity cannot be met because of a users leaving, the cluster should be merged with a neighbor that owns the minimum *STB*. Algorithm 6 is called.

It can be drawn from the second and third scenarios that $P_{need} = \frac{m-1}{m}$.

Last, if the equations of $m = k_m$ and $k_m = k_{m-1}$ are true, any user in the cluster leaves. The cluster size becomes $m-1$. The anonymity level of k_m or k_{m-1} cannot be met. Algorithm 6 will be implemented to rebuild clusters and hence $P_{need} = 1$.

The process is shown in the Algorithm 2 (Leaving part). It searches the neighbor cluster of c_i with the minimum *STB*. Users in c_i will be added into the neighbor cluster c_j, and then c_i is deleted from C_m. At last, it merges c_j into bigger ones.

4.3 Proof of Clustering K-Anonymity Algorithm

In our algorithms, K-Anonymity is considered to be successful if the number of users in a cluster is no less than k_m. Then, we will prove the correctness of our algorithms, which shows that the temporal requirement, spatial requirement and K-anonymity are met.

Let set $M = \{m_{s1}, m_{s2}, \ldots, m_{sn}\} \subset S$, $\forall 1 \leq i \neq j \leq n$, $m_{si}.u_{id} \neq m_{sj}.u_{id}$ and $\forall 1 \leq i \leq n$, $m_{ti} = \langle m_{si}.u_{id}, m_{si}.n_{id}, B_{STB}(M), m_{si}.C_t \rangle$, where $B_{STB}(M) = \{[x_{min}, x_{max}], [y_{min}, y_{max}], [t_{min}, t_{max}]\}$. Then, $\forall 1 \leq i \leq n$, m_{ti} is a valid K-anonymous perturbation of m_{si}, if and only if to the set M, $\forall 1 \leq i \leq n$, $m_{si}.k \leq n$.

Proof: First, we prove that the temporal requirement is met. During constructing a cluster, the expression of $m_{si} \in c_{temp}$ holds where $\forall 1 \leq i \leq p$ and $p \geq n$, if and only if $min\{m_{si}.(t+dt)\} \geq max\{m_{si}.t\}$ holds, which means $\forall 1 \leq i \neq j \leq p$, $[t_{min}, t_{max}] \sqsubseteq (m_{si}.dt,\ m_{sj}.dt) \sqsubseteq B_{STB}(M) \neq \Phi$.

Second, we prove that the spatial requirement is met. During constructing a cluster, the expression of $m_{si} \in c_{temp}$ holds where $\forall 1 \leq i \leq p$ and $p \geq n$ wherever u_i locates, because of $[x_{min}, x_{max}] = \Phi(c_x, \frac{1}{2}W_{STB}) \sqsubseteq B_{STB}(M)$ and $[y_{min}, y_{max}] = \Phi(c_y, \frac{1}{2}H_{STB}) \sqsubseteq B_{STB}(M)$.

Last, we prove that K-anonymity is met. For each $m_{si} \in M$, if $n \geq m_{si}.k$ holds where $\{m_{s1}, m_{s2}, \ldots, m_{sn}\} \subset T$, $\forall 1 \leq i \neq j \leq n$, $B_{STB}(M) = B_{STB}(m_{tj}) = B_{STB}(m_{ti})$ will be got.

Thus, if $m_{si}.k \leq n$ holds, STB will be the valid K-anonymous perturbation of S.

	Algorithm 2 Cluster Adjustment.
Joining:	user p join
	%Find cluster c_i which is the nearest cluster and can reach after p joins
	$c_i.add(p)$
	Update c_i in C_m
	If $0 < c_i.P_{need} \leq 1$ **Then**
	$c_i.P_{need} = 1, c_i.N_{ex} = 1$
	Else $c_i.N_{ex} = c_i.N_{ex} + 1$.
	ClusterAdjustment(c_i).
	Return *true*
Leaving:	user p leaving
	Find the cluster c_i in which p resides.
	If $c_i.N_{ex} > 1$ **Then**
	$c_i.del(p)$.
	ClusterAdjustment(c_i).
	Else If $c_i.N_{ex} = 1$ **Then**
	$c_i.N_{ex} = 0$.
	Adjust $c_i.P_{need} = 1$.
	Else *ClusterMerge*(c_i).
	Update C_m.
Ajustment:	Cluster Adjusting State
	Function *ClusterAdjustment*(c_i)
	$c_i.divided = true$.
	Adjust the *CDS* of c_i.
	Same as lines 17-23 in Algorithm 1 to iteratively divide c_i.
Mergence:	Clusters mergence
	Function *ClusterMerge*(c_i)
	Record the Neighbor Clusters of c_i with the largest N_{ex} in MC_m.
	If $\|MC_m\| \geq 1$ **Then** // Size of MC_m is bigger than 1
	Select the cluster c_j with minimum *STB*.
	Foreach $p_s \in c_i$
	$c_j.add(p_s)$.
	End
	Delete c_i
	ClusterAdjustment(c_j).

5. Experimental Analysis

In this section, we evaluate the performance of CK algorithm in terms of four important performance measures through simulated experiments. (1) Entropy. Entropy is as a measure of uncertainty of a system. Greater entropy means more uncertainty and indicates high privacy preservation. (2) Cloaked area size. This measure gives the average size of the cloaked areas generated by our algorithm. (3) Anonymization success rate. This is a ratio of the number of times that the clustering algorithm can successfully construct the cluster to satisfy the users k-anonymity privacy requirements. (4) QoS. This measure gives the relative temporal and spatial resolution.

We use the VANET (Vehicular Ad-Hoc Network) [11] system that simulates movement of cars and generates requests using the position information. Random Map Generator has been performed to create the geographical distribution of the map and the trace of the vehicles respectively. The number of mobile users is selected from the list {100, 200, 400, 600, 800, 1000}. The beginning time and temporal tolerance of each user randomly assigned. The K-anonymity level is 2 and 5 as the typical. For every combination of the different number of K, 10 data sets recording users' location information are used. Accordingly the sum of sets is $6 \times 2 \times 10$.

5.1 Entroy Analysis

Entropy is as a measure of our uncertainty about a system. Higher entropy means more uncertainty and indicates higher privacy level.

CK algorithm aims to protect location privacy with personalized K-anonymity. K-anonymity represents that the attacked probability of each user is 1/K in a region of K users. For any cluster, $\|C\|$ is defined as the number of users in the cluster and k_m is defined as the maximum K-anonymity level. In CK algorithm, each cluster is built based on $\|C\| \geq k_m$ which indicates that any person can get more privacy than he expects.

Let p_i denote the probability that the i-th user may be regarded as a target user T by attackers. The entropy of all users is defined as $H(p) = -\Sigma p_i \log_2 p_i$. Since it can be obtained that $\|C\| \geq k_m$, $p_1 = p_2 = \ldots = p_m = \frac{1}{\|C\|} \leq \frac{1}{k_m}$ and we can hold that $H(p) = \log_2 \|C\| \geq \log_2 k_m$. In Fig. 4, it show that the entropy of *MN* method is more higher, which indicates that CK algorithm adopting *MN* center choice method can provide more uncertainty, and can reduce the probability of being identified by center-of-cloaked-area privacy attack.

5.2 Anonymization Success Rate

In Fig. 5, the relationship between the number of clusters and the number of users is depicted. The number of clusters is linear with the number of users. Though the number of clusters in these methods is different, the slope of each method is constant, which indicates the size of the cluster is nearly a constant. Hence we can draw the conclusion that the size of clusters does not change with the increment of the number of users, which shows that CK algorithm can build clusters stably.

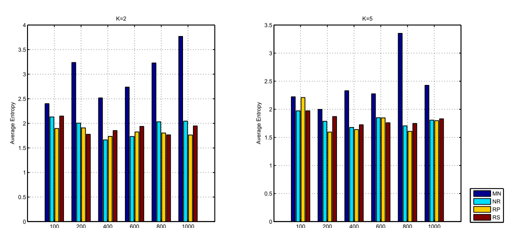

Fig. 4 Entropy analysis.

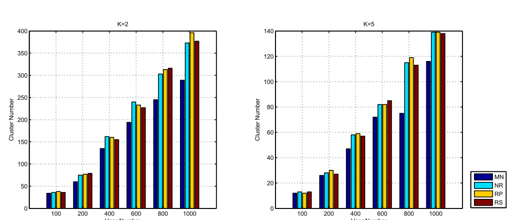

Fig. 5 The stability of cluster.

5.3 QoS Analysis

5.3.1 Relative Temporal Resolution

The relative temporal resolution is a measure of the temporal resolution provided by the cloaking algorithm, normalized by the minimum acceptable temporal resolution [9]. We define the relative temporal resolution as $R_t = \frac{2\Sigma d_t}{\|C\|(C.t_e - C.t_s)}$. The bigger R_t shows that K-anonymity can be achieved within a short time as shown in Fig. 6 and Fig. 7. Because mobile users always move, the clusters need being adjusted to meet the K anonymity. Figure 6 shows the relationship between the time consumption of adjusting clusters and the percentage of joining users, 5%, 10%, 15%, and 20% respectively. The maximum consumption time is no more than 0.15 s. Figure 7 shows the relationship between the time consumption of adjusting clusters and the percentage of leaving users, 5%, 10%, 15%, and 20% respectively. The maximum is less than 0.06 *s*. Figure 6 and Fig. 7 show that our algorithm can adjust the dynamic cluster to meet the

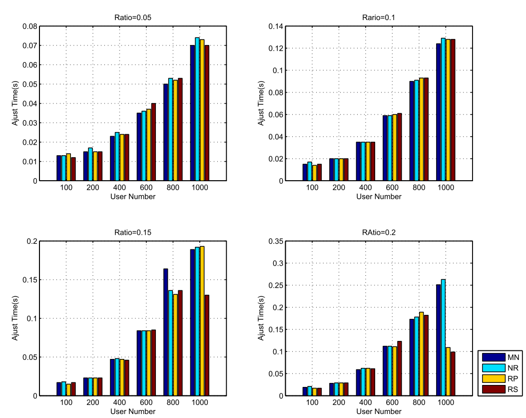

Fig. 6 Time Consumption of Users Joining.

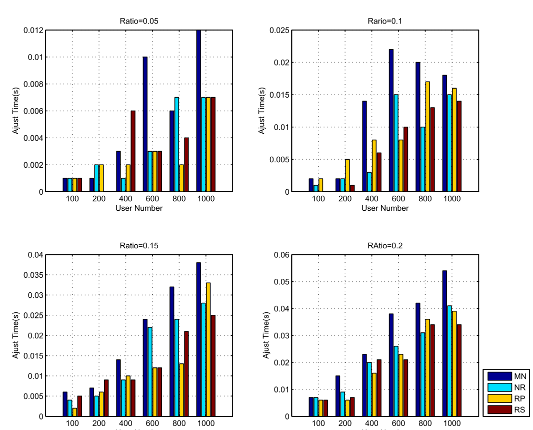

Fig. 7 Time Consumption of Users Leaving.

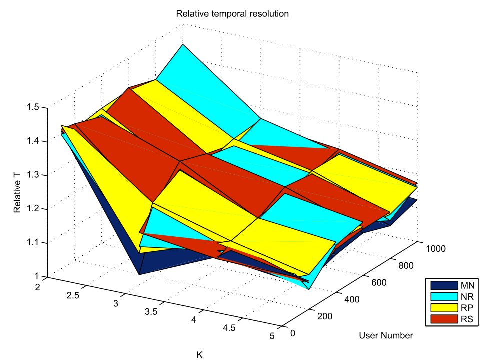

Fig. 8 Relative temporal resolution.

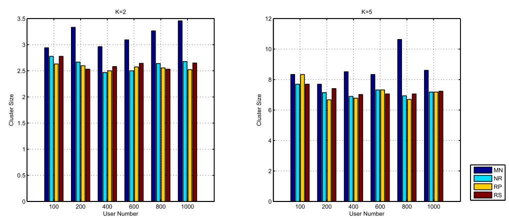

Fig. 9 The average size of clusters.

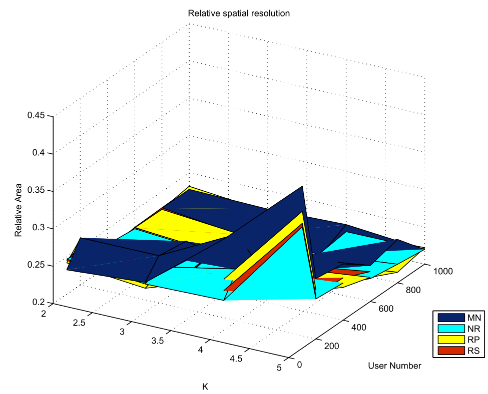

Fig. 10 Relative spatial resolution.

Table 2 Complexity of building clusters.

Algorithm	Complexity
`Nbr-k` [9]	$O(n^2)$
`Local-k` [9]	$O(n^2)$
`ARNN` [9]	$O(n^2)$
`TTP-free` [2]	$O(n^2)$
`CK`	$O(n \lg n)$
`Casper` [5]	$O(n \lg n)$
`HilbertCloak` [4]	$O(n \lg n)$

users temporal requirement, because the time consumpiton is in the range tolerated by users.

Figure 8 shows the relationship between R_t and K. *MN* can provide the smallest R_t. R_t is constant in the three methods of *NR, PR* and *RS*. $R_t \geq 1$ can be guaranteed in the four methods, which shows that our algorithm can provide higher anonymity level than the users expect.

5.3.2 Relative Spatial Resolution

Figure 9 illustrates the relationship between the cluster size and the total number of users. It indicates that the cluster size is almost stable. *MN* is higher than the other methods, but still less than $2k_m$. Therefore, the cluster size can be regarded as a constant. When a user joins, the cluster size is $\|C\|$+1. Once a member leaves, the cluster size is $\|C\|$-1. If two clusters merge, the maximum size is 2$\|C\|$-1 at most.

The relative spatial resolution is a measure of the spatial resolution provided by the cloaking algorithm, normalized by the minimum acceptable spatial resolution. We define the relative spatial resolution as $R_s = S_c/S$, and S_c is defined as the area of a cluster C. S is defined as the total area of all clusters. The relation between R_s and the cluster size is approximately linear. If a smaller region is sent to the LBS, a smaller list of results will be returned. Figure 10 shows R_s is much lower than one in all the methods. Though the *STB* area is small, K-anonymity can still be guaranteed. Therefore, CK algorithm can provide more high QoS.

5.4 Complexity Analysis

We compare the complexity of our algorithm with ARNN [9], Nbr-k and local-k [9], HilbertCloak [4] and Casper [5].

For a cluster containing n users, the complexity of one clustering procedure in CK method is *O(nt)*, which can be simplified to *O(n)*, since the number of iterations t is constant. In the worst case, the complexity of the recursion process is $T(n) = 2T(n/2) + O(n)$. Since *O(n)* is the complexity of each procedure, there must exist a constant satisfying $T(n) \leq 2T(n/2) + an$. Thus, the total complexity can be $O(n \lg n)$.

Table 2 shows the complexity comparisons. It can see that the complexity of CK is $O(n \lg n)$, which is lower than that of Nbr-k, Local-k and ARNN.

6. Conclusions

A clustering K-anonymity scheme (CK) for location privacy preservation has been proposed in this paper. CK can effectively protect location privacy without TTP entity. The User acts as an anonymity server between mobile users and LBS providers, the exact location information of a mobile user can be blurred into a cloaked spatial area by the clustering algorithms. Users can define personalized K-anonymity and temporal tolerance as needed. The primary contributions of this paper are summarized as follows.

1. In order to guarantee the QoS in LBS, personalization privacy profile is adopted, users can specify temporal and spatial constraints to meet personalization privacy preservation requirements.
2. The choice method for the cluster center can prevent center-of-cloaked-area privacy attack, thus, high privacy preservation is achieve. The experimental analysis proves that our approach can provide stringent pri-

vacy guarantees, strong robustness and high QoS.

Acknowledgments

This work was partially supported by the Natural Science Foundation of China under Grants No.60703101 and No.60903153, the Fundamental Research Funds for the Central Universities (DUT10ZD110), and the SRF for ROCS, SEM, China.

References

[1] Y. Sun, Thomas, F.L. Porta, and P. Kermani, "A flexible privacy-enhanced location-based services system framework and practice," IEEE Trans. Mobile Computing, vol.8, no.3, pp.304–321, March 2009.

[2] A. Solanas and M. Antoni, "A ttp-free protocol for location privacy in location-based services," Comput. Commun., vol.31, no.6, pp.1181–1191, 2008.

[3] L. Liu, "From data privacy to location privacy," VLDB '07: Proc. 33rd international conference on Very large data bases, pp.1429–1430, ACM, Sept. 2007.

[4] M. Gruteser and D. Grunwald, "Anonymous usage of location-based services through spatial and temporal cloaking," Proc. 1st International Conference on Mobile Systems, Applications and Services, pp.31–42, 2003.

[5] M. Mokbel, C. Chow, and W. Aref, "The new casper: Query processing for location services without compromising privacy," Proc. 32nd international conference on Very large data bases (VLDB 07), pp.763–774, ACM, Sept. 2007.

[6] M. Mokbel, C. Chow, and W. Aref, "The new casper:a privacy-aware location-based database server," Proc. International Conference on Data Engineering (ICDE'07), 2007.

[7] B. Bamba, L. Liu, P. Pesti, and T. Wang, "Supporting anonymous location queries in mobile environments with privacygrid," Proc. International World Wide Web Conference, WWW, 2008.

[8] B. Gedik and L. Liu, "Protecting location privacy with personalized k-anonymity: Architecture and algorithms," IEEE Trans. Mobile Computing, vol.7, no.1, pp.1–18, TMC, 2008.

[9] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Mobihide: A mobile peer-to-peer system for anonymous location-based queries," Proc. International Symposium on Advances in Spatial and Temporal Databases, Boston, MA, USA, SSTD, 2007.

[10] L. Yao, C. Lin, X. Kong, F. Xia, and G. Wu, "A clustering-based location privacy protection scheme for pervasive computing," The 2010 IEEE/ACM International Conference on Cyber, Physical and Social Computing (CPSCom-2010), pp.719–726, 2010.

[11] G. Ghinita, P. Kalnis, and S. Skiadopoulos, "Prive: Anonymous location based queries in distributed mobile systems," Proceedings of International Conference on World Wide Web (WWW07), pp.1–10, Banff, Alberta, Canada, 2007.

[12] Z. Xiao, X. Meng, and J. Xu, "Quality-aware privacy protection for location-based services," Proc. International Conference on Database Systems for Advanced Applications (DASFAA07), Bangkok, Thailand, 2007.

[13] C. Chow, M. Mokbel, and X. Liu, "A peer-to-peer spatial cloaking algorithm for anonymous location-based services," In: GIS 06: Proc. 14th annual ACM international symposium on Advances in geographic information systems, pp.171–178, ACM, Arlington, Virginia, USA, Nov. 2006.

[14] A. Solanas and M. Antoni, "Privacy protection in location-based services through a public-key privacy homomorphism," In: Fourth European PKI Workshop:theory and practice. Lecture Notes in Computer Science, Heidelberg Palma de Mallorca, Spain, pp.362–368, Springer Berlin, 2007.

[15] P. Samarati, "Protecting respondents identities in microdata release. ieee trans," Knowledge and Data Engineering, vol.13, no.6, pp.1010–1027, Dec. 2001.

[16] M. Duckham and L. Kulit, "Location privacy and location-aware computing," In: Dynamic and Mobile GIS: Investigating Changes in Space and Time, pp.35–52, CRC, 2007.

[17] M. Duckham and L. Kulit, "A formal model of obfuscation and negotiation for location privacy," In: Pervasive Computing, Berlin. Heidelberg, pp.152–170, LNCS, Springer, 2005.

[18] C. Ardagna, M. Cremonini, E. Damiani, D.C. di Vimercati, and Samarati, "Location privacy protection through obfuscation-based techniques," in Data and Applications Security, ed. S. Baker, G. Ahn, pp.47–60, IFIP, LNCS, 2007.

[19] M. Yiu, C. Jensen, X. Huang, and H. Lu, "Spacetwist: Managing the trade-offs among location privacy, query performance, and query accuracy in mobile services," n: IEEE 24th International Conference on Data Engineering ICDE08, pp.366–375, 2008.

[20] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K. Tan, "Private queries in location based services: Anonymizers are not necessary," In: SIGMOD 08: Proc. 2008 ACM SIGMOD international conference on Management of data, pp.121–132, ACM, Vancouver, BC, Canada, June 2008.

[21] C. Chow, M. Mokbel, J. Naps, and S. Nath, "Approximate evaluation of range nearest neighbor queries with quality guarantee," Lect. Notes Comput. Sci., vol.5644, pp.83–301, July 2009.

Lin Yao received B.E. and Master degrees from Harbin Engineering University, China, in 1998 and 2001, respectively, and received Ph.D. degree from Dalian University of Technology, China in 2011. She has been a lecturer in School of Software, Dalian University of Technology (DUT), China, since 2004. She has co-authored one book and over ten scientific papers. Her research interests include pervasive computing, cyber-physical systems (CPS), and wireless sensor networks.

Guowei Wu received B.E. and Ph.D. degrees from Harbin Engineering University, China, in 1996 and 2003, respectively. He was a Research Fellow at INSA of Lyon, France, from September 2008 to March 2010. He has been an Associate Professor in School of Software, Dalian University of Technology (DUT), China, since 2003. Dr. WU has authored three books and over 20 scientific papers. His research interests include embedded real-time system, cyber-physical systems (CPS), and wireless sensor networks.

Jia Wang received B.E. degree from Dalian University of Technology, China. Currently she is a Master student in School of Software, Dalian University of Technology. Her research interests include pervasive computing and wireless sensor networks.

Feng Xia received B.E. and Ph.D. degrees (with honors) from Zhejiang University, China, in 2001 and 2006, respectively. He is an Associate Professor in School of Software, Dalian University of Technology (DUT), China. He is the (Guest) Editor of several international journals. He serves as General Chair, PC Chair, Workshop Chair, Publicity Chair, or PC Member of a number of conferences. He received MDPI Certificate of Editorial Achievement in 2009. He was awarded IEEE/ACM GreenCom-2010 Outstanding Leadership Award in 2010. Dr. Xia has authored/co-authored one book and over 80 papers. His research interests include cyber-physical systems, mobile and social computing, and intelligent systems. He is a member of IEEE and ACM.

Chi Lin received B.E. degree from Dalian University of Technology, China. Currently he is a PH.D Candidate in School of Software, Dalian University of Technology. His research interests include embedded real-time system, cyber-physical systems (CPS), and wireless sensor networks.

Guojun Wang received B.Sc. degree in Geophysics, M.Sc. degree in Computer Science, and Ph.D. degree in Computer Science, from Central South University (CSU), China. He is currently Chairman and Professor of Department of Computer Science at CSU, and Director of Trusted Computing Institute at CSU. He has been an Adjunct Professor at Temple University, USA; a Visiting Scholar at Florida Atlantic University, USA; a Visiting Researcher at The University of Aizu, Japan; and a Research Fellow at The Hong Kong Polytechnic University. His research interests include network and information security, Internet of things, and cloud computing. He is a senior member of CCF, a member of IEEE, and ACM.