A Privacy-Aware Framework for Participatory Sensing

Leyla Kazemi
Information Laboratory, Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781
lkazemi@usc.edu

Cyrus Shahabi
Information Laboratory, Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781
shahabi@usc.edu

ABSTRACT

With the abundance and ubiquity of mobile devices, a new class of applications is emerging, called participatory sensing (PS), where people can contribute data (e.g., images, video) collected by their mobile devices to central data servers. However, privacy concerns are becoming a major impediment in the success of many participatory sensing systems. While several privacy preserving techniques exist in the context of conventional location-based services, they are not directly applicable to the PS systems because of the extra information that the PS systems can collect from their participants. In this paper, we formally define the problem of privacy in PS systems and identify its unique challenges assuming an un-trusted central data server model. We propose PiRi, a privacy-aware framework for PS systems, which enables participation of the users without compromising their privacy. Our extensive experiments verify the efficiency of our approach.

1. INTRODUCTION

With the advent of mobile technology, the area of participatory sensing (PS) [6] has attracted many researchers in different domains such as public health, urban planning, and traffic. The goal is to leverage sensor equipped mobile devices to collect and share data, which can later be utilized for analysis, mining, prediction or any other type of data processing. While many *unsolicited* PS systems exist (e.g., Flickr, Youtube), in which users participate by arbitrarily collecting data, other PS systems are *campaign*-based, which require a *coordinated* effort of the participants to collect a particular set of data that the server requires for any purpose. Some real-world examples of PS campaigns include [15; 20; 2], where users leverage their mobile devices to collect traffic information. In CycleSense [1], bikers document their trajectories along with other data modalities (e.g., pollution, traffic, accidents). In [24], the focus is on participatory texture documentation, where users, in a coordinated effort, aim to collect maximum amount of urban texture information from a set of predefined locations.

However, privacy concerns are the significant barriers to the success of any participatory sensing campaign, which delay the progress of massive deployment of such systems. Consider a scenario where the goal of the PS campaign is to collect pictures/videos from the anti-government riots at different locations of a city with the coordinated effort of the participants. Accordingly, each participant u should query the server for the set of closeby locations from which data (e.g., picture, video, temperature) needs to be collected (termed data collection points or *DC-points*). These are the DC-points that are closer to u than to any other participant. However, u may not be willing to disclose his identity due to safety reasons. An alternative is that u sends his query to a trusted server, known as *anonymizer*. The anonymizer removes the user's ID from the query and forwards the query to the server. However, the server requires u's location information to answer the query. Due to the strong correlation between people and their movements (see [12]), a malicious server can identify u by associating his location information to u. For example, if u issues the query from his home, his identity can be easily revealed by linking the home address to u using the online white page services. Thus, the server can identify a query issuer by associating the query to the location from which the query is issued. We refer to this process as a *location-based* attack. Our goal in this paper is to protect the campaign participants from location-based attacks by disassociating a query from the query location.

Existing privacy preserving techniques have been proposed to address these concerns in the context of location-based services (LBS) [18; 21; 7], one of which is *spatial K-anonymity (SKA)*. The idea behind SKA is that user blurs his location among K-1 other users, such that the probability of identifying the query issuer does not exceed $1/K$, even if in the worst case all the user locations are known to the adversary. The existing studies on cloaking techniques are classified into three categories: centralized, distributed, and peer-to-peer, of which the first two are not applicable to the highly adhoc mobile P2P environments because of their reliance on a fixed communication infrastructure and centralized/distributed servers. Thus, we focus on SKA approaches in P2P environments.

Unfortunately, certain characteristics of a PS campaign distinguish it from conventional LBS, and therefore, prevent a direct adaption of SKA to such systems. One characteristic of a PS campaign is that in order to collect data through a coordinated effort, *all* the participants query the PS server for the closeby DC-points. This is in contrast to LBS which serves millions of users from which any arbitrary subset of them might ask query at a given time and location. We refer to this as the *all-inclusivity* property. Another characteristic of a PS campaign, is that each participant queries for all the DC-points, which are closer to him than to any other participant. Thus, the second property of the PS campaign is that each participant asks a range query from the server which is dependent on the location of other users. We refer to this property as *range dependency.* These two properties, which reveal extra information to the server as compared to the conventional LBS, introduce major privacy leaks to the system. Thus, the system becomes unresilient to location-based attacks.

In this paper, we devise a privacy-aware framework for PS campaigns, which addresses these two major privacy leaks. Our approach, termed **PiRi** has the two following properties: **P**artial-**i**nclusivity and **R**ange **i**ndependence. PiRi is based on the observation that the range queries sent by participants have significant overlaps. Therefore, instead of each participant asking a separate query, only a group of the representative participants ask queries from the server, and share their results with those who have not posed any query. Moreover, instead of each participant submitting a range query, which is dependent on other participants' locations, we propose an adjustment technique that adjusts the range query such that the query becomes independent of the others.

A preliminary version of this work appeared as a short paper in [17], where the privacy problem in PS systems was introduced, and the PiRi approach was briefly discussed. This article subsumes [17] by delving into more details of the proposed approach as well as defining a new metric for quantifying the privacy leak in the PS campaigns, with which we can measure the resilience of our system to location-based attacks. Finally, in this paper we include our experimental studies that show the efficacy of our approach. Our extensive experiments show that our PiRi approach is 98% more resilient to such attacks, while the extra communication cost is tolerable.

The remainder of this paper is organized as follows. Section 2 reviews the related work. In Section 3, we discuss some background studies, formally define our problem, and discuss our system model. Thereafter, in Section 4 we explain our PiRi approach. Section 5 presents the experimental results. Finally, in Section 7 we conclude and discuss the future directions of this study.

2. RELATED WORK

Privacy preserving techniques have been studied in the context of location-based services. One category of techniques [9; 26; 18] focuses on evaluating the query in a transformed space, where both the data and query are encrypted, and their spatial relationship is preserved to answer the location-based query. However, many of the transformation techniques fail to guarantee practical query accuracy. Another group of well-known techniques in preserving users' privacy is the spatial cloaking technique [10; 7; 4; 8; 21; 16], where the user's location is blurred in a cloaked area, while satisfying the user's privacy requirements. An example of spatial cloaking is the spatial K-anonymity (SKA) [25], where the location of the user is cloaked among K-1 other users. While any of the privacy preserving techniques can be utilized to protect the users' privacy, in this paper without loss of generality we use cloaking techniques due to the following reasons: 1) accuracy and 2) popularity in different environments (i.e, centralized, distributed, peer to peer).

Most of the SKA techniques assume a *centralized* architecture [4; 8; 21; 16], which utilizes a trusted third party known as *location anonymizer.* The anonymizer is responsible for first cloaking user's location in an area, while satisfying the user's privacy requirements, and then contacting the location-based server. The server computes the result based on the cloaked region rather than the user's exact location. Thus, the result might contain false hits. The centralized approach has two drawbacks. First, the centralized approach does not scale because the users should repeatedly report their location to the anonymizer. Second, by storing all the users' locations, the anonymizer becomes a single point for attacks. To address these shortcomings, recent techniques [10] focus on distributed environments, where the users employ some complex data structures to anonymize their location among themselves via fixed infrastructures (e.g., base stations). However, because of high update cost, these approaches are not designed for the cases where users frequently move or join/leave the system. Therefore, alternative approaches have been proposed [7] for unstructured peer-to-peer networks where users cloak their location in a region by communicating with their neighboring peers without requiring a shared data structure. In this paper, we employ the P2P spatial cloaking techniques to hide the user's location when querying the PS server.

Despite all the studies about privacy in the context of LBS, only a few work [14; 23; 13] have studied privacy in participatory sensing. In [23], the concept of participatory privacy regulation is introduced, which allows the participants to decide the limits of disclosure. Moreover, in [14; 13], different approaches are proposed, which focus on preserving privacy in a PS campaign during the data contribution, rather than the coordination phase. That is, these approaches deal with how participants upload the collected data to the server without revealing their identity, whereas our focus is on how to privately assign a set of data collection points to each participant. The combination of private data assignment and private data contribution forms an end-to-end privacy-aware framework for the PS systems.

3. PRELIMINARIES

3.1 Background

As discussed in Section 2, we start by using the P2P SKA to address the privacy problem in participatory sensing. Here, we provide a background on the P2P SKA approach.

The idea of P2P SKA approach (see [7]) is that a user communicates with his neighboring peers via multi-hop routing to find at least K-1 other peers. Each user has two privacy requirements: K, and A. K is the minimum number of users in the cloaked region, and A is the minimum area of the cloaked region. After satisfying the K-anonymity requirement, the user extends the cloaked region to A, so that the minimum area privacy requirement is also satisfied. Consequently, the user sends his spatial query along with the cloaked region to the server. The server is equipped with a privacy-aware query processor, which computes a minimal answer set that contains the user's exact result. After receiving the answer set from the server, the user refines the answer set to retrieve the exact result.

Figure 1 illustrates an example of a privacy-aware range query, where user U_1 issues a query with $K = 4$ and a radius of 3 (i.e., $r = 3$). He first collaborates with his neighbors through multi-hop routing to form the cloaked region with 3 other peers. After sending the cloaked region (solid lined rectangle) along with the range query to the server, the query processor determines the minimal answer set (i.e., the answer to the range query for every point in the cloaked region). The reason is that the server does not know which of the 4 users asked the query. According to [7], the minimal answer set includes all the objects inside the region as well as all the objects within the radius of 3 from every point on the edges of the cloaked region (i.e., all the objects inside the dotted line rectangle). This guarantees no missing hits, but probably includes some false hits. Consequently, once U_1 receives the answer set, he can refine it to retrieve all the objects within the radius of 3 from his location.

3.2 Formal Problem Definition

A major focus in the PS campaign is to design a framework in which each participant is assigned to a set of data collection points (DC-points), where data should be collected. In this section, we formally define this problem.

Definition 1 (Participatory Assignment). *Given a campaign $C(P,U) \in R^2$, with P as the set of DC-points, and U as the set of participants, the Participatory Assignment (PA) problem is to assign to each participant $u \in U$ any DC-point $p \in P$, such that p is closer to u than to any other participant in U.*

Note that for simplification, we define the assignment problem for a given snapshot of time and location. That is, we do not assume the participants move during the assignment. This seems intuitive, since participants usually plan their paths from their residential location (e.g., home, office) before starting their movement. Moreover, participants are the current active users of the system willing to participate in the process.

In order to solve the PA problem, a straightforward solution is that each participant sends his location to the server. The server then assigns to each participant the set of DC-points close to him by computing the *Voronoi diagram* of the participants. Figure 2 depicts such scenario. The formal definition of the Voronoi diagram is as follows.

Definition 2 (Voronoi Diagram). *Given an environment $E(U) \in R^2$, with U as the set of participants, the Voronoi diagram of U is a partitioning of E into a set of cells, where each cell V_u belongs to a participant u, and any point $p \in E$ in the cell V_u is closer to u than to any other participants in the environment. Here, the closeness between two points is defined in terms of Euclidean distance.*

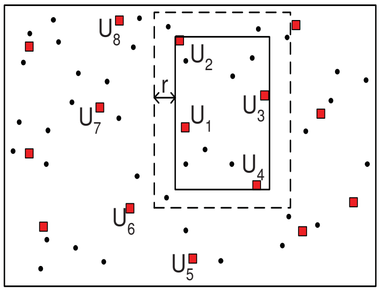
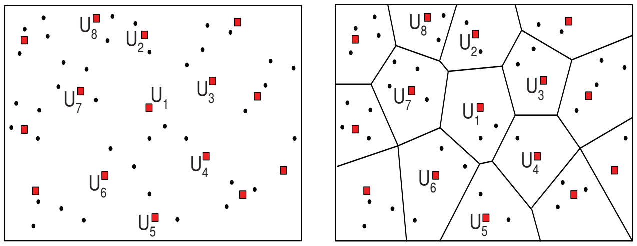


Figure 1: Illustrating an example of privacy-aware range query



a) $E(P,U)$ b) Voronoi Diagram of U

Figure 2: Illustrating the assignment of DC-points to the participants

Once the server computes the Voronoi diagram of the participants, it forwards to each participant u, all the DC-points lying inside the corresponding cell V_u. However, in many scenarios the server is not trusted, and therefore, a participant may not be willing to reveal his identity to the server. Even if the participant hides his identity from the server (i.e., only reveals his location), due to the strong correlation between people and their movements ([12]), a participant can still be identified by his location. In the following, we first formally define our privacy attack. Thereafter, we define the privacy problem.

Definition 3 (Location-based attack). *A location-based attack is to identify a query issuer by associating the query to the query location (i.e., location from which the query is issued).*

Definition 4 (Problem Definition). *The Privacy-Aware Participatory Assignment (PAPA) problem is a variation of the PA problem (Definition 1), in which the goal is to protect participants' identity from location-based attacks.*

3.3 System Model

In this section, we first describe our privacy threat model, and then discuss our system architecture which consists of two entities, participants and the PS server.

Our assumption is that participants trust each other, and do not reveal any sensitive information about their peers. However, they trust neither non-participant nor the PS server. We refer to any such entity as *adversary*. Moreover, the adversary, if needed, can obtain the locations of all participants [11]. The reason is that participants often issue their queries from the same locations (office, home), which can be identified through physical observation, triangulation, etc. In general, since it is difficult to model the exact knowledge available to the adversary, this is a necessary assumption to guarantee that the privacy preserving technique is secure under the most pessimistic scenario. That is, even though the participants' locations might be known to the adversary, it should not pose a threat (i.e., location-based attack) to the system if the system can successfully disassociate the queries from their locations. The adversary is also aware of the anonymization technique which is used by the participants. However, each participant determines his own privacy level, which is only available to himself. Moreover, each user must register with the server, receive the campaign password, and become the campaign participant before communicating with other campaign participants. Finally, in order to guarantee the pseudonymity of participants' location information, each query is assigned with a unique pseudonymous identity, which is totally unrelated to the participants's personal identity.

Our PS server which contains the list of DC-points is equipped with a privacy-aware query processor, which processes the queries issued by the participants. Each participant can determine his privacy level, by specifying two parameters: K, and A. K determines the K-anonymity, and A specifies the minimum resolution of the cloaked region. Each participant is equipped with two wireless network interface cards. One is dedicated to the communication with the PS server through a base station or wireless modem. The other one is dedicated to the P2P communication among the peers through a wireless LAN, e.g., Bluetooth or IEEE 802.11. Also, each participant is equipped with a positioning device, e.g., GPS, which can determine its current location.

4. PIRI APPROACH

As already discussed, to solve the PAPA problem, participants cannot share their locations with the untrustworthy server for the assignment of DC-points. Therefore, the centralized solution to the PA problem is no longer applicable to the PAPA problem. Thus, one baseline solution is that participants communicate among their peers to compute their Voronoi cell. Thereafter, each participant performs a privacy-aware range query to retrieve all the DC-points inside his Voronoi cell. The participant asks such query by applying the P2P SKA technique. That is, the participant blurs his location in a cloaked area among K-1 other peers, and sends the cloaked area, along with a radius r as the range query to the server. Note that the radius r represents the radius of the smallest circle which contains the participant's Voronoi cell. Consequently, the server responds to each participant by sending to him all the DC-points with respect to the range query submitted by any point inside the participant's cloaked area. Finally, the participant obtains the final assigned DC-points by refining the retrieved result from the server.

However, this baseline approach has major privacy leaks, which originates from the two characteristics of a PS campaign: all-inclusivity and range dependency. These properties give enough information to the server with which the server can easily identify each participant by linking his query to the query location. This gets even easier, if the server knows the exact locations of all the participants. The reason is that on one hand the server receives a set of query regions, and on the other hand, the server has the query locations. Each query region overlaps with a set of participants, one of which have issued the query. Therefore, the server can associate the query to its location by solving a matching problem between these two sets of data. As a result, the more information the server has, the more correct matches it can find between the queries and query locations. Consequently, the baseline approach is not applicable to our PAPA problem.

Our PiRi approach overcomes the drawbacks of the baseline approach by preventing these privacy leaks. The intuition is to avoid sharing any extra information with the server, as compared to conventional LBS, such that the adversary cannot use the gathered data in the server to compromise the system. Hence, our algorithm has two major steps. The first step is *Query Formation*, where each participant computes his Voronoi cell in a distributed fashion, and forms his cloaked region. In this step, an adjustment technique is applied to the query, which guarantees the range independency. In Section 4.1, we explain this step in more details. In the second step, *Query Selection* (Section 4.2), a voting mechanism is devised to select the set of representative participants, whose cloaked regions should be sent out to the server. These query results will later be shared with the rest of the participants. This step prevents the all-inclusivity leak.

4.1 Query Formation

To solve the PAPA problem, a set of DC-points those inside his Voronoi cell, should be assigned to each participant. This indicates that each participant should first compute his Voronoi cell to form the spatial range query. Thereafter, by employing the P2P SKA technique, the participant forms a privacy-aware range query. However, the problem is that the range query is dependent on the size of the participant's Voronoi cell (range dependency), which is a potential for information leak. Therefore, at this phase, we adjust the size of the range query, such that the privacy-awareness of the range query is guaranteed. In the following, we first briefly explain how the Voronoi cell of every participant is computed. Subsequently, we explain the cloaking step.

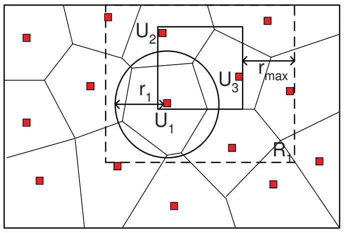


Figure 3: Illustrating an example of query formation for a single participant

In order to compute the Voronoi diagram, any distributed algorithm for Voronoi diagram computation can be applied [5; 3; 22]. In this paper, we employ the technique from [3], called *Completely Cooperative (CC)*. In order to compute the Voronoi cell of a participant, the CC approach has two major steps: 1) finding the Voronoi neighbors for the participant, and 2) computing the boundary of the cell by solving the geometric intersection of bisectors between the participant and the neighbors. The idea behind the CC approach is that instead of participants sending out queries to the network for discovering their Voronoi neighbors, the neighbors inform each other about any potential Voronoi neighbor. Once the Voronoi neighbors of a participant are identified, the participant computes his Voronoi cell by intersecting the bisectors of the neighbors.

Figure 5 depicts the pseudo-code for the query formation step. After the Voronoi cell computation, every participant u forms a spatial range query, which contains the Voronoi cell, along with a cloaked region to send out to the server. That is, u computes a radius r_u, which is the radius of the smallest enclosing circle of his Voronoi cell (line 2). This forms the spatial range query. Figure 3 depicts an example of the query formation for the participant U_1 of Figure 2, where U_1 computes his Voronoi cell, and the radius r_1, respectively. Thus, the circle with radius r_1 is the smallest enclosing circle of U_1's Voronoi cell. Next, as stated in line

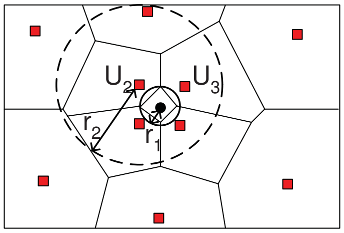


Figure 4: Illustrating an example of Range dependency

```
$QueryFormation$ (participant $U_i$)
01. let $V_i$= Voronoi cell of $U_i$;
02. let $r_i$ = radius of smallest enclosing circle of $V_i$;
03. let $r_{max} = 0$;
04. let $CLR_i$ = cloaked region of $U_i$;
05. for each peer $p$ inside $CLR_i$
06.     let $r_j$ = radius of smallest enclosing circle of $V_j$;
07.     let $r_{max}$ = max($r_{max}, r_j$);
08. return ($CLR_i$,$r_{max}$);
```

Figure 5: Query Formation algorithm

4 of Figure 5, the participant, using the technique explained in Section 3.1, forms a cloaked region, in which his location is blurred among K-1 other peers (the solid lined rectangle in Figure 3).

Consequently, the participant U_i can send the cloaked region along with the radius r_i to the server to retrieve all the DC-points, which lay inside his Voronoi cell. The problem is that each of the K participants in the cloaked region, termed *local peer*, has a different Voronoi size, and therefore, a different r is associated with each. Considering an extreme scenario where the server knows the locations of the participants, it also knows their Voronoi cells and therefore, the radius r for each of them. Consequently, the server can easily identify the query issuer (i.e, the set of all participants in the cloaked region with radius r). Figure 4 depicts such scenario, where U_1 (black-filled circle) cloaks himself with U_2, and sends the cloaked region along with radius r_1 to the server (see the size of r_1 as compared to r_2). The server, knowing the location of the participants, and hence their Voronoi cells (i.e., r_1, and r_2), relates the query with radius r_1 to its query location (i.e., the location of a participant with the Voronoi cell of the same radius).

In order to avoid the range dependency leak, each participant U_i should not only cloak his location among K-1 other peers, but also cloak his range query among that of the other K-1 peers. In other words, instead of forming his range query with radius r_i, the participant forms his query with radius r_{max}, where r_{max} is the maximum radius among all the K peers inside the cloaked region (lines 5-8 in Figure 5). This guarantees the K-anonymity at all times. In Figure 3, R_1 (the dotted line rectangle) shows the *query region* formed by r_{max}.

4.2 Query Selection

Once all participants formed their query region, they can send it out to the server. Since the server is receiving queries from all participants, it can utilize the gathered information (i.e, query regions) from all participants to attack the system (all-inclusivity leak). Figure 6 illustrates such scenario. For simplicity, we assume that only users $U_{1..3}$ participate in the campaign. The figure shows that U_1 cloaks himself with U_2. Similarly, U_2 forms a cloaked region with U_1. Consequently, both U_1 and U_2 form identical query regions. The figure also depicts that U_3 cloaks himself with U_1. Accordingly, the server can easily identify U_3 by relating it to the query region R_3, since U_3 appears only once (i.e., R_3) in all the three submitted query regions to the server. This indicates the more participants submit queries to the server, the more information server has to infer the participants' identities. Our algorithm attempts to prevent this leak by minimizing the number of queries submitted to the server, while assigning the closeby DC-points to *every* single participant.

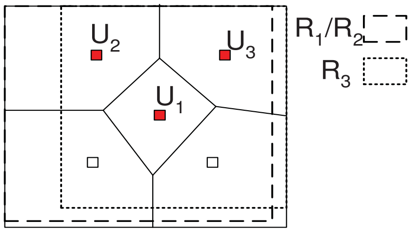


Figure 6: Illustrating an example of all-inclusivity leak

In order to address this issue, we observe that there is a large overlap among the query regions of the participants. Therefore, by receiving the result from the server, one can share his result with all the peers whose Voronoi cells lay completely inside his query region. The question is how to select the group of representative participants. To answer this question, we should solve the following optimization problem.

Definition 5 (V-Cover). *Given a campaign $C(P, U) \in R^2$, with P as the set of DC-points, and U as the set of participants, let R and V be the set of query regions and Voronoi cells for the set U, respectively, where R_i corresponds to the query region for user U_i, and V_i is the Voronoi cell for U_i. The V-Cover problem finds a set $W \subseteq R$ that covers the entire set V with minimum cardinality.*

We now prove that the V-Cover problem is NP-hard by reduction from the minimum set cover problem. First, we state the minimum set cover problem.

Definition 6 (Minimum Set Cover). *Let S=$\{s_1$,s_2, ...,$s_m\}$ be a collection of finite sets, s_i's, whose elements are drawn from a universal set U (i.e., $\bigcup_{i=1}^{m} s_i = U$). Minimum set cover finds a set C with minimum cardinality where $C \subseteq S$ and $\bigcup_{s \in C} s = U$.*

For example, assume U=$\{1, 2, 3, 4, 5\}$ and S=$\{s_1, s_2, s_3, s_4\}$, where s_1=$\{1, 2, 3\}$, s_2=$\{2, 4\}$, s_3=$\{3, 4\}$, and s_4=$\{4, 5\}$. The minimum set cover is C=$\{s_1, s_4\}$. The minimum set cover problem is NP-hard. Consequently, the following theorem is entailed.

Theorem 1. *The V-Cover problem is NP-hard.*

Proof. We prove the theorem by providing a polynomial time reduction from minimum set cover problem. Towards that end, we prove that given an instance of the minimum set cover problem, denoted by I_s, there exists an instance of the V-Cover problem, denoted by I_v, such that the solution to I_s can be converted to the solution of I_v in polynomial time. Consider a given I_s having U as the universal set, S={s_1,s_2,...,s_m} where $s_i \subseteq U$, and $\bigcup_{i=1}^{m} s_i = U$. To solve I_s, we select a set $C \subseteq S$, with minimum cardinality, to cover all the elements in U. Correspondingly, to solve I_v, we look for a $W \subseteq R$, with minimum cardinality, such that all the Voronoi cells in V are covered with the query regions in W. Therefore, we propose the following mapping from I_s components to I_v components to reduce I_s to I_v. Suppose the universal set U corresponds to the set of Voronoi cells V. The intuition behind this mapping is that with I_s we want to cover each element in U and accordingly we aim to cover all the Voronoi cells in V. Each $s_i \in S$ is mapped to a query region $R_i \in R$ as selection of sets in I_s corresponds to selecting the query regions in I_v. We next explain each mapping in detail.

For mapping S to R, we assume there exists a query region $R_i \in R$ corresponding to $s_i \in S$. Next, we assume a Voronoi cell $V_j \in V$ exists corresponding to $U_j \in U$. V_j is covered by $R_i \in R$ (i.e., it falls completely inside R_i) if and only if $U_j \in s_i$. It is easy to observe that if the answer to I_v is the set W, the answer to I_s will be the set C=$\{s_i | R_i \in W\}$. This completes the proof. □

According to the above theorem, we can employ any heuristic that solves the set cover problem to solve the V-Cover problem. One of the well-known approaches for solving the set cover problem is a greedy algorithm which is based on the following heuristic: at each stage of the algorithm, pick the set with the largest number of uncovered elements [19]. Consequently, in order to solve the V-Cover problem, during each step of the algorithm, we should pick a representative participant whose query region covers the largest number of uncovered Voronoi cells from V. However, this approach is applicable only in a centralized structure, where a global knowledge of the environment is available. In the V-Cover problem with a distributed architecture, each participant only has knowledge about his local peers and their Voronoi cells. Therefore, making globally optimal choices (i.e., picking the query region, which covers the largest number of Voronoi cells) at every step is nontrivial, and also costly.

To address this issue, our goal is to extend the greedy heuristic to support the distributed architecture. Hence, we implement a voting mechanism, so that the participants agree locally among their neighbors on selecting a set of representatives. That is, each participant picks a peer from the set of his local peers, based on the *score value* associated to them. Intuitively, the score value captures how significant a participant is in representing other peers, which is defined based on 1) the number of local peers covered[1] by his query region (K), and 2) the number of query regions covering each of his local peers. According to (1), a participant with large query region (i.e., large K) is assigned with a high score value. However, as (2) suggests, the number of query regions that cover each of those local peers also affects the score value. Consider the example of Figure 6, where the query region of each of the three participants covers two peers. However, as the figure shows, R_3 is the only query region covering U_3, and therefore a higher score should be assigned to U_3.

```
    $QuerySelection$ (participant $U_i$)
01. let $R_i$ = query region of $U_i$;
02. let $CR_i$ = set of container regions of $U_i$;
03. let $sum_k = 0$;
04. let $score_i = 0$;
05. let $max\text{-}score_i = 0$;
06. let $rep$ = null;
07. for each peer $U_j$ inside $R_i$
08.     send $K_i$ to $U_j$;
09. for each container region $R_j \in CR_i$
10.     let $U_j$ = owner of the region $R_j$;
11.     let $K_j$ = cloaking parameter for $U_j$;
12.     $sum_k = sum_k + K_j$;
13. for each container region $R_j \in CR_i$
14.     let $score_j^i = K_j / sum_k$ * 100;
15.     send $score_j^i$ to $U_j$;
16. for each peer $U_j$ inside $R_i$
17.     let $score_i = score_i + score_i^j$ ;
18. for each peer $U_j$ inside $R_i$
19.     send $score_i$ to $U_j$;
20. for each container region $R_j \in CR_i$
21.     if $score_j > max\text{-}score_i$
22.         $max\text{-}score_i = score_j$;
23.         $rep = U_j$;
24. return ($rep$);
```

Figure 7: Query Selection algorithm

The pseudo-code of our algorithm is shown in Figure 7. We explain the details of the voting mechanism with the example of Figure 2, where only the participants $U_{1..8}$ are the active users. The voting mechanism starts by assigning the score values to the participants. The score value for each participant is determined by his local peers based on the importance of the participant to any of them. Consequently, each participant computes the final score by summing up all the partial scores he receives from the local peers. The algorithm starts by each participant sending his cloaking parameter K to all his local peers (lines 7-8 of Figure 7). Table 1 depicts the value K for each participant U_i along with the set of peers that his query region R_i contains. For example, U_1 forms a 3-anonymous query, and his query region, R_1, contains U_1, U_2, and U_3. Therefore, U_1 sends the value $K = 3$ to both U_2 and U_3. Accordingly, every participant receives the parameter K with respect to all query regions in which he resides (termed *container regions*). Table 2 illustrates the container regions for every participant (e.g., R_1, R_2, and R_7 are the container regions for U_2).

Subsequently, each participant assigns a partial score value to all his container regions, based on their K value, such that regions with larger K values are assigned with higher scores. Note that the sum of the scores that each participant gives to his container regions is normalized to 100 (lines 9-15 of Figure 7). For example, as Table[2] 2 depicts, U_3 assigns score value of 50 to both of his container regions R_1 and R_3, since both have $K = 3$. Thereafter, each participant computes his final score by summing up all partial scores he receives from his local peers (lines 16-17 of Figure 7). The final scores of the users are shown in the last column of Table 1. As the table shows, U_1 receives the scores $\{25, 37, 50\}$ from his peers $\{U_1, U_2, U_3\}$ respectively, and therefore his final score adds up to 112.

Finally, every participant sends his final score to all his local peers. By receiving the final scores of the container regions, each participant U_i votes for the container region with the maximum score (lines 18-23 of Figure 7). Note that for container regions with equal scores, as tie breaker, the partici-

[1] Henceforth, for brevity we use the expression *covering a peer* to refer to *covering the Voronoi cell of that peer.*

[2] For simplifications, scores are rounded, and therefore, sum of the scores might not add up to 100.

User	Query Region	K	Users	Score
U_1	R_1	3	$\{U_1,U_2,U_3\}$	25+37+50=112
U_2	R_2	2	$\{U_2,U_8\}$	25+28=53
U_3	R_3	3	$\{U_1,U_3,U_4\}$	25+50+50=125
U_4	R_4	3	$\{U_1,U_4,U_5\}$	25+50+37=112
U_5	R_5	2	$\{U_5,U_6\}$	25+40=65
U_6	R_6	3	$\{U_1,U_5,U_6\}$	25+37+60=122
U_7	R_7	3	$\{U_2,U_7,U_8\}$	37+60+42=139
U_8	R_8	2	$\{U_7,U_8\}$	40+28=68

Table 1: Score assignment to the query regions

User	Container Regions	Score Distribution
U_1	$\{R_1,R_3,R_4,R_6\}$	{25,25,25,25}
U_2	$\{R_1,R_2,R_7\}$	{37,25,37}
U_3	$\{R_1,R_3\}$	{50,50}
U_4	$\{R_3,R_4\}$	{50,50}
U_5	$\{R_4,R_5,R_6\}$	{37,25,37}
U_6	$\{R_5,R_6\}$	{40,60}
U_7	$\{R_7,R_8\}$	{60,40}
U_8	$\{R_2,R_7,R_8\}$	{28,42,28}

Table 2: Score distribution among the container regions

pant randomly votes for one. The voting results is shown in Table 3. For example, participant U_4 chooses R_3 among his container regions, since it has the maximum score. According to Table 3, the final representatives are $\{U_3, U_6, U_7\}$. This indicates that only three of the participants should query the server. During the final process of voting, each peer U_i informs the corresponding elected participant by sending him a message, which also includes his radius r_i. The reason for sending the radius r_i is that once the representative receives the result from the server, he would know which part of the result set belongs to U_i (the representative already knows U_i's location during the SKA process). Once the query is issued, the representative filters the result on behalf of every local peer, and sends them the corresponding result.

5. PERFORMANCE EVALUATION

We conducted several simulation-based experiments to evaluate the performance of our proposed approaches. Below, first we discuss our experimental methodology. Next, we present our experimental results.

5.1 Experimental Methodology

We performed three sets of experiments. With the first set of experiments, we evaluated the scalability of our proposed technique. For the rest of the experiments, we evaluated the impact of the participant's privacy requirement and the transmission range on our approach. With these experiments, we used two performance measures: 1) communication cost, and 2) privacy leak. We measured the communication cost of our approach in terms of number of messages incurred by our algorithms per each participant. In order to measure the privacy leak, we defined a new metric for quantifying the privacy leak in the PS campaigns.

We propose a new privacy leak (PL) metric to determine how successful the server is in associating the submitted queries to the query locations. We assume the worst case scenario, where the server knows the locations of the participants. Consequently, on one hand the server receives a set of query regions (R), and on the other hand, the server has the query locations (L). Each query region overlaps with a set of participants, one of which have issued the query. Therefore, the server can associate the query to its location by solving a matching problem between these two sets of data. Accordingly, a bipartite graph is formed with vertices composed of two disjoint sets L and R, where an edge connects L_i to R_j, if L_i is located inside R_j. Since every R_j is issued by a participant j, finding a maximum bipartite matching assigns every query to exactly one query location. The PL metric which measures the percentage of correct matches between L and R is defined as follows.

User	Vote
U_1	Max$\{R_1(112),R_3(125),R_4(112),R_6(122)\}$:$\mathbf{R_3}$(125)
U_2	Max$\{R_1(112),R_2(53),R_7(139)\}$:$\mathbf{R_7}$(139)
U_3	Max$\{R_1(112),R_3(125)\}$:$\mathbf{R_3}$(125)
U_4	Max$\{R_3(125),R_4(112)\}$:$\mathbf{R_3}$(125)
U_5	Max$\{R_4(112),R_5(65),R_6(122)\}$:$\mathbf{R_6}$(122)
U_6	Max$\{R_5(65),R_6(122)\}$:$\mathbf{R_6}$(122)
U_7	Max$\{R_7(139),R_8(68)\}$:$\mathbf{R_7}$(139)
U_8	Max$\{R_2(53),R_7(139),R_8(68)\}$:$\mathbf{R_7}$(139)

Table 3: Voting result

$$PL = \frac{\text{Number of correct matches between } L \text{ and } R}{|R|} \times 100 \qquad (1)$$

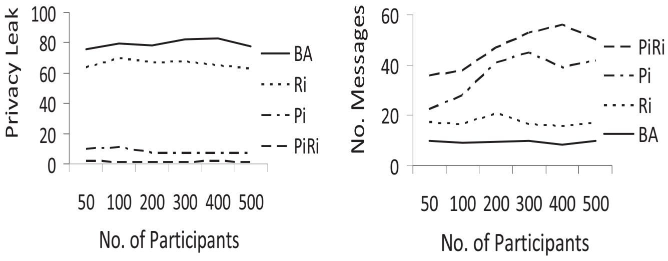
where $|R|$ is the number of query regions.

To compare with a competitive work, since no existing work has been found, we compared our work with the baseline approach (BA), proposed in Section 4. Moreover, in order to separately show the effectiveness of each of the two properties of the PiRi approach, we compared the baseline approach with each of our two proposed techniques corresponding to each property denoted by Pi and Ri. That is, the Pi approach is a variation of PiRi that only addresses the all-inclusivity property (i.e., is only partial-inclusive and does not have the range-independence feature). Similarly, Ri approach is a variation of the PiRi approach, which is only range-independent. We ran 1000 cases, and reported the average of the results.

We conducted our experiments with the objective of collecting a set of photos from 800 locations in part of Los Angeles county. These DC-points were randomly selected. Moreover, our participants dataset includes snapshot locations of 500 mobile users moving in the same area. Since usually a limited number of users participate in a PS campaign, we set the default number of participants to 300, and vary it between 50 to 500. Moreover, we set the transmission range to 250 meters, and vary it between 50 to 250 meters. The degree of anonymity (K) for each participant varies between 5 to 20, with 5 as the default value. The minimum area requirement was set to zero in all cases.

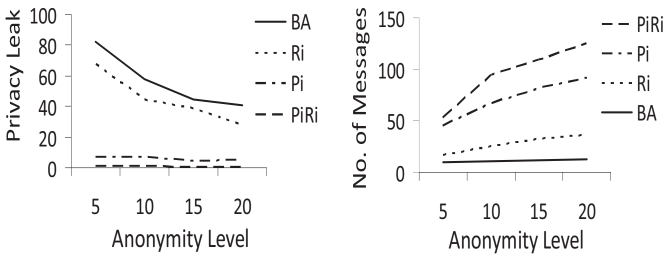
5.2 Scalability

With the first set of experiments, we evaluated the scalability of our PiRi approach by varying the number of par-



(a) (b)

Figure 8: Scalability



(a) (b)

Figure 9: Effect of privacy requirement

ticipants from 50 to 500. As Figure 8a depicts, the privacy leak is not much affected by the number of participants. The reason is that even though the overall information sent out to the server increases as the number of participants grows, the amount of information per participant remains the same, and hence, this does not affect the privacy leak. With the BA approach, the privacy leak is around 75% in all cases, whereas this value is decreased to 2% with the PiRi approach. This shows a significant improvement of PiRi over BA in preserving the privacy. Moreover, as the figure shows, the Pi approach is the next best approach with a huge effect on PL (PL $\simeq$ 10%). This confirms that the query selection has the most significant impact within the PiRi approach. The reason is that Pi focuses on minimizing the number of participants sending out queries, which lowers the chance of an accurate matching between the participants and the queries. The Ri approach, on the other hand, has the least impact on PL decreasing it to 65% as compared to the baseline approach. The reason is that query regions do not usually need a lot of expansion (i.e., adjustment) to meet the privacy requirements. This shows that the impact of the adjustment is not really substantial.

Figure 8b shows the impact of varying the number of participants on the number of messages. As the figure shows, the number of messages slightly increases in most cases. In a denser network, more communication is required among the peers to perform their queries. We observe the largest increase with PiRi and Pi approaches, whereas this only has a slight impact on the BA approach. Moreover, the figure shows that the number of messages in the PiRi approach (35-50 messages per participant) is 3.5 to 5 times more than that of the BA approach. This is because of applying the extra steps in PiRi to preserve the privacy.

5.3 Effect of privacy requirement

In the next set of experiments, we evaluated the performance of our PiRi approach with respect to the participant's privacy requirement K varied from 5 to 20. Figure 9a illustrates a decrease in the privacy leak as K grows. The reason is that an increase in K results in higher privacy-aware queries, and therefore less privacy leak. The PL value in the BA approach decreases from 80% to 45%, whereas this value remains almost fixed for the PiRi approach (i.e., PL $\simeq 0$). Similar to the previous experiments, Pi closely follows PiRi for having the most impact on the privacy leak in all cases. Moreover, Figure 9b shows the effect of varying K on the communication cost. The figure illustrates that the number of messages increases with an increase in K. This is because as K grows, the size of the query regions increases, and therefore, more messages are transmitted in both phases of the PiRi approach.

5.4 Effect of transmission range

In the final set of experiments, we measured the performance of our approach with respect to increasing the transmission range from 50 to 250 meters. As Figure 10 shows, the privacy leak is not affected by varying the transmission range. However, we see a decrease in the communication cost by increasing the transmission range. The reason is that with a higher transmission range, participants can communicate with their peers at a shorter hop distance. This reduces the communication cost. However, the overall information sent out to the server, which might reveal the query issuers' identity, remains the same.

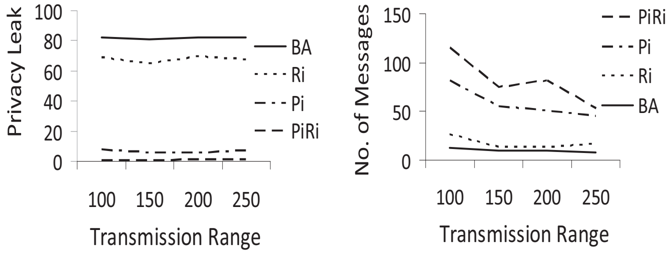
6. DISCUSSION

Our main observation from our experiments is that with an extra cost the privacy is achievable in PS systems. In general, there is a tradeoff between the privacy and the communication overhead. According to the experiments, we observed a significant drop (up to 90%) in the privacy leak of the PiRi approach compared to that of the BA approach, whereas the communication overhead was higher than that of the BA approach. However, we argue that this cost is not a burden to the participants since this is only a one-time cost associated to assigning DC-points to the participants during the planning phase. Moreover, this communication overhead can be interpreted in two ways: a) messaging charges and b) power consumption , of which (a) is negligible since most P2P communications (e.g., Bluetooth) are either free or users pay fixed monthly charges. For the case of (b), since the focus is on planning for participants with fixed locations (i.e., home or office), we can assume that most participants have access to stable power sources. Thus, the battery consumption is less critical than the times where participants are constantly moving.

7. CONCLUSION AND FUTURE WORK

In this paper, for the first time we introduced the problem of privacy-aware participatory assignment in PS systems. We proposed the PiRi approach, a solution to the PAPA problem, which addresses the major privacy leaks in PS system. We also defined a new metric for quantifying the privacy leak in the PS campaigns. With our experiments, we demonstrated the overall efficiency of our approach in preserving the privacy in PS campaigns.

As future work, we aim to extend the problem to the case



(a) (b)

Figure 10: Effect of transmission range

where participants have different constraints (e.g., time, source and destination). Our goal is to incorporate these constraints in the framework yet preserving the privacy of the participants.

Acknowledgment

This research has been funded in part by NSF grant CNS-0831505 (CyberTrust), the NSF Integrated Media Systems Center (IMSC), and unrestricted cash and equipment gift from Google and Microsoft. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

8. REFERENCES

[1] Center for embedded networked sensing (cens). `http://urban.cens.ucla.edu/projects/`.

[2] University of california berkeley, 2008-2009. `http://traffic.berkeley.edu/`.

[3] W. Alsalih, K. Islam, Y. Nú nez-Rodríguez, and H. Xiao. Distributed voronoi diagram computation in wireless sensor networks. In *SPAA'08*, pages 364–364.

[4] B. Bamba, L. Liu, P. Pesti, and T. Wang. Supporting anonymous location queries in mobile environments with privacygrid. In *WWW'08*, pages 237–246.

[5] B. A. Bash and P. J. Desnoyers. Exact distributed voronoi cell computation in sensor networks. In *IPSN'07*, pages 236–243.

[6] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. B. Srivastava. Participatory sensing. In *WSW'06*, pages 117–134.

[7] C.-Y. Chow, M. F. Mokbel, and X. Liu. Spatial cloaking for anonymous location-based services in mobile peer-to-peer environments. In *GeoInformatica'09*.

[8] B. Gedik and L. Liu. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE TMC'08*, 7(1):1–18.

[9] G. Ghinita, P. Kalnis, A. Khoshgozaran, C. Shahabi, and K.-L. Tan. Private queries in location based services: anonymizers are not necessary. In *SIGMOD'08*, pages 121–132.

[10] G. Ghinita, P. Kalnis, and S. Skiadopoulos. Mobihide: A mobilea peer-to-peer system for anonymous location-based queries. In *SSTD'07*, pages 221–238.

[11] G. Ghinita, K. Zhao, D. Papadias, and P. Kalnis. A reciprocal framework for spatial k-anonymity. *Inf. Syst.'10*, 35(3):299–314.

[12] M. C. Gonzalez, C. A. H. R., and A.-L. Barabási. Understanding individual human mobility patterns. *Nature'08*, 453:779–782.

[13] L. Hu and C. Shahabi. Privacy assurance in mobile sensing networks:go beyond trusted servers. In *PerCom 2010 Workshops*.

[14] K. L. Huang, S. S. Kanhere, and W. Hu. Towards privacy-sensitive participatory sensing. In *IEEE PerCom'09*.

[15] B. Hull, V. Bychkovsky, Y. Zhang, K. Chen, M. Goraczko, A. Miu, E. Shih, H. Balakrishnan, and S. Madden. Cartel: a distributed mobile sensor computing system. In *SenSys'06*, pages 125–138.

[16] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias. Preventing location-based identity inference in anonymous spatial queries. *IEEE TKDE'07*, 19(12):1719–1733.

[17] L. Kazemi and C. Shahabi. Towards preserving privacy in participatory sensing (short paper). In *IEEE PerCom'11*.

[18] A. Khoshgozaran, C. Shahabi, and H. Shirani-Mehr. Location privacy: going beyond k-anonymity, cloaking and anonymizers. *Knowledge and Information Systems'10*.

[19] J. Kleinberg and E. Tardos. *Algorithm Design*. 2005.

[20] P. Mohan, V. N. Padmanabhan, and R. Ramjee. Nericell: rich monitoring of road and traffic conditions using mobile smartphones. In *SenSys'08*, pages 323–336.

[21] M. F. Mokbel, C.-Y. Chow, and W. G. Aref. The new casper: query processing for location services without compromising privacy. In *VLDB'06*, pages 763–774.

[22] M. Sharifzadeh and C. Shahabi. Utilizing voronoi cells of location data streams for accurate computation of aggregate functions in sensor networks. *Geoinformatica'06*, 10(1):9–36.

[23] K. Shilton, J. Burke, D. Estrin, M. Hansen, and M. B. Srivastava. Participatory privacy in urban sensing. In *MODUS'08*.

[24] H. Shirani-Mehr, F. Banaei-Kashani, and C. Shahabi. Efficient viewpoint assignment for urban texture documentation. In *GIS'09*, pages 62–71.

[25] L. Sweeney. k-anonymity: a model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.'02*, 10(5):557–570.

[26] M. L. Yiu, G. Ghinita, C. S. Jensen, and P. Kalnis. Enabling search services on outsourced private spatial data. *VLDBJ'10*, 19(3):363–384.