Wireless Network with Privacy Protection and Location Monitoring

Arjun Deore, B. Mathew and B.K.Lande

* Department, Institute Name
** Department, Institute Name, if any

Abstract- Monitoring personal locations with a potentially untrusted server poses privacy threats to the monitored individuals. To this end, we propose a privacy-preserving location monitoring system for wireless sensor networks. In our system, we design two in network location anonymization algorithms, namely, *resource-* and *quality-aware* algorithms that aim to enable the system to provide high quality location monitoring services for system users, while preserving personal location privacy. Both algorithms rely on the well established k-anonymity privacy concept, that is, a person is indistinguishable among k persons, to enable trusted sensor nodes to provide the aggregate location information of monitored persons for our system. Each aggregate location is in a form of a monitored area A along with the number of monitored persons residing in A, where A contains at least k persons. The resource-aware algorithm aims to minimize communication and computational cost, while the quality-aware algorithm aims to maximize the accuracy of the aggregate locations by minimizing their monitored areas. To utilize the aggregate location information to provide location monitoring services, we use a spatial histogram approach that estimates the distribution of the monitored persons based on the gathered aggregate location information. Then the estimated distribution is used to provide location monitoring services through answering range queries. We evaluate our system through simulated experiments. The results show that our system provides high quality location monitoring services for system users and guarantees the location privacy of the monitored persons.

Index Terms- Location privacy, wireless sensor networks, location monitoring system, aggregate query processing.

I. INTRODUCTION

The advance in wireless sensor technologies has resulted in many new applications for military and/or civilian purposes. Many cases of these applications rely on the information of personal locations, for example, surveillance and location systems. These location-dependent systems are realized by using either identity sensors or counting sensors. For identity sensors, for example, Bat [1] and Cricket [2], each individual has to carry a signal sender/receiver unit with a globally unique identifier. With identity sensors, the system can pinpoint the exact location of each monitored person. On the other hand, counting sensors, for example, photoelectric sensors [3], [4], and thermal sensors [5], are deployed to report the number of persons located in their sensing areas to a server. Unfortunately, monitoring personal locations with a potentially untrusted system poses privacy threats to the monitored individuals, because an adversary could abuse the location information gathered by the system to infer personal sensitive information [2], [6], [7], [8]. For the location monitoring system using identity sensors, the sensor nodes report the exact location information of the monitored persons to the server; thus using identity sensors immediately poses a major privacy breach. To tackle such a privacy breach, the concept of *aggregate location information*, that is, a collection of location data relating to a group or category of persons from which individual identities have been removed [8], [9], has been suggested as an effective approach to preserve location privacy [6], [8], [9]. Although the counting sensors by nature provide aggregate location information, they would also pose privacy breaches. Figure 1 gives an example of a privacy breach in a location monitoring system with counting sensors. There are 11 counting sensor nodes installed in nine rooms R1 to R9, and two hallways C1 and C2 (Figure 1a). The nonzero number of persons detected by each sensor node is depicted as a number in parentheses. Figures 1b and 1c give the

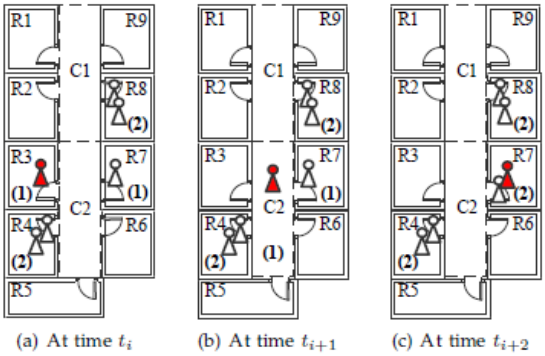


Fig. 1: A location monitoring system using counting sensors.

numbers reported by the same set of sensor nodes at two consecutive time instances ti+1 and ti+2, respectively. If R3 is Alice's office room, an adversary knows that Alice is in room R3 at time ti. Then the adversary knows that Alice left R3 at time ti+1 and went to C2 by knowing the number of persons detected by the sensor nodes in R3 and C2. Likewise, the adversary can infer that Alice left C2 at time ti+2 and went to R7. Such knowledge leakage may lead to several privacy threats. For example, knowing that a person has visited certain clinical rooms may lead to knowing the health records. Also, knowing that a person has visited a certain bar or restaurant in a mall building may reveal confidential personal information. This paper

proposes a privacy-preserving location monitoring system for wireless sensor networks to provide monitoring services. Our system relies on the well established k-anonymity privacy concept, which requires each person is indistinguishable among k persons. In our system, each sensor node blurs its sensing area into a *cloaked area*, in which at least k persons are residing. Each sensor node reports only aggregate location information, which is in a form of a cloaked area, A, along with the number of persons, N, located in A, where $N \geq k$, to the server. It is important to note that the value of k achieves a trade-off between the strictness of privacy protection and the quality of monitoring services. A smaller k indicates less privacy protection, because a smaller cloaked area will be reported from the sensor node; hence better monitoring services. However, a larger k results in a larger cloaked area, which will reduce the quality of monitoring services, but it provides better privacy protection. Our system can avoid the privacy leakage in the example given in Figure 1 by providing low quality location monitoring services for small areas that the adversary could use to track users, while providing high quality services for larger areas. The definition of a small area is relative to the required anonymity level, because our system provides better quality services for the same area if we relax the required anonymity level. Thus the adversary cannot infer the number of persons currently residing in a small area from our system output with any fidelity; therefore the adversary cannot know that Alice is in room R3.To preserve personal location privacy; we propose two in-network aggregate location anonymization algorithms, namely, *resource-* and *quality-aware* algorithms. Both algorithms require the sensor nodes to collaborate with each other to blur their sensing areas into cloaked areas, such that each cloaked area contains at least k persons to constitute a k-anonymous cloaked area. The resource-aware algorithm aims to minimize communication and computational cost, while the quality-aware algorithm aims to minimize the size of the cloaked areas, in order to maximize the accuracy of the aggregate locations reported to the server. In the resource-aware algorithm, each sensor node finds an adequate number of persons, and then it uses a greedy approach to find a cloaked area. On the other hand, the quality-aware algorithm starts from a cloaked area A, which is computed by the resource-aware algorithm. Then A will be iteratively refined based on extra communication among the sensor nodes until its area reaches the minimal possible size. For both algorithms, the sensor node reports its cloaked area with the number of monitored persons in the area as an aggregate location to the server. Although our system only knows the aggregate location information about the monitored persons, it can still provide monitoring services through answering aggregate queries, for example, .What is the number of persons in a certain area? The results show that the communication and computational cost of the resource-aware algorithm is lower than the quality-aware algorithm, while the quality-aware algorithm provides more accurate monitoring services (the average accuracy is about 90%) than the resource-aware algorithm (the average accuracy is about 75%). Both algorithms only reveal k anonymous aggregate location information to the server, but they are suitable for different system settings. The resource-aware algorithm is suitable for the system, where the sensor nodes have scarce communication and computational resources, while the quality-aware algorithm is favorable for the system, where accuracy is the most important factor in monitoring services.

II. SYSTEM MODEL

Figure 2 depicts the architecture of our system, where there are three major entities, *sensor nodes*, *server*, and *system users*. We will define the problem addressed by our system, and then describe the detail of each entity and the privacy model of our system. *Problem definition.* Given a set of sensor nodes $s_1, s_2, \ldots, s_n$ with sensing areas $a_1, a_2, \ldots, a_n$ respectively, a set of moving objects $o_1, o_2, \ldots, o_n$ and a required anonymity level k, (1) we find an aggregate location for each sensor node s_i in a form of $R_i = (Area_i, N_i)$, where $Area_i$ is a rectangular area containing the sensing area of a set of sensor nodes S_i and N_i is the number of objects residing in the sensing areas of the sensor nodes in S_i, such that $N_i \geq k$, $N_i = |\cup_{s_j \in S_i} s_i O_j| \geq k$, $O_j = \{ o_l \mid o_l \in a_j \}$, $1 \leq i \leq n$, and $1 \leq l \leq m$; and (2) we build a spatial histogram to answer an aggregate query Q that asks about the number of objects in a certain area Q. Area based on the aggregate locations reported from the sensor nodes.

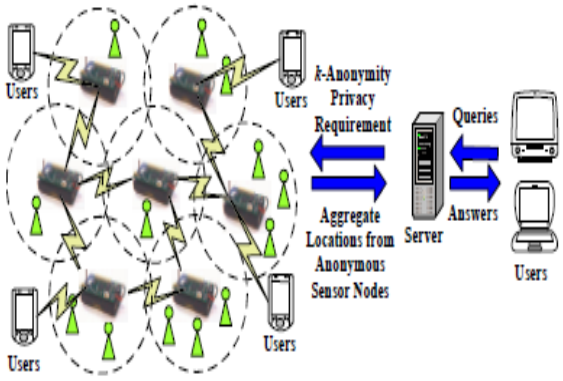


Fig. 2: System architecture.

Sensor nodes: Each sensor node is responsible for determining the number of objects in its sensing area, blurring its sensing area into a cloaked area A, which includes at least k objects, and reporting A with the number of objects located in A as aggregate location information to the server. We do not have any assumption about the network topology, as our system only requires a communication path from each sensor node to the server through a distributed tree [10]. Each sensor node is also aware of its location and sensing area.

Server: The server is responsible for collecting the aggregate locations reported from the sensor nodes, using a spatial histogram to estimate the distribution of the monitored objects, and answering range queries based on the estimated object distribution. Furthermore, the administrator can change the anonymized level k of the system at anytime by disseminating a message with a new value of k to all the sensor nodes.

System users: Authenticated administrators and users can issue range queries to our system through either the server or the sensor nodes, as depicted in Figure 2

Privacy model: In our system, the sensor nodes constitute a trusted zone, where they behave as defined in our algorithm and

communicate with each other through a secure network channel to avoid internal network attacks, for example, eavesdropping, traffic analysis, and malicious nodes [6], [11]. Since establishing such a secure network channel has been studied in the literature [6], [11], the discussion of how to get this network channel is beyond the scope of this paper. However, the solutions that have been used in previous works can be applied to our system. Our system also provides anonymous communication between the sensor nodes and the server by employing existing anonymous communication techniques [12], [13]. Thus given an aggregate location R, the server only knows that the sender of R is one of the sensor nodes within R. Furthermore, only authenticated administrators can change the k-anonymity level and the spatial histogram size. In emergency cases, the administrators can set the k-anonymity level to a small value to get more accurate aggregate locations from the sensor nodes, or even set it to zero to disable our algorithm to get the original readings from the sensor nodes, in order to get the best services from the system. Since the server and the system user are outside the trusted zone, they are untrusted. We now discuss the privacy threat in existing location monitoring systems. In an identity-sensor location monitoring system, since each sensor node reports the exact location information of each monitored object to the server, the adversary can pinpoint each object's exact location. On the other hand, in a counting-sensor location monitoring system, each sensor node reports the number of objects in its sensing area to the server. The adversary can map the monitored areas of the sensor nodes to the system layout. If the object count of a monitored area is very small or equal to one, the adversary can infer the identity of the monitored objects based on the mapped monitored area, for example, Alice is in her office room at time instance ti in Figure 1.

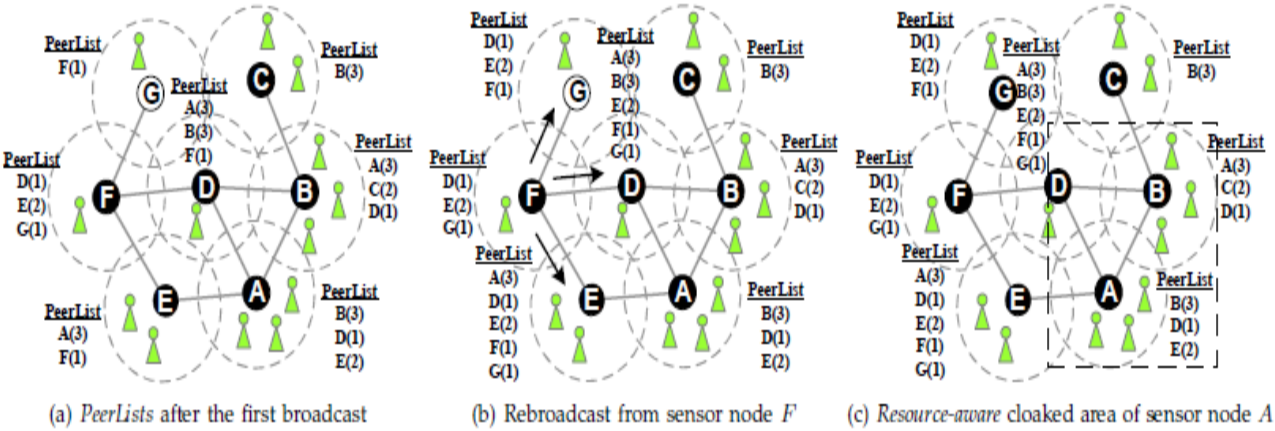
III. LOCATION ANONYMIZATION ALGORITHMS

In this section, we present our in-network resource- and quality-aware location anonymization algorithm that is periodically executed by the sensor nodes to report their k-anonymous aggregate locations to the server for every reporting period.

A. *The Resource-Aware Algorithm*

Algorithm 1 outlines the resource-aware location anonymization algorithm. Figure 3 gives an example to illustrate the resource-aware algorithm, where there are seven sensor nodes, A to G, and the required anonymity level is _ve, k = 5. The dotted circles represent the sensing area of the sensor nodes, and a line between two sensor nodes indicates that these two sensor nodes can communicate directly with each other. In general, the algorithm has three steps.

Step 1: The broadcast step. The objective of this step is to guarantee that each sensor node knows an adequate number of objects to compute a cloaked area. To reduce communication cost, this step relies on a heuristic that a sensor node only forwards its received messages to its neighbors when some of them have not yet found an adequate number of objects. In this step, after each sensor node m initializes an empty list *Peer List* (Line 2 in Algorithm 1), m sends a message with its identity m, ID, sensing area m. Area, and the number of objects located in its sensing area m. Count, to its neighbors (Line 3). When m receives a message from a peer p, i.e., (p:ID; p:Area; p:Count), m stores the message in its *Peer List* (Line 5). Whenever m finds an adequate number of objects, m sends a *notification* message to its neighbors (Line 7). If m has not received the notification message from all its neighbors, some neighbor has not found an adequate number of objects; therefore m forwards the received message to its neighbors (Line 10). Figures 3a and 3b illustrate the broadcast step. When a reporting period starts, each sensor node sends a message with its identity, sensing area, and the number of objects located in its sensing area to its neighbors. After the first broadcast, sensor nodes A to F have found an adequate number of objects (represented by black circles), as depicted in Figure 3a. Thus sensor nodes A to F send a notification message to their neighbors. Since sensor node F has not received a notification message from its neighbor G, F forwards its received messages, which include the information about sensor nodes D and E, to G (Figures 3b). Finally, sensor node G has found an adequate number of objects, so it sends a notification message to its neighbor, F. As all the sensor nodes have found an adequate number of objects, they proceed to the next step.



(a) *PeerLists* after the first broadcast (b) Rebroadcast from sensor node *F* (c) *Resource-aware* cloaked area of sensor node *A*

Fig. 3: The resource-aware location anonymization algorithm (k = 5).

Step 2: The cloaked area step. The basic idea of this step is that each sensor node blurs its sensing area into a cloaked area that includes at least k objects, in order to satisfy the k-anonymity privacy requirement.

To minimize computational cost, this step uses a greedy approach to find a cloaked area based on the information stored in *Peer List*. For each sensor node m, m initializes a set S = {m}, and then determines a score for each peer in its *Peer List* (Lines 13 to 14 in Algorithm 1). The score is defined as a ratio of the object count of the peer to the Euclidean distance between the peer and m. The idea behind the score is to select a set of peers from *Peer List* to S to form a cloaked area that includes at least k objects and has an area as small as possible. Then we repeatedly select the peer with the highest score from the *Peer List* to S until S contains at least k objects (Line 15). Finally, m determines the cloaked area (Area) that is a *minimum bounding rectangle* (MBR) that covers the sensing area of the sensor nodes in S, and the total number of objects in S (N) (Lines 16 to 17). An MBR is a rectangle with the minimum area (which is parallel to the axes) that completely contains all desired regions, as illustrated in Figure 3c, where the dotted rectangle is the MBR of the sensing area of sensor nodes A and B. The major reasons of our algorithms aligning with MBRs rather than other polygons are that the concept of MBRs have been widely adopted by existing query processing algorithms and most database management systems have the ability to manipulate MBRs efficiently. Figure 3c illustrates the cloaked area step. The *Peer List* of sensor node A contains the information of three peers, B, D, and E. The object count of sensor nodes B, D, and E is 3, 1, and 2, respectively. We assume that the distance from sensor node A to sensor nodes B, D, and E is 17, 18, and 16, respectively. The score of B, D, and E is 3/17 = 0:18, 1/18 = 0:06, and 2/16 = 0:13, respectively. Since B has the highest score, we select B. The sum of the object counts of A and B is six which is larger than the required anonymity level k = 5, so we return the MBR of the sensing area of the sensor nodes in S, i.e., A and B, as the resource-aware cloaked area of A, which is represented by a dotted rectangle.

Step 3: The validation step. The objective of this step is to avoid reporting aggregate locations with a containment relationship to the server. Let R_i and R_j be two aggregate locations reported from sensor nodes I and j, respectively. If R_i's monitored area is included in R_j 's monitored area, R_i. Area C R_j. Area or R_j. Area C R_i. Area, they have a containment relationship. We do not allow the sensor nodes to report their aggregate locations with the containment relationship to the server, because combining these aggregate locations may pose privacy leakage. For example, if R_i. Area C R_j. Area and R_i. Area ≠ R_j. Area, an adversary can infer that the number of objects residing in the non-overlapping area, R_j .Area – R_i. Area, is R_j. N – $_{Ri}$. N. In case that R_j. N –R_i. N < k, the adversary knows that the number of objects in the non-overlapping is less than k, which violates the k-anonymity privacy requirement. As this step ensures that no aggregate location with the containment relationship is reported to the server, the adversary cannot obtain any deterministic information from the aggregate locations. In this step, each sensor node m maintains a list R to store the aggregate locations sent by other peers. When a reporting period starts, m nullifies R. After m finds its aggregate location R_m, m checks the containment relationship between R_m and the aggregate locations stored in R. If there is no containment relationship between R_m and the aggregate locations in R, m sends R_m to the peers within R_m. Area and the server (Line 19 in Algorithm 1). Otherwise, m randomly selects an aggregate location R_p from the set of aggregate locations in R that contain m's sensing area, and m sends R_p to the peers within R_p. Area and the server (Lines 21 to 22). In case that no aggregate location in R contains m's sensing area, we find a set of aggregate locations in R that are contained by R_m, R', and N' is the number of monitored persons in R_m that is not covered by any aggregate location in R'. If N' ≥k, the containment relationship does not violate the k-anonymity privacy requirement; therefore m sends R_m to the peers within R_m. Area and the server. However, if N' < k, m cloaks the number of monitored persons of Rm, R_m. N, by increasing it by an integer uniformly selected between k and 2k, and sends R_m to the peers within R_m .Area and the server (Line 24). Since the server receives an aggregate location from each sensor node for every reporting period, it cannot tell whether any containment relationship takes place among the actual aggregate locations of the sensor nodes.

A. *The Quality-Aware Algorithm*

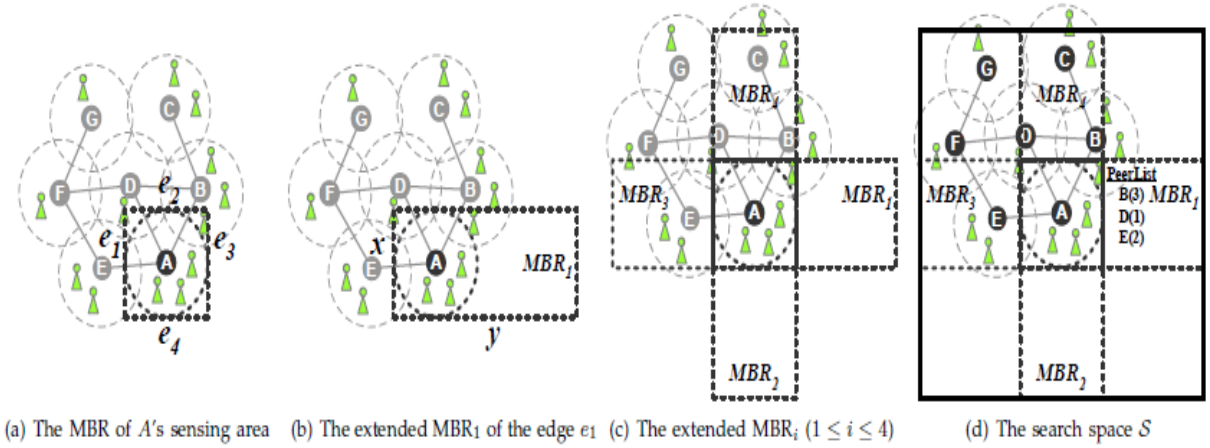
Algorithm 2 outlines the quality-aware algorithm that takes the cloaked area computed by the resource-aware algorithm as an *initial solution*, and then refines it until the cloaked area reaches the minimal possible area, which still satisfies the k-anonymity privacy requirement, based on extra communication between other peers. The quality-aware algorithm initializes a variable *current minimal cloaked area* by the input initial solution (Line 2 in Algorithm 2). When the algorithm terminates, the *current minimal cloaked area* contains the set of sensor nodes that constitutes the minimal cloaked area. In general, the algorithm has three steps.

Algorithm 2 Quality-aware location anonymization

```
1: function QUALITYAWARE (Integer k, Sensor m, Set init_solution, List R)
2: current_min_cloaked_area ← init_solution
   // Step 1: The search space step
3: Determine a search space S based on init_solution
4: Collect the information of the peers located in S
   // Step 2: The minimal cloaked area step
5: Add each peer located in S to C[1] as an item
6: Add m to each itemset in C[1] as the first item
7: for i = 1; i ≤ 4; i ++ do
8:     for each itemset X = {a_1, ..., a_{i+1}} in C[i] do
9:         if Area(MBR(X)) < Area(current_min_cloaked_area) then
10:            if N(MBR(X)) ≥ k then
11:                current_min_cloaked_area ← {X}
12:                Remove X from C[i]
13:            end if
14:        else
15:            Remove X from C[i]
16:        end if
17:    end for
18:    if i < 4 then
19:        for each itemset pair X={x_1,...,x_{i+1}}, Y={y_1,...,y_{i+1}} in C[i]
           do
20:            if x_1 = y_1, ..., x_i = y_i and x_{i+1} ≠ y_{i+1} then
21:                Add an itemset {x_1, ..., x_{i+1}, y_{i+1}} to C[i + 1]
22:            end if
23:        end for
24:    end if
25: end for
26: Area ← a minimum bounding rectangle of current_min_cloaked_area
27: N ← the total number of objects in current_min_cloaked_area
   // Step 3: The validation step
28: Lines 18 to 25 in Algorithm 1
```

Step 1: The search space step. Since a typical sensor network has a large number of sensor nodes, it is too costly for a

sensor node m to gather the information of all the sensor nodes to compute its minimal cloaked area. To reduce communication and computational cost, m determines a *search space*, S, based on the input initial solution, which is the cloaked area, computed by the resource-aware algorithm, such that the sensor nodes outside S cannot be part of the minimal cloaked area (Line 3 in Algorithm 2). We will describe how to determine S based on the example given in Figure 4. Thus gathering the information of the peers residing in S is enough for m to compute the minimal cloaked area for m (Line 4). Figure 4 illustrates the search space step, in which we compute S for sensor node A. Let Area be the area of the input initial solution. We assume that Area = 1000. We determine S for A by two steps. (1) We find the *minimum bounding rectangle* (MBR) of the sensing area of A. It is important to note that the sensing area can be in any polygon or irregular shape.



(a) The MBR of A's sensing area (b) The extended MBR_1 of the edge e_1 (c) The extended MBR_i $(1 \leq i \leq 4)$ (d) The search space S

Fig. 4: The search space S of sensor node A.

In Figure 4a, the MBR of the sensing area of A is represented by a dotted rectangle, where the edges of the MBR are labeled by e_1 to e_4. (2) For each edge ei of the MBR, we compute an MBRi by extending the opposite edge such that the area of the extended MBRi is equal to Area. S is the MBR of the four extended MBRi. Figure 4b depicts the extended MBR_1 of the edge e_1 by extending the opposite edge e_3, where MBR1.x is the length of MBR_1, $MBR_1.y = Area/MBR_1.x$ and Area = 1000. Figure 4c shows the four extended MBRs, MBR_1 to MBR_4, which are represented by dotted rectangles. The MBR of the four extended MBRs constitutes S, which is represented by a rectangle (Figure 4d). Finally, the sensor node only needs the information of the peers within S.

Step 2: The minimal cloaked area step. This step takes a set of peers residing in the search space, S, as an input and computes the minimal cloaked area for the sensor node m. Although the search space step already prunes the entire system space into S, exhaustively searching the minimal cloaked area among the peers residing in S, which needs to search all the possible combinations of these peers, could still be costly. Thus we propose two optimization techniques to reduce computational cost. The basic idea of the first optimization technique is that we do not need to examine all the combinations of the peers in S; instead, we only need to consider the combinations of at most four peers. The rationale behind this optimization is that an MBR is defined by at most four sensor nodes because at most two sensor nodes define the width of the MBR (parallel to the x-axis) while at most two other sensor nodes define the height of the MBR (parallel to the y-axis). Thus this optimization mainly reduces computational cost by reducing the number of MBR computations among the peers in S.

IV. RELATED WORK

Straightforward approaches for preserving users' location privacy include enforcing privacy policies to restrict the use of collected location information [15], [16] and anonymizing the stored data before any disclosure [17]. However, these approaches fail to prevent internal data thefts or inadvertent disclosure. Recently, location anonymization techniques have been widely used to anonymize personal location information before any server gathers the location information, in order to preserve personal location privacy in location-based services. These techniques are based on one of the three concepts. (1) *False locations*. Instead of reporting the monitored object's exact location, the object reports n different locations, where only one of them is the object's actual location while the rest are false locations [18]. (2) *Spatial cloaking*. The spatial cloaking technique blurs a user's location into a cloaked spatial area that satisfy the user's specified privacy requirements [19], [20], [21], [22], [23], [24], [25], [26], [27], [28]. (3) *Space transformation*. This technique transforms the location information of queries and data into another space, where the spatial relationship among the query and data are encoded [29]. Among these three privacy concepts, only the spatial cloaking technique can be applied to our problem. The main reasons for this are that (a) the false location techniques cannot provide high quality monitoring services due to a large amount of false location information; (b) the space transformation techniques cannot provide privacy-preserving monitoring services as it reveals the monitored object's exact location information to the query issuer; and (c) the spatial cloaking techniques can provide aggregate location information to the server and balance a trade-off between privacy protection and the quality of services by tuning the specified privacy requirements, for example, k-anonymity and minimum area privacy requirements [17], [27]. Thus we adopt the spatial

cloaking technique to preserve the monitored object's location privacy in our location monitoring system.

V. CONCLUSION

In this paper, we propose a privacy-preserving location monitoring system for wireless sensor networks. We design two in-network location anonymization algorithms, namely, *resource*- and *quality-aware* algorithms that preserve personal location privacy, while enabling the system to provide location monitoring services. Both algorithms rely on the well established k-anonymity privacy concept that requires a person is indistinguishable among k persons. In our system, sensor nodes execute our location anonymization algorithms to provide k-anonymous aggregate locations, in which each aggregate location is a cloaked area A with the number of monitored objects, N, located in A, where $N \geq k$, for the system. The resource-aware algorithm aims to minimize communication and computational cost, while the quality-aware algorithm aims to minimize the size of cloaked areas in order to generate more accurate aggregate locations. We evaluate our system through simulated experiments. The results show that our system provides high quality location monitoring services (the accuracy of the resource-aware algorithm is about 75% and the accuracy of the quality aware algorithm is about 90%), while preserving the monitored object's location privacy.

REFERENCES

[1] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, .The anatomy of a context-aware application,. in Proc. of MobiCom,1999.

[2] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan,. The cricket location-support system,. in Proc. of MobiCom, 2000.

[3] B. Son, S. Shin, J. Kim, and Y. Her, .Implementation of the realtime people counting system using wireless sensor networks,. IJMUE, vol. 2, no. 2, pp. 63.80, 2007.

[4] Onesystems Technologies, .Counting people in buildings. http://www.onesystemstech.com.sg/index.php?option=comcontent&task=view%&id=10..

[5] Traf-Sys Inc., .People counting systems. http://www.trafsys. com/products/people-counters/thermal-sensor.aspx..

[6] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald, .Privacy-aware location sensor networks,. in Proc. of HotOS, 2003.

[7] G. Kaupins and R. Minch, .Legal and ethical implications of employee location monitoring,. in Proc. of HICSS, 2005.

[8] Location Privacy Protection Act of 2001, http://www. techlawjournal.com/cong107/privacy/location/s1164is.asp..

[9] Title 47 United States Code Section 222 (h) (2), http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname= browse usc&do%cid=Cite:+47USC222..

[10] D. Culler and M. S. Deborah Estrin, .Overview of sensor networks, . IEEE Computer, vol. 37, no. 8, pp. 41.49, 2004.

[11] A. Perrig, R. Szewczyk, V. Wen, D. E. Culler, and J. D. Tygar, .SPINS: Security protocols for sensor netowrks,. in Proc. of MobiCom, 2001.

[12] J. Kong and X. Hong, .ANODR: Anonymous on demand routing with untraceable routes for mobile adhoc networks,. in Proc. Of MobiHoc, 2003.

[13] P. Kamat, Y. Zhang,W. Trappe, and C. Ozturk, .Enhancing sourcelocation privacy in sensor network routing,. in Proc. of ICDCS, 2005.

[14] S. Guo, T. He, M. F. Mokbel, J. A. Stankovic, and T. F. Abdelzaher, .On accurate and ef_cient statistical counting in sensor-based surveillance systems,. in Proc. of MASS, 2008.

[15] K. Bohrer, S. Levy, X. Liu, and E. Schonberg, .Individualized privacy policy based access control,. in Proc. of ICEC, 2003.

[16] E. Snekkenes, .Concepts for personal location privacy policies,. in Proc. of ACM EC, 2001.

[17] L. Sweeney, .Achieving k-anonymity privacy protection using generalization and suppression,. IJUFKS, vol. 10, no. 5, pp. 571. 588, 2002.

[18] H. Kido, Y. Yanagisawa, and T. Satoh, .An anonymous communication technique using dummies for location-based services,. In Proc. of ICPS, 2005.

[19] B. Bamba, L. Liu, P. Pesti, and T. Wang, .Supporting anonymous location queries in mobile environments with privacygrid,. In Proc. of WWW, 2008.

[20] C. Bettini, S. Mascetti, X. S. Wang, and S. Jajodia, .Anonymity in location-based services: Towards a general framework,. in Proc. of MDM, 2007.

[21] C.-Y. Chow, M. F. Mokbel, and X. Liu, .A peer-to-peer spatial cloaking algorithm for anonymous location-based services,. In Proc. of ACM GIS, 2006.

[22] B. Gedik and L. Liu, .Protecting location privacy with personalized k-anonymity: Architecture and algorithms,. IEEE TMC, vol. 7, no. 1, pp. 1.18, 2008.

[23] G. Ghinita, P. Kalnis, and S. Skiadopoulos, .PRIV ´ E: Anonymous location-based queries in distributed mobile systems,. in Proc. Of WWW, 2007.

[24] G. Ghinita1, P. Kalnis, and S. Skiadopoulos, .MobiHide: A mobile peer-to-peer system for anonymous location-based queries,. In Proc. of SSTD, 2007.

[25] M. Gruteser and D. Grunwald, .Anonymous usage of location based services through spatial and temporal cloaking,. in Proc. Of MobiSys, 2003.

[26] P. Kalnis, G. Ghinita, K. Mouratidis, and D. Papadias, .Preventing location-based identity inference in anonymous spatial queries,. IEEE TKDE, vol. 19, no. 12, pp. 1719.1733, 2007.

[27] M. F. Mokbel, C.-Y. Chow, and W. G. Aref, .The New Casper: Query procesing for location services without compromising privacy, . in Proc. of VLDB, 2006.

[28] T. Xu and Y. Cai, .Exploring historical location data for anonymity preservation in location-based services,. in Proc. of Infocom, 2008.

AUTHORS

First Author – Arjun Deore
Second Author – B. Mathew
Third Author – B.K.Lande