Optimizing Placement of Mix Zones to Preserve Users' Privacy for Continuous Query Services in Road Networks

Kamenyi Domenic M., Yong Wang, Fengli Zhang,
Yankson Gustav, Daniel Adu-Gyamfi, and Nkatha Dorothy

School of Computer Science and Engineering
University of Electronic Science and Technology of China
Chengdu, P.R. China
{cla,fzhang}@uestc.edu.cn,
dkamenyi@yahoo.co.uk

Abstract. Location Based Services (LBS) are becoming very popular with today's users. Some of these LBS require users to continuously send requests for services. This lead to leakages of both location and query contents to malicious adversaries. Further, if users are constrained by the nature of the road networks, an adversary can follow their path trajectory with ease. Most of the current privacy preserving solutions focus on temporal and spatial cloaking based methods to protect users' location privacy. However, these solutions are vulnerable when subjected to continuous query environments. In this paper, we propose an optimized solution that preserves privacy for users' trajectory for continuous LBS queries in road networks. First, we deploy a trusted third party architecture to provide anonymity for users as they use LBS services. Second, we utilize mix zone techniques and design two algorithms. The first algorithm, Abstraction Graph (AG), selects a sample of mix zones that satisfy the user desired privacy level under the acceptable service availability condition. The second algorithm, Optimized Decision Graph (ODG), utilizes the generated graph to find an optimal solution for the placement of mix zones through decomposition, chunking and replacement strategies. Finally, we analyze the capability of our algorithms to withstand attacks prone to mix zones and carry out experiments to verify this. The experiments results show that our Algorithms preserve privacy for users based on their privacy and service availability conditions.

Keywords: Location-based Services (LBSs), Privacy Preservation, User Trajectory, Continuous Query, Decision Graphs.

1 Introduction

In recent time, there has been widespread use of LBS through mobile technology with GPS capability [9]. There are two types of privacy concerns in LBS; *location privacy* where the aim is to protect sensitive location from being linked to a specific user and *query privacy* where query is protected from being linked to a user.

H. Motoda et al. (Eds.): ADMA 2013, Part II, LNAI 8347, pp. 323–335, 2013.

Further, query can either be *snapshot* or *continuous* [11,10]. Our concern is to preserve privacy for continuous query that is difficult to achieve. For example, in continuous query, an adversary can follow users' trajectory over Euclidean space and break their security. Worse still, an attacker can easily use the constrained road network setup to follow users' trajectory with ease. However, privacy can be achieved by use of mix-zone frameworks [11]. When users simultaneously enter an intersection (designated as a mix-zone), they change to a new unused pseudonym. In addition, while in a mix zone, they do not send their location information to any location-based application. An adversary tracking a user will not distinguish users entering a mix-zone with those coming out of it.

To illustrate how mix zone work, take an example shown in Fig. 1. vehicle numbers 1, 2 and 3 enter an intersection using road segment A. Vehicle numbers 4, 5 and 6 enter the same intersection through road segment C. Vehicles entering the junction from A can exit through road segment B, C or D. Likewise, vehicles accessing the junction from C can exit either through A, B or D. The adversary cannot distinguish or even pinpoint correctly vehicles leaving this junction.

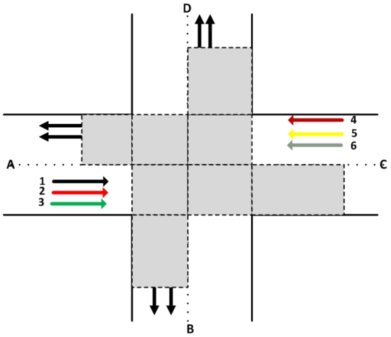


Fig. 1. Non-Rectangular Mix Zone

However, under certain conditions, users' trajectory may be exposed that may lead to three types of attacks. First, *timing attack* [10] occur when a group of users enter and leave a mix-zone within a small difference in their time. Second, *transition attack* [10] occur when an adversary utilize the users' transition of turning (right, left or going straight), if the number entering an intersection simultaneously is not large enough. Finally, *continuous query attack* occur where the speed and transition of a user requesting for continuous service is different from other users. Attacker may utilize these phenomenon to infer the target user.

In this paper, we offer our contributions towards finding a solution to these privacy exposing attacks. First, we deploy a trusted third party architecture of anonymizing servers to preserve users' privacy while accessing LBS as well as maintaining desired service availability. Second, we use mix zone approach to design two algorithms. The first algorithm (Abstraction Graph - AG) select a sample of mix zones that satisfy the user desired privacy and service levels, while the second algorithm (Optimized Decision Graph - ODG) finds an optimal solution for the placement of mix zones through decomposition, chunking and replacement strategies. Finally, we analyze the capability of our algorithms to withstand mix zones attacks and carry out experiments to verify this.

The rest of the paper is organized as follows: In section 2, we present related work followed by system design in section 3. A detailed explanation of the proposed algorithms is given in section 4. Section 5 presents Security and Privacy Analysis with experiments and evaluations in section 6. Finally, in section 7, we conclude with a proposal for future work.

2 Related Work

We categorize related work into two parts. The first part explores recent research on privacy preservation in road networks and the second part considers privacy preservation in road network using mix zone approach.

2.1 Privacy Preservation Techniques in Road Networks

In recent times several techniques have been proposed on how to preserve privacy in road network. One of these techniques is Ting Wangs' [14] et al. *X-Star*. They regard the *attack resilience* and the *query-processing cost* as two critical measures for designing location privatization solutions. However, *X-Star* Framework incurs low anonymization success rate and it's computation cost is quite high. Al-Amin Hossain [4] et al. proposed Hilbert-order based star network expansion cloaking algorithm (*H-Star*) that guarantees K-anonymity under the strict reciprocity condition and increases anonymization success rate by reducing computation overhead. However, this framework does not support continuous location based queries. Further, Joseph T. Meyerowitz [8] et al. developed *CacheCloak*, a system that anonymize a user by camouflaging their current location with various predicted paths. They extended the idea of *path confusion* and *predictive path confusion* to enable *caching* that generates a predicted path for the user. They considered applications that can operate using user's location. However, some applications require more than just the user's current location to operate.

Chi-Yin Chow [2] et al. noted that applying the above techniques directly to the road network environment lead to privacy leakage and inefficient query processing. They proposed "*query-aware*" algorithm designed specifically for the road network environment that takes into account the query execution cost at a database server and the query quality during the location anonymization process. However, the proposed algorithm only works with snapshot locations. Further, Chi-Yin Chow [1] et al. gave a survey on the state-of-the-art privacy-preserving techniques in snapshot and continuous LBS. They noted that protecting user location privacy for continuous LBS is more challenging than snapshot LBS.

2.2 Privacy Preservation Using Mix Zones in Road Networks

One of the recent techniques for preserving privacy in road network is mix zone approach. Balaji Palanisamy [11] et al. proposed *MobiMix* framework that protect location privacy of mobile users on road networks. They provided the formal analysis on the vulnerabilities of directly applying theoretical rectangle mix-zones to road networks in terms of anonymization effectiveness and attack resilience. They proposed use of non-rectangular mix zones. Further, Kai-Ting Yang [15] et al. argues that the concept of continuous location privacy should be transferred to users' path privacy, which are consecutive road segments that need to be protected and proposed a novel M-cut requirement to achieve this.

Xinxin Liu [7] et al. investigated a new form of privacy attack to the LBS system where an adversary reveals user's true identity and complete moving trajectory with the aid of *side information*. They proposed a new metric to quantify the system's resilience to such attacks, and suggested using multiple mix zones as a cost constrained optimization problem. Further, Murtuza Jadliwala [5] et al. studied the problem of determining an optimal set of mix-zones such that the degree of mixing in the network is maximized and the overall network-wide mixing cost is minimized. They modeled the optimal mixing problem as a generalization of the vertex cover problem. In the *Mix Cover*, as it was called, they proposed three approximation algorithms and two heuristics. However, the problem of Continuous Query (C-Q) attacks [10] was not tackled by above research.

3 System Design

3.1 Designing Goals

First, we adopt trusted third party architecture to achieve anonymity. Second, we introduce two terms; 1) *demand* - d for a road segment that represents the average number of users in a road segment (traffic capacity on a segment), and 2) *cost* - c at each vertex that represents the average cost (per user) of mix-zone deployment at that intersection (intersection mixing cost). We use *cost* to select a sample of mix zones to act as the population of promising solution. We then use *demand* to maximize on mix zones entropy in order to confuse the adversary. We optimize this solution using decomposition, chunking and replacement strategies [13]. Finally, we deal with mix zone attacks [11] by deploying an Optimized Decision Graph that achieves greater anonymity and service availability.

3.2 System Architecture

The proposed architecture uses secure communication channel composing of mobile users, anonymizing server and LBS as in Fig. 2. AS consists of 3 components: 1) **Optimizing Decision Engine** that use hierarchical Bayesian Optimization Algorithm - *hBOA* to generate Graphs from road networks that satisfy client service availability and feeling safe conditions. The generated graph and query content are then forwarded to LBS for service. 2) **Repository** that stores generated Graphs that are tagged with time, date and the month they were generated. 3) **Result Refiner** that is responsible for refining the accurate result from candidate ones sent by the LBS according to the knowledge of the client's position.

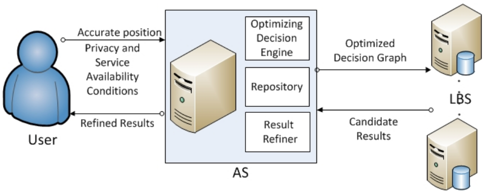


Fig. 2. System Architecture

A user will submit to AS, its position, query content, privacy profile and service availability condition. AS will retrieve and forward to LBS, the relevant Optimized Decision Graph and query content. Based on this information, LBS will figure out candidate results and return them to AS. AS will calculate refined results and send them to client. Updates to AS will be in form of: 1) privacy conditions, 2) new service availability conditions, and 3) new areas to be visited. AS will then use the new updates to generate Optimized Decision Graphs for storage. For a continuous query, AS will continue to receive location information updates from the client until the expiry of the query. Based on the updated client location, AS will continue to retrieve relevant decision graphs that preserve users privacy and update LBS. Further, we treat LBS as un-trusted. We discuss how to deal with attacks by malicious adversary in Section 5.

4 Algorithms

4.1 Preliminaries

The problem of optimizing mix zone placement is NP-Hard as discussed in [7,3]. However, we can have a Linear Programming relaxation of this problem. If LP relaxation has an integral solution then that can also be a solution. For example, let assume that we have a binary decision variable for each edge e and its corresponding vertex v which indicates whether the vertex v is included in the selected population of promising solutions for edge e or not. That is, if included we represent it with binary 1 and if not a 0. Let d_v be the decision variable indicating the demand on a road segment $e \varepsilon E$ and c_v indicating the cost of each vertex $v \varepsilon V$. The linear Programming of this problem can be presented as:

$$\text{Min} \sum_{v\epsilon V} d_v.c_v$$
$$\text{Subject to: } d_e^v du + d_e^u dv \geq d_e^u d_e^v, \forall e \equiv (u,v)\epsilon E$$
$$d_v \geq 0, \forall v\epsilon V$$

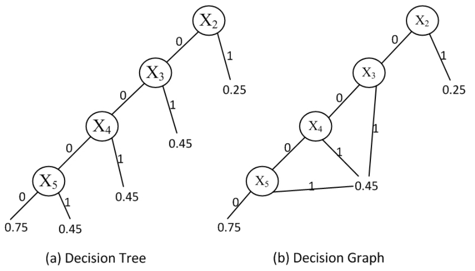
To solve this problem, we model the location map as a directed weighted graph *G(V;E;d;c)*, where V is the set of vertices, E is the set of road segments, d is the *demand* on a road segment given by the average number of users in that road segment, and c is the *cost* at each vertex given by the average cost (per user) of mix-zone deployment at that vertex. Further, two vertices are said to be pairwise connected when there is at least one path connecting them. We introduce a mix-zone to break pairwise connectivity in order to achieve anonymity.

First, We classify the road network by clustering according to traffic in road segments. For example, segments in major roads carry more traffic than roads feeding major roads. We generate a Decision Graph with the top most layer having segments from major roads, followed by other smaller roads. Second, we find an optimized solution on the placement of mix zones to achieve the desired privacy for users as well as acceptable service availability. The ideal situation is to place mix zones in every road intersection. However, this is not practical. The actual intersection cost resulting from a road segment with intersection at v depends on the intersection mixing cost c and the traffic capacity d. We perform a selection that minimizes the intersection mixing cost as well as maximizing on traffic capacity on road segment that affect entropy (to be introduced later).

To achieve optimized solution we propose to use Hierarchical Bayesian Optimization Algorithm *(hBOA)* [13] that captures hierarchical nature of our problem at hand. In this case, higher layers captures traffic on major roads and lower layers that of streets. *hBOA* accomplishes 3 steps; 1) *Decomposition* - in each level the problem is decomposed properly by identifying most important interactions between the problem variables and modeling them appropriately, 2) *Chunking* - partial solutions are represented at each level compactly to enable the algorithm to effectively process partial solutions of large order, and 3) *Diversity maintenance* - alternative partial solutions are preserved until it becomes clear which partial solutions may be eliminated (*niching*). To ensure *decomposition* and *chunking*, we use decision graphs to build Bayesian networks from the selected population of promising solutions and capture local structures to represent parameters of the learned networks. To ensure *diversity maintenance*, we use *restricted tournament replacement (RTR)* [13] to satisfy *niching.*

For example, consider a binary variable X_1 which is conditioned on 4 other binary variables denoted by X_2, X_3, X_4 and X_5. A fully conditional probabilities table for X_1 containing 16 entries (2^4) is generated and after proper decomposition and chunking, we get a Decision Tree and Graph as in Fig. 3. A sequence of splits and merges are done without losing the original meaning and a further reduction on storage space is achieved. We only store values (0.25, 0.45, 0.75).

X_2	X_3	X_4	X_5	P($X_1 = 0 \mid X_2, X_3, X_4, X_5$)
0	0	0	0	0.75
0	0	0	1	0.45
0	0	1	0	0.45
0	1	0	1	0.45
0	0	1	0	0.45
0	0	1	1	0.45
0	1	0	0	0.45
0	1	1	1	0.45
1	1	1	0	0.25
1	1	0	1	0.25
1	0	1	0	0.25
1	0	0	1	0.25
1	0	0	0	0.25
1	1	0	1	0.25
1	1	1	0	0.25
1	1	1	1	0.25



(a) Decision Tree

(b) Decision Graph

Fig. 3. Decision Tree and Decision Graph that encodes the probabilities in the Table

A common metric for evaluating an adversary's uncertainty in finding out the link between a user's old and new pseudonym in a mix zone is calculating its entropy. Consider a sequence of entering/exiting nodes traversing a mix zone i over a period of T time steps, the entropy is given by:

$$H_T(i) = -\sum_i (P_i.log_2 P_i) \tag{1}$$

where P_i are probabilities of possible outcomes. The higher they are, the more the uncertainty. The conditional probability of $X_i = x_i$ given that $\Pi_i = \pi_i$ is:

$$p(x_i|\pi_i) = \frac{p(\pi_i|x_i).P(x_i)}{p(\pi_i|x_i).P(x_i) + p(\pi_i|x_i^c).P(x_i^c)} \tag{2}$$

where $P(x_i)$ is the probability of event x_i occurring; $p(\pi_i|x_i^c)$ is the conditional probabilities of event π_i given that event $P(x_i)$ has not occurred; and $P(x_i^c)$ is the probability of event $P(x_i)$ not occurring. The condition entropy is:

$$H(X_i|\Pi_i) = -\sum_{x_i,\pi_i} p(x_i,\pi_i) log_2 p(x_i|\pi_i), \tag{3}$$

where $p(x_i,\pi_i)$ is the marginal probability of $X_i = x_i$ and $\Pi_i = \pi_i$. For a total number of m mix zones, the Overall Conditional Entropy (OCE) is given by:

$$OCE = \frac{1}{m}\sum_{i=1}^{m} H(X_i|\Pi_i) \tag{4}$$

It therefore follows that for a sample of k mix zones selected to offer user abstraction, their Sample Conditional Entropy (SCE) is thus given by:

$$SCE = \frac{1}{k}\sum_{i=1}^{k} H(X_i|\Pi_i) \tag{5}$$

When constructing optimized placement graphs, we need to measure quality of competing network structures. We do this by calculating the *Bayesian Metrics* (for quality) that is given by *Bayesian-Dirichlet metric (BD)* [13]:

$$BD(B) = p(B)\prod_{i=1}^{n}\prod_{\pi_i}\frac{\Gamma(m'(\pi_i))}{\Gamma(m'(\pi_i)+m(\pi_i))}\prod_{x_i}\frac{\Gamma(m'(x_i,\pi_i)+m(x_i,\pi_i))}{\Gamma(m'(x_i,\pi_i))} \tag{6}$$

and *Minimum description length (MDL) metrics* (for compression) given by *Bayesian information criterion (BIC)* [13]:

$$BIC(B) = \sum_{i=1}^{n}(-H(X_i|\Pi_i)N - 2^{|\Pi_i|}\frac{log_2(N)}{2}) \tag{7}$$

4.2 Abstraction Graph - AG

We propose AG (Algorithm 1) that is generated off-line. Assume that we have n vertices (i.e v = 1, 2 ....., n). Sort vertices v ∈ V in ascending order based on values of *mixcost c* (steps 7 - 12). Starting with the lowest value v_1 , check if there exist a pairwise connection between this vertex and the next closest neighbor in the list and if there is, place a mix zone. Repeat the above until all vertices are covered (steps 16 - 20). To capture highest entropies, we select the segment with maximum traffic capacity and calculate entropy (steps 21 - 25). Let Q represent user's feeling safe value which is associated with conditional entropy. Let K represent minimum acceptable level of service availability that is estimated over the previously recorded service availability requests. If $Q < H(X_i|\Pi_i)$, then we say that the user is safe, otherwise not. We group users according to similarity in their; locations, feeling safe*(*Q) and service availability*(*K)-(step 26). We select a sample of mix zones satisfying minimum service availability condition K and where $(Q < H(X_i|\Pi_i))$ (steps 28 - 32). We store $Z_{(AG)}$ (step 34).

Algorithm 1. Abstraction Graph *(AG)*

```
Require: <Graph G ≡ $(V,E,d,c)$>
Ensure: <Abstraction Graph $Z_{(AG)}$>
 1: Let $P_{(t)}$ := n be the initial population;
 2: let $V'$ := Sorted vertices v ∈ V in ascending order based on values of mixcost c;
 3: let $S_{(t)}$ := selected population of promising solutions;
 4: let $Z_{(AG)}$ := Abstraction Graph;
 5: let mixcost c := Intersection Mixing Cost at any intersection $v$;
 6: Initialize $V'$, $S_{(t)}$ and $Z_{(AG)}$;
 7: for all $v \in V$ do
 8:     for $i = 1$ to $n$ do
 9:         sort vertices $v \in V$ in ascending order based on values of mixcost c;
10:         $V'$ := Sorted Vertices
11:     end for
12: end for
13: let $v_1$ be the least value;
14: add $v_1$ to $S_{(t)}$;
15: Starting at $v_1$
16: for $i = 2$ to $n$ do
17:     if there are more Vertices in $V'$ to be covered then
18:         check and locate a pairwise connection with the next closest neighbor in the sorted list
            and add this vertex to $S_{(t)}$;
19:     end if
20: end for
21: for all $S_{(t)}$ do
22:     for $i = 1$ to $n$ do
23:         for every vertex, select the segment with maximum traffic capacity;
24:     end for
25: end for
26: group users according to their similarity in locations, feeling safe value of $Q$ and service avail-
    ability value of $K$;
27: calculate condition entropies (Formula 3): $H(X_i|\Pi_i) = -\sum_{x_i,\pi_i} p(x_i,\pi_i) log_2 p(x_i|\pi_i)$;
28: while $(Q \geq H(X_i|\Pi_i))$ do
29:     select a sample of mix zones from $S_{(t)}$ that satisfy minimum service availability condition K
30:     $Z_{(AG)} = Z_{(AG)} + 1$
31:     until $(Q < H(X_i|\Pi_i))$;
32: end while
33: construct the resulting Graph $Z_{(AG)}$;
34: return and store Abstraction Graph $Z_{(AG)}$;
```

4.3 Optimized Decision Graph - ODG

We propose ODG (Algorithm 2) that is generated off-line and uses decomposition, chunking and replacement strategy to achieve an optimal solution from AG. The network B is initialized to an empty network that contains no edges. The decision graph $G_{(ODG)}(i)$ for each variable X_i is initialized to a single-leaf graph, containing only probabilities $p(X_i)$. In each iteration, all operators that can be performed on all decision graphs $G_{(ODG)}(i)$ are examined (steps 9 - 16). The quality of competing network structures is achieved by use of Bayesian metrics (Formula 6) as well as calculating Minimum Description Length using BIC (Formula 7) for compression. To allow for diversity, a replacement strategy is used (steps 17 - 33) where a random subset of candidate solutions is first selected from the original population. The new solution is then compared to each candidate solution in the selected subset and the fitness of the most similar candidate solution determined. The new solution will replace the most similar solution of the subset, or if otherwise, discarded. We store B and G_{ODG} (step 36).

Algorithm 2. Optimized Decision Graph *(ODG)*

```
Require: <Abstraction Graph $Z_{(AG)}$>
Ensure: <Bayesian Network B, Optimized Decision Graph $G_{(ODG)}$>
 1: t := 0;
 2: Let $P_{(t)}$ := n be the initial population;
 3: let $Z_{(AG)}$ := Abstraction Graph;
 4: let $G_{(ODG)}(i)$ := single-node decision graph;
 5: let $B_{(t)}$ := empty Bayesian network;
 6: let $O_{(t)}$ := Generated Offspring;
 7: let *mixcost* $c$ := Intersection Mixing Cost at any intersection $v$;
 8: while ($t \leq n$) /greedy algorithm for network construction using decision graphs do
 9:    for each variable $i$ do
10:       for each leaf $l$ of $G_{(ODG)}(i)$ do
11:          add all applicable splits of $l$ into $O_{(t)}$;
12:          for each pair of leaves $l_1$ and $l_2$ of $G_{(ODG)}(i)$ do
13:             add merge($l_1$, $l_2$) into $O_{(t)}$;
14:          end for
15:       end for
16:    end for
17:    for each offspring X from $O_{(t)}$ /Restricted Tournament Replacement for diversity maintenance do
18:       Y := random individual from $P_{(t)}$;
19:       CXY := *mixcost*(X,Y);
20:       for i=2 to n do
21:          Y' := random individual from $P_{(t)}$;
22:          CXY' := *mixcost*(X,Y');
23:          if (CXY' < CXY) then
24:             Y := Y';
25:             CXY := CXY';
26:          end if
27:       end for
28:    end for
29:    if (fitness(X) > fitness(Y)) then
30:       replace Y in $P_{(t)}$ with X (Y is discarded);
31:       else
32:       discard X;
33:    end if
34:    t := t+1;
35: end while
36: return and store B, $G_{(ODG)}$;
```

5 Security and Privacy Analysis

Our goal is to preserve privacy for users. If security of users is guaranteed, they end up feeling safe. Assume that a malicious adversary knows; 1) the exact location of querying user, 2) continuous query content, 3) locations of some POI, 4) AG and ODG Algorithms. The adversary may carry out 3 types of attacks; 1) *Timing attack*, 2) *Transition attack* and 3) *Continuous Query attacks* [10].

In *Timing attack* [10], the adversary observe the time of entry say t_1 and exit time say t_2 for each user entering and exiting the mix zone. For example, consider a case where the time of entry and exit from a mix zone is represented by a and b respectively. The adversary can calculate the cumulative distribution function of the continuous random variable x (x represent a user) as follows:-

$$P(a \leq X \leq b) = \int_a^b f(x)dx \tag{8}$$

Again, an adversary can calculate the skewness (which is a measure of symmetry or lack of it) of the distribution function of data set as follow:-

$$Skewness = \frac{\sum_{i=1}^{n}(y_i - \overline{y})^3}{(n-1)s^3} \tag{9}$$

where $\overline{y}$ is the mean, s is the standard deviation, and n is the number of data points. The calculated distribution could either be skewed to the right or left. In such a case, the attacker will use this to eliminate some low probable mappings and narrow down to only the high probable ones. In our proposed algorithms, we use non-rectangular mix zone approach proposed in [11] to confuse the adversary.

In *Transition attack* [10] the adversary will use Markov Chain to estimate the transition probabilities of either turning left, right or going straight for each possible turn in the intersection based on previous observations and use these similarities to infer the target user. For example, let m-step transition probability be the probability of transitioning from state i to state j in m steps. Using total probability theorem we get Chapman Kolmogorov equation [12]:

$$p_{ij}^{m+n} = \sum_{k=1}^{\infty} p_{kj}^{n} p_{ik}^{m} = P^{(m+n)} = P^{(m)}P^{(n)} \tag{10}$$

Increased anonymity strength in ODG drastically reduce chances of this attack.

On the other hand, *Continuous Query (CQ) attacks* [11] are mitigated by increasing the anonymity strengths of the mix-zones [10]. As supported by experiments, the proposed Optimized Decision Graph Algorithm offer greater anonymity strength and therefore minimizing chances of all kinds of *CQ* attacks.

6 Experiments and Evaluations

We adopt the real world mobility trace of human mobility data collected from five different sites from CRAWDAD [6] to generate our trajectory. We rank the sites according to traffic density starting with the highest densities in New York City, then Disney World - Florida, North Carolina State University (NCSU) Campus, South Korea University (KAIST) Campus and lastly State Fair streets with the lowest density. We form hierarchical decision graph with New York City (mostly with major roads) as our root node and generate ODG.

6.1 Privacy Metrics

1) **Conditional Entropy** - We utilize Formula (1) to calculate individual entropies. In a decision graph, a leaf node will depend on its parent node. we capture dependencies by calculating conditional entropies using Formula (3).
2) **Conditional Probability** (Formula (2)) measures resilience of the proposed algorithms to withstand attacks like Continuous Query (CQ) attacks.
3) **Cost** - Let R be the cost (per user) that results from mixing at a particular vertex with n participants. The average cost (AC) (per user) is given by:-

$$AC = \frac{\sum_{i=1}^{n} R_i}{n} \tag{11}$$

4) **Quality of Service (QoS)** - Let m represent the total number of sampled mix zones and t_1, t_2, ... t_m be the time take to mix per mix zone. The minimum average service availability K that achieves the desired QoS is given by:-

$$K = \frac{\sum_{i=1}^{m} t_i}{m} \tag{12}$$

6.2 Optimization Effectiveness Strategy

For each graph, we select a sample of 10 mix zones with least cost as per Formula (11) and sort them in ascending order to capture hierarchical attribute. We calculate their entropies as shown in Fig. 4. As such, highest entropy of 7.3 is observed in New York City as compared to lowest entropy of 3.3 in State Fair, North Carolina. High entropy is attributed to traffic capacity in road segments

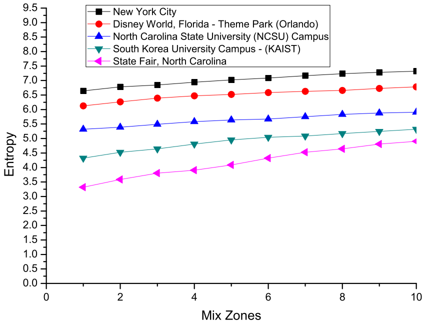


Fig. 4. Entropy Evaluation

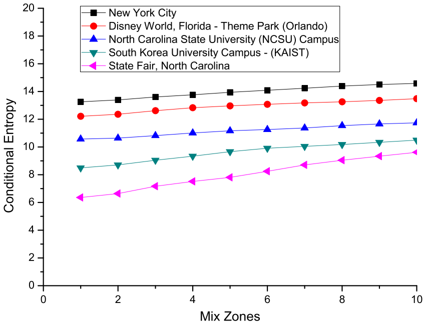


Fig. 5. Conditional Entropy Evaluation

in New York City. More traffic means higher entropy and better privacy. We use Formula (3) to calculate conditional entropies as shown in Fig. 5. We observe that entropy increases to a maximum of 14.6. This is due to dependency factor.

ODG Mix effect is tested in Table 1. The Overall Conditional Entropy (OCE) is calculated using Formula (4). OCE increases to 11.15, achieving greater anonymity.

Table 1. Optimized Decision Graph Mixing effectiveness

Mix Zones	New York	Disney World	NCSU USA	KAIST S Korea	State Fair
1	13.26	12.22	10.57	8.50	6.36
2	13.40	12.36	10.64	8.71	6.64
3	13.60	12.62	10.82	9.05	7.17
4	13.76	12.83	11.01	9.34	7.52
5	13.94	12.97	11.17	9.66	7.81
6	14.09	13.08	11.26	9.91	8.25
7	14.24	13.18	11.37	10.05	8.71
8	14.39	13.26	11.54	10.17	9.05
9	14.51	13.36	11.67	10.34	9.34
10	14.59	13.48	11.74	10.50	9.61
CE	**13.98**	**12.94**	**11.18**	**9.62**	**8.05**
				OCE	**11.15**

6.3 Quality of Service (QoS)

We use Formula (12) to test QoS when K = 0.6 and 1.0. In Fig. 6, we observe that a sample of 8 and 3 mix zones satisfy the minimum set service availability when K = 0.6 and 1.0 respectively. QoS is less when K = 1.0 as opposed to when K = 0.6, indicating a trade-off between privacy and utility (cost and QoS).

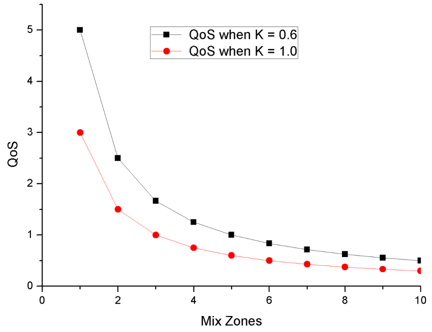


Fig. 6. Quality of Service Evaluation

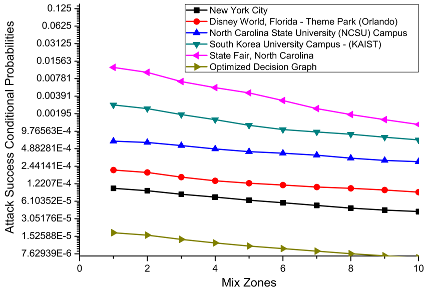


Fig. 7. Attack Resilience Evaluation

6.4 Attack Resilience

From generated ODG, we use Formula (2) to calculate the corresponding conditional probabilities as the user traverse the decision graph from leaf to root nodes and observe that the probability of successful attack is almost zero as shown in Fig. 7. With optimization and proper abstraction, we achieve greater anonymity strength, making it impossible to launch all kinds of C-Q attacks.

7 Conclusion and Future Work

We have presented a mix zone solution of preserving user's privacy using trusted third party architecture. We have proposed AG algorithm that selects a sample of mix zones satisfying user's desired privacy and service availabitity conditions, and ODG algorithm that finds an optimal solution for the placement of mix zones. The results of our experiment show that these Algorithms preserve privacy for users based on their privacy and service availability conditions. We are looking at how to implement decentralized architectures as opposed to centralized ones that are prone to single point of vulnerability as our future work.

References

1. Chow, C.-Y., Mokbel, M.F.: Trajectory privacy in location-based services and data publication. ACM SIGKDD Explorations Newsletter (2011)
2. Chow, C.-Y., Mokbel, M.F., Bao, J., Liu, X.: Query-aware location anonymization for road networks. Geoinformatica (2011)
3. Freudiger, J., Shokri, R., Hubaux, J.-P.: On the optimal placement of mix zones. In: Goldberg, I., Atallah, M.J. (eds.) PETS 2009. LNCS, vol. 5672, pp. 216–234. Springer, Heidelberg (2009)
4. Al-Amin, H., Amina, H., Hye-Kyeom, Y., Jae-Woo, C.: H-star: Hilbert-order based star network expansion cloaking algorithm in road networks. In: 14th IEEE International Conference on Computational Science and Engineering, CSE (2011)
5. Jadliwala, M., Bilogrevic, I., Hubaux, J.-P.: Optimizing mix-zone coverage in pervasive wireless networks. In: JCS (2013)
6. Kotz, D., Henderson, T.: Crawdad, ncsu/mobilitymodels (2009), `http://crawdad.cs.dartmouth.edu/meta.php?name=ncsu/`
7. Xinxin, L., Han, Z., Miao, P., Hao, Y., Xiaolin, L., Yuguang, F.: Traffic-aware multi-mix-zone placement for protec. location priv. In: IEEE INFOCOM (2012)
8. Meyerowitz, J.T., Choudhury, R.R.: Realtime location privacy via mobility prediction: Creating confusion at crossroads. In: Proceedings of the 10th Workshop on Mobile Computing Systems and Applications (2009)
9. Wichian, P., Walisa, R., Nucharee, P.: Navigation without gps: Fake location for mobile phone tracking. In: 11th Intern. Conf. on ITS Telecomm. (2011)
10. Palanisamy, B., Liu, L., Lee, K., Singh, A., Tang, Y.: Location privacy with road network mix-zones. In: IEEE MSN (2012)
11. Palanisamy, B., Liu, L.: MobiMix. Protecting location privacy with mix-zones over road networks. In: IEEE 27th ICDE (2011)
12. Papoulis, A.: Prob., Random Processes and Stoch. Processes. Mc-Graw Hill (2003)
13. Pelikan, M.: Hierarchical Bayesian Optimization Algorithm: Toward a New Generation of Evolutionary Algorithms. STUDFUZZ, vol. 170. Springer, Heidelberg (2005)
14. Wang, T., Liu, L.: Privacy-aware mobile services over road networks. Proc. of Very Large Databases (VLDB) Endowment 2(1) (2009)
15. Yang, K.-T., Chiu, G.-M., Lyu, H.-J., Huang, D.-J., Teng, W.-C.: Path privacy protection in continuous location-based services over road networks. In: IEEE 8th International Conference on WiMob (2012)