On Safety of Pseudonym-Based Location Data in the Context of Constraint Satisfation Problems

Tomoya Tanjo[1], Kazuhiro Minami[1], Ken Mano[2], and Hiroshi Maruyama[1]

[1] Institute of Statistical Mathematics, Tokyo, Japan
[2] NTT Corporation, Kanagawa, Japan
{tanjo,kminami,hm2}@ism.ac.jp, mano.ken@lab.ntt.co.jp

Abstract. Pseudonymization is a promising technique for publishing a trajectory location data set in a privacy-preserving way. However, it is not trivial to determine whether a given data set is safely publishable against an adversary with partial knowledge about users' movements. We therefore formulate this safety decision problem based on the framework of constraint satisfaction problems (CSPs) and evaluate its performance with a real location data set. We show that our approach with an existing CSP solver outperforms a polynomial-time verification algorithm, which is designed particularly for this safety problem.

1 Introduction

Nowadays, a location data set, which is obtained by collecting GPS data from people's mobile devices, can be used for various analytic purposes, such as real-time traffic monitoring [5] and urban planning for future sustainable cities [13]. However, due to the significant concern about location privacy [1], the sharing of mobile users' location traces has largely been restricted to k-anonymized data sets [6], which degrade the granularity of location data to ensure that every location contains more than k people. Such a k-anonymized data set provides little information on trajectory patterns of mobile users.

We, therefore, consider a dynamic pseudonym scheme for constructing a location data set that retains users' path information while preserving their location privacy. The basic idea is to exchange multiple users' pseudonyms randomly when they meet at the same location to eliminate the linkability of their pseudonyms before and after that exchange. Roughly speaking, a user's location privacy is preserved if we can find enough number of plausible alternate paths for that user in the data set. We believe that such a dynamic pseudonym approach is effective enough to publish large segments of the users' whole trajectory paths in a privacy-preserving way if the data set involves a large number of users whose trajectory paths intersect with each other many times.

However, it is not trivial to count the numbers of users' alternate paths under the presence of an adversary who owns partial information on users' movements (e.g., a user's home location). Such an adversary can eliminate some of the

Linawati et al. (Eds.): ICT-EurAsia 2014, LNCS 8407, pp. 511–520, 2014.

Fig. 1. Pseudonymized location data publishing. The data publisher replaces a user's identity u_i with a pseudonym p_i before releasing location data to data set users.

users' alternate paths that are inconsistent with his external knowledge. We thus need to address the issue of multi-path inconsistencies among multiple users and formulate this problem in the context of constraint satisfaction problems (CSPs) [4].

A CSP is defined as a set of variables whose values must satisfy a number of constraints expressed with arithmetic and logical operators. We can declaratively define constraints on each user's pseudonym assignments considering the possibility of pseudonym exchanges at mix zones and consistency requirements with an adversary's external knowledge. Once we formulate all constraints on users' plausible trajectory paths, we can compute the number of possible alternate paths with an existing CSP solver; that is, the number of different pseudonyms that are possibly taken by the same user corresponds to the uncertainty about that user's possible destinations.

Although the time complexity of solving a CSP is exponential at the worst case, our experimental results with a real location data set show that our CSP-based approach outperforms a polynomial-time algorithm we previously developed for this problem [9]. Therefore, we believe that our CSP-based approach is effective in many realistic situations.

2 Privacy Model

We first define our system model for a psedonymized location data publishing service, and introduce a technique of dynamic pseudonym exchanges at a mix zone. Next, we establish our privacy metrics we consider in this paper. Figure 1 shows our system model. We assume that each user u_i carrying a GPS-enabled mobile device periodically reports a triplet (u_i, l_k, t_k), which indicates that user u_i is at location l_k at time t_k. The data publisher receives identifiable location data from multiple users, replaces their identities with pseudonyms, and provides a dataset user with a pseudonymized location data set. This data set is an output from the data publisher in Figure 1.

To replace a user's identity on a given moving path with a static pseudonym does not necessarily protect the user's location privacy. The danger is that if an adversary knowing that a target user u is at location l at time t finds a data point (p, l, t) where p is a pseudonym from the received data set, the adversary can associate p with the user's identity u. Furthermore, he also learns that all the data points with the same pseudonym p in the data set belong to the same user u; that is, the adversary can identify user u's whole trajectory path.

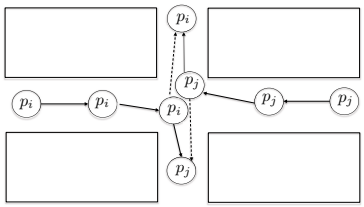

Fig. 2. Example pseudonym exchange. Two users exchange their pseudonyms p_i and p_j at the intersection. The solid lines denote each user's actual path while the dotted lines denote an alternate possible path.

To limit undesirable information disclosure from the above inference attack, we take an approach of changing each user's pseudonym dynamically when multiple users meet at the same location, which we call a *mix zone*. The basic idea is to divide a whole path of the same user into multiple segments with different pseudonyms such that the linkability of any neighboring segments is eliminated. Figure 2 shows an example of two users' exchanging their pseudonyms. Two users who own pseudonyms p_i and p_j, respectively, randomly exchange their pseudonyms when meeting at the intersection. Although the user who previously owned pseudonym p_i actually turns right at the corner, we consider that the alternate path of the users' turning left is also possible. The other user similarly has the two possible paths after passing the intersection.

If we consider the possible paths of a single user, whenever the user meets another user, we can add a new branch as a possible segment of the path. However, such a possible path must be consistent with an adversary's external knowledge. Suppose that the adversary knows users' home location and that every user starts its path with his home location and eventually returns home. We need to eliminate some possible branches if taking that direction makes it impossible for the user to return home. Furthermore, even if one user u_i is able to return home along a possible path, another user u_j who exchanged her pseudonym with u_i might lose a possible route to her home location. We thus need to consider possible pseudonym sequences for multiple users simultaneously. We call this requirement the *multi-path* consistency requirement, which is expressed as a set of constraints in a CSP in Section 3. We assume that an adversary learns a user u's location only at some mix zones; that is, the adversary observes user u where many people get together (e.g., a zebra zone on the street or a public space such as a hospital).

We consider the number of pseudonyms at a given time t on possible pseudonym sequences satisfying the multi-path consistency requirement as our location privacy metrics. Figure 3 shows such multiple pseudonym sequences of user u_i. If an adversary knows u_i's location at times t_0 and t^*, there is no uncertainty about a pseudonym taken by u_i at both times. However, user u_i is likely to have some uncertainty about his pseudonym in the middle of his trajectory after passing multiple mix zones. We now define the notion of (k, t)-pseudonym location privacy as follows.

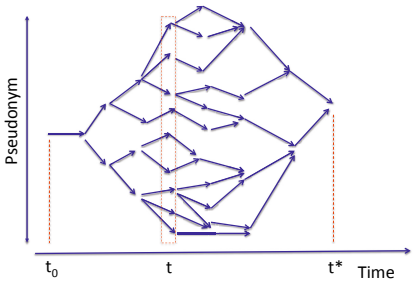

Fig. 3. Concept of (k, t)-pseudonym location privacy

Definition 1 ((k, t)-pseudonym location privacy). *If user u_i can take k or more pseudonyms at a given time t while satisfying the multi-path consistency requirement, we say that user u_i satisfies (k, t)-pseudonym location privacy.*

3 Background on Constraint Satisfaction Problems

We give a brief overview of a constraint satisfaction problem (CSP) [12], which is sufficient to formulate the safety problem of location data psuedonymization in the context of CSP in Section 4. A CSP is a problem of finding a solution satisfying all the given conditions on a set of variables. We define a CPS in a declarative way such that a CSP solver finds a solution in a computationally efficient way. Although solving a CSP is known as NP-hard, existing CSP solvers, which have been widely used in many areas (e.g., [2,8]), usually show good performance for practical purposes.

We first define a constraint network and a CSP as follows.

Definition 2 (Constraint network). ***Constraint network*** *(or Network) N is a tuple (X, D, C) where*

- *X is a finite set of integer variables,*
- *D is a mapping from X to a set of all possible finite subsets of integers which represents their possible values (domain), and*
- *C is a finite set of constraints over X which represents a conjunction of constraints ($\wedge$). Each constraint consists of:*
  - *Arithmetic operators (such as $+$, $-$)*
  - *Arithmetic comparisons (such as $=$, $\neq$, $\leq$)*
  - *Logical operators (such as $\wedge$, $\vee$, $\Rightarrow$)*
  - *Global constraints (such as **same**: described later)*

A global constraint is a constraint between non-fixed number of variables. For example, the **same** constraint takes two integer sequences $X =< x_1, x_2, \ldots, x_n >$ and $Y =< y_1, y_2, \ldots, y_n >$, and it represents X is a permutation of Y.

An *assignment* is a mapping from X to a set of integers and a *partial* assignment is a mapping from a subset of X to a set of integers.

Definition 3 (Constraint Satisfaction Problem). *Let $N = (X, D, C)$ be a constraint network. A constraint satisfaction problem (CSP) is a problem to find an assignment α such that*

- *α satisfies all the constraints $c \in C$ and*
- *$\alpha(x) \in D(x)$ holds for all integer variables $x \in X$.*

If there exists such assignment α, it is called a solution *of the CSP.*

An existing CSP solver typically finds a solution in the following way. First, it picks one variable x_i from X and defines a partial assignment to x_i. Second, the solver removes inconsistent values from the other domains by applying a constraint propagation algorithm. If another domain for a variable x_j becomes an empty set (i.e., the partial assignment cannot be extended to any solution), the solver picks a different value x_i' from $D(x_i)$ and repeats the same process to extend the partial assignment to variable x_j. This process is iterated until the solver finds a solution by extending the partial assignment to an assignment for all the variables.

4 Formalizing the CSP Safety Problem

We formalize the safety problem in Section 2 as the k-pseudonym decision problem and show how to solve that decision problem using a CSP solver.

4.1 The *k*-Pseudonym Decision Problem

We use letters u and $u_1, u_2, \ldots$ for users and p and $p_1, p_2, \ldots$ for pseudonyms respectively. We denote N_{t^*} as a finite set of integers $\{1, \ldots, t^*\}$.

Definition 4 (Mix Zone). *Let U be a finite set of users. A mix zone m over U is a subset of U whose size is greater than one. We denote by $\mathbb{M}_{\mathbb{U}}$ all the set of mix zones over U.*

We next define the mix zone function that takes a time t as an input and outputs a finite set of mix zones, which occur at time t as follows.

Definition 5 (Mix zone function). *Let U be a finite set of users, t^* be a positive integer, and $\mathbb{M}_{\mathbb{U}}$ be a set of all possible mix zones over U. The mix zone function $f_{U,t^*} : N_{t^*} \to 2^{\mathbb{M}_{\mathbb{U}}}$ is a mapping from N_{t^*} to a finite subset of $\mathbb{M}_{\mathbb{U}}$ where $\mathbb{M}_{\mathbb{U}}$ is the set of all mix zones.*

To formulate a k-pseudonym decision problem, we express an adversary's external knowledge and each user u's security requirements as follows.

Definition 6 (External knowledge). *External knowledge is a finite set of pairs (u, t), which represents the fact that an adversary knows a user u's location at time t.*

Definition 7 (Security requirement). *A security requirement is a tuple (u, t, k) where $u \in U$ is a user and $1 \leq t \leq t^*$ and $k \geq 1$ are integers.*

This requirement represents the fact that a user u can possibly take more than k different pseudonyms at time t. We expect that each user specifies multiple security requirements on a given pseudonymized data set.

We next define the pseudonym function which returns a pseudonym for the user u at time t.

Definition 8 (Pseudonym function). *Let U be a finite set of users, $P = \{p_1, p_2, \ldots, p_{|U|}\}$ be a finite set of pseudonyms, and t^* be a positive integer. The function **pseudonym** $s : U \times N_{t^*} \to P$ maps a pair of a user u and time t to a pseudonym $p \in P$ such that a user u has a pseudonym p at time t.*

Finally, we define the k-pseudonym decision problem as follows.

Definition 9 (k-pseudonym decision problem). *Let U be a finite set of users, t^* be a positive integer, and E be an adversary's external knowledge. Let (u, t, k) be a security requirement. The k-pseudonym decision problem $(f_{U,t^*}, (u, t, k), E)$ is a problem to decide whether there exist k candidates for $s(u, t)$ that are consistent with the external knowledge E.*

4.2 Solving k-Pseudonym Decision Problem

We first represent the k-pseudonym decision problem as a constraint network and show how we solve it with a CSP solver in an incremental way. Figure 4 shows the function *generateCSP* that generates a constraint network from the given mix zone function. Quoted variables or constraints such as 's_u^t' in Figure 4 show the variables or constraints in the network. Here are the overview of the function *generateCSP*:

- We introduce an integer variable $s_u^t \in P$ which represents a pseudonym $s(u, t)$ for each user u and each time t. The domain of s_u^t is $\{1, 2, \ldots, i, \ldots, |U| - 1, |U|\}$ where each domain value i corresponds to the pseudonym p_i in the set of pseudonyms P.
- Without loss of generality, we add constraints for specifying the pseudonyms of users at time $t = 0$.
- For each time $t \in \{1..t^*\}$, we add the following constraints.
  - $s(u, t) = s(u, t - 1)$ holds if the user u is not included in any mix zones at time t.
  - **same**$(< s(u_i, t - 1), s(u_j, t - 1), \cdots >, < s(u_i, t), s(u_j, t), \cdots >)$ holds if there is a mix zone $\{u_i, u_j, \ldots\}$ at time t.

```
// U: a finite set of users, f: mix zone function, t*: maximum time
def generateCSP(U, f, t*)
  X = ∅
  C = ∅
  // introduce integer variables
  foreach u in U
    foreach t in 0..t*
      X = X ∪ {‘s_u^t’}
    end
  end

  // pseudonyms at time t = 0
  i = 0
  foreach u in U
    C = C ∪ {‘s_u^0 = i’}
    i = i + 1
  end

  foreach t in 1..t*
    // The same constraint should hold for each mix zone.
    foreach M in f(t)
      P = ∅
      foreach ‘s_u^t’ in M
        P = P ∪ {‘s_u^(t−1)’}
      end
      C = C ∪ {‘same(P, M)’}
    end
    // If an user u is not included in any mix zones,
    // ‘s_u^t’ is same as the pseudonym at the previous time.
    foreach u in U − ⋃ f(t)
      C = C ∪ {‘s_u^t = s_u^(t−1)’}
    end
  end
  return (X, D, C) where D(s) = {1..|U|} for all s in X
end
```

Fig. 4. A pseudo code to generate a constraint network from the given mix zone function

In addition to the constraint network generated by the function *generateCSP*, we need to express extra constraints for the external knowledge. If an adversary knows only one external knowledge (u, t), she knows that one of the possible values in $D(s_u^t)$ corresponds to the user u. She cannot infer further information because she only knows the external knowledge at mix zones as described in Section 2. Therefore we do not need additional constraints in this case.

If the adversary knows two external knowledge $\{(u, t_1), (u, t_2)\}$ (i.e., she know the information about $s(u, t_1)$ and $s(u, t_2)$), she can infer more information from the external knowledge at the worst case where $s(u, t_1) = s(u, t_2)$ holds. To consider this worst case, we add an extra constraint $s_u^{t_1} = s_u^{t_2}$ to the generated constraint network. If there is more than two elements in the external knowledge, we add the corresponding constraints in the same way.

Let N be a constraint network which is generated with the function $generateCSP(U, f, t^*)$ and let us consider $(X_G, D_G, C_G) = \Phi_G(N)$ where $\Phi_G(N)$ is the function for removing all inconsistent values from the domains. Each domain value $d \in D_G(s_u^t)$ corresponds to a possible pseudonym which is computed with $s(u, t)$. Therefore, we can solve k-pseudonym decision problem by checking whether $|D_G(s_u^t)| \geq k$ holds for the security requirement (u, t, k).

However, many CSP solvers do not use Φ_G in practice because it requires too much computation time. Those solvers usually use the algorithms for other consistencies which are weaker but more reasonable with respect to execution time or memory consumption. Therefore, we propose an incremental solving method, which can be applicable to existing CSP solvers.

In the incremental solving method, we first generate a constraint network (X, D, C) from the given mix zone function, and check whether the network $(X, D, C \cup \{s_u^t = i\})$ has a solution for each domain value $i \in D(s_u^t)$.

5 Experimental Results

We develop the safety verification program for solving the k-pseudonym decision problem from a given mix zone function. It is written in Groovy with 409 lines using the Choco library [14] as an external CSP solver. Using this program, we further develop an optimization program based on dynamic programming that finds the minimum number of mix zones satisfying a given safety requirement.

We use the dataset [11] containing mobility traces of taxi cabs in San Francisco, USA. It contains GPS coordinates of approximately 500 taxis collected over 30 days in the San Francisco Bay Area. When we conduct our experiments with a given number of users, we randomly pick a specified number of users from the dataset.

Figure 5 shows performance results of finding the minimum set of mix zones satisfying all the security requirements changing the number of users in a dataset. We randomly define security requirements of up to five to randomly chosen users. We compare results with our safety verification program using the CSP solver with those using our previously developed polynomial-time algorithm[10]. Although the time complexity of the CSP solver is exponential at the worst

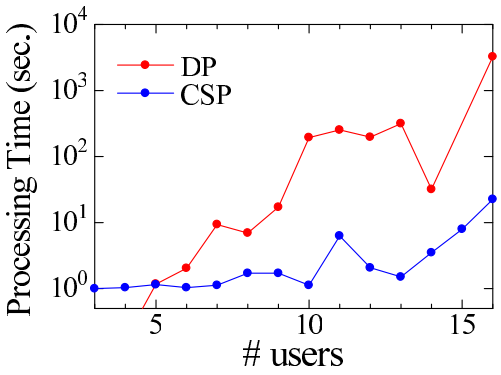

Fig. 5. Comparison of the processing time. *CSP* is the result with a CSP solver and *DP* is the results with our previously developed polynomial algorithm.

case, our safety versifier outperforms our previously algorithm. This results show that the CSP solver, which has been done with various performance turning, efficiently makes a safety decision with realistic location data sets.

6 Related Work

Using pseudonyms is a promising way to make location data unlinkable to a particular user. Beresford and Stajano [3] were the first to discuss the idea of dynamically changing pseudonyms in a mix zone where multiple people meet, in order to prevent an adversary from linking two pseudonyms of the same user. However, they only consider the situation where an adversary has just a local view of users' movements and observes pseudonyms of entering or leaving the same mix zone. Hoh and Gruteser [7] present a path perturbation algorithm that adds noises to original location data so that each user can construct alternate possible paths by exchanging his pseudonym with those of other users when they meet at the same place. However, their scheme does not consider an adversary's external knowledge that can associate each user with a particular location, as we assume in this paper.

7 Conclusions

In this paper, we introduce the safety definition of pseudonym-based location data and show how to represent the original safety problem in the context of constraint satisfaction problem. Our experimental results with a real location data set show that our approach with an existing CSP solver outperforms a polynomial-time verification algorithm, which is designed particularly for this safety problem.

Acknowledgments. This research is supported by the Strategic Joint Research Grant for NTT and Research Organization of Information and Systems (ROIS) and by the Grants-in-Aid for Scientific Research C, 11013869, of Japan Society for the Promotion of Science.

References

1. Anthony, D., Henderson, T., Kotz, D.: Privacy in location-aware computing environments. IEEE Pervasive Computing 6(4), 64–72 (2007)
2. Backofen, R., Gilbert, D.: Bioinformatics and constraints. In: Rossi, F., van Beek, P., Walsh, T. (eds.) Handbook of Constraint Programming, ch. 26, pp. 903–942. Elsevier Science Inc. (2006)
3. Beresford, A.R., Stajano, F.: Location Privacy in Pervasive Computing 2(1), 46–55 (January-March 2003)
4. Freuder, E.C., Mackworth, A.K.: Constraint satisfaction: An emerging paradigm. In: Rossi, F., van Beek, P., Walsh, T. (eds.) Handbook of Constraint Programming, ch. 2, pp. 11–26. Elsevier Science Inc. (2006)
5. Google maps, `http://maps.google.com/`
6. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: Proceedings of Mobisys 2003: The First International Conference on Mobile Systems, Applications, and Services. USENIX Associations, San Francisco (2003)
7. Hoh, B., Gruteser, M.: Protecting location privacy through path confusion. In: First International Conference on Security and Privacy for Emerging Areas in Communications Networks, SecureComm 2005, pp. 194–205 (September 2005)
8. Hooker, J.N.: Operations research methods in constraint programming. In: Rossi, F., van Beek, P., Walsh, T. (eds.) Handbook of Constraint Programming, ch. 15, pp. 525–568. Elsevier Science Inc. (2006)
9. Mano, K., Minami, K., Maruyama, H.: Privacy-preserving publishing of pseudonym-based trajectory location data set. In: Proceedings of the 2nd International Workshop on Security of Mobile Applications, IWSMA (2013)
10. Mano, K., Minami, K., Maruyama, H.: Protecting location privacy with k-confusing paths based on dynamic pseudonyms. In: Proceedings of the 5th IEEE International Workshop on SEcurity and SOCial Networking (SESOC), pp. 285–290 (2013)
11. Piorkowski, M., Sarafijanovic-Djukic, N., Grossglauser, M.: CRAWDAD data set epfl/mobility (v. 2009-02-24) (February 2009), `http://crawdad.cs.dartmouth.edu/epfl/mobility`
12. Rossi, F., van Beek, P., Walsh, T.: Handbook of Constraint Programming. Elsevier Science Inc., New York (2006)
13. Seike, T., Mimaki, H., Hara, Y., Odawara, R., Nagata, T., Terada, M.: Research on the applicability of "mobile spatial statistics" for enhanced urban planning. Journal of the City Planning Institute of Japan 46(3), 451–456 (2011)
14. The choco team: choco: An open source Java constraint programming library. In: Proceedings of the 3rd International CSP Solver Competition, pp. 7–13 (2008)