LETTER *Special Section on Trust, Security and Privacy for Pervasive Applications*

An Optimal Algorithm towards Successive Location Privacy in Sensor Networks with Dynamic Programming

Baokang ZHAO[†a)], ***Student Member***, **Dan WANG**[††], **Zili SHAO**[††], **Jiannong CAO**[††], **Keith C.C. CHAN**[††], ***and*** **Jinshu SU**[†], ***Nonmembers***

SUMMARY In wireless sensor networks, preserving location privacy under successive inference attacks is extremely critical. Although this problem is NP-complete in general cases, we propose a dynamic programming based algorithm and prove it is optimal in special cases where the correlation only exists between p immediate adjacent observations.

key words: *wireless sensor networks, location privacy, algorithms, dynamic programming*

1. The Successive Location Privacy Problem

Since Kamat's seminal work in 2005 [1], so far several literatures have been concentrated on analyzing, modeling and building the location privacy concerns in WSNs (wireless sensor networks) [2]–[4]. These schemes mainly focus on protecting the current location of interested objects. However, to date, given a sequence of past observations, abundant techniques are available to infer successive locations of an object, such as particle filter [6], mobility pattern prediction [7], etc. With location inference techniques listed above, the adversary is able to infer current location via successive disclosed locations, which bring serious successive location privacy threat to WSNs.

To deal with the aforementioned successive privacy threats, by analyzing classical location inferring techniques, we aim at depicting the basic characteristics of these inference techniques. Intrinsically, each past observation will contribute to the accuracy of the inference of the future locations. The more observations, the higher the accuracy. Therefore, we generalize it into a weighted representation model for the successive location privacy problem [5]. Within this model, given that the base station has the observation sequence $Z = (z_1, z_2, \ldots, z_n)$, we have the following observations:

1. Each observation z_i is associated with a weight w_i which denotes the impact of this observation on the inference of z_{n+1}.
2. There is a *cross weight* for a set of observations, e.g., we have $w_{i,j}$ for z_i and z_j. Intuitively, publishing both z_i and z_j will contribute more to the final inference of z_{n+1} than the combined weight of publishing them individually.
3. The joint weights usually have higher impact if the observations are taken closer in a time period, i.e., the $w_{ij} > w_{ik}$ if $|t_i - t_j| < |t_i - t_k|$.

We call it a *publishing subsequence* or *publishing sequence* for a subset of the observation sequence that are published to the users. Formally, we define indicator variables x_i, such that

$$x_i = \begin{cases} 1, & z_i \text{ is published} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Note that the sensor network is deployed for monitoring the objects's locations. Therefore, the objective is to maximize the number of observations that can be published to the public; after all, the sensor network is deployed for data collection. However, the weights of the observation sequence indicate the inference capability of the adversary. Consequently, there is a trade-off between the number of observations that can be released to the public and the successive privacy of the object. Accordingly, motivated by the classical *K-anonymity model*, we can formulate it into the *maximum publishing sequence problem (MPS)* as follows:

$$Maximize: \sum_{1 \le i \le n} x_i \tag{2}$$

$$s.t. \sum_{\forall j_l \in [1,\ldots,n], 1 \le k \le n} w_{j_1,\ldots,j_k} x_{j_1} \cdots x_{j_k} \le K \tag{3}$$

$$\forall 1 \le j \le n \quad x_j \in \{0, 1\} \tag{4}$$

Where K denotes the pre-determined threshold value, and $w_{j_1,\ldots,j_k}$ represents the weight for $x_{j_1} \ldots x_{j_k}$. For a publishing sequence P, if the combined weight of these observations is greater than a pre-determined threshold value K, we say that the location privacy (i.e., next location) of the object is broken; otherwise, the location privacy of the object is maintained.

This is a typical non-linear Integer Programming problem. Since the parameters of weights are generic, this problem is NP-complete [8]. We can apply several general non-linear Integer Programming solutions, such as introducing penalty functions, using relaxation, etc [9], [10].

As the problem is intractable, we are more interested in developing optimal solutions for special cases together with heuristics for common cases.

Manuscript received July 16, 2009.
Manuscript revised October 7, 2009.

[†]The authors are with the School of Computer Science, National University of Defense Technology, Changsha, Hunan, China.
[††]The authors are with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China.
a) E-mail: bkzhao@nudt.edu.cn

DOI: 10.1587/transinf.E93.D.531

2. Our Proposed Optimal Algorithm

We first consider a simple variant of the MPS problem where all the observations are independent in the contribution of the inference, i.e., there is no cross weights. This problem can thus be solved by a greedy algorithm where the observations are selected in the order (from low to high) of their associated weights, until the summation of the weights exceeds K.

2.1 The p-Relation Observation Sequence

We consider a special case where the correlation only exists between p immediate adjacent observations. Formally, let Q be a set of observations that has property such that $\forall z_i, z_j \in Q, |t_i - t_j| \leq p$. Thus,

Definition 1: ***p*-relation observation sequence**: An observation sequence $Z = (z_1, z_2, \ldots, z_n)$ is defined to be p-relation observation sequence if $\forall Q \subseteq Z$, the weighted function w on Z has the following feature:

$$\begin{cases} w(Q) \geq 0, & Q \in \mathcal{Q} \\ w(Q) = 0, & \text{otherwise} \end{cases}$$

Two examples for 2-relation and 3-relation observation sequences are shown in Fig. 1 (a) and (b), respectively.

In what follows, we study the 2-relation observation sequence for MPS problem (in short, the 2-relation MPS problem). Similar techniques can be used for p-relation MPS problem. The following theorem shows that optimal solution exists for 2-relation MPS problem.

Theorem 1: 2-relation MPS problem is solvable by dynamic programming.

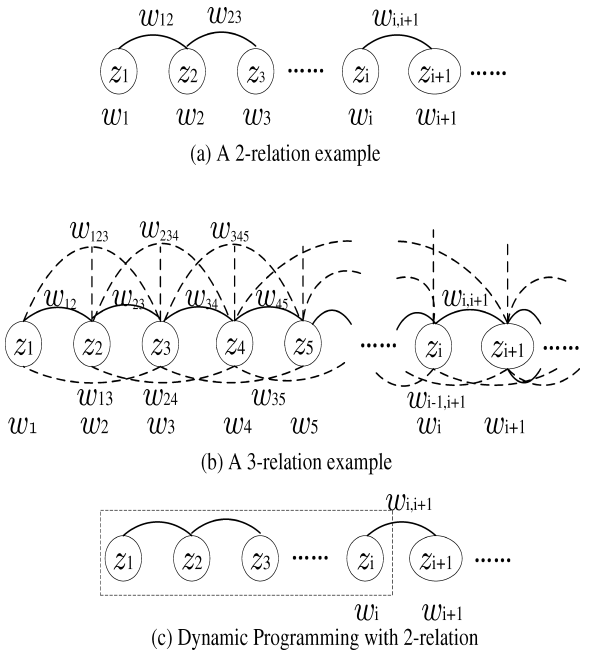

Fig. 1 The p-relation observation sequence.

proof Let $Q_i = (z_1, z_2, \ldots, z_i)$, i.e., a subsequence of Z from z_1 to z_i. Let $V_i = (v_1, v_2, \ldots, v_i)$ be the (sub)-assignment of $X = (x_1, x_2, \ldots, x_i)$. Let $\mathcal{W}(V_i)$ be the total weight of this assignment.

Let $h(Q_i, K, v_i)$ be the maximum utility function for subsequence Q_i with privacy constraint K and $x_i = v_i$. Formally,

$$h(Q_i, K, v_i) = max\left\{ \sum_{1 \leq j \leq i} v_j \;\middle|\; \mathcal{W}(V_i) \leq K \;\&\; x_i = v_i \right\}$$

The optimal data utility is thus

$$max\{h(Q_n, K, 0), h(Q_n, K, 1)\}$$

Notice that in 2-relation, the cross weight only exists between observations z_{i-1} and z_i. We first consider the case when $x_i = 0$. In this case, the corresponding weight w_i and $w_{i-1,i}$ has no impact on the result. Hence, $h(Q_i, K, 0)$ can be calculated by:

$$h(Q_i, K, 0) = max\{h(Q_{i-1}, K, 0), h(Q_{i-1}, K, 1)\} \tag{5}$$

We then consider $x_i = 1$. We have:

$$\begin{aligned} h(Q_i, K, 1) = max\{&h(Q_{i-1}, K - w_i, 0), \\ &h(Q_{i-1}, K - w_{i-1,i} - w_i, 1)\} + 1 \end{aligned}$$

We add one additional to the maximum utility as x_i will be published. To make sure that $h(\cdot)$ takes valid values, we finally have:

$$h(Q_i, K, 1) = max \begin{cases} 1 + max\{h(Q_{i-1}, K - w_i, 0), & \\ h(Q_{i-1}, K - w_{i-1,i} - w_i, 1)\} & K \geq w_{i-1,i} + w_i \\ 0, & K < w_i \\ h(Q_{i-1}, K - w_i, 0) + 1, & w_i \leq K < w_{i-1,i} + w_i \end{cases} \tag{6}$$

Clearly the 2-relation MPS problem shows optimal substructure and thus is solvable by dynamic programming.

2.2 Dynamic Programming Algorithm and Its Complexity

To develop a dynamic programming-based algorithm, we initialize the $h(\cdot)$ function for the following special cases:

$$h(Q_0, i, 0) = 0 \quad \forall 0 \leq i \leq n, \tag{7}$$
$$h(Q_j, 0, 0) = 0 \quad \forall 0 \leq j \leq K, \tag{8}$$
$$h(Q_i, j, 0) = -\infty \quad \forall i, j < 0, \tag{9}$$
$$h(Q_i, j, 1) = -\infty \quad \forall i, j < 0, \tag{10}$$
$$h(Q_0, i, 1) = 0 \quad \forall 0 \leq i \leq w_0, \tag{11}$$
$$h(Q_0, i, 1) = 1 \quad \forall w_0 < i \leq K, \tag{12}$$
$$h(Q_j, 0, 1) = 0 \quad \forall 0 \leq j \leq K, \tag{13}$$

Where Eqs. (7), (8), (11)–(13) denotes the starting values; Eqs. (9) and (10) denotes that the utility is infinity when there is no feasible solution. Then, $h(\cdot)$ can be iteratively solved as shown in Fig. 2.

We estimate the complexity of the DP-2 algorithm.

```
1 Initialization;
2 for ∀1 ≤ i ≤ n, 1 ≤ j ≤ K do
3 |   calculate h(Q_i, j, 0) and h(Q_i, j, 1) according to
  |   Equation (5) and Equation (6), respectively ;
4 end
5 Output: max{h(Q_n, K, 0), h(Q_n, K, 1)} and V_n;
```

Fig. 2 DP-2 algorithm for 2-relation MPS problem.

Since it should maintain two tables for $h(Q_i, K, 0)$ and $h(Q_i, K, 1)$, the time complexity for initialization process of line 1 is $2 \times n \times K$. From line 2 to line 4, the complexity maintains $2 \times n \times K$. Hence, the time complexity of DP-2 is $O(n \times K)$. Note that although dynamic programming provides optimal solutions for 2-relation problem, it occupies two tables to store intermediate data. Therefore, the space complexity required is $(2 \times n \times K)$.

The above dynamic programming can be easily extended to the p-relation MPS problem.

Theorem 2: p-relation MPS problem is solvable by dynamic programming.

We further consider the space complexity of our proposed algorithm. When we consider the case for p-relation MPS problem, more tables should be stored. General speaking, the number of equations for DP-p corresponding to Eqs. (5) and (6) are increased to $2 \times p$. As a result, the time and space complexity are $O(2^p \times n \times K)$.

Acknowledgment

The work described in this paper is partially supported by the grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (CERG 526007 (PolyU 5305/08E, PolyU 5102/08E)), the National Basic Research Program of China (973 project) under Grant No.2005CB314802, No.2009CB320503; the National 863 Development Plan of China under Grant No. 2006AA01Z401, No. 2008AA01A32, No. 2009AA01Z423; and the key project of National Science Foundation of China under grant No.90604006.

References

[1] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," Proc. IEEE ICDCS '05, Washington, DC, USA, 2005.

[2] H. Wang, B. Sheng, and Q. Li, "Privacy-aware routing in sensor networks," Computer Netw., vol.53, no.9, pp.1512–1529, 2009.

[3] K. Mehta, D. Liu, and M. Wright, "Location privacy in sensor networks against a global eavesdropper," Proc. IEEE ICNP '07, Oct. 2007.

[4] Y. Jian, S. Chen, Z. Zhang, and L. Zhang, "Protecting receiver-location privacy in wireless sensor networks," Proc. IEEE INFOCOM '07, May 2007.

[5] B. Zhao, D. Wang, Z. Shao, J. Cao, K.C.C. Chen, and J. Su, "Towards successive privacy protection in sensor networks," Proc. IEEE TSP '08, Dec. 2008.

[6] L. Liao, J. Patterson, D. Fox, and H. Kautz, "Learning and inferring transportation routines," Artif. Intell., vol.171, no.9, pp.311–331, 2008.

[7] P. Pathirana, V. Savkin, and J. Sanjay, "Mobility modelling and trajectory prediction for cellular networks with mobile base stations," Proc. MobiHoc '03, Dec. 2003.

[8] M. Garey and D. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman and Company, San Francisco, CA. 1979.

[9] D.P. Bertsekas, Nonlinear Programming, 2nd ed., Athena Scientific, Nashua, NH, USA, 2000.

[10] G.P. McCormick, Nonlinear Programming: Theory, Algorithms, and Applications, John Wiley & Sons, New York, NY, USA, 1983.