Computer Communications 33 (2010) 1414–1427

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Measuring long-term location privacy in vehicular communication systems

Zhendong Ma [a,*], Frank Kargl [b], Michael Weber [a]

[a] Institute of Media Informatics, Ulm University, Albert-Einstein-Allee 11, 89081 Ulm, Germany
[b] Distributed and Embedded Security, University of Twente, P.O.-Box 217, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Available online 6 March 2010

Keywords:
Vehicular communication systems
Location privacy
Metric
Entropy
Accumulated information

ABSTRACT

Vehicular communication systems are an emerging form of communication that enables new ways of cooperation among vehicles, traffic operators, and service providers. However, many vehicular applications rely on continuous and detailed location information of the vehicles, which has the potential to infringe the users' location privacy. A multitude of privacy-protection mechanisms have been proposed in recent years. However, few efforts have been made to develop privacy metrics that can provide a quantitative way to assess the privacy risk, evaluate the effectiveness of a given privacy-enhanced design, and explore the full possibilities of protection methods.

In this paper, we present a location privacy metric for measuring location privacy in vehicular communication systems. As computers do not forget and most drivers of motor vehicles follow certain daily driving patterns, if a user's location information is gathered and stored over a period of time, e.g., weeks or months, such cumulative information might be exploited by an adversary performing a location privacy attack to gain useful information on the user's whereabouts. Thus to precisely reflect the underlying privacy values, in our approach we take into account the *accumulated information*. Specifically, we develop methods and algorithms to process, propagate, and reflect the accumulated information in the privacy measurements. The feasibility and correctness of our approaches are evaluated by various case studies and extensive simulations. Our results show that accumulated information, if available to an adversary, can have a significant impact on location privacy of the users of vehicular communication systems. The methods and algorithms developed in this paper provide detailed insights into location privacy and thus contribute to the development of future-proof, privacy-preserving vehicular communication systems.

1. Introduction

Vehicular communication systems are an emerging form of communication that enables new ways of cooperation among vehicles, traffic operators, and service providers. Based on Dedicated Short Range Communications (DSRC) technology, vehicles can communicate among each others and with the entities in infrastructure networks via Roadside Units (RSU) to deliver and exchange high-definition information about themselves and their environments. As one of the key technologies to build an Intelligent Transportation System (ITS) in the near future, vehicular communication systems are envisioned to significantly improve road safety, traffic efficiency, and driver convenience. Example vehicular communication applications include collision warning, floating car data, and location-based services. If deployed, vehicular communication systems will become one of the biggest implementations of Mobile Ad Hoc Networks (MANET).

1.1. Motivation

However, many envisioned vehicular communication (VC) applications rely on continuous and detailed location and time information of the vehicles. This requires all vehicles to frequently send their location information in terms of current positions, speeds, and headings, combined with a time stamp in so-called "beacon" or "heartbeat" messages openly to all of their neighbors. The message from a vehicle can be eavesdropped by anyone within the radio transmission range. By establishing a network of receivers, an adversary, i.e., any individual or public, private, commercial, or criminal organization, can collect and abuse the location information to its advantage. Vehicles are personal devices and usually owned for a long period of time. The whereabouts of a vehicle reveal the movements and activities of its driver and passengers. Sending and disseminating location information has the potential to infringe location privacy[1] of the users of vehicular communication systems.

* Corresponding author.
E-mail addresses: zhendong.ma@uni-ulm.de (Z. Ma), f.kargl@utwente.nl (F. Kargl), michael.weber@uni-ulm.de (M. Weber).

[1] In [3], location privacy is defined as *the ability of an individual to move in public space with the expectation that under normal circumstances their location will not be systematically and secretly recorded for later use.*

0140-3664/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2010.02.032

Privacy issues in vehicular communication systems have been identified in recent years and a number of privacy-protection mechanisms have been proposed [20,26,6,28,16]. Clearly, without proper privacy protection, vehicular communication systems pose a severe privacy threat to potential users.

Nevertheless, to assess a system's ability to preserve the users' location privacy and to evaluate the effectiveness of any protection mechanism, a metric for measuring the level of users' location privacy is crucial and indispensable. In other words, a location privacy metric that numerically expresses privacy will be a more precise and rigorous way to reflect the provided privacy level than just stating that a system provides "adequate privacy". For example, we need a metric to tell us that a user's privacy level has been increased by 20% after applying one of the protection mechanisms. Furthermore, privacy does not come for free. Privacy-protection mechanisms usually have side-effects in terms of communication and computation overhead [7] and deployment cost. In addition, privacy requirements are among a set of requirements for vehicular communication systems [29], which are sometimes conflicting, e.g., the need for strong identification for authentication and accountability versus the need for anonymity for privacy. This determines that future vehicular communication systems will have to consider and harmonize a conglomeration of requirements from different stakeholders. Therefore, a privacy metric can greatly contribute to an overall system design process that balances and optimizes various requirements and finds the best privacy protection available.

However, so far the main focus in the literature is on the development of privacy-protection mechanisms. In contrast, the effort to develop an appropriate metric that reflects the underlying level of location privacy of the users of vehicular communication systems has been overlooked at large. Hence the privacy values related to location privacy cannot be quantitatively and explicitly expressed. Consequently, the usefulness of any given privacy-protection mechanisms cannot be rigorously evaluated, the trustworthiness and the privacy risks of future vehicular communication systems cannot be strictly assessed, and the range of possible protection methods cannot be fully explored.

1.2. Problem statement

In our previous work [23], we proposed a *trip-based location privacy metric* to measure the level of location privacy of individual users of vehicular communication systems. We identified that the most privacy-relevant location information in vehicular communication systems is the origin and destination of a vehicle trip,[2] which reveal the driver's identity and social activities and are susceptible to a number of attacks such as home identification [16,12] and inference attack [21]. Based on the observation that the uncertainty of a potential adversary and a user's privacy level are indeed two sides of the same coin, the metric measures the level of location privacy as the *linkability* of vehicle trips to the individuals who generate them. Taking an information-theoretic approach, the uncertainty in the linkability is expressed in probabilities and quantified into entropy. To be able to take meaningful measurements on dynamic and continuous systems like vehicular communication systems, we introduced the concept of a *snapshot*. A snapshot limits the privacy-related information to an arbitrary defined period of time such that we can base our measurements on a set of stable and confined information. Section 2 gives a more detailed description of our previous work.

However, our previous work only considers a *single* snapshot. To precisely reflect the level of location privacy, it is reasonable to assume that information available to an adversary is not limited to only a short period of time. Instead, it is reasonable to assume that a determined adversary will do its best to obtain as much information as possible and to decrease the uncertainty of the obtained information. Thus the adversary will take advantage of the *accumulated information*, i.e., privacy-related information captured for a long period of time, e.g., weeks or months.

To reflect such assumption in our metric, we need to take *time* into consideration and measure location privacy in a long-term perspective. Hence, instead of one single snapshot, now the metric should be able to base its measurements on multiple snapshots, i.e., a sequence of snapshots taken at successive times with equal interval among them. Measurements based on multiple snapshots should reflect the impact of the accumulated information on the level of location privacy in vehicular communication systems. Intuitively, the more information an adversary obtains, the easier it can draw conclusions with less uncertainties.

For measuring long-term location privacy, several issues need to be addressed such as the challenges to model, process, and reflect the accumulated information in the privacy measurements.

1.3. Contribution

The relation and the impact of the accumulated information on location privacy have not been investigated previously. In this paper, we identify and address this issue by extending the current location privacy metric to take into account accumulated information. Therefore, the metric will become more precise to reflect location privacy of the users of vehicular communication systems. Our contributions in this paper are:

- to develop methods to model accumulated information,
- to design approaches and algorithms to process, propagate, and reflect the accumulated information in location privacy measurements,
- to devise approaches to evaluate the feasibility and correctness of our approaches by various case studies and extensive simulations.

Notice that this paper includes significant extensions of our previous conference paper [24]. In the extensions, we develop a heuristic algorithm to propagate and reflect accumulated information in the metric under extremely dynamic situations. We further evaluate the feasibility of the heuristic algorithm by extensive simulations. With the heuristic algorithm, the location privacy metric is more robust in processing accumulated information and thus more precise to reflect long-term location privacy. Moreover, due to the heuristic algorithm's ability to process accumulated information under dynamic situations, we gain more insights into location privacy, which contributes to the design of future-proof, privacy-preserving vehicular communication systems.

In the remainder of this paper, Section 2 gives the background information on the basics of the trip-based location privacy metric. Section 3 describes the method to model accumulated information in multiple snapshots. Section 4 introduces two exact approaches to process and reflect accumulated information in the metric. Section 5 evaluates the two approaches by case studies and simulations. Section 6 presents the heuristic algorithm followed by the corresponding feasibility evaluation in Section 7. Section 8 discusses the related work, followed by the conclusion in Section 9.

[2] In [18], a vehicle trip is defined as a trip by a single privately operated vehicle (POV) regardless of the number of persons in the vehicle.

2. Metric fundamentals

This section provides the necessary background information on the trip-based location privacy metric introduced in [23].

In vehicular communication systems, each time a vehicle sends a message, it reveals its location in the system. Although there are different levels of granularities, the location information in vehicular communication systems can be categorized into three types, i.e., single locations, tracks, and trips. A single message reveals a single location of a vehicle. A track reveals a vehicle's movement in space and time. To obtain the information on tracks, an adversary can use various algorithms and methods [34,14,11] to "link the dots", i.e., to track a segment of a vehicle's movement by linking the messages belong to the same vehicle. Due to uncertainty, the relation of the messages and the tracks are commonly expressed in probabilities. If an adversary can follow a vehicle from end to end, i.e., from origin to destination, the adversary obtains the information on vehicle trips. Location information only becomes privacy-relevant if it can be linked to identifiable individuals. Since for privacy concerns vehicles are very likely to use pseudonyms in communications [25,22], information on single locations and tracks will be less privacy-sensitive than the information on trips, which can be used to infer an individual's identity and activities.

To measure privacy, we let the metric capture the information on trips and individuals in an arbitrary defined area and time period. Hence the metric virtually takes a "snapshot" of the dynamic vehicular communication systems. The information captured in the snapshot is then modeled in a weighted tripartite graph, shown in Fig. 1. The graph contains three distinct sets of vertices, i.e., *I*, *O*, and *D*, which represent *I*ndividuals, *O*rigins and *D*estinations of the trips. An adversary's knowledge on the linkability of an individual to a set of trips is expressed in probability distributions. The probabilities are used as the weights on the directed edges. For example, p_{jk} is a weight on an edge (v_j, v_k) between the vertices v_j and v_k.

For an individual to make a trip (e.g., $o_1 \rightarrow d_1$), he or she must start from one of the origins, e.g., i_1 from o_1. If the trip from o_1 ends at one of the destinations, it must be possible to link i_1 to d_1 as well. Due to the uncertainty in the information, there can be many of such possible linkings among the vertices. A closed walk or a cycle starting from a vertex i_s and passing vertices $\{o_j, d_k\}$ in the graph has the semantics of i_s's probability p_{jk} to make a trip with origin o_j and destination d_k. By collecting all cycles connected to a particular individual in the graph, we can extract the probability distribution of the linkability of that individual to a set of trips. The probability distribution can be graphically expressed as a hub-and-spoke structure, shown in Fig. 2. The last spoke with probability p^c in the clock-wise order denotes the probability of an individual not making any trips, i.e., "staying at home".

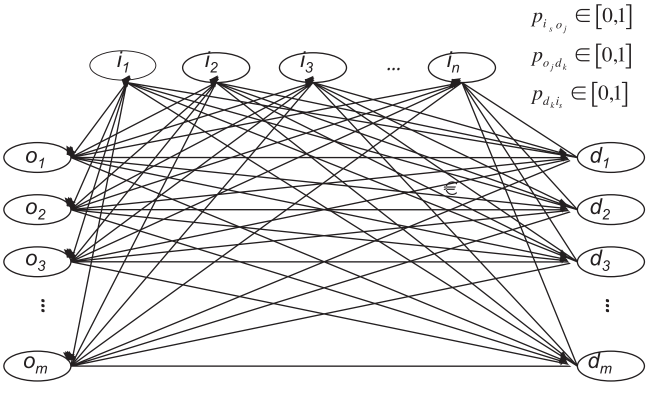


Fig. 1. Snapshot information modeled in weighted tripartite graph.

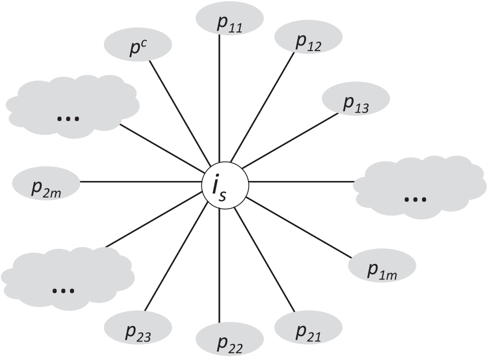


Fig. 2. Extracted probability distribution as hub-and-spoke.

Using the notations specified by the tripartite graph (see Fig. 1), the normalized probabilities $\hat{p}_{jk}$ on each of the spokes are calculated as

$$\hat{p}_{jk} = \frac{p(i_s, o_j)p(o_j, d_k)p(d_k, i_s)}{\sum_{j=1}^{m}\sum_{k=1}^{m} p(i_s, o_j)p(o_j, d_k)p(d_k, i_s)}(1 - p^c)$$

where $p(i_s, o_j)p(o_j, d_k)p(d_k, i_s)$ is the product of the three probabilities on the cycle with vertices i_s, o_j, d_k. The rest of the equation normalize the probability distribution to 1. The complementary probability p^c is calculated as

$$p^c = 1 - \sum_{j=1}^{m} p(i_s, o_j)$$

Applying Shannon's entropy [31], we can quantify the uncertainty in the information about i_s in entropy as

$$H(i_s) = -\left(\sum_{j=1}^{m}\sum_{k=1}^{m} \hat{p}_{jk} log(\hat{p}_{jk}) + p^c \log(p^c)\right)$$

where the logarithm is taken to base 2 to have a unit of *bit*. $H(i_s)$ is used as a quantitative measure of i_s's level of location privacy. The privacy level is directly proportional to the value of entropy, i.e., the higher the entropy, the higher the privacy level, and vice versa. Entropy reaches its maximum if all trips are equally probable. For a snapshot with m^2 O/D pairs, the maximum entropy for each individuals in the snapshot is

$$H_{max} = \log(m^2 + 1)$$

with 1 accounting for the individual not making any trips [23].

3. Accumulated information

Using snapshots enables us to capture privacy-relevant information from vehicular communication systems, which are continuous and dynamic in nature. However, privacy measurements based on a single snapshot only reflect the privacy values in a short period of time. It is reasonable to assume that a determined adversary will collect as much information as possible over a long period of time to work for its advantage. Intuitively, information accumulated over time should help to reveal more facts about the individuals and their vehicle movements.

To reflect this more realistic assumption on the adversary, instead of one snapshot, we extend the metric to include consecutive snapshots. Thus the metric yields measurements on "multiple snapshots". In a single snapshot, the information needed for measuring each individual can be represented by a hub-and-spoke structure shown in Fig. 2. When more snapshots are added to the metric, we can imagine that the information related to an individual i becomes a sequence of hub-and-spoke structures ordered in time as shown in Fig. 3. Notice that only one individual is shown in Fig. 3. But we can imagine that for each of the individuals captured in the snapshots, we can extract the information and build a similar sequence of hub-and-spoke structures. For simplicity in formulations, we will only consider *one* individual i in the rest of the paper. The same formulas and procedures are applicable to any of the other individuals captured in the snapshots. However, in our future work, we will further investigate the *interrelations* among individuals and their impacts on the level of location privacy.

There are several observable characteristics of the consecutive hub-and-spoke structure (in Fig. 3) and the accumulated information contained within. First, i can be linked to different trips from snapshot to snapshot. The differences are in the number, as well as the origins and destinations of the trips. We name the assortment of trips related to i in a snapshot a *trip constellation*. Second, the accumulated information has two dimensions, i.e., the one extends into the diversity of trip constellations, and the other extends along the timeline. Third, given the fact that many individuals use vehicles to fulfill demands on activities on a daily basis [1], accumulated information is likely to contain an individual's *trip patterns*, i.e., regularly occurring trips with the same origins and destinations. By *same trip* we mean two or more trips have the same origin and destination, e.g., the same garage, parking lot, or street parking space, etc.

To model accumulated information in multiple snapshots, we represent the hub-and-spoke structure in a more compact way. Let S be the set of all snapshots and let T be the set of all trips considered for an individual i, then snapshot S_t reflects the relation of i to a set of trips at the time period t. We define S_t to be

$$S_t := \left\{ (T_k, p_k) | T_k \in T, p_k \in]0,1],\ \sum_k p_k = 1, k = 1, \dots, n_t \right\} \tag{1}$$

where (T_k, p_k) is a tuple in which T_k denotes a specific trip (i.e., the kth trip) and p_k is the corresponding probability of that trip. Only trips with probabilities bigger than 0 are assigned to i. As trip constellations can vary in snapshots, we denote the number of possible trips at t by a variable n_t. For the tth snapshot, each T_k represents a spoke and each p_k represents the corresponding probability on that spoke. For simplicity, the last spoke denoting the probability of an individual "staying at home" is also represented as one of the trips. As the metric uses entropy to quantify the uncertainty in the information (cf. Section 2), the calculation of entropy of i at time t can be simplified as

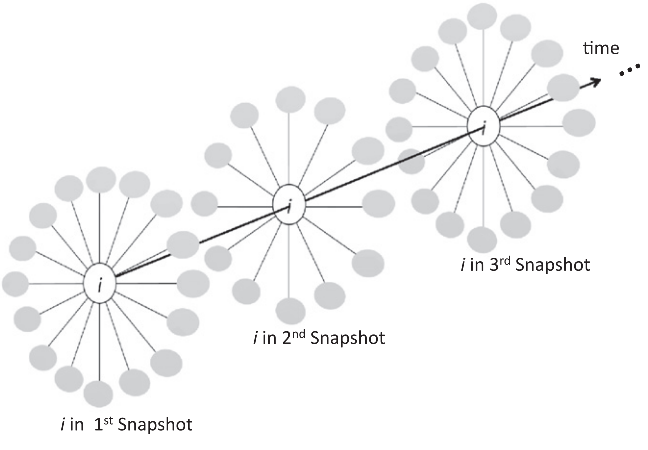


Fig. 3. Multiple snapshots of i in timely-ordered sequence.

$$H_t = -\sum_k p_k log(p_k) \tag{2}$$

where p_k is the probability of the kth trip in S_t.

Consider a simple example in Table 1. We have five consecutive snapshots of an individual i, $t = 1, \dots, 5$. In the 1st snapshot, i is probable to make one of the trips $\{T_1, T_2, T_3, T_4\}$ with corresponding probabilities given in the table. In the 2nd snapshot, i is observed to make a new trip T_5. In the 4th and 5th snapshot, T_3 disappears from the observation. For clarity, non-existing trips (or tuples) are shown as blanks in the table. The probabilities show the adversary's information on the linkability of the vehicle trips to a particular individual over time. However, only one trip at each time (i.e., each row in the table) has actually happened.

Now imagine that the 6th snapshot is captured. Without considering snapshots accumulated in the past, the information contained in S_6 represents the highest uncertainty because all trips are equally probable. However, if we also take into account the five already existing snapshots, our intuition tells us that the historical data might provide us with some useful information.

Based on the observed characteristics, we are aware that to include accumulated information in the metric, we need approaches to *process* the information contained in the snapshots, *propagate* such information along the timeline to the following snapshots, and *reflect* the information in the measurement results.

4. Measurements based on multiple snapshots

In this section, we propose two approaches to measure location privacy in multiple snapshots. Specifically, the existing trip-based location privacy metric is extended from a single snapshot to multiple timely-ordered snapshots. The extension to multiple snapshots takes into account the impact of accumulated information on location privacy.

4.1. Frequency based approach

One way to "learn from the past" is to check whether the same trip has already been observed. Normally vehicle trips have some patterns. For example, we might drive from home to work on a daily basis. Hence the information on the frequency of a particular trip in the past gives hints on how probable the same trip will be repeated in a later point in time. For this we define an auxiliary variable f_k^t that counts how often trip T_k has been linked to i over all snapshots up to time t, i.e., $f_k^t = |\{S_i | S_i \in S, i = 1, 2, \dots, t, \exists (T_k, p_k) \in S_i\}|$. For example, in Table 1, at time $t = 6$, T_1 has occurred 6 times so $f_1^6 = 6$, whereas $f_3^6 = 4$ holds. Then the frequency-adjusted snapshot $\widehat{S}_t^f$ of snapshot $S_t = \{(T_k, p_k) | \dots\}$ can be calculated as

$$\widehat{S}_t^f = \left\{ \left(T_k, \alpha p_k f_k^t\right),\ k = 1, 2, \dots, n_t \right\} \tag{3}$$

where $\alpha = 1 / \sum_k p_k f_k^t$ is a normalization constant calculated by requiring that all probabilities in $\widehat{S}_t^f$ sum up to 1. Consequently, the frequency-adjusted S_6 is

Table 1
A simple example with six consecutive snapshots of i.

t	T_1	T_2	T_3	T_4	T_5
$t = 1$	0.2	0.2	0.3	0.3	
$t = 2$	0.2	0.2	0.3	0.2	0.1
$t = 3$	0.2	0.1	0.3	0.2	0.2
$t = 4$	0.2	0.3		0.2	0.3
$t = 5$	0.2	0.2		0.3	0.3
$t = 6$	0.2	0.2	0.2	0.2	0.2

$$\widehat{S}_6^f \approx \{(T_1, 0.22), (T_2, 0.22), (T_3, 0.15), (T_4, 0.22), (T_5, 0.19)\}$$

Comparing $\widehat{S}_6$ with S_6, the probability distribution changes from equal to unequal. The corresponding entropy calculated by (2) is also decreased from 2.32 for S_6 to 2.31 for $\widehat{S}_6$, i.e., the accumulated information helps to slightly reduce the uncertainty of the current information.

However, using only the frequency of a particular trip does not consider the actual probability of that trip in each snapshot. Therefore, we lose information if we use only frequencies to adjust a snapshot. For example, in Table 1, though T_1 and T_4 have the same value of f_k^t, T_4 has a higher average probability than T_1. To also include actual probability values in the frequency-adjustment, we rewrite (3) as

$$\widehat{S}_t^w = \left\{ \left(T_k, \alpha p_k w_k^t\right), k = 1, 2, \ldots, n_t \right\} \tag{4}$$

in which we replace f_k^t by the average probability of the same trip, i.e., $w_k^t = \left(\sum_i p_k^i\right) / f_k^t$ for $i = 1, 2, \ldots, t$. The normalization constant α is changed to $\alpha = 1/\sum_k p_k w_k^t$, accordingly. The probability of a non-existing trip (e.g., T_5 at $t = 1$) is treated as 0, so the equation can be kept in a generic form. Using (4), $\widehat{S}_6^w$ turns out to be

$$\widehat{S}_6^w \approx \{(T_1, 0.18), (T_2, 0.18), (T_3, 0.24), (T_4, 0.21), (T_5, 0.19)\}$$

with an entropy value of 2.31. The result again shows that accumulated information, in terms of average probabilities of specific trips, can change the current probability distribution and thus modify the level of uncertainty. Furthermore, the result reflects the value of probabilities of the trips in the past. For example, T_3 has the highest probability because it has been associated with high probabilities in the past (i.e., 0.3 at $t = 1, 2, 3$). On the other hand, even though T_1 and T_2 appear at all snapshots, the relatively low probabilities in the past cause these two trips to have the lowest value in the probability distribution of $\widehat{S}_6^w$ (i.e., both are 0.18). A more extensive evaluation of this approach will be given in Section 5.

4.2. Bayesian approach

Our second approach to process, propagate, and reflect the accumulated information is to use the Bayesian method to infer information from the historical data.

4.2.1. Bayesian method

In principle, Bayesian method uses evidence to update a set of hypotheses expressed numerically in probabilities. The core of Bayesian method is the Bayes' theorem. Let h_k be the *kth hypothesis* of a complete set of *hypotheses H*,[3] the Bayes' theorem can be written as a function of h_k as

$$P(h_k|E) = \frac{P(E|h_k)P(h_k)}{\sum_k P(E|h_k)P(h_k)} \tag{5}$$

in which E is the evidence. $P(h_k|E)$ is the *posterior probability* of h_k because it is the conditional probability of h_k given the evidence E. $P(E|h_k)$ is the conditional probability of observing the evidence E if the hypothesis h_k is true. $P(h_k)$ is the *prior probability* of h_k because it is the probability of h_k before it is updated by E. The denominator in (5) is the sum of probabilities of observing the evidence E under all possible hypotheses.

The above description accounts for updating the hypotheses once. When applying Bayes' theorem to situations in which hypotheses are continuously updated by new evidence, the following steps are usually involved:

- Initially define an *exhaustive* and *mutually exclusive* hypotheses H^0.
- Before receiving new evidence E, generate a prior hypotheses H^-. H^- is the same as H^0 before the first update.
- After receiving the evidence E, calculate the posterior hypotheses H^+ using (5). H^+ will be used as the prior hypotheses H^- for the next update.

In Bayesian method, the initial hypotheses can be subjective, i.e., we can assign probabilities to a hypotheses according to some preliminary knowledge. If there are enough evidence, the hypotheses will eventually be updated towards the objective truth.

The characteristics of the modeled accumulated information make it appropriate to apply Bayesian method. Specifically, S_t contains a set of possible trips and the corresponding probabilities. Each of the trips can be regarded as a hypothesis of an individual making that trip. S_t includes all the possible trips and only one of them can be true. Therefore, the hypotheses are complete and mutually exclusive. The corresponding probabilities in the snapshots are the evidence of those trips from observations. At each time period, S_t contains a new set of evidence, which can be used to update the hypotheses.

However, there is still an issue to be solved before we can apply Bayesian method. It is very likely that S_t contains dynamic trip constellations, e.g., $\{T_1, T_2, T_3, T_4\}$ in S_1 and $\{T_1, T_2, T_3, T_4, T_5\}$ in S_2 (see Table 1). The implication of such dynamics is that the set of hypotheses H will be different from snapshot to snapshot. As Bayesian method works on a fixed set of hypotheses, i.e., it does not consider adding or removing one or more hypotheses during the evidence updating process, we need a "smart" solution to apply Bayesian method to solve this problem.

4.2.2. Exact algorithm

The solution is Algorithm 1 shown below. In general, for a given snapshot at time t, the algorithm calculates the modified probability distribution for this snapshot using the Bayesian method. Specifically, for each existing snapshot S_j, $j = 1, 2, \ldots, t$, the algorithm generates a prior hypotheses H_j^- and uses the probability in S_j to calculate the posterior hypotheses H_j^+. The algorithm stores each H_j^+ in a belief table B. Entries in B are called *Belief* because they are posterior hypotheses updated by evidence that express the level of confidence of the algorithm on their "correctness". The algorithm also keeps tracks of *the latest posterior hypotheses with the same trip constellation*. For example, S_6 has the same trip constellation as S_3 in Table 1, so H_3^+ will be the latest posterior hypotheses with exactly the same trip constellation to S_6. Informally, we use $H_j^+ \equiv_{lph} S_i, j < i$ to denote that H_j^+ is the latest posterior hypotheses of S_j in B with a trip constellation that exactly matches the one in snapshot S_i.

Algorithm 1. Calculate $\widehat{S}_t$ using Bayesian method

```
Input: snapshots until time $t, S_1, \ldots, S_t$
Output: snapshot at time $t$ with modified probability distribution, $\widehat{S}_t$
for $i = 1$ to $t$ do
    if found $H_j^+ \equiv_{lph} S_i$ then
        use $H_j^+$ as $H_i^-$
    else
        assign equal probabilities to $H_i^-$
    end if
    update $H_i^-$ with the probabilities in $S_i$, the result is $H_i^+$
    add $H_i^+$ to $B$
end for
replace the probability distribution in $S_t$ with $H_t^+$ to obtain $\widehat{S}_t$,
  return $\widehat{S}_t$
```

[3] Notice that the notation H is conventionally used for both entropy and hypotheses. We keep the convention and assume that the meaning should be clear from the context.

To calculate $\widehat{S}_t$, the algorithm takes all existing snapshots up to time t. Before processing a new snapshot S_i, the algorithm first consults B for the latest posterior hypotheses with the same trip constellation as S_i. If found, the posterior hypotheses H_j^+ will be used as the prior hypotheses H_i^- for the current snapshot S_i. If not found, the algorithm assigns H_i^- with equally distributed probabilities. The rationale is that we assign probabilities without any prejudices to the initial hypotheses, believing that the evidence will eventually update the hypotheses towards the objective truth. Then H_i^- is updated by S_i to generate H_i^+. Afterwards, H_i^+ is added to B. Notice that for efficiency, B only needs to keep the latest H^+ with a unique trip constellation. Finally, H_t^+ replaces the probability distribution in S_t to have $\widehat{S}_t$. $\widehat{S}_t$ reflects the current beliefs expressed in probabilities, which have been continuously updated by new evidence, on each of the trips in the trip constellation in S_t. In line 7 of the algorithm, when using the probabilities in S_i to update the prior hypotheses, the notions in (5) can be substituted and rewritten as

$$p_k^{H_i^+} = \frac{p_k^{S_i} p_k^B}{\sum_k p_k^{S_i} p_k^B} \tag{6}$$

in which $p_k^{H_i^+}$ and $p_k^{S_i}$ are the probabilities of the kth trip in H_i^+ and S_i, respectively. p_k^B is defined as

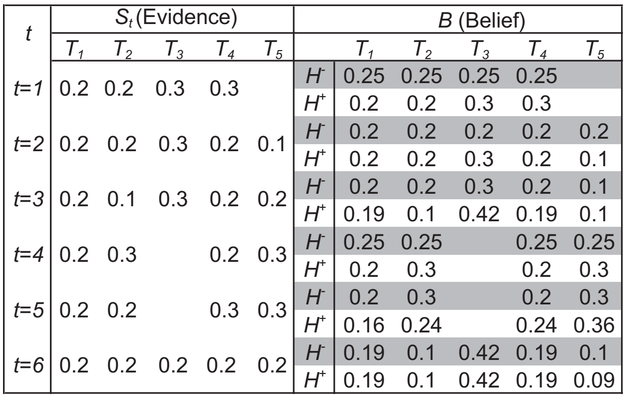
$$p_k^B = \begin{cases} p_k^{H_j^+} & \text{if } H_j^+ \equiv_{lph} S_i \text{ found} \\ \frac{1}{n_i} & \text{if } H_j^+ \equiv_{lph} S_i \text{ not found} \end{cases} \tag{7}$$

in which $p_k^{H_j^+}$ is the probability of the kth trip of the latest posterior hypotheses in B with the same trip constellation as S_i, and n_i is the number of trips in S_i.

We demonstrate how the algorithm works by calculating the same example from Table 1. The results at each time period are shown in Fig. 4. We also include H^- at each time period to show how they are assigned and how they are updated by S to generate H^+. For example, at $t = 2$, since the trip constellation of S_2 appears for the first time, H^- is assigned a equal probability distribution. Look further down, at $t = 6$, the latest snapshot with the same trips constellation can be found at $t = 3$. So the posterior probabilities H^+ at $t = 3$ is copies to the prior probabilities H^- at $t = 6$. $\widehat{S}_6$ has the same value as H^+ at $t = 6$

$$\widehat{S}_6 \approx \{(T_1, 0.19), (T_2, 0.1), (T_3, 0.42), (T_4, 0.19), (T_5, 0.09)\}$$

with entropy of 2.08. Comparing with the results from the frequency based approaches in Section 4.1, we witness a more dramatic change in the probability distribution, as well as a sharp decrease in entropy. The results show that Bayesian approach is more effective to reflect the impact of accumulated information than the frequency based approaches. We will further compare and evaluate these approaches in the next section.



t	S_t (Evidence)					B (Belief)					
	T_1	T_2	T_3	T_4	T_5		T_1	T_2	T_3	T_4	T_5
t=1	0.2	0.2	0.3	0.3		H^-	0.25	0.25	0.25	0.25	
						H^+	0.2	0.2	0.3	0.3	
t=2	0.2	0.2	0.3	0.2	0.1	H^-	0.2	0.2	0.2	0.2	0.2
						H^+	0.2	0.2	0.3	0.2	0.1
t=3	0.2	0.1	0.3	0.2	0.2	H^-	0.2	0.2	0.3	0.2	0.1
						H^+	0.19	0.1	0.42	0.19	0.1
t=4	0.2	0.3		0.2	0.3	H^-	0.25	0.25		0.25	0.25
						H^+	0.2	0.3		0.2	0.3
t=5	0.2	0.2		0.3	0.3	H^-	0.2	0.3		0.2	0.3
						H^+	0.16	0.24		0.24	0.36
t=6	0.2	0.2	0.2	0.2	0.2	H^-	0.19	0.1	0.42	0.19	0.1
						H^+	0.19	0.1	0.42	0.19	0.09

Fig. 4. Example of Algorithm 1.

5. Evaluation

5.1. Evaluation criteria

Our goal is to evaluate whether the privacy metric, now with the extension for accumulate information, can *really* reflect the underlying value of user location privacy in vehicular communication systems. For this purpose, we define two use-case-based evaluation criteria. The use cases specify scenarios likely to happen in vehicular communication systems. The criteria are the expected impacts of the scenarios on user location privacy. We simulate the use cases. The simulation results will then be compared with the criteria. The results give us clues as how good the metric can be used to measure the long-term location privacy in vehicular communication systems. We define the evaluation criteria as

1. if an individual has irregular trips with quite different origins and destinations at each time, accumulated information should provide less or even no additional information;
2. if an individual has regular trip patterns, accumulated information should provide additional information. With this additional information, it should be possible to detect an individual's trip patterns.

In our metric, the uncertainty of information is quantified in entropy. A decrease in entropy indicates that additional information leads to a decrease in uncertainty, i.e., a decrease in user location privacy.

5.2. Evaluation setup

We identify three parameters to have main influences on the outcome of our location privacy metric. Among them are the trip constellations in each snapshot, their corresponding probability distributions, and the number of snapshots. First, the trip constellation specifies the number of trips and their appearances observed in a snapshot. Second, the probability distribution of the corresponding trips specifies the information captured by a snapshot. Third, the number of snapshots specifies the duration of the measurement. Implicitly, it specifies the amount of accumulated information available to the metric. By specifying these parameters, we can create use cases to check whether the metric meets the evaluation criteria. The use cases are the mock-ups of scenarios in the real world. We have created a set of use cases to evaluate the metric. However, due to the page limit, we include only three selected use cases in this paper.

The first two use cases represent two opposite extremes. In the 1st use case, each of the snapshots has different trip constellations. A series of such snapshots contain irregular trips. We imagine that such scenario will happen, if either an individual makes different trips each time or the observation of an adversary is of very bad quality such that there are high confusions or uncertainties associated with the obtained information. For each snapshot, the simulation first generates a random trip index in the range of 1 to 100, then it generates the corresponding probabilities. To avoid any subjectiveness in the probability assignment, the probabilities are randomly generated from the uniform distribution. The process is repeated to generate 60 snapshots with dynamic trip constellations.

In the 2nd use case, all snapshots have the same trip constellation. However, only one trip in the constellation actually happens. Hence the snapshots contain a regular trip hidden among other observed trips. This scenario happens if an adversary has correctly observed the regular trip such as driving from home to work, but somehow cannot distinguish it from other trips observed at the same time. To simulate such scenario, we generate 60 snapshots with a trip index from 1 to 100. We set trip T_1 in the constellation as the one actually happened and assign a fixed probability, called the *p*-value, to it. The remaining 99 trips are assigned with probabilities from the uniform distribution. We set the *p*-value to be the average, i.e., $p = 0.01$, and normalize the probabilities of the remaining 99 trips to $(1 - p_1) = 0.99$. The choice and impact of the *p*-value will be further elaborated in Section 5.3.

The 3rd use case locates on the spectrum between the two extreme cases described before, and contains several re-occurring trips. It is a mock-up of a more realistic and common scenario as specified in Table 2. Imagine there is a series of snapshots capturing an individual's vehicle trips for several weeks. All snapshots cover a time period somewhen in the morning, so all the trips are from home to somewhere. We simulate this by four trip constellations. The first trip constellation for snapshots (Mon.–Wed.) contains trips $\{T_1, T_4, \ldots, T_{100}\}$. We set T_1 as the trip actually happened and assign a *p*-value of 0.012. The corresponding probabilities of $\{T_4, T_5, \ldots, T_{100}\}$ are assigned with probabilities from the uniform distribution, and normalized to $(1 - p_1) = 0.988$. The second trip constellation for snapshots (Thur.–Fri.) contains trips $\{T_2, T_4, \ldots, T_{100}\}$. We set T_2 as actually happened and also assign a *p*-value of 0.012, and the normalized probabilities to $\{T_4, T_5, \ldots, T_{100}\}$. The third trip constellation for snapshots (Sat.) contains trips $\{T_3, T_4, \ldots, T_{100}\}$. We assign a *p*-value of 0.012 to T_3 and the normalized probabilities to $\{T_4, T_5, \ldots, T_{100}\}$. The last trip constellation for snapshots (Sun.) has trips $\{T_4, T_5, \ldots, T_{100}\}$. To simulate random destinations on Sundays, we assign all the trips with probabilities from the uniform distribution. We repeat the process and generate 56 snapshots to simulate 8 weeks of snapshots with re-occurring trips.

During the simulation, we generate snapshots corresponding to the use cases and feed them to the location privacy metric. The outcome of the metric is analyzed along the evaluation criteria. For our analysis, we choose the following entropy values: (1) H_{max}, the theoretical maximum entropy based on each single snapshot; (2) H, the entropy based only on single snapshot; (3) H_f, the entropy based on the snapshots modified by frequencies of occurrence; (4) H_w, the entropy based on the snapshots modified by average probabilities; and (5) H_B, the entropy based on the snapshots modified by Bayesian approach.

To analyze the impact of accumulated information on the actual level of uncertainty, we further define H_d as a measurement of the *decrease in uncertainty*

$$H_d = \frac{H_B - H}{H} 100\% \tag{8}$$

which bases the calculation on the difference of the entropy using Bayesian approach and the entropy based on single snapshot without any additional information.

Table 2
Third use case setup.

Scenario	Simulation	
Vehicle trips	Trip constellation	Probability assignment
Home to office A (Mon.–Wed.)	$\{T_1, T_4, \ldots, T_{100}\}$	$p_1 = 0.012$, $\sum_{i=4}^{i=100} p_i = 1 - p_1$
Home to office B (Thur.–Fri.)	$\{T_2, T_4, \ldots, T_{100}\}$	$p_2 = 0.012$, $\sum_{i=4}^{i=100} p_i = 1 - p_2$
Home to shopping mall C (Sat.)	$\{T_3, T_4, \ldots, T_{100}\}$	$p_3 = 0.012$, $\sum_{i=4}^{i=100} p_i = 1 - p_3$
Home to a random destination (Sun.)	$\{T_4, T_5, \ldots, T_{100}\}$	Random, $\sum_{i=4}^{i=100} p_i = 1$

5.3. Simulation

Fig. 5 shows the simulation result from the 1st use case, in which each snapshot contains a randomly generated trip constellation. We can see from the figure that the entropies of H, H_f, H_w, and H_B are so close that they overlap each other most of the time. This means neither frequency based approaches nor Bayesian approach are able to benefit from the accumulated information. Besides, these entropies are very close to the upper-bound H_{max}, due to the fact that the probabilities in each snapshot are from uniform distributions. For illustrative reason, the lower part of the figure includes a bar chart showing the number of trips in each of the snapshots. Notice that the actually trip constellations are not shown in the bar chart.

Fig. 6 shows what the metric result of the 2nd use case, which simulates the scenario that a regular trip is blurred by other false observations in each snapshot. The result shows that the frequency based approaches can barely reflect the accumulated information. As a result, H_f and H_w mostly overlap H, with the exception that H_w has slightly lower entropies at the first few snapshots. On the other hand, Bayesian approach has significantly decreased the en-

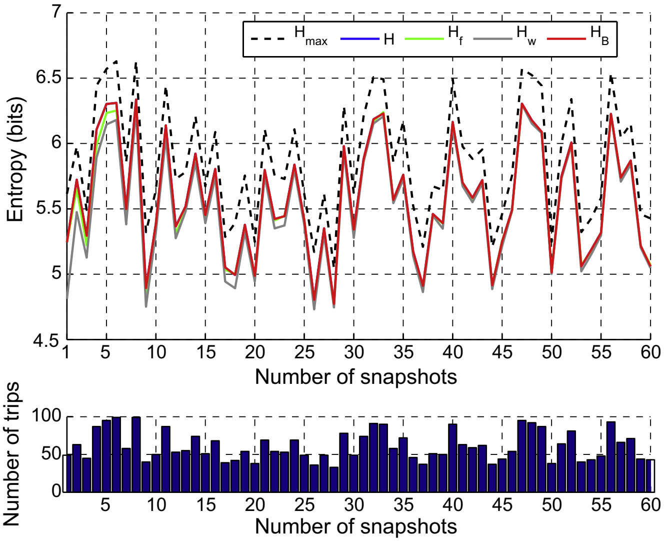


Fig. 5. Entropy of irregular trips.

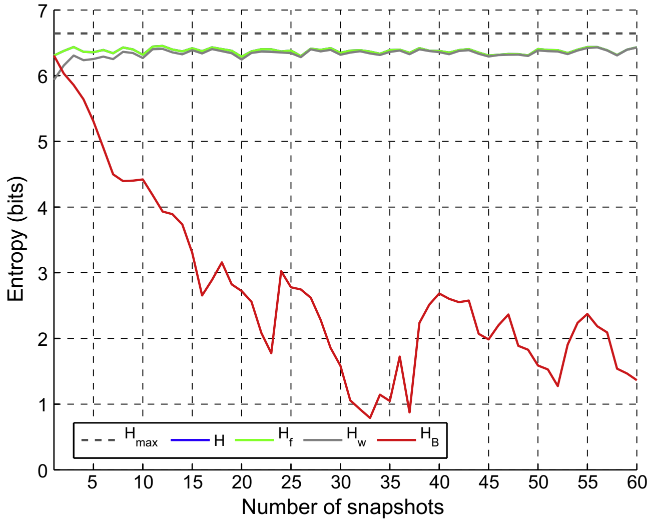


Fig. 6. Entropy of regular trips.

tropy level from 6.3 bits to as low as 0.79 bits at the 33rd snapshot. Obviously, at 0.79 bits, the uncertainty is very low, i.e., the privacy level is very low. The shape of the curve of H_B suggests that Bayesian approach is able to process and benefit from the accumulated information.

Fig. 7 shows the simulation result from the 3rd use case. The 3rd use case simulates weekly re-occurring trips. H_f and H_w have similar outcomes as those in Fig. 6, i.e., frequency based approaches cannot really benefit from accumulated information in the long run. Again, Bayesian approach has significantly decreased the entropy value. Interestingly, this time the curve of H_B has a cascading and downward shape. The reason is that we have simulated four types of re-occurring trips in this use case. The first three trips are regularly occurred trips and the fourth one (i.e., the Sunday trip) is chosen to be random. Therefore, while the overall curve of H_B demonstrates a downward trend, the entropies corresponding to the first three trips decrease much faster than the entropy of the Sunday trip. Notice that the entropy of the Sunday trip also exhibits a downward trend. The reason is that even though the probability distributions of the Sunday trip are from the uniform distribution, their values are slightly different among each others. As a result, the probabilities are modified by Bayesian approach towards a non-uniform distribution. In other words, given consecutive snapshots, Algorithm 1 regards some of the trips are "more likely to have happened" than others. The result again demonstrates that Bayesian approach can take advantage of the accumulated information caused by regularly occurring trips.

As the next step, we use H_d to analyze the decrease in uncertainty in each of the use cases. Since a new set of random values is generated each time a use case is simulated, we run each use case 100 times and calculate the mean values to take into account the effects of the variations of random variables. The results are plotted in Fig. 8. For irregular trips, taking more snapshots into the metric does not decrease information uncertainty. In some cases, it even increases the level of uncertainty. This means based on the metric, accumulated information does not provide any additional information due to the randomness in the captured information. For regular trips, we can see that there is a constant decrease in uncertainty as more and more snapshots are added in the sequence. The decrease reaches −84.6% at the 60th snapshot. The outcome of the metric shows that with regular trips, accumulated information can significantly reduce the uncertainty in the information related to user location privacy. For re-occurring trips, despite the spikes on each Sunday due to the randomness of the trips on that day, there is also a constant decrease in uncertainty as the time elapses. Because there are several regular trip patterns involved in this use case, the speed of the decrease in uncertainty is slower than the use case with regular trips. The result demonstrates again that the accumulated information can cause considerable decreases in the level of uncertainty, i.e., users' location privacy. Notice that the shape of the curves in Fig. 8 correspond to those appeared in Figs. 5–7, i.e., the observations we made before on single simulation result also hold in general cases.

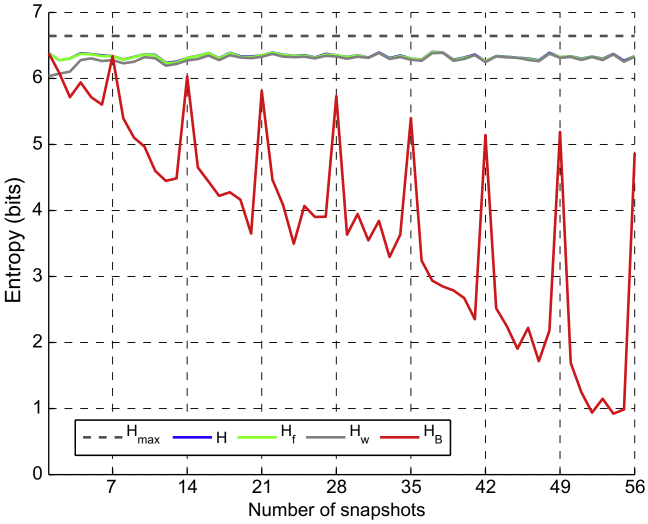


Fig. 7. Entropy of re-occurring trips.

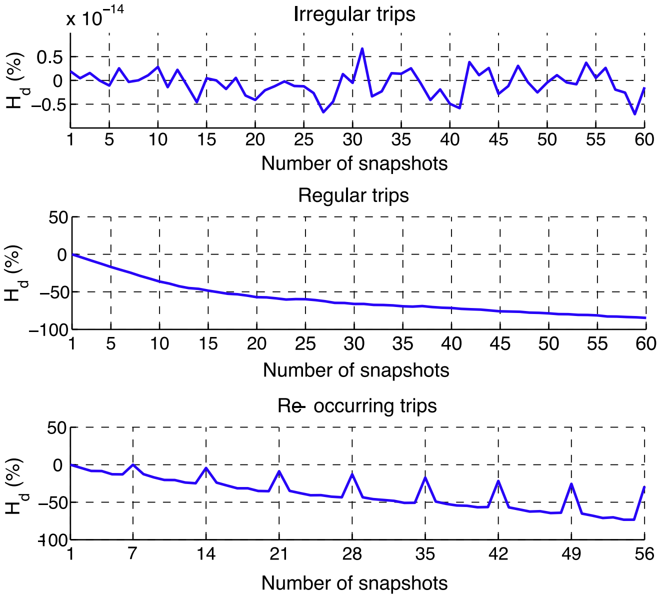


Fig. 8. Change of uncertainty.

We know that the main reason behind the significant decrease in uncertainty is because of the application of Bayesian method in Algorithm 1. Algorithm 1 processes, propagates, and reflects the accumulated information by continuously updating the probabilities in each hypotheses after a new set of evidence contained in a snapshot is received. The updated hypotheses are kept in the belief table B. As a result, the probability distributions in the belief table converge toward the "real happened" trips. The changing of probability distributions leads to lower entropy values and hence a decrease in uncertainty. However, so far we have not shown whether the algorithm is able to update probability distributions in a correct way. We test the correctness of Algorithm 1 by tracing the change of beliefs in the belief table. In this sense, the second and the third use case are quite similar. Therefore, we only show the study on the 2nd use case here. Same as before, we assign the first trip as the one actually happened. Furthermore, we assign different probabilities to study the effect of the p-values on the performance of the algorithm. The p-values are $\{0.009, 0.01, 0.011\}$, which correspond to 10% lower than the average, the average, and 10% higher than the average of the probability of the 100 trips in the trip constellation. Again, we run the simulation 100 times to account for the variations in the random dataset and calculate the means of the first trip in the 100 simulation runs.

Fig. 9 shows the result. At 10% below the average, Algorithm 1 almost fails to detect the trip. However, as soon as the p-value is of the average value, there is a steady rise of the probability. If we assume that 0.5 is the threshold to select a trip as the one really happens, the first trip will be selected at the 59th snapshot. Only slightly increase the p-value 10% higher, the probability of the first trip exhibits a sharp rise and passes the 0.5 threshold at the 32nd snapshot.

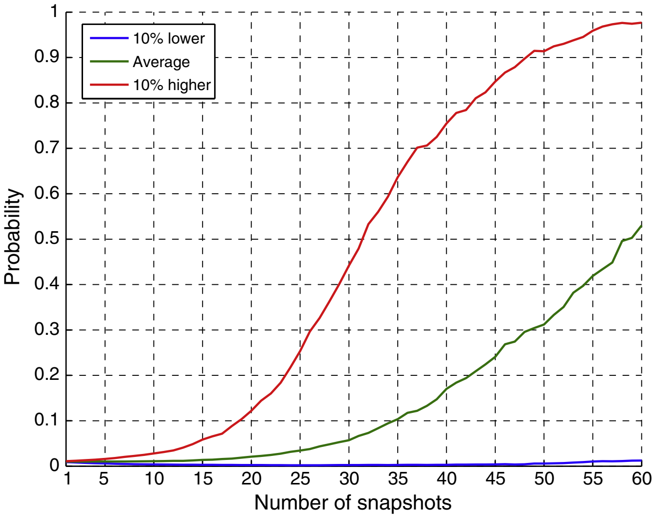


Fig. 9. Change of beliefs with different *p*-values.

From the simulation results, we conclude that our location privacy metric and the related approach meet both evaluation criteria defined in Section 5.1.

6. Heuristic algorithm for dynamic trip constellations

Algorithm 1 in Section 4.2.2 relies on finding posterior hypotheses (i.e., H^+) of the previous snapshots with exactly the same trip constellations to propagate the beliefs. Therefore, it functions well on snapshots containing regular trip patterns, in which snapshots with same trip constellations appear frequently. Imagine if an individual can be linked to different sets of trips in each of the snapshots, Algorithm 1 will likely wait for a very long period of time until it has the same trip constellation again. In the worst case, a specific trip constellation might even never happen more than once. The simulation results in Figs. 5 and 8 have already shown the negative effect of snapshots with dynamic trip constellations.

To have a robuster way to process and reflect accumulated information in the privacy measurements, in this section, we develop a heuristic algorithm as an important extension to Algorithm 1 and evaluate its feasibility to work with dynamic trip constellations in Section 7.

6.1. Finding an adequate measurement of similarity

A trip constellation is a set of trips associated with a specific individual in a snapshot. The biggest difference in the heuristic algorithm is that, instead of searching for a snapshot with an *identical* trip constellation, now the heuristic algorithm searches for a snapshot with the most *similar* trip constellation. Then the beliefs (i.e., the posterior hypotheses) from the previous snapshot are used an input to construct the prior hypotheses of the later snapshot. Recall that originally, Bayesian method is intended to work on a *fixed* set of exhaustive and mutually exclusive hypothesis during the evidence update process (cf. Section 4.2.1), our solution to tackle the trip dynamics is an heuristic approach. However, our rationale is that, if the beliefs are propagated between two snapshots with the most similarities, the distortions during the belief propagation will be kept at a minimum. In fact, because two identical trip constellations are the most "similar" ones, a search for the most similar will return the identical trip constellation, if it exists.

The question arises as "how to find an adequate notion of similarity?" Intuitively, two snapshots are more similar, the more trips they have in common. To quantitatively express the concept of "similarity", we can count the number of trips presented in both snapshots, as well as those only appeared at respective ones. An elegant way to count the occurrence of trips in a snapshot is to convert the set-based snapshot representation in (1) to binary strings. Let n be the number of all *unique* trips appeared in all snapshots up to S_t, formally: $n = |\bigcup S'_i|$, $i = 1, 2, \dots, t$ with $S'_i = \{T_k | \exists (T_k, p_k) \in S_i\}$. Then the trip constellation of S_i expressed by a binary string c_i is

$$c_i = [T_1, T_2, \dots, T_n] \quad \text{with } T_k = \begin{cases} 1 & \text{if } \exists (T_k, p_k) \in S_i \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

in which we use 1 for an existing trip and 0 for a non-existing trip within snapshot S_i. Notice that n is a constant, so all binary strings will have the same length of n bits. This also means that to convert the trip constellation in a snapshot to a binary string, we might need to pad all snapshots retrospectively to have the same length for all c_i, $i = 1, 2, \dots, t$. For example, in Table 1, at $t = 1$, c_1 will be $[1, 1, 1, 1]$, while at $t = 2$, by retrospective padding, c_1 becomes $[1, 1, 1, 1, 0]$ and c_2 will be $[1, 1, 1, 1, 1]$.

For two binary strings with equal length, the *hamming distance* [15] is a measure of the number of positions where there are different bits. For example, the hamming distance of $[1, 1, 1, 1, 0]$ and $[1, 1, 1, 1, 1]$ is 1. Therefore, we can use hamming distance to measure the similarity of two snapshots. Hence, the hamming distance between two snapshots (or more precisely, the trip constellations in the two snapshots) expresses explicitly the difference in their trip constellations. The more trips in common, the smaller the hamming distance, hence the more similar are the two snapshots.

Therefore, for S_t, we can calculate the hamming distances from S_t to each of the previous snapshots $S_1, S_2, \dots, S_{t-1}$. We regard the snapshot with the smallest hamming distance *the most similar snapshot* to S_t. In case more than one snapshot have the same hamming distances, we choose the latest one. This is also in accordance with Algorithm 1, which looks for the latest posterior hypotheses with the same trip constellation.

6.2. Constellation fitting

After finding the most similar snapshot, the next question is "how to propagate the beliefs between two snapshots so there will be minimum distortions?" In case of the exact match as in Algorithm 1, this is done by taking the whole posterior hypotheses H^+ of the previous snapshot from the belief table B, and using them as the prior hypotheses H^- of the current snapshot.

Knowing that the trip constellations in the two snapshots will most likely to match only partially, we need to find a solution to align the hypotheses so we can propagate the probabilities from H^+ to H^-. We call this the "constellation fitting" problem, i.e., to shape and fit the current trip constellation into the previous one such that the current snapshot can heuristically inherit the associated hypotheses of the previous one with minimum distortions.

To propagate beliefs between two sets of similar but not exact matching hypotheses with minimum distortions, we made two decisions in our heuristic algorithm. The feasibility will be evaluated by simulations in Section 7. The two decisions are:

1. if a posterior hypothesis of a trip exists, it will be used as the prior hypothesis for the same trip in the current snapshot;
2. otherwise, the prior hypothesis of the trip in the current snapshot will be given an equally distributed probability.

As the probabilities in a hypotheses should sum up to 1, we also normalize the probability distribution in a hypotheses in the process when it is necessary.

Although two snapshots might be similar with respect to their trip constellations, there are various ways that such similarities can be. Because the various relations between two trip constellations directly influence the probability assignment for prior hypotheses in the heuristic algorithm, we will first elaborate on the possible relations and their corresponding probability assignments, and present the detailed description of the algorithm afterwards.

Let S_i be the current snapshot and S_j be the most similar snapshot in the past, $j < i$, we can derive five kinds of relations between S_i and S_j. The first one is the exact match, i.e., the trip constellations in S_i and S_j are identical, which is the case considered in Algorithm 1. Beside the exact match, the other four relations are illustrated in Fig. 10. For simplicity, in the following description, we treat a snapshot as a set containing only trips, and omit the corresponding probabilities (cf. (1)), e.g., $S_i = \{T_1, T_2, \ldots, T_{n_i}\}$. Moreover, we use H_i^- to denote the prior hypotheses of S_i and H_j^+ for the posterior hypotheses of S_j stored in the belief table B. Hence we have four relations as:

(a) *Disjoint relation* might happen when S_j is most similar to S_i, despite S_j and S_i have completely different sets of trips. For example, if $S_1 = \{T_1, T_2\}$ and $S_2 = \{T_3, T_4\}, S_1$ will be the "choice" for S_2 because c_1 has the smallest hamming distance to c_2. In this case, S_i has a complete new trip constellation, and H_i^- will not inherit any beliefs from S_j. Hence H_i^- are assigned equal probabilities, which is similar to line 5 in Algorithm 1.

(b) *Intersected relation* might be the most occurring relation for two similar-but-not-identical snapshots. In this relation, S_i and S_j will share some trips in common, but have different sets of trips to their own at the same time. For example, $S_1 = \{T_1, T_2, T_3, T_4\}$ and $S_2 = \{T_1, T_3, T_5, T_6\}$ have an intersection of $\{T_1, T_3\}$. The unique trips to S_2 is $S_2 \setminus S_1 = \{T_5, T_6\}$. To assign probabilities to H_i^-, we let the trips in the intersection inherit the probabilities of the same trips in H_j^+, and the rest of the trips in H_i^- are equally assigned the remaining probability.

(c) In *subset relation*, S_i is a subset of S_j, i.e., all trips in S_i are also in S_j. The trips in H_i^- will inherit all corresponding probabilities of the same trips in H_j^+. Since we have only a subset of H_j^+, we need to normalize the probabilities in H_i^- to 1.

(d) In *superset relation*, S_i is a superset of S_j, i.e., S_i includes all trips in S_j plus some other trips. To assign probabilities, we first let all trips in S_i but not in S_j (i.e., $S_i \setminus S_j$) to have the equal probabilities, so these trips can have unbiased initial hypotheses. Then we let all trips also in S_j inherit the corresponding probabilities from H_j^+. We further normalize the inherited probabilities to the remaining probability in H_i^-. For example, for $S_1 = \{T_1, T_2, T_3\}$ and $S_2 = \{T_1, T_2, T_3, T_4, T_5\}, T_4$ and T_5 in H_2^- will each have a probability of $\frac{1}{5}$, the probabilities of T_1, T_2, T_3 will be taken from H_1^+ and normalized to $\frac{3}{5}$.

Notice that at any time, S_i will contain only two possible sets of trips: the trips as also in S_j and the trips not in S_j. The design of the probability assignment for H_i^- reflects our idea to use the existing beliefs while avoiding prejudicing the hypotheses of "newly-appeared" trips.

6.3. Heuristic algorithm

The heuristic algorithm has a similar structure as Algorithm 1, except the search for similar snapshots and the probability assignment for the prior hypotheses. The details of the heuristic algorithm are given in Algorithm 2.

Notice that line 5 in Algorithm 2 searches for the *latest* snapshot with the trip constellation of *minimum* hamming distance to S_i. Line 6 to line 16 is the probability assignment for the prior hypotheses H_i^-. Also notice that line 8 to line 15 correspond to the four relations outlined in Fig. 10. Furthermore, because two snapshots with an identical trip constellation have a hamming distance of 0, and Algorithm 2 always searches for the latest snapshot with the smallest hamming distance, the heuristic algorithm will function exactly as the "exact" algorithm (i.e., Algorithm 1) when there are snapshots with same trip constellations in the series. In other words, Algorithm 2 is fully compatible with Algorithm 1.

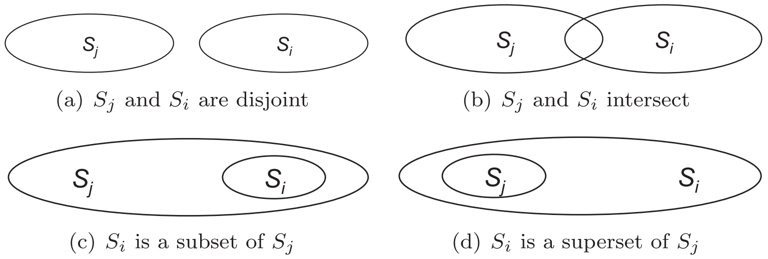
Algorithm 2. Heuristic algorithm to calculate $\widehat{S}_t$

```
Input: snapshots until time $t, S_1, \ldots, S_t$
Output: snapshot at time $t$ with modified probability distribution,
    $\widehat{S}_t$
 1: for $i = 1$ to $t$ do
 2:   for $l = 1$ to $i$ do
 3:     convert trip index in $S_l$ to binary string $c_l$ and pad to equal
        length
 4:   end for
 5:   find $c_j$ with minimum hamming distance to $c_i, j < i, i - j$ is
      minimum
 6:   if hamming distance = 0 then
 7:     $H_i^- = H_j^+$
 8:   else if $S_i \bigcap S_j = \emptyset$ then
 9:     assign trips with probability of $\frac{1}{|S_i|}$
10:   else if $S_i \bigcap S_j \neq \emptyset$ then
11:     assign trips in $S_i \bigcap S_j$ with probabilities from $H_j^+$, and trips
        in $S_i \setminus S_j$ with probability of $\frac{(1-\sum p_k)}{|S_i \setminus S_j|}$
12:   else if $S_i \subseteq S_j$ then
13:     assign trips with probabilities of $\frac{p'_k}{\sum p'_k}, p'_k$ are probabilities
        from $H_j^+$
14:   else if $S_i \supseteq S_j$ then
15:     assign trips in $S_i \setminus S_j$ with probability of $\frac{1}{|S_i|}$, and trips in $S_j$
        with probabilities of $(1 - \sum p_c)p'_k, p'_k$ are probabilities from
        $H_j^+$
16:   end if
17:   update $H_i^-$ with the probabilities in $S_i$, the result is $H_i^+$
18:   add $H_i^+$ to $B$
19: end for
20: replace the probability distribution in $S_t$ with $H_t^+$ to obtain $\widehat{S}_t$,
    return $\widehat{S}_t$
```

To demonstrate how Algorithm 2 works, we show a simple example in Fig. 11. Similar to the example in Fig. 4, the figure shows the snapshot and their corresponding prior and posterior hypotheses. Besides, there is an extra column to show the latest most similar snapshot (LMSS) of each snapshot. The example includes six snapshots with very dynamic trip constellations. The snapshots include all five relations we outlined in Section 6.2. For example, S_2 and S_1 have disjoint relation, S_3 and S_2 have intersected relation, S_4 and S_2 have subset relation, S_5 and S_3 have superset relation, and S_6 and S_1 match exactly. The prior hypotheses H^- at each time period demonstrate how prior probabilities are assigned according to Algorithm 2. Notice that the calculation of posterior hypotheses H^+ is the same in both Algorithms 1 and 2.

7. Evaluation of heuristic algorithm

Comparing to Algorithm 1, the heuristic algorithm involves more variables that are of interest in the evaluation, such as trip constellations and the dynamics of the constellations. Due to the page limit, we cannot evaluate all the variables and their combinations. Because our focus is on the feasibility of the heuristic algorithm, we choose the most important aspects related to the



(a) S_j and S_i are disjoint

(b) S_j and S_i intersect

(c) S_i is a subset of S_j

(d) S_i is a superset of S_j

Fig. 10. Various relations of S_j and S_i.

feasibility and use simulations to evaluate them. In the following, we will evaluate the heuristic algorithm with respect to the constellation dynamics, the probability of the "real" trip, and clusters of re-appearing trips, respectively.

7.1. Evaluation with respect to constellation dynamics

Snapshots with dynamic trip constellations model the scenario in which an adversary is able to "correctly" link an individual to a specific trip. However, due to uncertainties, the real trip is mixed with a set of false trips in each of the snapshots, such that from the adversary's perspective, the correct information is submerged and concealed by incorrect information. To make things worse, in each snapshot, the real trip is presented with a different set of false trips that form a different trip constellation. The consequence is a sequence of snapshots with dynamic trip constellations.

The heuristic algorithm is developed to cope with constellation dynamics. Hence, we expect that Algorithm 2 can propagate beliefs under dynamic trip constellations. Furthermore, we are also interested in the performance of the algorithm under different degrees of constellation dynamics. Following the same approach in Section 6.1, we express the degree of constellation dynamics between two snapshots by their hamming distance. The bigger the hamming distance, the more dynamic are the trip constellations along the timeline.

In order to simulate such scenario, we generate a dataset of 60 snapshots with 100 trips each. We specify the first trip T_1 as the "real" trip. If all trips are assumed to be equally probable, they will have a probability of 0.01 each. However, we assume that the real trip will have a slightly higher than the average probability if it really occurs. Therefore, we assign 0.011 to T_1, which is 10% higher than the average probability. Another reason for the 10% higher is that it yields a good result in the previous simulation of Algorithm 1 (cf. Fig. 9). Other trips (i.e., $\{T_2, T_3, \dots, T_{100}\}$) are given random probabilities from the uniform distribution.

The next step is to find a way to distribute the 100 trips, so we can have a sequence of snapshots with dynamic trip constellations. One possibility is to distribute the trips randomly. However, in this case, it is difficult to have a clear picture of the relation between the trip dynamics to the results from the heuristic algorithm. Therefore, we control the degree of constellation dynamics so we can evaluate the heuristic algorithm in a controlled manner. We achieve this by shifting all trips after T_1 to the right, each time a new snapshot is generated. For example, if we want the 2nd snapshot to have 10% constellation dynamics to the 1st snapshot with trips $\{T_1, T_2, \dots, T_{100}\}$, we shift the trip block of $\{T_2, T_3, \dots, T_{100}\}$ of the 2nd snapshot 10 trips to the right, so the trip index becomes $\{T_1, T_{12}, T_{13}, \dots, T_{110}\}$. Consequently, 10% of the trips in the 2nd snapshot (i.e., $\{T_{101}, T_{102}, \dots, T_{110}\}$) are different from the 1st snapshot. The idea is illustrated in Fig. 12. For simplicity, we show an example of only 6 snapshots with 10 trips each. In the figure, a black square indicates an existing trip. All snapshots in the figure have a 10% constellation dynamics to the one before, i.e., each latter snapshot has one trip different from the former snapshot. In other words, each two neighboring snapshots have $10 * 90\% = 9$ trips in common.

We construct snapshots with different constellation dynamics and observe the change of beliefs on the real trip T_1. The goal is to evaluate the performance of the heuristic algorithm under various constellation dynamics. Fig. 13 shows some of the selected results with constellation dynamics of 1%, 10%, and 50%. The results are averaged over 100 simulation runs for each value of constellation dynamics to account for the variations in the random dataset. For 1% constellation dynamics, Algorithm 2 has a similar good result as Algorithm 1 (cf. Fig. 9). This means that the heuristic algorithm is able to propagate beliefs among snapshots with dynamic trip constellations, resulting in an increase in the belief on the real trip. Notice that because the hamming distances are fixed between any two consecutive snapshots, the heuristic algorithm will always find the directly precedent snapshot as the most similar one and use H^+ from that one as the basis for the construction of H^-. Therefore, the hypotheses are continuously updated and the two algorithms yield similar results. However, the beliefs on T_1 go down when the constellation dynamics increase. This matches our intuition that if there are more dynamics in the trip constellations (i.e., a real trip is associated with a different set of false trips

t	S_t(Evidence)					LMSS		B (Belief)				
	T_1	T_2	T_3	T_4	T_5			T_1	T_2	T_3	T_4	T_5
t=1	0.4	0.6				1	H^-	0.5	0.5			
							H^+	0.4	0.6			
t=2			0.3	0.5	0.2	1	H^-			0.33	0.33	0.33
							H^+			0.3	0.5	0.2
t=3	0.1		0.3	0.6		2	H^-	0.2		0.3	0.5	
							H^+	0.049		0.22	0.73	
t=4				0.6	0.4	2	H^-				0.71	0.29
							H^+				0.79	0.21
t=5	0.1		0.3	0.4	0.2	3	H^-	0.04		0.16	0.55	0.25
							H^+	0.01		0.15	0.68	0.16
t=6	0.7	0.3				1	H^-	0.4	0.6			
							H^+	0.61	0.39			

Fig. 11. Example of Algorithm 2.

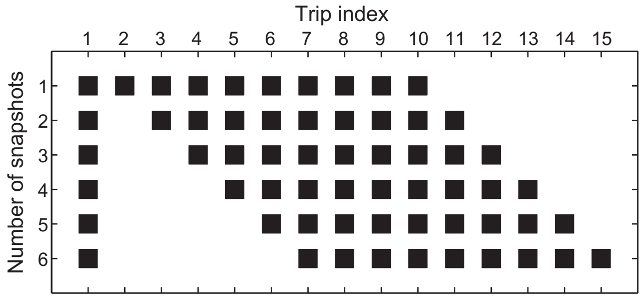


Fig. 12. Snapshots with 10% constellation dynamics.

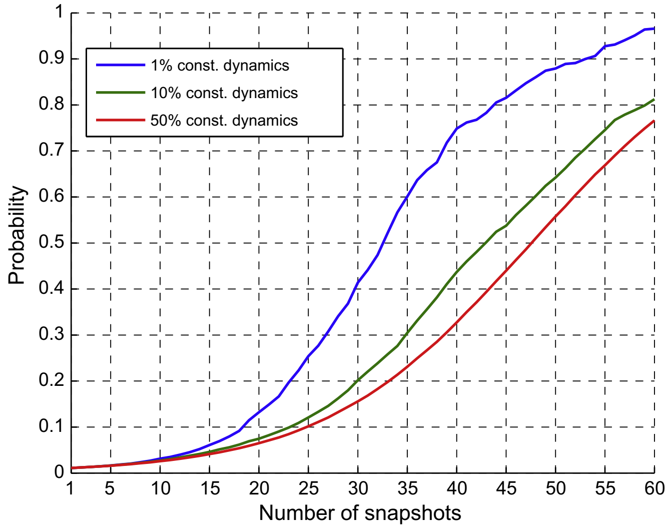


Fig. 13. Changes of beliefs on T_1 with different degree of constellation dynamics.

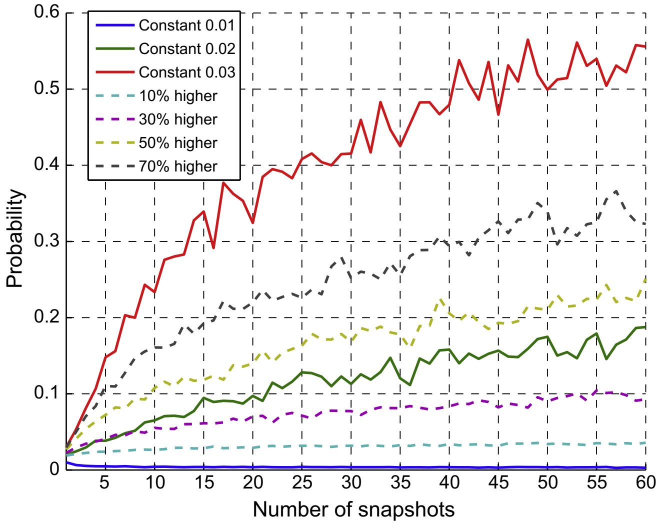


Fig. 14. Changes of beliefs with different p-values.

at each snapshot), there are more uncertainties and thus less possibilities to detect a really happened trip.

7.2. Evaluation with respect to p-value

The p-value (cf. Fig. 9) is the probability assigned to the real trip in each of the snapshots in the dataset. By specifying different probabilities of the real trip, we model an adversary's ability to link an individual to his or her vehicle movements.

In Section 7.1, we have shown that the heuristic algorithm performs well with a 10% higher than the average p-value under fixed constellation dynamics. However, we can imagine that a fixed constellation dynamics will be rare in most realistic scenarios. Therefore, we use snapshots with totally random trip constellations to evaluate Algorithm 2 with respect to different p-values. Total randomness also means that the constellation dynamics is at its maximum.

For the dataset, we randomly generate trips in the range from 2 to 100 for each of the snapshot. Hence each snapshot has a random number of trips and the trip indices are random as well. Same as before, we specify T_1 as the real trip so it appears in all snapshots. Furthermore, we assign T_1 with the p-value and probabilities from the uniform distribution to the rest of the trips. The rest of the trips are then normalized to $(1 - p_1)$. We choose two kinds of p-values: absolute values and variable values. Since now each snapshot contains a various amount of trips, the absolute p-value is a constant probability throughout all snapshots, and the variable p-value is the average probability at each snapshot (i.e., $p_1 = \frac{1}{|S_t|}$ at time t) multiplied by a scaling factor. The p-values are: 0.01, 0.02, 0.03 for the absolute and 10%, 30%, 50%, and 70% higher than the average for the variable. For each of the p-values, we run the simulation 100 times and take the averages of the beliefs on T_1 from the belief table B. The results are shown in Fig. 14.

By observing the simulation results, we made several interesting observations. First, the curves with high p-values have ripples in the short-term and exhibit an upward trend in the long-term. The ripples are due to the fluctuations in the hypotheses because the heuristic algorithm searches for the most similar snapshot in the past. For example, for the 10th snapshot, the algorithm might find that the 2nd snapshot has the most similar trip constellation, and construct H^-_{10} based on H^+_2. As a result, the updated beliefs on T_1 between the 3rd and 9th snapshots are not involved in the construction of H^-_{10}. However, in the long-term, the heuristic algorithm is able to benefit from the accumulated information. Thus the long-term beliefs on T_1 increase.

Second, if the probability of the real trip is below a certain threshold, the heuristic algorithm is unable to detect the trip. This is demonstrated by the curves representing p-values of absolute 0.01 and variable 10% higher in the figure. Notice that in previous evaluations, 10% higher p-value gives a very quick rise to the beliefs on T_1. The reason for the slow rise here is that the hypothesis of T_1 in the previous settings is *continuously* updated, while in our current setting, due to the same reason that causes the ripples, the hypothesis of T_1 is updated based on the posterior hypothesis from a randomly found snapshot with most similar trip constellation. However, looking closely, we can see that the curve of 10% higher actually increases. A measurement on the 10% higher curve confirms that there is an 88% increase at the 60th snapshot comparing to the value at the 1st snapshot.

Third, the relation of low p-values to low beliefs corresponds to our intuition that if an adversary fails to capture correct information on a real trip and give it an “outstanding treatment” in the probability assignment, the trip will be concealed among others and *no* adversaries can derive any useful information from that. In this sense, our findings here provide two interesting privacy thresholds for the design of privacy-protection mechanisms. If each time an individual has a trip and the trip can be mistaken by an adversary with no more than 99 other trips, a privacy-protection mechanism should be able to conceal the real trip among the others, in which the probability of the real trip is no more than 0.01 or no higher than 10% of the average of the trips at the same time.

7.3. Evaluation with respect to cluster of re-appearing trips

The evaluation in Section 7.2 specifies T_1 as the real trip through out all the snapshots. All other trips are generated randomly and hence might not appear in every snapshot. Thus a question arises as whether the high occurrence of T_1 biases the heuristic algorithm?

To answer this question, we use clusters of re-appearing trips to evaluate the fairness of the heuristic algorithm. Specifically, when generating the dataset, instead of placing only T_1 in each of the snapshots, we specify a set of trips to appear in all snapshots as well. Thus T_1 and other trips in the set form a trip cluster among other randomly generated trips in each of the snapshots. Conse-

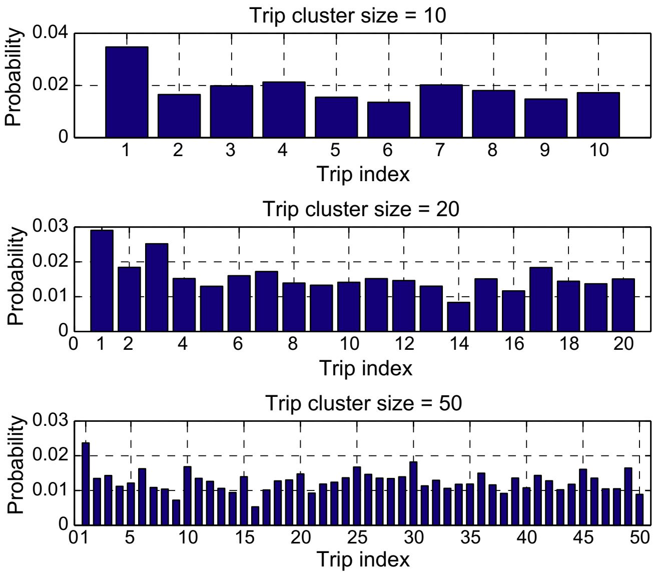


Fig. 15. Beliefs on trip clusters at 60th snapshot.

quently, the hypotheses of all trips in the cluster will be updated by the heuristic algorithm at the same time. Then we can check whether the hypothesis of T_1 is treated equally as the others in the cluster.

For simulations, we generate 60 snapshots with maximum 100 trips each. We specify three cluster sizes, i.e., 10 trips from T_1 to T_{10}, 20 trips from T_1 to T_{20}, and 50 trips from T_1 to T_{50}. Each snapshot includes a trip cluster together with other randomly generated trips. We assign a 10% higher than the average probability to T_1. The rest of the trips are assigned probabilities from the uniform distribution. For each cluster size, we run the simulation 100 times. Then we take the averaged beliefs corresponding to the trips in the cluster at the 60th snapshot from the belief table B. The results are shown in Fig. 15. As clearly demonstrated by the figure, T_1 has the highest belief at the 60th snapshot for all three cluster sizes. Thus we conclude that the occurrence of a trip does not bias the heuristic algorithm, and the algorithm performs correctly.

Based on the simulation results, we conclude that the heuristic algorithm is robust and powerful to process, propagate, and reflect the accumulated information in the location privacy metric under dynamic situations.

8. Related work

Anonymity and (un)linkability are two of the common approaches to express user privacy in communication systems. A definition on these two terms is given in [27] and unlinkability is further refined in [32].

The size of the anonymity set, e.g., *k*-anonymity [33], is a popular quantitative measurement of anonymity. Gruteser and Grunwald [13] apply *k*-anonymity to the design of a cloaking algorithm for anonymous usage of location information, and show that a quantitative privacy measurement is crucial in the development of privacy-protection mechanisms. The authors of [30,5] point out that the size of the anonymity set does not reflect different probabilities of the members in the set, and propose to use entropy of the anonymity set as the metric for anonymity for mix networks.

A main feature of location privacy is user movements, which is not explicitly captured and expressed by anonymity set. Hung et al. [19] propose geographical anonymity set, which is a set of user identifiers that forms an anonymity set geographically due to their indistinguishable movements in space and time. Beresford et al. [2] propose the concept of mix zone, an area in which users' movements cannot be observed, and to use entropy of the mix zone to quantify the information obtained by an adversary on the user movements through mix zones. Applying the same principle, the authors in [4,10] use the entropy provided by the mix zones to evaluate the level of location privacy achieved by the vehicles in vehicular ad hoc neworks (VANET). In [9], a flow-based metric is proposed to measure the effectiveness of mix zones and use it as a basis for the optimal placement of mix zones in mobile networks.

Tracking, which learns a vehicle's movement by linking a series of messages from that vehicle, is another common approach to measure location privacy. Gruteser and Hoh [14] propose to use tracking algorithms to characterize the level of location privacy. Sampigethaya et al. [28] use maximum tracking time to evaluate the location privacy of vehicles in VANET. Hoh et al. [17] use the mean time to confusion to measure the privacy level of vehicles sending GPS traces in a traffic monitoring system. Fischer et al. [8] propose to measure unlinkability of sender-message relations based on the outer and inner structures of the set partitions of the observed messages. Most approaches to location privacy focus on location information. In [23], we propose that metrics for location privacy in vehicular communication systems should take both individuals and their vehicle trips into consideration.

The impact of accumulated information on location privacy has not been explicitly addressed in most of these approaches so far. Mostly, it is assumed that an adversary's knowledge on a system already reflects its long-term observations at the time of attack. For example, in most of the mix zone approaches, an adversary is assumed to have the statistical data on the user mobility in the mix zone. Empirical studies such as [21] use two weeks of recorded pseudonymous location tracks to infer home addresses and identities of the drivers with partial successes. Outside the communication domain, the authors of [35] find out that snapshot-based, time-invariant approaches cannot cope with the emergence of time series data mining, and propose to add the time dimension to the current research on privacy-preserving data mining.

9. Conclusion and outlook

In this paper, we present our approaches and algorithms in the form of a trip-based location privacy metric for measuring long-term location privacy of the users of vehicular communication systems. To precisely reflect underlying privacy values in vehicular communication systems, we take accumulated information into consideration. We develop approaches and algorithms to model, process, propagate, and reflect the impact of accumulated information in privacy measurements. Moreover, in this paper we present two algorithms to apply the Bayesian method to process and propagate accumulated information among multiple snapshots along the timeline. Specifically, the first algorithm propagates information among snapshots with exactly matching trip constellations. The second algorithm is a heuristic extension to the first one, which is robust to function on snapshots with highly dynamic trip constellations. We also design methods to evaluate the feasibility and correctness of the approaches and algorithms by various case studies and extensive simulations. We show in this paper that accumulated information can have significant impacts on the level of location privacy. The results and findings in this paper provides some valuable insights into location privacy, which contribute to the development of future-proof, privacy-preserving vehicular communication systems.

An important aspect of our work is to demonstrate the practicability and feasibility of using the trip-based location privacy metric to measure privacy values and improve privacy-enhanced designs in vehicular communication systems. For this reason, we have gathered a collection of datasets of realistic vehicle trips from various sources. In our future work, based on the realistic datasets, we will further evaluate our approaches and algorithms by more use-case-based scenarios. The evaluation will also include existing privacy-protection mechanisms proposed to vehicular communication systems. The current metric only measures privacy of individual users. The possible interrelations among individuals and their impacts on the level of location privacy will be investigated to determine location privacy in a global view. The metric is extensible, which means we can take other identified attacks on location privacy into respect in our metric whenever necessary.

Acknowledgment

This work was partially supported by the European Commission through the PRECIOSA project (IST-224201).

References

[1] Moshe E. Ben-akiva, John L. Bowman, Activity based travel demand model systems, in: Equilibrium and Advanced Transportation Modeling, Kluwer Academic Publishers, Dordrecht, 1998, pp. 27–46.

[2] Alastair R. Beresford, Frank Stajano, Location privacy in pervasive computing, IEEE Pervasive Computing 2 (1) (2003) 46–55.

[3] Andrew J. Blumberg, Peter Eckersley, On locational privacy, and how to avoid losing it forever. Technical report, Electronic Frontier Foundation, August 2009.

[4] Levente Buttyan, Tamas Holczer, Istvan Vajda, On the effectiveness of changing pseudonyms to provide location privacy in VANETs. in: ESAS 2007, July 2007.

[5] C. Diaz, S. Seys, J. Claessens, B. Preneel, Towards measuring anonymity, in: Workshop on Privacy Enhancing Technologies, 2002.

[6] Florian Dötzer, Privacy issues in vehicular ad hoc networks, in: Workshop on Privacy Enhancing Technologies, 2005.

[7] Elmar Schoch, Frank Kargl, Tim Leinmüller, Stefan Schlott, Panagiotis Papadimitratos, Impact of pseudonym changes on geographic routing in vanets, in: ESAS, November 2006.

[8] Lars Fischer, Stefan Katzenbeisser, Claudia Eckert, Measuring unlinkability revisited, in: WPES'08: Proceedings of the Seventh ACM workshop on Privacy in the electronic society, Alexandria, Virginia, October 27 2008.

[9] Julian Freudiger, Reza Shokri, Jean-Pierre Hubaux, On the optimal placement of mix zones, in: PET, 2009.

[10] Julien Freudiger, Maxim Raya, Mark Felegyhazi, Panos Papadimitratos, Jean-Pierre Hubaux, Mix-zones for location privacy in vehicular networks, in: WiN-ITS, 2007.

[11] Matthias Gerlach, Felix Güttler. Privacy in VANETs using changing pseudonyms – ideal and real, in: VTC2007-Spring, 2007, pp. 2521–2525.

[12] Philippe Golle, Kurt Partridge, On the anonymity of home/work location pairs, Pervasive, vol. 5538, Springer, Berlin, 2009, pp. 390–397.

[13] Marco Gruteser, Dirk Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking, in: MobiSys'03: Proceedings of the First International Conference on Mobile Systems, Applications and Services, ACM Press, New York, NY, 2003, pp. 31–42.

[14] Marco Gruteser, Baik Hoh, On the anonymity of periodic location samples, in: Security in Pervasive Computing 2005, Boppard, Germany, vol. 3450, 2005, pp. 179–192.

[15] R.W. Hamming, Error detecting and error correcting codes, Bell System Technical Journal 29 (2) (1950) 147–160.

[16] Baik Hoh, Marco Gruteser, Hui Xiong, Ansaf Alrabady, Enhancing security and privacy in traffic-monitoring systems, IEEE Pervasive Computing 5 (4) (2006) 38–46.

[17] Baik Hoh, Marco Gruteser, Hui Xiong, Ansaf Alrabady, Preserving privacy in GPS traces via density-aware path cloaking, in: ACM Conference on Computer and Communications Security (CCS), 2007.

[18] Pat S. Hu, Timothy R. Reuscher, Summary of travel trends, 2001 national household travel survey. US Department of Transportation, Federal Highway Administration, December 2004.

[19] Leping Huang, Hiroshi Yamane, Kanta Matsuura, Kaoru Sezaki, Towards modeling wireless location privacy, in: Privacy Enhancing Technologies, 2005, pp. 59–77.

[20] Jean-Pierre Hubaux, Srdjan Čapkun, Jun Luo, The security and privacy of smart vehicles, IEEE Security and Privacy 4 (2004) 49–55.

[21] John Krumm, Inference attacks on location tracks, in: Fifth International Conference on Pervasive Computing, Toronto, Canada, May 2007, pp. 127–143.

[22] Zhendong Ma, Frank Kargl, Michael Weber, Pseudonym-on-demand: a new pseudonym refill strategy for vehicular communications, in: WiVeC 2008, Calgary, Canada, September 2008.

[23] Zhendong Ma, Frank Kargl, Michael Weber, A location privacy metric for v2x communication systems, in: IEEE Sarnoff Symposium, Princeton, NJ, USA, March 2009.

[24] Zhendong Ma, Frank Kargl, Michael Weber, Measuring location privacy in V2X communication systems with accumulated information, in: IEEE MASS'09, Macau, China, October 2009.

[25] P. Papadimitratos, L. Buttyan, T. Holczer, E. Schoch, J. Freudiger, M. Raya, Z. Ma, F. Kargl, A. Kung, J.-P. Hubaux, Secure vehicular communications: Design and architecture, IEEE Communications Magazine 46 (11) (2008) 100–109.

[26] P. Papadimitratos, A. Kung, Jean-Pierre Hubaux, F. Kargl, Privacy and identity management for vehicular communication systems: A position paper, in: Workshop on Standards for Privacy in User-Centric Identity Management, Zurich, Switzerland, July 2006.

[27] Andreas Pfitzmann, Marit Hansen, Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management – a consolidated proposal for terminology. Technical report, TU Dresden, v0.31, February 2008.

[28] Krishna Sampigethaya, Mingyan Li, Leping Huang, Radha Poovendran, Amoeba: robust location privacy scheme for vanet, IEEE Journal on Selected Areas in Communications 25 (8) (2007) 1569–1589.

[29] Florian Schaub, Zhendong Ma, Frank Kargl, Privacy requirements in vehicular communication systems, in: IEEE International Conference on Privacy, Security, Risk, and Trust (PASSAT 2009), Symposium on Secure Computing (SecureCom09), Vancouver, Canada, August 2009.

[30] A. Serjantov, G. Danezis, Towards an information theoretic metric for anonymity, in: Workshop on Privacy Enhancing Technologies, 2002.

[31] C.E. Shannon, A mathematical theory of communication, Bell System Technical Journal 27 (1948) 379–423. 623–656.

[32] Sandra Steinbrecher, Stefan Köpsell, Modelling unlinkability, in: Workshop on Privacy Enhancing Technologies, 2003, pp. 32–47.

[33] L. Sweeney, *k*-Anonymity: a model for protecting privacy, International Journal on Uncertainty, Fuzziness and Knowledge-based Systems 10 (5) (2002) 557–570.

[34] Björn Wiedersheim, Frank Kargl, Zhendong Ma, Panagiotis Papadimitratos, Privacy in inter-vehicular networks: why simple pseudonym change is not enough, in: Proceedings of the Seventh International Conference on Wireless On-demand Network Systems and Services (WONS 2010), February 2010.

[35] Ye Zhu, Yongjian Fu, Huirong Fu, On privacy in time series data mining, in: PAKDD, 2008, pp. 479–493.