Camouflaging Mobility for Itinerary Privacy in Mobile Ad-hoc Networks

Lei Tang, Xiaoyan Hong, Susan Vrbsky
Department of Computer Science, University of Alabama, Tuscaloosa, AL 35487
{*ltang,hxy,vrbsky*}*@cs.ua.edu*

Abstract

The privacy of wireless communications is becoming an important issue due to the open nature of wireless medium. Many research work have been proposed to address the anonymity of communicating parties, the location privacy of the message source and destination, and the privacy of the network routing paths. However, with the advent of new radio identification and localization techniques, more advanced privacy attacks are possible. We describe a new privacy attack in which the adversary tries to infer the itineraries of the nodes in the network. To protect itinerary privacy, we design an algorithm, called the Δ*-mobility camouflaging algorithm, which can be applied upon any mobility model by changing the original motion segments into* Δ*-shaped camouflaging paths. Our analysis results show that* Δ*-mobility camouflaging algorithm is cost-effective, which in most cases decreases more than 80% itinerary exposure probability at a cost of less than* 3% *extra travel distance. Through comparing DSR routing performance under different mobility models and their counterparts with* Δ*-motion, we find* Δ*-mobility camouflaging algorithm does not degrade network layer routing performance in terms of message delivery ratio, delivery latency and routing overhead.*

1 Introduction

In a mobile ad-hoc network(MANET), to send a message to another node, the message is forwarded hop-by-hop through wireless links from the message source to the destination. Due to the open nature of the wireless medium, adversaries in the network are able to eavesdrop wireless communications to obtain the information interested. For example, in the non-anonymous routing schemes such as AODV [14] and DSR [8], the IP addresses of the message source and destination are explicitly contained in the message. Hence, for a message detected, the adversary is able to determine the identities of the message source and destination. With the knowledge of message source/destination, the adversary may launch the attacks targeted to the specific nodes. Possible attacks include discarding message, altering message content and flooding the message [21].

Therefore, privacy issues are becoming increasingly important for MANET wireless communications. Many privacy preserving schemes ([10, 18, 2, 13, 9, 1, 16]) have been proposed to address *correspondent privacy*, *route privacy* and *location privacy*. The objective of *correspondent privacy* is to prevent adversaries from discovering who are the message source and destination (e.g [9]) whereas the objective of *route privacy* is to prevent adversaries from tracing the network routes of the messages (e.g [10, 18]). And the objective of *location privacy* may include preventing the adversary from determining the location of message source and destination or preventing the adversaries from tracing nodes in the network. Two approaches are widely used in these work. One approach uses cryptography to generate pseudonyms to hide the real identities of correspondents or the identities of the nodes on the routes (e.g [10, 18, 2, 13]). One issue of this approach is that decrypting and encrypting cryptographical pseudonyms may cause large computational overhead [13]. The other approach is to mix the real correspondents among a set of the nodes to make it difficult for the adversary to pinpoint a specific node (e.g [9, 1, 16]).

With the advent of new wireless localization techniques (e.g [19]) and radio identification techniques (e.g [4]), adversaries are able to launch more advanced privacy attacks. The localization technique in [19] is able to localize indoor radio transmitters with 2-meter accuracy based on the radio signals received. And the radio identification technique proposed in [4] by D.B.Faria *et al.* can robustly identify a radio transmitter by its *signalprint* (a set of signal strength values collected).

These localization techniques and radio identifica-

tion techniques enable new privacy attacks. In this article, we describe a new privacy attack, which aims at determining the mobile hosts' itineraries based on the above localization technique and radio identification technique. Nodes in the wireless network are unlikely to move total randomly but follow a certain schedule [12]. We define *itinerary privacy* as a property that it is difficult for attackers to determine the itinerary of a node (i.e. when and where it will appear). The itinerary privacy differs from location privacy in that its emphasis is on discovering nodes' repeatedly-occurring mobility patterns, which most likely reveal a node's routine activities and the paths taken by the node when conducting the activities. We call the path from one activity to another activity as an *itinerary segment*. We believe itinerary privacy is important because we want to prevent unauthorized tracking of wireless-communicating mobile hosts (e.g. patrol cars, cash-in-transit vehicles, etc). Once the itinerary information of a mobile host is mastered by the adversaries, adversaries may launch many attacks more intelligently and precisely with greater damages.

In our adversarial model, when a radio transmission is detected by the adversary, the adversary stores the *transmission print* of the radio transmission in its database, which includes the signalprint, location, and the time of the radio transmission. The adversary infers a node's itinerary by associating the *transmission prints* with the node.

Itinerary privacy is important for protecting correspondent privacy and route privacy. This is because after the itineraries of the nodes are exposed, the adversary can correlate a message detected with the message sender. For example, if the adversary detects a message M is transmitted at location p at time t and the adversary knows that a node x was at p at t based on its knowledge of x's itinerary, then the adversary can infer that x is the sender of M even if M does not expose any identity information of the sender.

Therefore, the motivation for this paper is to design a scheme for protecting itinerary privacy. We design an algorithm, called the Δ-*mobility camouflaging algorithm*, that protects itinerary privacy by camouflaging the nodes' mobility. Δ-mobility camouflaging algorithm can be applied upon any mobility model by changing the original motion segments into Δ-shaped camouflaging paths. Δ-mobility camouflaging algorithm is effective because it significantly increases the number of possible motion traces. Also the motion traces of the nodes are "mixed" and become less distinctive. Furthermore, it reduces the probability of generating matchable transmission prints since nodes are unlikely to move in the same path, thereby making it difficult for the adversary to confirm or eliminate a hypothetical motion trace.

Our mathematical analysis shows that Δ-mobility camouflaging algorithm is cost-effective, which decreases more than 80% itinerary exposure probability at a cost of less than 3% extra travel distance in the cases we studied. We also conduct simulations under Qualnet Network Simulator [17] to compare DSR routing performance under mobility models with/without Δ-mobility camouflaging. Our simulation results show that Δ-mobility camouflaging algorithm does not degrade routing performance in terms of message delivery ratio, delivery latency and routing overhead.

This rest of this article is structured as follows. In section 2, we introduce prior work on MANET privacy issues and classify them according to their approaches and objectives. Section 3 introduces itinerary privacy attack and adversarial model. Section 4 presents the Δ-mobility camouflaging algorithm and analyzes its effectiveness on reducing itinerary exposure probability and its overhead. In section 5, we compare DSR routing performance under mobility models using and not using the Δ-mobility camouflaging algorithm. We summarize our works and outline future plan in section 6.

2 Related Work

Many works on MANET privacy are to protect the following three types of privacy: *correspondent privacy*, *route privacy* and *location privacy*. To realize these anonymity goals, two approaches are widely used. One approach uses cryptographic pseudonyms to identify routes and hide real identities of the nodes. The other approach is to "mix" the nodes among other nodes that conduct radio transmissions. Our scheme can be categorized into the type using MIX methodology, which "mixes" the motion traces of the nodes and makes radio transmissions less identifiable so that it is difficult for the adversary to identify a node's itinerary.

The idea of MIX is first presented in [3], in which D. Chaum presented a technique to hide the correspondences between input emails and output emails by encrypting correspondent information and sending emails randomly. MIX technique can be used in MANET. For example, we can "mix" the real correspondents among a set of nodes so that it is difficult for the adversaries to pinpoint the real correspondents.

J. Kong *et al.* proposed a technique called *Motion-MIX* to hide the motion pattern of the nodes through adding decoy messages [11]. A. Beresford *et al.* designed a method, called the *mix zone*, to enhance user location privacy when using location-aware services [1]. M. Gruteser *et al.* proposed protecting path pri-

vacy by truncating longer paths into multiple segments and switching pseudonyms during traveling the path [6]. AO2P [20] is a position-based anonymous routing protocol which uses destination's present position as the destination identifier to protect correspondent anonymity. In *phantom* routing scheme [9], messages are first sent to a fake source and then are flooded to the destination to protect location privacy of source. B. Hoh *et al.* present an algorithm [7] that protects location privacy by adding acceptable perturbations to the original location data. In [16], K. Sampigethaya *et al.* introduce random silent periods in vehicle broadcast communications to mitigate unauthorized tracking of vehicles.

Different from [16], our mobility algorithm does not affect the way nodes conducting wireless communications. Our Δ-mobility model is also different from [20, 1] since we do not assume the existences of *mix zones* in which the nodes' movements are anonymous nor do we assume that the locations of message destinations can be obtained from trustable location server to be used as destination identifiers. Instead, in our system, nodes proactively change their mobility to hide their itineraries.

3 Itinerary Privacy Threat

Since nodes in the network are unlikely to move total randomly but follow a certain schedule [12], we design a mobility model, called *rendezvous visiting* or RV in short to model the mobility scenario when the nodes move according to their itineraries. RV mobility model can be viewed as a simplified *agenda mobility model* in [22]. In RV model, a node visits the rendezvous in the network based on its itinerary and the path from one rendezvous to another rendezvous is defined as an *itinerary segment*. The itinerary of a node consists of itinerary segments. An itinerary segment comprises a starting and an ending rendezvous. For instance, an itinerary segment of a patrol car starts from location A to location B, which should be prevented from unauthorized tracking.

In this section, we describe a new privacy attack that discovers the itineraries of the mobile hosts in the network by using two important techniques, i.e. *signalprint* and *multiple target tracking (MTT)* [15]. The network scenario considered in the paper is a wireless mobile ad-hoc network and we assume wireless communications are symmetric (i.e. if node A can hear node B, then B can hear A).

The *signalprint* of a wireless transmission is a vector of signal strength measurements [4]. Signalprint has the following properties [4]. First, it is hard to spoof because radio transmitters have no control over signal attenuations within the network [4]. Second, signalprints are strongly correlated with the location of the radio transmitter and a stationary transmitter generates similar signalprints with high probability [4]. In [4], D.B.Faria *et al.* propose a radio identification technique that robustly identifies a radio-transmitting device by its *signalprints*.

Multi Target Tracking (MTT) algorithm is a well-studied technique to link location samples of the nodes to individual nodes based on the temporal and spatial correlation between successive location samples [7]. Multiple target tracking algorithm proposed in [15] generates a set of hypotheses about the possible motion traces of the nodes. The hypothetical motion traces are confirmed or eliminated when more location samples are processed. The algorithm stops when all location samples have been processed. For the details of the multiple target tracking algorithm, interested readers are referred to [15].

In our system, we assume that the adversary deploys a sufficient number of *snoopers* in the network to cover the entire network, which passively eavesdrop on the radio transmissions in the network and send the collected signalprints to the adversary. Meanwhile, the adversary uses a localization technique similar to the one proposed by Tao *et al.* in [19] to determine the location of the radio transmitter with a precision of 2 meters. The localization system in [19] uses the Markov localization algorithm [5] to determine the location of the radio transmitter based on the signal strength information.

We assume that the adversary divides the network into a grid of equal-sized cells. From the experiment results of [4], a node generates similar signalprints with high probability at locations less than 5 meters apart [4]. So we select 5×5 meters as the area of a grid cell such that a node will generate the same signalprint if it conducts multiple radio transmissions at the same cell. Once a radio transmission occurs in the network, the adversary locates the radio transmitter to a grid cell. Fig. 1 illustrates our adversarial model.

We define some of the terms used in the paper as follows.

- ***itinerary segment:*** The itinerary of a node comprises a set of *itinerary segments*. When a node conducts an itinerary segment, it moves from one rendezvous to another rendezvous. An itinerary segment is represented as {*start*, *end*, *time*, *speed*}, in which *start*, *end* is the starting rendezvous and ending rendezvous, respectively. *time* specifies when the node starts the itinerary segment and *speed* specifies the moving speed of

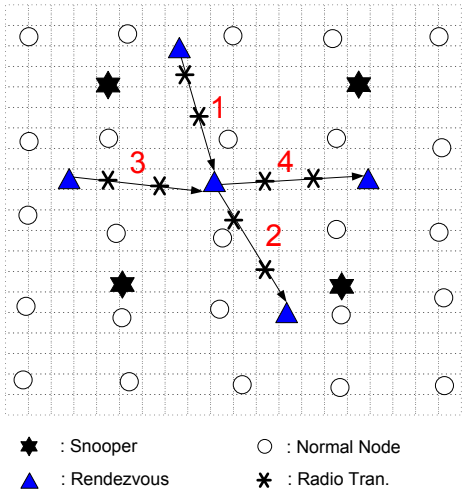

Figure 1. Adversarial Model

the node on the itinerary segment. Segments 1, 2, 3, and 4 in Fig. 1 are some exemplary itinerary segments.

- ***transmission print:*** The adversaries store the information of each radio transmission as a *transmission print*, which includes the following information $\{signalprint, location, time\}$. *location* and *time* record when and where the radio transmission occurs, respectively.

- ***matchable transmission print:*** We call the transmissions prints generated by the same node at the same cell as *matchable transmission prints*.

We now describe how the adversary launches the itinerary privacy attack to discover the itineraries of the nodes in the network.

After a radio transmission is detected, the snoopers send the signal strength measurements of the radio transmission to the adversary. These signal strength measurements are used by the adversary to calculate the location and the signalprint of the radio transmission. Finally, the adversary stores the transmission print of the detected radio transmission in database. For some non-anonymous routing schemes (e.g. DSR), the identity of the radio transmitter can be obtained from the message transmitted. Therefore, the adversary can easily draw the motion traces of the nodes based on the locations of the radio transmissions and the identities of the radio transmitters. So here we assume that the nodes can use existing anonymous routing technique such as ANODR [10] to prevent the adversary from obtaining the identity of the radio transmitter from the content of the radio messages.

Since the adversary can not obtain the identity information directly from the messages detected, it uses the MTT technique to associate the transmission prints with individual radio transmitters. First the adversary uses the MTT algorithm to construct all the possible motion traces by exploiting the temporal and spatial correlations between subsequent transmission prints. Then the adversary uses signalprint information in the transmission prints to confirm or eliminate the hypothetical traces by finding the matchable transmission prints.

Signalprint technique makes it much easier for the adversary to confirm or eliminate the hypothetical motion traces during MTT computation process. If without matchable transmission prints, MTT algorithm can only exploit the temporal and spatial correlations between radio transmissions to determine nodes' possible traces. For example, in Fig. 2, without using signalprints, ten radio transmissions can be from ten different transmitters. Since a node will generate the same signalprints at the same cell, with signalprints the adversary can figure out which radio transmissions are from the same node to reduce the number of possible transmitters and hypothetical motion traces.

The transmission prints at a cell can be categorized into two types: matchable/non-matchable transmission prints. From the matchable transmission prints, the adversary learns when the node travels the cell and the interval between its tours. On the other hand, the number of non-matchable transmission prints reveals the number of transmitters touring the cell. With the above information, the adversary run MTT algorithm to eliminate or confirms the hypothetical motion traces. After the adversary is able to associate transmission prints with the motion traces, the itineraries of the nodes are exposed. For example, after a radio transmission of node x is detected by the adversary, the adversary can associate the transmission print of the radio transmission to a determined motion trace and predict the future motion of node x.

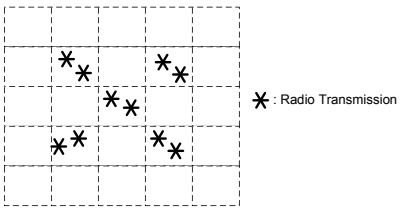

Figure 2. Signalprints

4 Δ-mobility Camouflaging algorithm

We propose Δ-mobility camouflaging algorithm to make it difficult for the adversary to determine the itineraries of the nodes. Δ-mobility camouflaging algorithm is effective because of the following reasons.

1. With Δ-mobility camouflaging algorithm, nodes take random camouflaging movements to cover their itinerary segments. Hence the number of possible motion traces is significantly increased. Also the motion traces of the nodes are "mixed" and become less distinctive.

2. The probability of generating matchable transmission prints is reduced since nodes are unlikely to move in the same path. Hence it becomes more difficult for the adversary to confirm or eliminate a hypothetical motion trace.

In the section, we first analyze the probability of exposing a node's itinerary when the node takes no camouflaging movements (i.e move straightly from one rendezvous to another). Then we propose our Δ-mobility camouflaging algorithm and analyze its improvement on reducing itinerary exposing probability.

4.1 Terminology and Notations

The terminologies and notations used in the paper are defined as follows.

- d_i: the distance from the start to the end of an itinerary segment i.
- e_i^1: the mobility displacement of an itinerary segment i when using Δ-mobility.
- ξ_i: the number of grid cells on an itinerary segment i when the node uses straight-line mobility.
- ξ_i': the number of grid cells on an itinerary segment i when the node uses Δ-mobility.
- ω_i: the average number of radio transmissions conducted by a node on an itinerary segment i.
- μ_i: the travel overhead of an itinerary segment i when using Δ-mobility.
- P_τ: the probability of exposing an itinerary segment when using straight-line mobility.
- P_τ': the probability of exposing an itinerary segment when using Δ-mobility.
- α_i: camouflaging angle of an itinerary segment i when using Δ-mobility.
- α_{max}: the maximum camouflaging angle in Δ-mobility.
- S,D: source and destination.

4.2 Analysis of Non-camouflaging Mobility

The scenario when the nodes take no camouflaging movements (i.e straight-line mobility from the start to the end of the itinerary segment) is illustrated in Fig. 3. Based on the collected transmission prints, the adversary calculates all the hypothetical motion traces of the nodes using MTT algorithm. Then it confirms/eliminates the hypothetical motion traces using the information mined from transmission prints.

In our adversarial model, we assume that the adversary is able to associate two matchable transmission prints with the transmitter based on the temporal and spatial correlation of the transmission prints. Here we give a simple example to illustrate how the adversary exploits temporal and spatial information in the matchable transmission prints. For example, the adversary collected two matchable transmission prints: $\{signalprint_1,\ location_1,\ time_1\}$ and $\{signalprint_2,\ location_2,\ time_2\}$. The adversary can calculate the traveling speed between these two matchable transmission prints as: $\frac{|location_1 - location_2|}{|time_1 - time_2|}$. If the traveling speed approximates the estimated speed of a node, then the adversary tentatively associates these two transmission prints with the node.

Since it requires only two different points on a line to determine the line, the adversary will be able to discover an itinerary segment of a node if the node leaves two matchable transmission prints on the itinerary segment. For the MTT details of constructing hypothetical motion traces, confirming and eliminating hypothetical motion traces, interested readers are referred to [15].

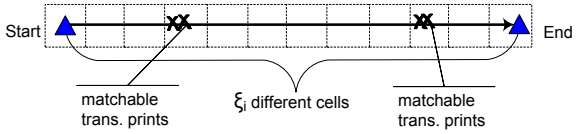

Figure 3. Straight-line Mobility

Let P_{match}^0 be the probability of generating no matchable transmission print on an itinerary segment and P_{match}^1 be the probability of generating only one matchable transmission print. We have:

$$P_\tau(\xi_i, \omega_i) = 1 - P_{match}^0(\xi_i, \omega_i) - P_{match}^1(\xi_i, \omega_i). \quad (1)$$

Based on (1), equation (2) calculates P_τ when ω_i is smaller than ξ_i, in which $\frac{\binom{\omega_i}{k} \times \xi_i \times (\xi_i - 1)^{P(\omega_i - k)}}{\xi_i^{\omega_i}}$ is the probability of k radio transmissions of a node occurring in one cell and the other $\omega_i - k$ transmissions occurring in different cells. And $\prod_{k=2}^{\omega_i} (\frac{\xi_i - (k-1)}{\xi_i})$ calculates the probability that in ω_i transmissions, no two or more transmissions occur in the same cell.

$$P_\tau(\xi_i, \omega_i) = 1 - \prod_{k=2}^{\omega_i} (\frac{\xi_i - (k-1)}{\xi_i}) - \sum_{k=2}^{\omega_i} \frac{\binom{\omega_i}{k} \times \xi_i \times (\xi_i - 1) P(\omega_i - k)}{\xi_i^{\omega_i}}, \ \xi_i \geq \omega_i \geq 2 \quad (2)$$

Fig. 4 shows that when $\xi_i = 400$, it takes only 80 radio transmission for P_τ to approach 1. And when ω_i approaches ξ_i, P_τ becomes 1. From the above results, we know that a node exposes an itinerary segment quickly when there is no camouflaging mobility. Next we will analyze the itinerary exposure probability when using our Δ-mobility camouflaging algorithm.

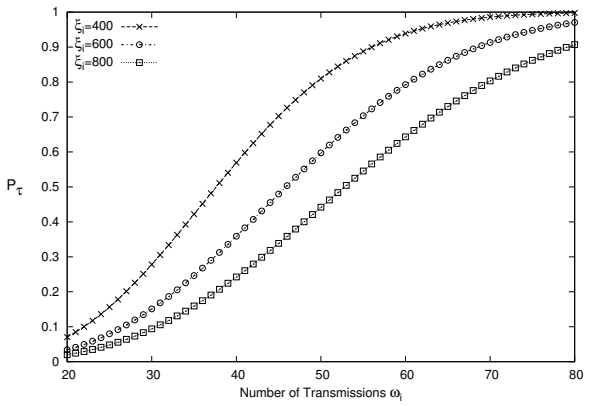

Figure 4. P_τ when taking no camouflaging movements

4.3 Analysis of Δ-mobility Camouflaging Algorithm

Δ-mobility camouflaging algorithm makes it difficult for the adversary to distinguish the motion traces of the nodes by randomly distributing the transmission prints of the nodes and by avoiding generating matchable transmission prints.

Δ-mobility camouflaging algorithm is showed in Fig. 5. Each time a node tours an itinerary segment i, the node randomly selects a camouflaging angle $\alpha \leq \alpha_{max}$ and a displacement e_i^1. The node will first walk along the camouflaging angle α for e_i^1 then head for the end of the segment. Thus when using Δ-mobility camouflaging, an itinerary segment i is represented as:$\{start_i, end_i, \alpha_i, e_i^1, time_i, speed_i\}$, in which $start_i$, end_i and $time_i$ are the starting rendezvous, ending rendezvous and starting time of the itinerary i, respectively.

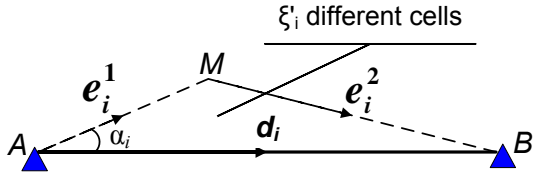

Figure 5. Δ-mobility for an itinerary segment

In Fig. 6, a node x has an itinerary segment starting from A and going to B whereas a node y has an itinerary segment from C to D. Fig. 6 shows the transmission prints left by x and y when they take random triangle-shaped pathes generated by Δ-mobility camouflaging algorithm and the straight-line paths. We can see that the transmission prints of Δ-mobility camouflaging algorithm are "mixed" together whereas the transmission prints of straight-line mobility show a distinctive pattern. Moreover, the transmission prints of Δ-mobility camouflaging algorithm are distributed over a larger area while the transmission prints of straight-line mobility are focused on a narrow straight-line area. Hence Δ-mobility camouflaging algorithm reduces the probability of generating matchable transmission prints.

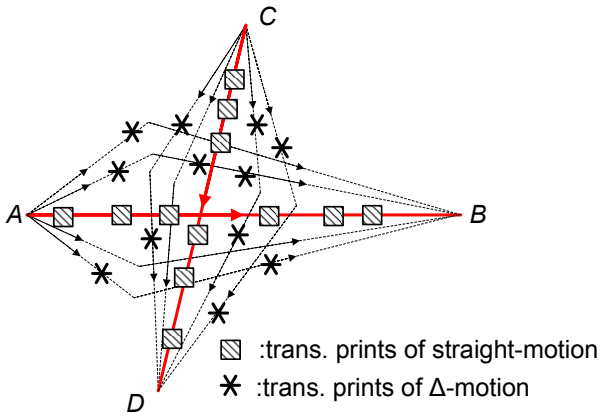

Figure 6. Δ-mobility Camouflaging

Next we analyze the overhead of Δ-mobility, the probability of exposing an itinerary segment and the probability of the nodes generating the same Δ-mobility.

4.3.1 Overhead Analysis of Δ-mobility

Compared with straight-line mobility, Δ-mobility incurs extra travel distance, which is called *travel overhead*. As illustrated in Fig.5, we can compute the travel overhead μ_i for an itinerary segment i with travel distance d_i ($d_i = |start_i - right_i|$) as follows:

$$\mu_i = \frac{e_i^1 + e_i^2}{d_i} = \frac{e_i^1 + \sqrt{(e_i^1)^2 + (d_i)^2 - 2d_i e_i^1 \cos\alpha_i}}{d_i} = \frac{e_i^1}{d_i} + \sqrt{\left(\frac{e_i^1}{d_i}\right)^2 + 1 - 2\frac{e_i^1}{d_i}\cos\alpha_i}\ . \quad (3)$$

From equation (3), we know that overhead μ_i is determined by the largest possible α_i (i.e α_{max}) and $\frac{e_i^1}{d_i}$. Fig. 7 shows that when $\alpha_i \leq 15°$ and $\frac{e_i^1}{d_i} \leq 0.4$, the travel overhead is always smaller than 0.03. Plus, it is easy to control the overhead μ_i by tuning α_{max} and $\frac{e_i^1}{d_i}$. For instance, if $\frac{e_i^1}{d_i}$ is known and we want travel overhead to be no larger than μ_i, we can set α_{max} as

$$-\arccos\left(\frac{(d_i)^2 - (\mu_i d_i)^2 + 2\mu_i d_i e_i^1}{2e_i^1 d_i}\right) \leq \alpha_{max} \leq \arccos\left(\frac{(d_i)^2 - (\mu_i d_i)^2 + 2\mu_i d_i e_i^1}{2e_i^1 d_i}\right) \quad (4)$$

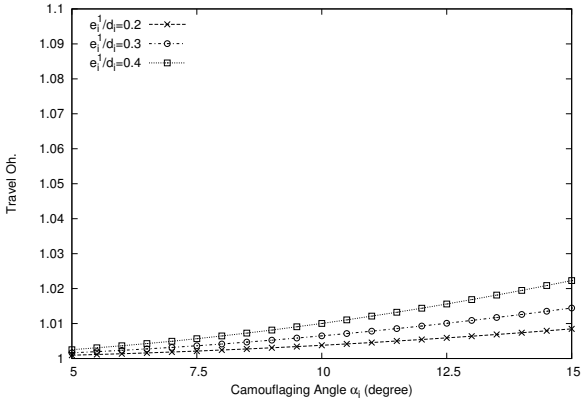

Figure 7. Travel Overhead versus α_i and $\frac{e_i^1}{d_i}$

4.3.2 Privacy Analysis of Δ-mobility

Now we calculate itinerary exposure probability. In our evaluation of itinerary privacy, we assume a sufficient number of snoopers are distributed to cover the whole network. Each grid cell is of a dimension 5×5 meters.

As illustrated in Fig. 5, for an itinerary segment i, Δ-mobility has smaller probability of generating matchable transmission prints than straight-line mobility because its transmission prints are distributed to large triangle shapes with area $e_i^1 d_i \sin\alpha_{\max}$ instead of a small straight-line shape with area $5d_i$. Hence, equation (5) calculates the expected value of $\frac{\xi_i'}{\xi_i}$, in which ξ_i' and ξ_i are the number of cells that a node possibly visits when touring itinerary segment i using Δ-mobility and straight-line mobility, respectively. Notice that when using Δ-mobility, it is easy to choose e_i^1 and α_{max} to ensure $\frac{\xi_i'}{\xi_i} > 1$.

$$\frac{\xi_i'}{\xi_i} = \frac{e_i^1 d_i \sin\alpha_{\max}}{5d_i} = 0.2e_i^1 \sin\alpha_{\max} \quad (5)$$

The calculation of P_τ' is similar to the calculation of P_τ, the result of which is showed in equation (6).

$$P_\tau'(\xi_i',\ \xi_i,\ \omega_i) = 1 - \frac{\sum_{s=0}^{\omega_i} \binom{w}{s} \xi_i P_s \times (\xi_i' - \xi_i)^{\omega_i - s}}{\xi_i'^{\omega_i}} - \frac{\sum_{s=2}^{\omega_i}\sum_{k=2}^{s} \binom{w}{s}\binom{s}{k} \xi_i \times {}_{(\xi_i-1)}P_{(s-k)} (\xi_i' - \xi_i)^{\omega_i - s}}{\xi_i'^{\omega_i}} \quad (6)$$

Combining equation (5) and (6), we have:

$$P_\tau'(\xi_i,\ e_i^1,\ \alpha_{max},\ \omega_i) = 1 - \frac{\sum_{s=0}^{\omega_i} \binom{w}{s} \xi_i P_s \times [0.2e_i^1\xi_i \sin(\alpha_{\max}) - \xi_{\mathrm{i}}]^{\omega_{\mathrm{i}} - \mathrm{s}}}{[0.2e_i^1\xi_i \sin(\alpha_{\max})]^{\omega_{\mathrm{i}}}} - \sum_{s=2}^{\omega_i}\sum_{k=2}^{s} \left[\frac{\binom{w}{s}\binom{s}{k} \xi_i \times {}_{(\xi_i-1)}P_{(s-k)} [0.2e_i^1\xi_i \sin(\alpha_{\max}) - \xi_{\mathrm{i}}]^{\omega_i - s}}{[0.2e_i^1\xi_i \sin(\alpha_{\max})]^{\omega_i}}\right] \quad (7)$$

Using equation (7) and equation (2), we compare itinerary exposure probability of straight-line mobility and Δ-mobility. We set $\frac{e_i^1}{d_i} = 0.4$ and $\xi_i = 80$. The result is showed in the Fig. 8. Our results indicate that as ω_i increases both P_τ and P_τ' increase while P_τ' is significantly smaller than P_τ. For example, when $\omega_i = 32$, P_τ is 0.99 whereas P_τ' is only 0.014. Also a larger α_{max} will achieve a smaller P_τ'. From Fig.8 and 7, we can see that in general cases Δ-mobility reduces more than 80% itinerary exposure probability with a travel overhead less than 0.03.

4.3.3 Analysis of Δ-mobility Collision

Now we measure the probability of having a mobility collision, i.e two or more nodes generate the same Δ-mobility. In our model, we assume there are N_r rendezvous in the network and the number of nodes in the

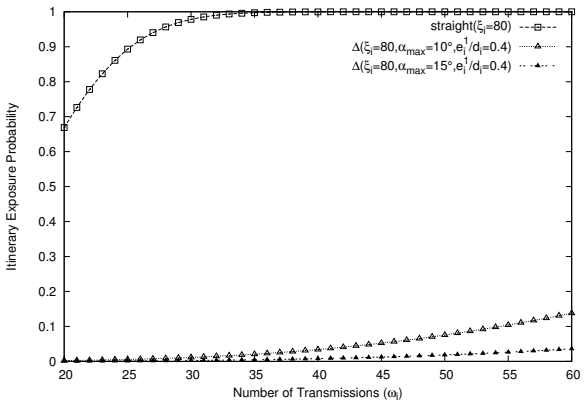

Figure 8. Itinerary Exposure Probability Comparison

network is n. We assume a node has equal probabilities of choosing any two rendezvous as the endpoints of an itinerary segment.

Equation (8) calculates the probability of having a Δ-mobility collision assuming all nodes may simultaneously generate their Δ-mobility. Fig. 8 shows the collision probability when we vary N_r and n and set $\alpha_{max} = 30^\circ$ and $e_i^1 = 160\ m$. From Fig. 9, we know that the probability of collision is negligible ($< 0.02\%$).

$$P_{collision}(N_r, \alpha_{\max}, e_i^1, n) = 1 - \frac{\prod_{i=1}^{n} (N_r \times N_r \times \alpha_{\max} \times e_i^1 - i + 1)}{(2 \times N_r \times N_r \times \alpha_{\max} \times e_i^1)^n} \quad (8)$$

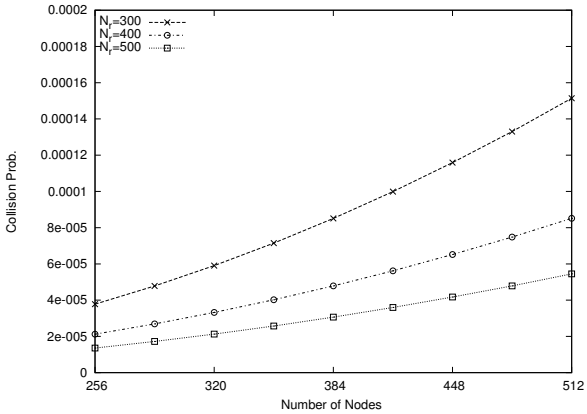

Figure 9. Δ-mobility Collision Probability

5 Evaluation of Routing Performance

In this section, we evaluate the influences of Δ-mobility algorithm on the routing performance. We conduct simulations using Qualnet Network Simulator [17] and use DSR [8] routing protocol. In DSR routing, when a source does not have the route to the destination, it launches route discovery to find the route to the destination.

We focus on comparing the following three metrics:

- *message delivery ratio:* This metric measures the percentage of data messages that are delivered to destinations.
- *message delivery latency:* This metric measures how long it takes for a data message to be delivered.
- *routing overhead:* This metric measures the ratio of the number of bytes sent (data plus control messages) to the number of delivered data bytes.

Through comparing DSR routing performance under different mobility models, nodes moving speeds and traffic conditions, we assess the influence of Δ-mobility camouflaging algorithm on routing performance. Table 1 summarizes the simulation configurations. For each simulation, we run 6 times (each with a different random seed) and obtain the results by taking average.

Table 1. Simulation Configuration

Number of Nodes	64
Terrain Dimension	1024×1024 (m)
Simulation Time	800 S
Radio Model	IEEE 802.11a 24Mb/s
Mobility Model	RW, RW-Δ, RV, RV-Δ
α_{max}	15°
e_i^1/d_i	$0.2 \leq e_i^1/d_i \leq 0.4$
Node Speed	2∼14m/s
Message Sending Speed	6 concurrent traffic flows, each with speed 20∼ 80 pkts/second
Data Message Size	512 bytes

One mobility model we use in our simulations is *random-waypoint* mobility model or RW in short. In random waypoint mobility model, a node randomly selects a position, moves to the position, stays there for a period time and repeat the process again [8]. We then add Δ-mobility to RW model for each movement segment. We denote DSR-RW as DSR routing using RW

mobility model and denote DSR-RW-Δ as DSR routing using RW mobility model plus Δ-mobility camouflaging algorithm.

Another mobility model we designed for testing Δ-mobility camouflaging algorithm is RV mobility model. In contrast to the random movement of RW mobility model, the movements of the nodes in RV mobility model follow their itineraries. And here we predefine the itineraries for the nodes. Δ-mobility is also added to this model for each itinerary segment. DSR using RV and its Δ-mobility alternative are denoted as DSR-RV and DSR-RV-Δ respectively.

In our communication model, at any time there are six concurrent transmission flows with randomly-chosen message source and destination. In the first set of the simulations, we fix data message sending speed to be 40 messages per second and evaluate the routing performance of the above-mentioned mobility models under different mobility speeds. In the second set of the simulations, we measure the routing performance of these mobility models under increasing traffic loads when the mobility speed is 4m/s.

The delivery ratio curves showed in Fig. 10 and Fig. 13 show that all four mobility models are able to deliver more than 99% of the data messages. More importantly, the figures show that adding Δ-motion almost causes no change in the packet delivery ratio.

From Fig.11 and Fig.14, we can see that the delivery latency change caused by Δ-mobility camouflaging algorithm is neglible. Plus it can be noticed that when RW has a slightly larger latency when node moving speed increases. This is because the connectivity of nodes using RW mobility model is easier to be influenced by increasing mobility due to the randomness of node movements. Because of the movements of nodes using RV mobility model are determined by their itineraries, the connectivity among the nodes are more stable and the delivery latency shows little variation as mobility speed increases.

Fig. 12 and Fig. 15 show that Δ-mobility camouflaging algorithm does not increase routing overhead, which is about 4 on average in our simulations. The routing overhead comprises the overehead of transmitting data messages hop by hop on the route from the source to the destination and the message overhead to construct routes (i.e route request and route reply packets in DSR). Since the data message size (i.e. 512 bytes) is much larger than the control message size (it takes 2 bytes to describe a hop on a route), in our simulation the routing overhead is primarily determined by the average length of routes. Thus, the overhead difference between RV and RW mobility model is resulted from their difference on the average length of routes.

In summary, from the simulation results of RW and RV mobility model under different traffic loads and node moving speeds, we can see that Δ-mobility camouflaging algorithm does not degrade DSR's routing performance.

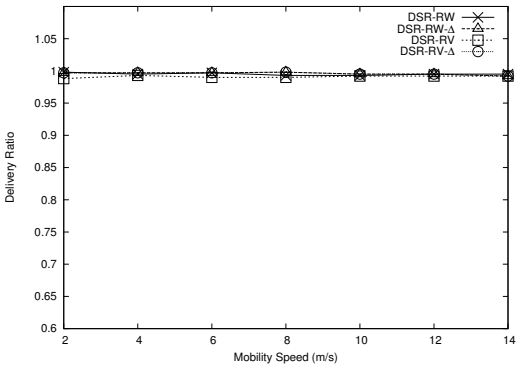

Figure 10. Delivery Ratio VS. Mobility

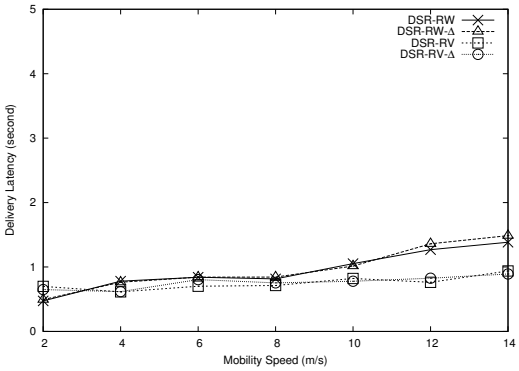

Figure 11. Delivery Latency VS. Mobility

6 Conclusion

The paper has described the itinerary privacy attack and a model to measure itinerary privacy. To protect itinerary privacy, we designed Δ-mobility camouflaging algorithm. Through mathematical analysis, we have showed that in general cases Δ-mobility decreases more than 80% itinerary exposure probability with less than 3% extra travel distance. Our simulation results show that using Δ-mobility camouflaging algorithm does not lead to DSR routing performance degradation under different mobility models and different traffic load conditions.

In our future work, we will extend Δ-mobility camouflaging algorithm to other network scenarios such as

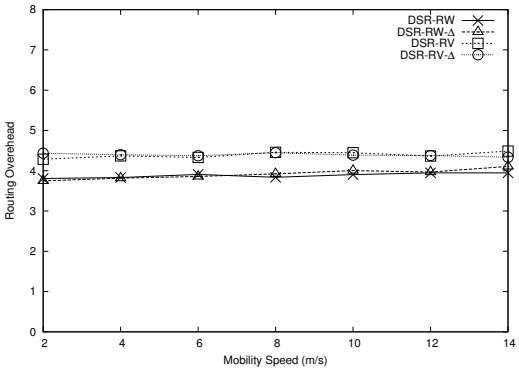

Figure 12. Routing Overhead VS. Mobility

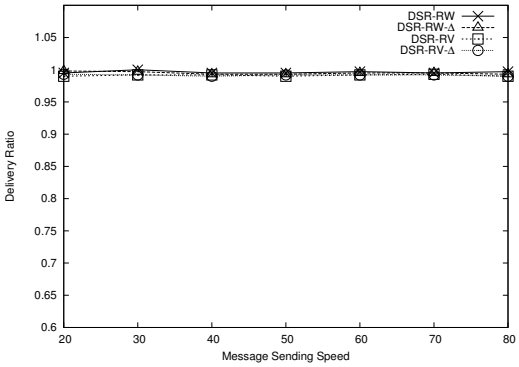

Figure 13. Delivery Ratio VS. Traffic Load

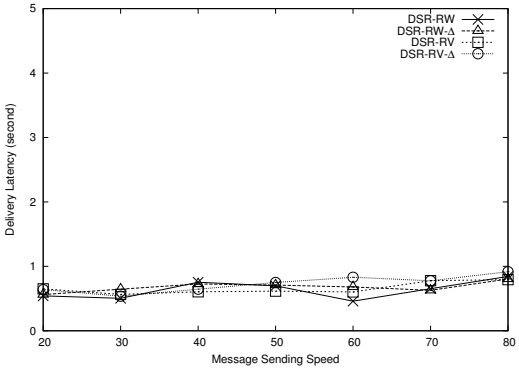

Figure 14. Delivery Latency VS. Traffic Load

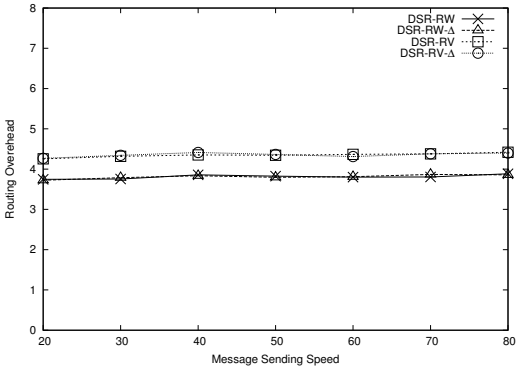

Figure 15. Routing Overhead VS. Traffic Load

delay-tolerant networks. Based on motions generated by Δ-mobility camouflaging algorithm, we plan to construct a routing scheme that provides comprehensive privacy protection.

References

[1] A. R. Beresford and F. Stajano. Mix Zones: User Privacy in Location-aware Services. In *PERCOMW '04*, page 127, Washington, DC, USA, 2004.

[2] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba. SDAR: A Secure Distributed Anonymous Routing Protocol for Wireless and Mobile Ad Hoc Networks. In *LCN '04: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*, pages 618–624, USA, 2004.

[3] D. L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Commun. ACM*, 24(2):84–90, 1981.

[4] D. B. Faria and D. R. Cheriton. Detecting identity-based attacks in wireless networks using signalprints. In *WiSe '06*, pages 43–52, New York, NY, USA, 2006.

[5] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427, 1999.

[6] M. Gruteser, J. Bredin, and D. Grunwald. Path Privacy in Location-aware Computing. In *MobiSys 2004*, June 2004.

[7] B. Hoh and M. Gruteser. Protecting Location Privacy Through Path Confusion. *Securecomm*, 0:194–205, 2005.

[8] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, volume 353, pages 153–181, 1996.

[9] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk. Enhancing Source-Location Privacy in Sensor Network Routing. In *ICDCS '05*, pages 599–608, USA, 2005.

[10] J. Kong and X. Hong. ANODR: ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks. In *ACM MOBIHOC'03*, pages 291–302, 2003.

[11] J. Kong, D. Wu, X. Hong, and M. Gerla. Mobile traffic sensor network versus motion-mix: tracing and protecting mobile wireless nodes. In *SASN '05*, pages 97–106, New York, NY, USA, 2005.

[12] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. In *SIGMOBILE Mobile Computing Communications Review*, pages 7:19–20, July 2003.

[13] S. W. Liu Yang, Markus Jakobsson. Discount Anonymous On Demand Routing for Mobile Ad hoc Networks. In *SecureComm*, MD, USA, Sep. 2006.

[14] C. E. Perkins and E. M. Royer. Ad-hoc On-Demand Distance Vector Routing. In *IEEE WMCSA '99*, pages 90–100, Washington, DC, USA, 1999.

[15] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, AC-24:843–854, 1979.

[16] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran. AMOEBA: Robust Location Privacy Scheme for VANET. *IEEE Journal on Selected Areas in Communications*, 25:1569–1589, 2007.

[17] Scalable Network Technologies (SNT). Qualnet Network Simulator, http://www.qualnet.com/.

[18] R. Song, L. Korba, and G. Yee. AnonDSR: efficient anonymous dynamic source routing for mobile ad-hoc networks. In *SASN '05: Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pages 33–42, New York, NY, USA, 2005.

[19] P. Tao, A. Rudys, A. M. Ladd, and D. S. Wallach. Wireless LAN location-sensing for security applications. In *WiSe '03*, pages 11–20, New York, NY, USA, 2003.

[20] X. Wu and B. Bhargava. AO2P: Ad Hoc On-Demand Position-Based Private Routing Protocol. *IEEE Transactions on Mobile Computing*, 4(4):335–348, 2005.

[21] Y. Zhang, Y. Huang, and W. Lee. An Extensible Environment for Evaluating Secure MANET. In *SecureComm*, pages 339– 352, Greece, Sep. 2005.

[22] Q. Zheng, X. Hong, and J. Liu. An agenda based mobility model. In *39th Annual Simulation Symposium*, 2006.