4PR: Privacy Preserving Routing in Mobile Delay Tolerant Networks

Jingwei Miao[a], Omar Hasan[a], Sonia Ben Mokhtar[a], Lionel Brunie[a], Ammar Hasan[b]

[a] *University of Lyon, CNRS, INSA-Lyon, LIRIS, UMR5205, F-69621, France*
[b] *SEECS, National University of Sciences and Technology, Islamabad, Pakistan*

Abstract

Message routing is one of the major challenges in Mobile Delay Tolerant Networks (MDTNs) due to frequent and long-term network partitions. A number of routing protocols for MDTNs belong to the category of prediction-based routing protocols, which utilize the social encounter probability of nodes to guide message forwarding. However, these prediction-based routing protocols compromise the privacy of the nodes by revealing their mobility patterns. In this paper, we propose the Privacy Preserving Probabilistic Prediction-based Routing (4PR) protocol that forwards messages by comparing aggregated information about communities instead of individual nodes. Specifically, it compares the probability that at least one node in a community will encounter the destination node. We present theoretical security analyses as well as practical performance evaluations. Our simulations on a well established community-based mobility model demonstrate that our routing protocol has comparable performance to existing prediction-based protocols. Additionally, the community information is computed efficiently and independently of the routing protocol.

Keywords: privacy, routing protocols, mobile computing, delay tolerant networks

1. Introduction

Mobile Delay Tolerant Networks (MDTNs) (also referred to as Mobile Opportunistic Networks) are constructed by the intermittent connection of co-located mobile devices. The MDTN architecture caters to the rapidly expanding cyber-physical space where mobile and socially connected human users are coupled with smart portable devices forming mobile network nodes. The short range networking interfaces (e.g., Bluetooth) of these devices enable Mobile

Email addresses: jingwei.miao@insa-lyon.fr (Jingwei Miao), omar.hasan@insa-lyon.fr (Omar Hasan), sonia.benmokhtar@insa-lyon.fr (Sonia Ben Mokhtar), lionel.brunie@insa-lyon.fr (Lionel Brunie), ammar.hasan@seecs.edu.pk (Ammar Hasan)

Networking in Proximity (MNP), where neighboring devices interact through short-range communications. However, routing messages between two nodes that are not within communication range is a challenge in MDTNs since an end-to-end routing path cannot be guaranteed. The applications developed in these networks are often geo-localized with no critical time constraint, e.g., advertisement dissemination, recommendation of points of interest, and asynchronous communication.

In order to deal with the lack of end-to-end connectivity between nodes, message routing in MDTNs is often performed in a "store-carry-and-forward" manner [1], in which a node may store and carry a message for some time before opportunistically forwarding it to another node [2]. In order to better choose intermediary nodes, a number of routing protocols [3, 4] forward a message from one intermediate node to another if the latter has higher probability of encountering the destination node. Such routing protocols are called prediction-based routing protocols. It has been shown that these protocols perform better than other protocols when nodes exhibit well-known mobility patterns [3, 4]. However, prediction-based routing protocols implicitly assume that nodes accept revealing their mobility patterns to other nodes. In practice, the disclosure of mobility patterns can result in the unwillingness of nodes to participate in MDTNs due to privacy concerns [5].

In this paper, we present the Privacy Preserving Probabilistic Prediction-based Routing (4PR) protocol for MDTNs. For routing a message, 4PR distinguishes the routing inside a community from the routing between communities. A community is defined as a set of nodes that frequently encounter each other (see Section 3). For disseminating a message inside a community, 4PR relies on the epidemic protocol [6], which by construction preserves the privacy of nodes and is efficient as communities are small. The main challenge addressed by 4PR is thus the routing of a message between communities in a privacy preserving manner. To do so, each node in the network calculates the probability that at least one of the nodes in its community will encounter the destination. When two nodes from different communities encounter, instead of comparing their respective probabilities to encounter the destination node, they compare the aforementioned probabilities to determine the message forwarding decision. The probability that at least one node in a community will encounter a given node in the network is computed in a privacy preserving manner within the community using the MDTN-Private-Probability protocol, presented in Section 5.

To the best of our knowledge, only our previous work (the 3PR protocol [7]) has addressed the privacy issue of prediction-based routing protocols. In 3PR, message routing is guided by the maximum probability that nodes in a community will encounter a destination node. In contrast, in the 4PR protocol, message routing is guided by the probability that at least one node in a community will encounter the destination node. This fundamental difference in how messages are forwarded provides 4PR some significant advantages over 3PR. As we discuss in further detail in Section 2.1, the advantages of 4PR over 3PR include: 1) better privacy preservation since the true upper bound of encounter probabilities is not revealed; 2) private computation of probability is more ef-

ficient than private computation of maximum; 3) the probability that at least one node in a community will encounter the destination node is a more accurate measure for routing path prediction than the maximum probability in the community.

We evaluate 4PR both theoretically by providing security analyses (Sections 4 and 5) and practically through extensive simulations (Section 6). We have conducted our simulations based on a well established community-based mobility model [8, 9]. We compare the performance of 4PR against five state-of-the-art protocols, i.e., epidemic [6], Direct [10], PRoPHET [11], Bubble [12], and the 3PR protocol [7]. Epidemic and Direct are traditionally considered to achieve the upper and lower bounds of routing performance. PRoPHET and Bubble are representatives in prediction-based and community-based routing protocols respectively. Results show that 4PR has comparable performance to existing prediction-based protocols while preserving the privacy of the nodes.

The remainder of this paper is structured as follows. Section 2 discusses related work on privacy preserving protocols in MDTNs. The system model is described in Section 3. We describe the 4PR protocol in Section 4 followed by the MDTN-Private-Probability protocol presented in Section 5. The performance evaluation is subsequently presented in Section 6. We conclude in Section 7.

2. Related Work

Recent years have seen considerable research addressing the issues of privacy in delay tolerant networks. The protocols in the literature are mainly concerned with preserving the privacy of one or more of the following sensitive user aspects [5]: (1) identity, (2) location, (3) message content, and (4) relationships. In contrast, our protocol 4PR is a novel type of protocol, which has the specific goal of hiding the encounter probabilities of nodes. Therefore, 4PR differs fundamentally from other existing privacy preserving routing protocols for MDTNs due to the difference in objectives.

Hasan et al. [7] proposed the Privacy Preserving Prediction-based Routing (3PR) protocol for MDTNs, which is the predecessor of the 4PR protocol presented in this paper. This is the only other work that we are aware of that has the same objective as 4PR, i.e., hiding the encounter probabilities of nodes. In 3PR, when two nodes from different communities encounter, they compare the *maximum probability in their community* that a given node will encounter the destination. However, compared with 4PR, the forwarding decision mechanism of 3PR has the following two shortcomings. Firstly, 3PR consumes much more resources in terms of the number of message copies to compute the maximum probability in the community (see Section 6.1). Secondly, the maximum probability in the community cannot accurately measure the probability of all nodes in the community delivering the message to the destination node, due to the message being flooded inside the community.

We note some protocols that attempt to preserve privacy in the other aforementioned categories. In the category of identity privacy, the identity of nodes

participating in message delivery is considered as private information. Papapetrou et al. [13] propose the SimBet-BF routing protocol for MDTNs. This anonymized routing protocol represents all node identities using Bloom filters. The desired effect is that two nodes can exchange information while maintaining the privacy of their identity and their past encounters. However, the protocol described by Papapetrou et al. is not a prediction-based protocol. In fact, a direction of future work described by the authors is to use encounter information to enhance the routing protocol.

Kate et al. [14] presented an anonymous communication architecture for MDTNs using Identity-Based Cryptography (IBC). This is one of the first anonymous communication solutions specifically for MDTNs. Kate et al. use a construct called MDTN gateways, which are entities assumed to be trusted and to be aware of user identities. In the routing process, a MDTN gateway replaces the identity of a source node with a pseudonym unlinkable to the identity. The advantage of the protocol is that there is not much overhead for routing. However, the protocol relies on the assumption that trusted MDTN gateways are present, which is a strong assumption for MDTNs.

In the category of location privacy in MDTNs, the discovery of the user location by the adversary is considered as the main privacy threat. Zakhary and Radenkovic [15] presented a location privacy protocol that is based on the utilization of social information of nodes. In this protocol, each node maintains a social profile, which includes n profile attributes. The social relationship between nodes are inferred by the matching of profile attributes. For each message, the forwarding is guided by the obfuscated attributes in the first k hops. After that, the message can be routed by any routing protocols. Therefore, an adversary cannot distinguish the location of the source node from the other k relay nodes. However, nodes that have strong social relationships are generally considered to be frequently co-located. Thus, the adversary can still detect the approximate location of the source node. Moreover, the routing performance is degraded, due to the extra k forwarding hops.

Since messages are relayed by intermediary nodes in MDTNs, the content of messages can be unintentionally disclosed to these nodes in the routing process. Thus, in the category of message content privacy, the content of messages is considered as private information. Shi and Luo [16] proposed an anonymous communication mechanism called ARDEN based on onion routing [17], multicast dissemination and Attribute-Based Encryption (ABE) [18]. In ARDEN, before sending a message, the source node determines a path of disjoint groups, one of which includes the destination node. The message is then encrypted by the keys of the destination node and the grouping keys. Compared with the traditional onion routing, the advantage of ARDEN is that it encrypts messages with the keys of groups rather than the keys of individual intermediate nodes. The performance in terms of delivery ratio and delivery latency can be improved, since all nodes in the same group can participate in message forwarding. On the other hand, the arbitrary group partitioning manner may result in performance degradation in terms of delivery ratio and delivery latency.

In the category of relationships privacy in MDTNs, the social relationships

of nodes is considered as personal and private thus users may hesitate in participating in such protocols. Parris and Henderson [19] presented the Privacy-enhanced Social-network Routing protocol. This protocol takes advantage of obfuscated social information rather than accurate social information to guide the message forwarding. The original social information of a node is obfuscated by modifying the friend list, i.e., adding or removing some items into or from the friend list. The advantage of the protocol is that the presence of a public key infrastructure is not necessary. However, message routing may be guided less accurately due to the utilization of obfuscated social information.

2.1. 4PR vs. 3PR

In [7], we presented our Privacy Preserving Prediction based Routing protocol, abbreviated as the 3PR protocol. In this paper, we propose the Privacy Preserving Probabilistic Prediction based Routing protocol, which we abbreviate as the 4PR protocol. The original 3PR protocol provided significant advantages over state of the art routing protocols, notably preservation of user privacy while maintaining comparable routing performance. Our newer 4PR protocol proposes further improvements to privacy preserving prediction based routing. In this section we will present an architectural comparison between 4PR and 3PR, describe the shortcomings of the 3PR protocol, and an overview of how the 4PR protocol overcomes those shortcomings and implements an even stronger privacy preserving routing protocol.

4PR and 3PR share some commonalities, which include routing a message inside a community using the epidemic protocol [20], which by construction preserves the privacy of nodes and is efficient as communities are assumed to be small. The main difference between 4PR and 3PR is how the routing of a message between communities in a privacy preserving manner is handled.

In 3PR, when two nodes from different communities encounter, they compare the *maximum probability in their community* that a given node will encounter the destination. However, in 4PR, when two nodes from different communities encounter, instead they compare the *probability of at least one node in their community* encountering the destination node.

There are a number of advantages to comparing the *probability of at least one node in the community* as in 4PR over comparing the *maximum probability in the community* as in 3PR.

Firstly, in 3PR, although the maximum probability in the community is an aggregate value and does not reveal the precise private probability $P_{a_i,d}$ of an individual node a_i encountering the destination node d, it still divulges some undesirable information about the private probability. Specifically, the maximum value reveals the upper bound on the private value. For example, if the maximum is given as 0.4, then the adversary learns that the private value $P_{a_i,d}$ is no higher than 0.4. On the other hand, the 4PR protocol demonstrates the probability of at least one node in the community encountering the destination. This aggregate value does not reveal the true upper bound or any lower bound on the private value of an individual node.

Secondly, the protocol for computing the probability of at least one node in the community encountering the destination is much more efficient than the protocol for computing maximum probability in the community. As described in Section 5, the protocol for the former requires one round of multiplication, whereas as described in [7], the protocol for the latter requires several rounds of summation depending on the number of bits that represent the private number. The network resources required for multiplication and summation required in the two protocols being equal, the protocol for 4PR is much more resource efficient.

Thirdly, the probability of at least one node in the community encountering the destination (as in 4PR) is a more accurate measure for the likelihood of some node in the community encountering the destination than the maximum probability in the community (as in 3PR). Let's take an example to demonstrate this difference. Let's say that there are two communities C_1 and C_2. Community C_1 has three nodes each of which has a probability of 0.8 of encountering the destination node, whereas community C_2 has three nodes, one with probability of 0.8, and the remaining two with probability 0 of encountering the destination node. The maximum probability in both communities is 0.8 (as in 3PR), whereas the probability of at least one node in the community encountering the destination is 0.99 and 0.8 in C_1 and C_2 respectively, according to Equation (3) (as in 4PR). Clearly, 4PR provides a more accurate measure of the likelihood.

The above stated advantages offered by 4PR over 3PR make 4PR a major improvement over 3PR, which was to the best of our knowledge, the first privacy preserving prediction based routing protocol for mobile delay tolerant networks in the literature.

3. System Model

3.1. A Mobile Delay Tolerant Network Model

We consider a set $\mathbb{A}$ of N nodes with communication facilities that can freely roam in a physical environment. The communication facilities consist of a short range wireless connection. Two nodes can communicate only if they are adjacent to each other, i.e, if they are physically within each other's transmission range. We assume that the communication is unreliable, i.e., a message sent from a node to an adjacent node may not arrive. However, we assume that a node knows whether the transmission of a message has been interrupted by a network failure or whether the message correctly reached the intended recipient.

To send a message to a destination node that is not within the transmission range of the source node, the latter uses a routing protocol. The routing strategy that we consider in this work is prediction-based routing [21]. We generalize prediction-based routing protocols as follows: Consider a node a that has a message for a destination node d. When the node a encounters another node b, it forwards a copy of the message to the node b if the probability of b encountering d (given as $P_{b,d}$) is higher than the probability of a encountering d (given as $P_{a,d}$). Thus the probability that a node with a copy of the message will encounter the

destination node continues to rise until the message is delivered or the Time To Live (TTL) of the message expires.

As demonstrated in many studies of real human mobility traces, we assume that nodes belong to communities [12]. We define a community C as a set of nodes such that $C \subset \mathbb{A}$. We assume that the nodes in a community are frequently physically collocated and thus a high probability exists of successful message delivery from any source node in a community to any destination node in the community. A node $l \in C$ is designated as the leader of the community. A consensus protocol may be used for the election of the leader node within a community. The leader node maintains the list of the nodes in the community. Let the set of nodes in a community $C = \{a_1, a_2, \ldots, a_n\}$, where $n = |C|$. We consider a community to comprise of at least three nodes, that is, $n \geq 3$. The topic of community management has been discussed in detail in the literature by several authors including Hui et al. [22], Dang and Wu [9], and Miao et al. [23].

We consider the probability that a node a will encounter a node d as private information. Nodes are willing to let this private information be used for routing of messages. However, nodes require that their private information is not revealed to any other node in the network, which includes fellow nodes in a community.

In this paper, we consider the semi-honest adversarial model [24]. The nodes in this model always execute the protocol according to the specification. However, the adversary passively attempts to learn the private information of nodes by using intermediate information gleaned during the execution of the protocols.

3.2. Computation of Encounter Probabilities of Nodes

In this paper, the encounter probabilities of nodes are computed according to the method proposed by Lindgren et al. [11]. The computation of the encounter probabilities of nodes is driven by events. There are two kinds of events: (1) *Connect Event*, and (2) *Update Event*.

(1) *Connect Event*. It happens at the moment when two nodes, e.g., nodes a and b, encounter each other. When a connect event takes place, two encountering nodes compute their encounter probabilities for a given time window according to Equation (1), where $P_{init} \in [0, 1]$ is an initialization constant, and $P^{'}_{a,b}$ is the previous probability that node a may encounter node b.

$$P_{a,b} = P^{'}_{a,b} + (1 - P^{'}_{a,b}) \times P_{init} \tag{1}$$

(2) *Update Event*. The update event is periodically invoked by all nodes every δ time units. When an update event happens, each node in the network utilizes an aging equation to reduce the probabilities of encountering the other nodes. The intuition behind such a strategy is that a pair of nodes are less likely to encounter each other in the future if they have not encountered in a while. The aging equation is expressed in Equation (2), where $\alpha \in [0, 1)$ is an aging constant.

$$P_{a,b} = P^{'}_{a,b} \times \alpha \tag{2}$$

It is worth pointing out that the computation of the encounter probabilities of nodes are based on their own histories. This implies that nodes compute the encounter probabilities locally. Therefore, the computation of these probabilities is carried out in a privacy preserving manner.

4. 4PR: Privacy Preserving Probabilistic Prediction-based Routing

4.1. Protocol Description

As stated in Section 3, C is a community, such that $C = \{a_1, a_2, \ldots, a_n\}$, and $n = |C|$. Let $P_{C,d} = prob(C, d)$ be the probability that at least one node a_i in community C will encounter the destination node d, given as Equation (3).

$$P_{C,d} = 1 - \prod_{i=1}^{n}(1 - P_{a_i,d}) \tag{3}$$

We now present an overview of 4PR, our Privacy Preserving Probabilistic Prediction-based Routing protocol. A routing example is depicted in Figure 1. This figure shows a number of nodes belonging to three communities C_1, C_2 and C_x. A source node s that belongs to the community C_1 wants to send a message to a node d that belongs to the community C_x.

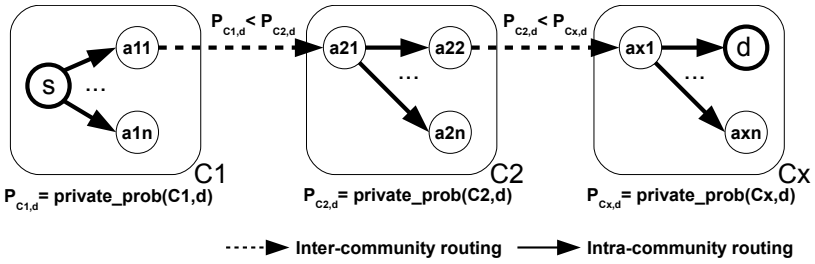


Figure 1: 4PR Protocol Overview

In 4PR, we distinguish the routing inside a community from the routing between communities. Specifically, when two nodes that belong to the same community encounter each other, they exchange all the messages that they each have. On the other hand, if two nodes a_{11} and a_{21} that belong to different communities C_1 and C_2 respectively encounter each other, node a_{11} forwards a message intended for a destination node d to node a_{21}, only if the probability of at least one node in community C_2 encountering d (given as $P_{C_2,d}$) is higher than that in community (given as $P_{C_1,d}$). In Figure 1, when node a_{11} encounters node a_{21}, node a_{11} forwards the message intended for d to node a_{21} because $prob(C_2, d) > prob(C_1, d)$.

In other words, to route a message m from s to d, m is first disseminated in an epidemic manner inside the community C_1. Message m then moves from a community to another such that: (1) at each forwarding step, the probability

Protocol: MDTN-4PR
Participants: Node a and node b, where $a, b \in \mathbb{A}$.
Input: (1) m, a message. (2) d, the destination node of message m. (3) C_a, the set which denotes the community of node a. (4) C_b. (5) $P_{C_a,d} = prob(C_a, d)$, that is the probability that at least one node in community C_a will encounter the destination node d. (6) $P_{C_b,d}$.
Output: Message m is delivered to the node b if $b = d$, or $b \in C_a$, or $P_{C_b,d} > P_{C_a,d}$.
Setup: Node a has a message m whose destination is node d.
Events and Associated Actions:

node a **encounters a node** b

1 **if** $b = d$
2 **then** node a sends message m to node b
3 **elseif** $b \in C_a$
4 **then** node a sends a copy of the message m to node b
5 **elseif** $P_{C_b,d} > P_{C_a,d}$
6 **then** node a sends a copy of the message m to node b

Figure 2: Protocol: MDTN-4PR

of at least one node in the next community to reach the destination is higher than that in the previous community, (2) as soon as it reaches a community, m is disseminated in an epidemic manner within the community.

A key characteristic of 4PR is that $P_{C_a,d} = prob(C_a, d)$, the probability that at least one node in community C_a will encounter the destination node d, is computed in a privacy preserving manner, that is without revealing the individual probabilities of the nodes in the community. $prob(C_a, d)$ is therefore denoted as $private\_prob(C_a, d)$ in Figure 1.

Our protocol 4PR for Privacy Preserving Probabilistic Prediction-based Routing in Mobile DTNs is specified in Figure 2. The computation of $private\_prob(C_a, d)$ is performed using a decentralized protocol for privately computing the function over a set of values in a delay tolerant manner without revealing the individual values, i.e., MDTN-Private-Probability, further described in Section 5.

The probability is computed periodically in the community independently from the routing protocol. Therefore, the complexity of the MDTN-Private-Probability protocol has no direct impact on the performance of the routing protocol.

4.2. Security Analysis: Correctness

With each forwarding of the message, the conventional prediction-based routing strategy delivers a copy of the message to a node that has a higher probability of encountering the destination node. We consider our protocol 4PR to be correct if it achieves the same effect as the conventional prediction-based routing strategy.

In 4PR, a node a in community C_a sends message m to node b in a community C_b if a and b encounter and $P_{C_b,d} > P_{C_a,d}$, i.e., if the probability of at least one node in C_b encountering the destination node d is higher than that in C_a (lines 7 and 8). Upon receiving the message m, node b floods the message to all nodes in C_b (lines 4 and 5). In Section 3, we stated the assumption that a high probability exists of successful message delivery from any source node in a

community to any destination node in the community. Given this assumption, the message m reaches all nodes in C_b with high probability. As $P_{C_b,d} > P_{C_a,d}$, the protocol succeeds (with high probability) in delivering the message m to a node that has a higher probability of encountering the destination node than the node a.

4.3. Security Analysis: Privacy

A node a reveals to an outsider node only the probability that at least one node in its community C_a will encounter the destination node. This probability is a function of the community as a whole and thus hides the probability of any individual node in the community encountering the destination. Moreover, the probability is computed within the community in a privacy preserving manner using the MDTN-Private-Probability protocol, thus individual probabilities also remain confidential from the nodes inside the community.

The reader may refer to Section 5 for the security analyses of the MDTN-Private-Probability protocol.

5. Privacy Preserving Computation of Probability

5.1. Protocol Description

We describe a protocol for computing the probability $P_{C,d}$, that at least one node a_i in community C in a mobile delay tolerant network will encounter the destination node d, as given in Equation (3).

Our protocol for private computation of probability is inspired by the protocols by Kreitz et al. [25], Sheikh and Mishra [26], and Hasan et al. [7, 27, 28]. However, our protocol addresses specific challenges in MDTNs listed below that the protocols by Kreitz et al. and Sheikh and Mishra do not. Moreover, unlike the protocol by Sheikh and Mishra, our protocol does not require Trusted Third Parties (TTPs).

The mobile delay tolerant network environment presents the following challenges: (1) Mobility implies that the nodes a node will encounter (neighbor nodes in the terminology of graph theory) are not known beforehand. (2) Connectivity is intermittent, messages arrive after long and variable delays, and message transmission is asynchronous.

Each node a_i in the set C participates in the protocol with a private number p_i as an input, where $p_i = 1 - P_{a_i,d}$, that is the complement of the probability that node a_i will encounter the destination node d. The nodes participating in the protocol learn the probability $P_{C,d}$, that at least one node a_i in community C in a mobile delay tolerant network will encounter the destination node d, as given in Equation (3). The protocol is specified in Figure 3.

The protocol is initiated by the leader node of a community given as the set of nodes C. The leader node floods an *init* message (Figure 3: protocol initiation: line 3) to all nodes. After a node receives the *init* message, it sends and receives a random number from each node belonging to C that it encounters (PROBINIT: lines 5 and 6). A node can send the *init* message to an encountered

node if it has not received it yet (PROBINIT: lines 3 and 4). After a node has encountered k nodes (PROBINIT: lines 1 and 2), where k is a constant, the node sends a partial product to the leader node (PROBINIT: line 8). A node computes the partial product as the product of its private number and all random numbers received divided by the product of all random numbers sent (PROBINIT: line 7). The leader node maintains a running product of all partial products received (PROBPARTIAL: line 2). When the partial products are received from all nodes in C (PROBPARTIAL: line 3), the leader node computes the final product γ_C and floods $1-\gamma_C$ to all nodes (PROBPARTIAL: line 4). $1-\gamma_C = P_{C,d}$ is the required probability that at least one node a_i in community C will encounter the destination node d, as given in Equation (3).

Protocol: MDTN-Private-Probability
Participants: Nodes in a community denoted by the set C. One node in C is the leader node denoted by l.
Input: Each node a_i has a private input $p_i = 1 - P_{a_i,d}$.
Output: The nodes in C learn $P_{C,d} = 1 - \prod_{a_i \in C} p_i$.
Setup: (l, g) uniquely identifies a session of the protocol, where g is an integer. k is a constant such that $2 \leq k < n$, and $n\%(k+1) = 0$, where $n = |C|$. Nodes are not ordered, that is, a_i denotes any given node in C.
Events and Associated Actions:

leader node l initiates the protocol

1 $R \leftarrow \phi$
2 $\gamma_C \leftarrow 1$
3 l floods $\langle$PROBINIT, $l, g\rangle$ to all nodes in C

node $a_i \in C$ receives $\langle$PROBINIT, $l, g\rangle$

1 **for** $j \leftarrow 1$ **to** k
2 **do** a_i encounters node $a_j \in C$
3 **if** a_j has not received $\langle$PROBINIT, $l, g\rangle$
4 **then** a_i sends $\langle$PROBINIT, $l, g\rangle$ to a_j
5 a_i sends a random number r_{ij} to a_j
6 a_i receives a random number r_{ji} from a_j
7 $\gamma_i \leftarrow p_i(\prod_{j=1}^{k} r_{ji})/(\prod_{j=1}^{k} r_{ij})$
8 a_i sends $\langle$PROBPARTIAL, $l, g, \gamma_i\rangle$ to l

leader node l receives $\langle$PROBPARTIAL, $l, g, \gamma_i\rangle$ from a_i

1 $R \leftarrow R \cup \{a_i\}$
2 $\gamma_C \leftarrow \gamma_C \times \gamma_i$
3 **if** $R = C$
4 **then** $P_{C,d} = 1 - \gamma_C$
5 l floods $\langle$PROBFINAL, $l, g, P_{C,d}\rangle$ to all nodes in C

Figure 3: Protocol: MDTN-Private-Probability

5.2. The Value of Constant k

The choice of the value of constant k depends on the value of n, where $n = |C| \geq 3$. As stated in Section 3, we consider a community C to comprise of at least three nodes. Since a node in the protocol can exchange random numbers with at most all other nodes in its community, the interval of the constant k can be given as $[2, n)$, i.e., $2 \leq k < n$.

Additionally, when $k = 2$, whatever the value of n, these n nodes can always make a pair. Therefore, k can always be set as 2. When $2 < k < n$, according to the mechanism of our protocol, each node should exchange random numbers with k distinct nodes in its community. Hence, there are nk random numbers generated in each execution of our protocol. These nk random numbers should be divisible by $k + 1$. That is $n(k + 1 - 1) = n(k + 1) - n$ is divisible by $k + 1$. Therefore, the value of the constant k should also be compatible with: $n\%(k + 1) = 0$.

Summarizing, the value of the constant k should meet the following two requirements: 1) $2 \leq k < n$, and 2) $k = 2$ or $n\%(k + 1) = 0$.

5.3. Security Analysis: Correctness

The first challenge for the protocol due to the mobile delay tolerant network environment is that the nodes a node will encounter (neighbor nodes) are not known beforehand. To address this challenge, the protocol allows a node $a_i \in C$ to encounter any other k nodes in C (PROBINIT: lines 1 and 2). The encountered nodes, given as a_j, where $j \in \{1, 2, \ldots, k\}$, are considered as the neighbors of node a_i.

Each node $a_i \in C$ sends a random number r_{ij} to each encountered node a_j (PROBINIT: lines 5 and 6). Node a_i divides its product γ_i by r_{ij}, whereas node a_j multiplies its product γ_j by r_{ij} (PROBINIT: line 7). Each node a_i also multiplies its private value p_i to its product γ_i (PROBINIT: line 7). When the leader node computes $\gamma_C = \prod_{a_i \in C} \gamma_i$, the product γ_C is the required product $\prod_{a_i \in C} p_i$ because γ_i and γ_j are divided by and multiplied by r_{ij} respectively which results in being multiplied by the multiplicative identity 1 (PROBPARTIAL: lines 1 – 4).

The second set of related challenges of mobile delay tolerant network environments are as follows: connectivity is intermittent, messages arrive after long and variable delays, and message transmission is asynchronous. The following two elements of the protocol address this set of challenges: (1) The *init* message (PROBINIT) reaches all nodes in C with high probability and thus they all participate in the protocol. This is due to the assumption that a high probability exists of successful message delivery from any source node to any destination node in a community. (2) If a node $a_i \in C$ that has received the *init* message encounters a node $a_j \in C$ that has not yet received the *init* message then a_i sends a copy of the message to a_j to initiate it to the protocol (PROBINIT: lines 3 and 4). Nodes consider an encounter successful only if they exchange all messages according to the specification during their period of contact. Otherwise, they ignore any partial messages sent and received.

5.4. Security Analysis: Privacy

Let's consider a node $a_i \in C$. In an ideal protocol [24], the node would submit its private value p_i to a TTP. The TTP is considered trustworthy therefore it would not disclose the private value p_i of node a_i to any other party. It would only reveal the output of the protocol, which is the product of the private

values received from all nodes in C, and consequently the probability as defined in Equation (3).

In the MDTN-Private-Probability protocol, node a_i discloses the following information: (1) One random number to each of the k nodes that it encounters after receiving the PROBINIT message. (2) The value γ_i to the leader node l as part of the PROBPARTIAL message. The value γ_i is also revealed to the intermediate nodes that participate in the delivery of the message to the leader node.

The random numbers r_{ij}, where $j \in \{1, 2, \ldots, k\}$, are independent of p_i therefore they reveal no information about p_i.

$\gamma_i = p_i\theta_i$, where $\theta_i = (\prod_{j=1}^{k} r_{ji})/(\prod_{j=1}^{k} r_{ij})$. Let's assume that the interval of the random numbers is large compared to the interval of p_i and that the random numbers are distributed uniformly. This implies that the interval of θ_i is also large and that it is distributed uniformly. Thus there is high probability that the adversary can learn no information about p_i from γ_i.

The adversary can learn p_i if it learns θ_i in addition to γ_i. To learn θ_i, the adversary must learn all values r_{ij} and r_{ji}. This is possible only if all k nodes a_j that encountered node a_i are dishonest and collude to reveal all of their individual r_{ij} and r_{ji} values and consequently the value of θ_i.

As in the ideal protocol, the output of the protocol is the product of the private values of all nodes in C, and consequently the probability as defined in Equation (3). The MDTN-Private-Probability protocol thus does not reveal any more information about the private value p_i of node a_i than the ideal protocol if the following assumptions hold true: (1) the interval of the random numbers r_{ij} and r_{ji} is large compared to the interval of p_i and the random numbers are distributed uniformly, and (2) at least one of the k nodes that encountered node a_i is honest.

5.5. Security Analysis: Probability of Privacy Breach

As we described in the previous section, the adversary can learn p_i if all k nodes a_j that encountered node a_i and the leader node l are dishonest. Let P_D denote the probability that the private value of a node a_i is disclosed by the collusion of dishonest nodes. Let P_l denote the probability that the leader node l is dishonest. Let P_k denote the probability that the k encountered nodes of node a_i are dishonest. According to the above analysis, we can see that $P_D = P_l \times P_k$. Hence, P_D depends on the number of nodes in community C, the value of k, and the number of dishonest nodes in community C. Let's denote the number of dishonest nodes as h, where $0 \leq h \leq n-1$.

Hence, if we assume that the leader l is randomly chosen from the community, then P_l can be expressed as Equation (4).

$$P_l = \frac{h}{n-1} \tag{4}$$

Moreover, due to the random mobility model, we can assume that the encounters are random and cannot be scripted by the adversary. According to the

values of k, h, and n, the analysis of P_k can be divided into the following two cases: (1) $0 < h < k$; (2) $k \leq h \leq n-1$. In the first case, the private information of node a_i cannot be learned by the adversary node, i.e., $P_k = 0$. In the second case, there are C_h^k combinations that all the k encountered nodes met by node a_i are dishonest, while there are C_{n-1}^k combinations that node a_i encounters k distinguish nodes inside community C, i.e., $P_k = C_h^k / C_{n-1}^k$. Hence,

$$P_k = \begin{cases} 0, & \text{if } 0 \leq h < k \\ \frac{C_h^k}{C_{n-1}^k}, & \text{if } k \leq h \leq n-1 \end{cases} \tag{5}$$

Combining (4) and (5), the probability P_D can then be expressed as Equation (6).

$$P_D = \begin{cases} 0, & \text{if } 0 \leq h < k \\ \frac{h}{n-1} \times \frac{C_h^k}{C_{n-1}^k}, & \text{if } k \leq h \leq n-1 \end{cases} \tag{6}$$

In addition, one unavoidable side-effect of the protocol is that the adversary learns that node a_i's probability (i.e., $P_{a_i,d}$) of encountering the destination node d is not higher than $P_{C,d}$, since $P_{C,d} = P(\bigcup_{x=1}^{n} P_{a_x,d}) \geq P_{a_i,d}$, where $n = |C|$, $1 \leq i \leq n$. However, in contrast to the previous protocol (3PR), the 4PR protocol does not reveal the true upper bound of any individual node.

5.6. Complexity Analysis

In this section, we discuss the complexity or the overhead of computing $P_{C,d}$ using the MDTN-Private-Probability protocol. According to the mechanism of MDTN-Private-Probability protocol (see Figure 3), the information, which is utilized to compute $P_{C,d}$ in a given community C, is transmitted between nodes in the following four sub-processes: (1) the leader node floods the *PROBINIT* message to all other nodes in community C; (2) each node in community C exchanges k random values (with the first k distinct nodes in community C; (3) each node directly sends the mixed value (*PROBPARTIAL*) to the leader node; and (4) the leader node floods the final result (*PROBFINAL*) to all other nodes.

Let's consider that each field (i.e., an integer or a real) of each message occupies β bits (i.e., of the same size). In the sub-processes (1), all the nodes in community C (except the leader node) get a copy of the message which contains three fields. That is, $n-1$ messages exchanged between nodes. In the sub-process (2), each of the nodes in community C sends k messages which contains only one field to the first k community members. Therefore, there are kn messages exchanged in this sub-process. In sub-process (3), all the nodes in community C (except the leader node) sends a message with four fields to the leader node. That is, $n-1$ messages exchanged in the sub-process (3). Since sub-process (4) utilizes the same method as in sub-process (1) to disseminate messages which contain four fields, the amount of messages transmitted between nodes in this sub-process is $n-1$. Consequently, the overhead of computing $P_{C,d}$ in community C is $\beta((k+11)n-11)$. That is, the protocol requires $O(k\beta n)$ bits

Table 1: Protocol MDTN-Private-Probability - Complexity

Sub-process	**(1)**	**(2)**	**(3)**	**(4)**
No. of bits	$3\beta(n-1)$	$k\beta n$	$4\beta(n-1)$	$4\beta(n-1)$
Complexity	$O(\beta n)$	$O(k\beta n)$	$O(\beta n)$	$O(\beta n)$

to be exchanged, where k and β are constants, and $n = |C|$. Table 1 represents an analysis of the communication complexity of the MDTN-Private-Probability protocol.

6. Performance Evaluation

In this section, we first present a comparison between private probability and private maximum, the background protocols employed by 4PR and 3PR, respectively. We then present the simulation settings and the utilized mobility model for our experimental performance evaluation in Sections 6.2 and 6.3, respectively. Next, we introduce the routing protocols against which we compare the performance of 4PR and the performance metrics that we use in Sections 6.4 and 6.5, respectively. Finally, we present the results of our experiments in Section 6.6.

6.1. Private Probability vs. Private Maximum

Private probability and private maximum are computed for 4PR and 3PR respectively by the nodes in a community in the background independently of the routing protocols. In this section, we compare the efficiency of these background protocols. We observe that computing private probability, as presented in this paper, is significantly more efficient than computing private maximum, as was proposed previously for the 3PR protocol [7].

In order to compute the value of maximum in a privacy preserving manner in [7], the protocol needs to run $2 + \lambda$ (where $\lambda \geq 7$) rounds of another privacy preserving protocol (named private_sum), which computes the sum of the probability that nodes in a community will encounter a destination node. In each round of the private_sum protocol, kN messages are exchanged among the nodes in a community, where N is the number of nodes in the community, and k is a constant with $2 \leq k < N$.

In comparison, the protocol presented in the previous section in this paper for computing probability in a privacy preserving manner requires only one round of kN messages to be exchanged among the nodes in a community. The order of the size of the messages being similar in the two protocols, the private probability protocol used for 4PR is at least 9 times more efficient than the private maximum protocol used for 3PR in terms of messages exchanged and the bandwidth utilized.

Since the private probability and private maximum protocols are executed in the background, they do not have a direct impact on the performance of the two routing protocols (as evident in the subsequent experimental evaluation).

Table 2: Parameter settings

Parameter Name	**Value**
Simulation area	2000 m × 1500 m
Transmission range	10 m
Simulation duration	13 h + TTL
Warm-up period	1 hour
Message generation rate	1 message per 30 seconds
Number of communities	12
Number of nodes in a community	from 10 to 50
Node speed	1.34 m/s
p_l	0.8
p_r	0.2

However, considering that these background protocols need to be executed regularly, and that private probability is significantly more efficient than private maximum, the 4PR approach has the potential to globally conserve substantial network resources.

6.2. Simulation Settings

We have implemented 4PR as a module of the Opportunistic Network Environment simulator (ONE) [29]. We summarize the simulation parameters that we used in Table 2.

We have used a simulation area of 2000 m × 1500 m. This area is equally divided into twelve regions each measuring 500 m × 500 m. In each region we initially deploy a varying number of nodes (from ten to fifty). Each node considers the region in which it has been deployed as its *local region*. According to the mobility that model we used, further described below, a node is more likely to visit its local region than other places. Nodes associated to a region constitute a community. This simulation scenario is very similar to the one used in PRoPHET [11].

The communication between nodes is performed using the Bluetooth protocol since modern mobile devices are commonly equipped with this technology. According to the specification of Bluetooth version 2.0 [29], the transmission range and bandwidth are set as 10 m and 2 Mb/s, respectively. Furthermore, the speed of nodes is set to 1.34 m/s, since this is an average human walking speed [30]. Each experiment that we run lasts approximately thirteen hours (simulation time). The first hour is a warm up period during which no message is generated. After this period, every thirty seconds, a random node sends a message to a random destination node. We have considered only messages for which the source and the destination belong to different communities.

6.3. Mobility Model

In our evaluation, we adopt the community-based mobility model proposed in [8], which has been widely utilized for the evaluation of community-based routing protocols [31, 9]. In this mobility model, each community is associated with a geographical area. The movement of node i, which belongs to the community C_i consists of a sequence of *local* and *roaming* epochs. A local epoch

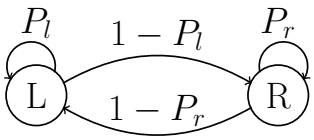


Figure 4: Community-based Mobility Model

is a random direction movement restricted inside the area associated with the community C_i. A roaming epoch is a random direction movement inside the entire network. If the previous epoch of a node i was a local one, the next epoch is a local one with probability p_l, or a roaming epoch with probability $1 - p_l$. Similarly, if the previous epoch of node i was a roaming one, the next epoch is a roaming one with probability p_r, or a local one with probability $1 - p_r$. The state transition between local and roaming epochs is shown in Figure 4. In our simulations, we adopt the same values for p_l and p_r as in [11], i.e., p_l=0.8 and p_r=0.2.

6.4. Routing Protocols

We have compared the performance of 4PR against the following protocols:

Epidemic: in this protocol, a node forwards a copy of each unexpired message it holds to every node it encounters, which does not already have a copy of the message. Epidemic routing achieves the upper bounds of delivery ratio and delivery cost, and achieves the lower bound of delivery latency.

Direct: in this protocol, the source node only forwards the message to the destination node. Contrary to Epidemic, Direct routing achieves the lower bounds of delivery ratio and delivery cost, and achieves the upper bound of delivery latency.

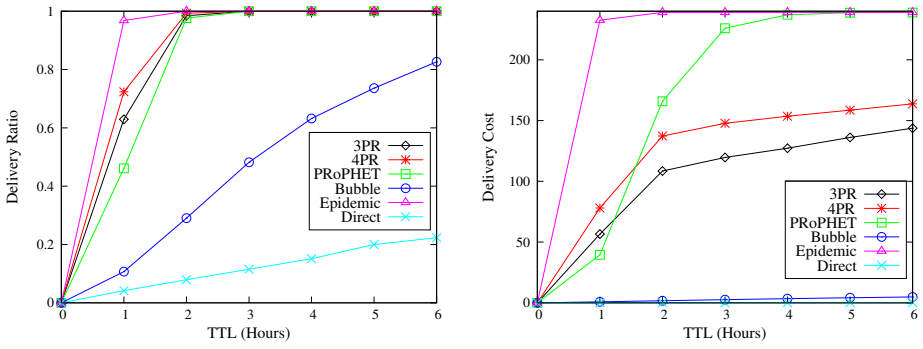
PRoPHET: in this protocol, a node forwards a copy of a message that it holds to a node that it encounters, only if the latter has a higher probability of encountering the destination node of the message. The parameters of the protocol are set as described in [11]. PRoPHET is a well known prediction-based routing protocol.

Bubble: this is a community-based protocol that utilizes social information about nodes, such as their centrality and the community to which they belong.

3PR: in this protocol, the message forwarding decision is made by comparing the maximum probability that a node in the community of a potential intermediate node will encounter the destination node. The parameters of the protocol are set as described in [7].

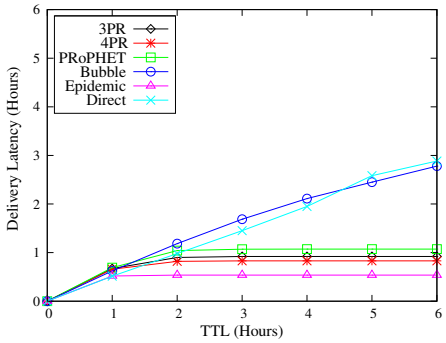
6.5. Performance Metrics

To evaluate 4PR we used three well known metrics: the delivery ratio, the delivery cost and the delivery latency defined as follows.



(a) Delivery Ratio

(b) Delivery Cost



(c) Delivery Latency

Figure 5: (a) delivery ratio, (b) delivery cost, and (c) delivery latency w.r.t. the increasing TTL of messages.

Delivery ratio: is the proportion of messages that have been delivered out of the total unique messages created.

Delivery cost: is the total number of messages transmitted in the simulation. To normalize this, we divide it by the total number of unique messages created.

Delivery latency: is the average time needed to finish transmitting messages to their destinations.

6.6. Performance Results

We performed two experiments. First, we compared the performance of 4PR against the protocols introduced above, with respect to the above three performance metrics. We then analyze the impact of the community size on the performance of 4PR.

6.6.1. Performance Comparison of Routing Protocols

Figure 5a shows the delivery ratio of the compared protocols as a function of the Time-To-Live (TTL) of the generated messages. As expected, Epidemic and Direct achieve the best and worse delivery ratio, respectively, for all values of TTL. We also observe that 4PR achieves a better delivery ratio than PRoPHET and 3PR when the TTL is less then 2 hours, and achieves a similar delivery ratio to that of PRoPHET and 3PR when the TTL is greater than 2 hours. Finally, 4PR has a much higher delivery ratio than Bubble. The difference between the performance of the two protocols rises up to 70.29% for a TTL of 2 hours. This is because 4PR floods a message inside the communities which are on the path from the community of its source node to the community of its destination node.

Figure 5b, shows the delivery cost of the compared routing protocols. As expected, Epidemic and Direct have the highest and lowest delivery cost, respectively, whatever the value of TTL. Compared to other protocols, Bubble has a low delivery cost, which remains stable when the TTL increases. The delivery cost of 4PR is higher than that of Bubble and 3PR, but much lower than that of PRoPHET.

Figure 5c shows the delivery latency of the compared routing protocols. Epidemic has the lowest delivery latency, whatever the TTL. Further, 4PR follows the same trend as Epidemic with higher latencies (around 0.29 hour). 3PR and PRoPHET achieve a little higher delivery latency than 4PR. The performance of Bubble and Direct increases linearly with the increase of the TTL.

6.6.2. Influence of the Number of Nodes in a Community

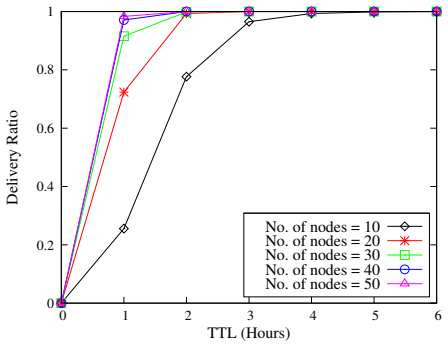
In order to investigate the impact of the number of nodes in each community on the routing performance of our protocol, we run an experiment in which we vary the number of nodes in each community from 10 to 50.

Figure 6a, 6b and 6c show the impact of the increasing community size on the delivery ratio, the delivery cost and the delivery latency, respectively of the 4PR protocol. The results show that the larger the communities, the higher the delivery ratio and cost and the lower the delivery latency. Since 4PR floods a message inside the community of the message carriers, the delivery cost increases as the communities become larger. However, more message copies increase the delivery probability and reduce the delivery latency.

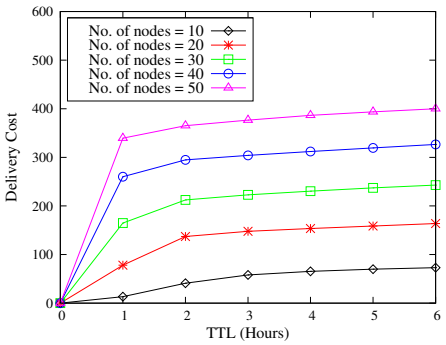
6.6.3. Impact of the Settings of the Mobility Model

In this section, we investigate the impact of the settings of the adopted mobility model on the routing performance of 4PR. We run an experiment in which we vary the value of p_l from 0.5 to 0.9 and set the value of p_r as $1 - p_l$.

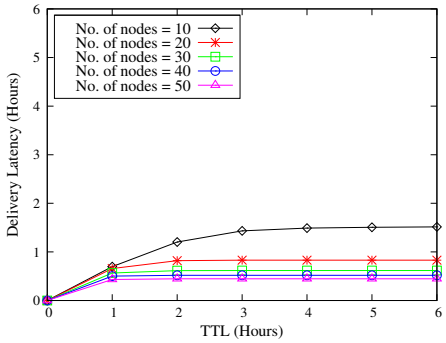
First, we look at the impact of the settings of the adopted mobility model on the delivery ratio. As shown in Figure 7a, we can observe that 4PR achieves similar results with different settings of p_l and p_r. The performance of delivery ratio increases as the increment of the value of p_l when the TTL is not greater than 3 hours. The performance of delivery ratio with different settings is the



(a) Delivery Ratio

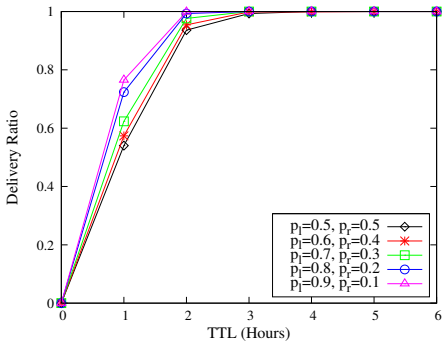


(b) Delivery Cost

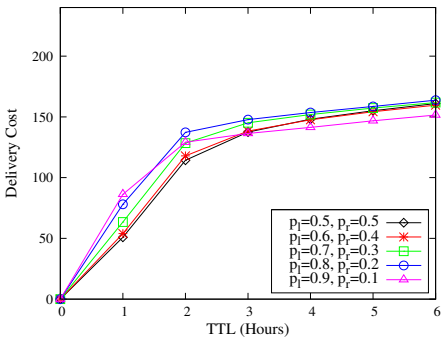


(c) Delivery Latency

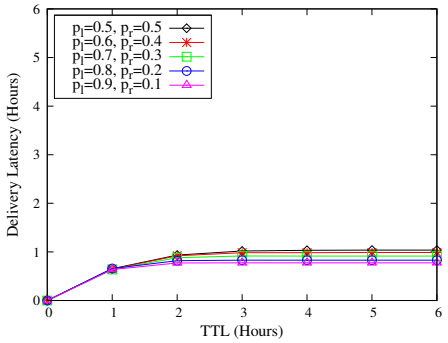
Figure 6: (a) delivery ratio, (b) delivery cost, and (c) delivery latency w.r.t. the increasing size of communities.



(a) Delivery Ratio



(b) Delivery Cost



(c) Delivery Latency

Figure 7: The impact of the settings of the mobility model on the (a) delivery ratio, (b) delivery cost, and (c) delivery latency of 4PR

same, when the TTL is greater than 3 hours. Since 4PR floods messages inside a community, under the pre-condition that messages can be transferred among communities, the higher the probability that a node stays inside its community, the higher probability that the node gets a message flooded inside its community.

Next, we compare the delivery cost of 4PR with different settings of the adopted mobility model. From the results illustrated in Figure 7b, we can observe that the performance of delivery cost increases as the value of p_l increases when the TTL is not greater than 3 hours. When the TTL is greater than 3 hours, the performance of delivery cost decreases as the increment of the value of p_l. This is because that the higher probability that a node stays inside its community, the higher probability that the node gets a message flooded inside its community. In our case, for a given message, most of the nodes on the routing path from the community of its source node to the community of its destination node can get a copy of the message within 3 hours. Therefore, when the TTL is greater than 3 hours, the delivery cost increases slowly for the simulations with high values of p_l. This is consistent with the results of the delivery ratio.

Lastly, we investigate the results of delivery latency of 4PR with different settings of the adopted mobility model. As shown in Figure 7c, we can see that the delivery latency decreases as the increment of p_l. For each setting, the delivery latency increases as the TTL increases, when the TTL is less than 3 hours; the delivery latency stays the same as the TTL increase, when the TTL is greater than 3 hours. For the case that the TTL is less than 3 hours, the messages that need more time can be delivered as the TTL increases. As for the case where the TTL is greater than 3 hours, the latency stays the same, since the messages are delivered within 3 hours. Note that this is consistent with the results of the delivery ratio.

7. Conclusion

This article describes the 4PR protocol, which provides privacy preserving probabilistic prediction-based routing in mobile delay tolerant networks. 4PR is similar to prior prediction-based protocols (e.g., PRoPHET and Bubble), which take advantage of the mobility patterns of nodes to route messages. Our experimental evaluation using a well established community-based mobility model demonstrates that 4PR is comparable to the above noted protocols in terms of performance. Yet, 4PR preserves the privacy of nodes by hiding their individual mobility patterns, whereas the prior protocols do not.

The 4PR protocol is the successor of our 3PR protocol, which to the best of our knowledge, was the first protocol to hide the encounter probabilities of nodes in MDTNs. However, 4PR differs fundamentally from 3PR in how messages are exchanged and the protocols that execute in the background. 4PR's approach gains multiple advantages over 3PR, which include 1) the upper bound of encounter probabilities is not divulged, thus better privacy preservation; 2) private computation of probability requires a single round of computation, whereas private maximum in 3PR required multiple rounds; 3) the probability of at least one node in the community encountering the destination (as in 4PR) is a more

accurate measure for routing path prediction than the maximum probability in the community (as in 3PR).

We foresee three opportunities for future work. First, we would like to reinforce the protocol for preservation of privacy in the malicious adversarial model, where nodes may take disruptive actions such as dropping messages, modifying the protocol, etc. Second, we would like to study the effect of conditions such as network churn and overlapping communities on the protocol. Third, we would like to analyze the energy consumption of privacy preserving routing protocols in MDTNs.

References

[1] K. Fall, A delay-tolerant network architecture for challenged internets, in: Proc. of ACM SIGCOMM, 2003, pp. 27–34.

[2] T. Spyropoulos, T. Turletti, K. Obraczka, Routing in delay-tolerant networks comprising heterogeneous node populations, IEEE Transaction on Mobile Computing 8 (8) (2009) 1132–1147.

[3] Q. Yuan, I. Cardei, J. Wu, An efficient prediction-based routing in disruption-tolerant networks, IEEE Transactions on Parallel and Distributed Systems 23 (1) (2012) 19–31.

[4] Z. Zhang, Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges, IEEE Communications Surveys and Tutorials 8 (1) (2006) 24–37.

[5] J. Miao, O. Hasan, S. Ben Mokhtar, L. Brunie, K. Yim, An investigation on the unwillingness of nodes to participate in mobile delay tolerant network routing, International Journal of Information Management 33 (2) (2013) 252–262.

[6] A. Vahdat, D. Becker, Epidemic routing for partially connected ad hoc networks, Tech. rep., Citeseer (2000).

[7] O. Hasan, J. Miao, S. Ben Mokhtar, L. Brunie, A privacy preserving prediction-based routing protocol for mobile delay tolerant networks, in: Proc. of IEEE AINA, 2013, pp. 546–553.

[8] T. Spyropoulos, K. Psounis, C. Raghavendra, Performance analysis of mobility-assisted routing, in: Proc. of ACM MobiHoc, 2006, pp. 49–60.

[9] H. Dang, H. Wu, Clustering and cluster-based routing protocol for delay-tolerant mobile networks, IEEE Transactions on Wireless Communications 9 (6) (2010) 1874–1881.

[10] T. Spyropoulos, K. Psounis, C. S. Raghavendra, Efficient routing in intermittently connected mobile networks: The single-copy case, IEEE/ACM Transactions on Networking 16 (1) (2008) 63–76.

[11] A. Lindgren, A. Doria, O. Schelén, Probabilistic routing in intermittently connected networks, ACM SIGMOBILE Mobile Computing and Communications Review 7 (3) (2003) 19–20.

[12] P. Hui, J. Crowcroft, E. Yoneki, Bubble rap: Social-based forwarding in delay-tolerant networks, IEEE Transactions on Mobile Computing 10 (11) (2011) 1576–1589.

[13] E. Papapetrou, V. F. Bourgos, A. G. Voyiatzis, Privacy-preserving routing in delay tolerant networks based on bloom filters, in: Proc. of IEEE WoWMoM, 2015, pp. 1–9.

[14] A. Kate, G. Zaverucha, U. Hengartner, Anonymity and security in delay tolerant networks, in: Proc. of IEEE SECCOM, 2007, pp. 504–513.

[15] S. Zakhary, M. Radenkovic, Utilizing social links for location privacy in opportunistic delay-tolerant networks, in: Proc. of IEEE ICC, 2012, pp. 1059–1063.

[16] C. Shi, X. Luo, P. Traynor, M. H. Ammar, E. W. Zegura, Arden: Anonymous networking in dtns, Ad Hoc Networks 10 (6) (2012) 918–930.

[17] M. Reed, P. Syverson, D. Goldschlag, Anonymous connections and onion routing, IEEE Selected Areas in Communications 16 (4) (1998) 482–494.

[18] V. Goyal, O. Pandey, A. Sahai, B. Waters, Attribute-based encryption for fine-grained access control of encrypted data, in: Proc. of ACM CCS, 2006, pp. 89–98.

[19] I. Parris, T. Henderson, Privacy-enhanced social-network routing, Computer Communications 35 (1) (2012) 62–74.

[20] A. Vahdat, D. Becker, et al., Epidemic routing for partially connected ad hoc networks, Tech. rep., CS-200006, Duke University (2000).

[21] M. Liu, Y. Yang, Z. Qin, A survey of routing protocols and simulations in delay-tolerant networks, in: Proc. of WASA, 2011, pp. 243–253.

[22] P. Hui, E. Yoneki, S. Y. Chan, J. Crowcroft, Distributed community detection in delay tolerant networks, in: Proc. of ACM/IEEE MobiArch, 2007, pp. 7:1–7:8.

[23] J. Miao, O. Hasan, S. B. Mokhtar, L. Brunie, G. Gianini, A delay and cost balancing protocol for message routing in mobile delay tolerant networks, Ad Hoc Networks 25 (3) (2015) 430–443.

[24] O. Goldreich, The Foundations of Cryptography - Volume 2, Cambridge University Press, 2004.

[25] G. Kreitz, M. Dam, D. Wikstrom, Practical private information aggregation in large networks, in: Proc. of NordSec, 2010, pp. 89–103.

[26] R. Sheikh, D. K. Mishra, Protocols for getting maximum value for multi-party computations, in: Proc. of IEEE AMS, 2010, pp. 597–600.

[27] O. Hasan, L. Brunie, E. Bertino, Preserving privacy of feedback providers in decentralized reputation systems, Computers & Security 31 (7) (2012) 816 – 826.

[28] O. Hasan, L. Brunie, E. Bertino, N. Shang, A decentralized privacy preserving reputation protocol for the malicious adversarial model, IEEE Transactions on Information Forensics and Security 8(6) (2013) 949–962.

[29] A. Keränen, T. Kärkkäinen, J. Ott, Simulating mobility and dtns with the one, Journal of Communications 5 (2) (2010) 92–105.

[30] M. Kim, D. Kotz, S. Kim, Extracting a mobility model from real user traces, in: Proc. of IEEE INFOCOM, 2006, pp. 1–13.

[31] K. . R. C. S. Spyropoulos, Thrasyvoulos; Psounis, Efficient routing in intermittently connected mobile networks: the multiple-copy case, IEEE/ACM Transactions on Networking 16 (1) (2008) 77–90.