Computer Communications 48 (2014) 30–43

Contents lists available at ScienceDirect

Computer Communications

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

# Privacy-aware message exchanges for HumaNets

Adam J. Aviv [a,*], Matt Blaze [b], Micah Sherr [c], Jonathan M. Smith [b]

[a] United States Naval Academy, Annapolis, MD, United States
[b] University of Pennsylvania, Philadelphia, PA, United States
[c] Georgetown University, Washington, DC, United States

ARTICLE INFO

*Article history:*
Available online 16 April 2014

*Keywords:*
Opportunistic networking
Privacy
Geographic routing
Location privacy
Anonymous communication

ABSTRACT

This paper describes a novel privacy-aware geographic routing protocol for *Human Movement Networks* (HumaNets). HumaNets are fully decentralized opportunistic store-and-forward, delay-tolerant networks composed of smartphone devices. Such networks allow participants to exchange messages *phone-to-phone* and have applications where traditional infrastructure is unavailable (*e.g.*, during a disaster) and in totalitarian states where cellular network monitoring and censorship are employed. Our protocol leverages self-determined *location profiles* of smartphone operators' movements as a predictor of future locations, enabling efficient geographic routing over metropolitan-wide areas. Since these profiles contain sensitive information about participants' *prior movements*, our routing protocol is designed to minimize the exposure of sensitive information during a message exchange. We demonstrate via simulation over both synthetic and real-world trace data that our protocol is highly scalable, leaks little information, and balances privacy and efficiency: messages are approximately 20% more likely to be delivered than similar random walk protocols, and the median latency is comparable to epidemic protocols while requiring an order of magnitude fewer messages.

Published by Elsevier B.V.

## 1. Introduction

The ubiquity of smartphones enable new communication models beyond those provided by cellular carriers. While standard cellular communication uses a centralized infrastructure that is maintained by the service provider, smartphones have communication interfaces such as ad-hoc WiFi and Bluetooth that allow direct communication between devices. Since smartphone owners often carry their devices, leave them on, and encounter other individuals (and their smartphones) in their daily routines, *smartphones enable fully decentralized store-and-forward networks that completely avoid the cellular infrastructure.*

### 1.1. Human movement networks

(HumaNets) [1,2] fit this model and are designed to allow participants to exchange messages phone-to-phone without using any centralized infrastructure. HumaNets' "out-of-band" message passing is applicable when cellular networks are unavailable or if the networks are untrusted (*i.e.*, operated by a totalitarian state that censors [3], shuts down [4], or otherwise leverages its communication systems to restrict its citizenry [5]).

Rather than rely on network addresses, HumaNets route messages using *geocast* – an addressing scheme that directs messages towards a particular geographic region. Such a messaging system could be used, for example, to notify a group of people in a targeted area of an upcoming event, or to warn them of some impending crisis. To cope with mobility, HumaNet routing protocols route messages based on message carriers' predicted *future* locations. This is accomplished by leveraging self-determined *location profiles* that approximate the smartphone owners' routine movements. The patterns of human mobility – for example, the daily commute to and from work – serve as predictors of future locations. HumaNets take advantage of this observation by greedily forwarding messages to smartphones whose owners' location profiles indicate that they are good candidates for delivery.

Privacy issues must be central when designing a HumaNet routing protocol since location profiles contain sensitive information about participants' *prior movements*. The disclosure of such information is particularly dangerous when HumaNets are used for covert communication in totalitarian regimes. Existing decentralized routing approaches that do not consider privacy [6,7], rely on trusted third parties [8], or assume *a priori* trust relationships [9] are also unsuitable for HumaNets.

* Corresponding author. Address: 572M Holloway Road, Stop 9F, Annapolis, MD 21402-5002, United States. Tel.: +1 410 293 6655; fax: +1 410 293 6800.
*E-mail addresses:* aviv@usna.edu (A.J. Aviv), blaze@cis.upenn.edu (M. Blaze), msherr@cs.georgetown.edu (M. Sherr), jms@cis.upenn.edu (J.M. Smith).

http://dx.doi.org/10.1016/j.comcom.2014.03.013
0140-3664/Published by Elsevier B.V.

This paper proposes a novel routing protocol for HumaNets that protects participants' location profiles from an adversary who wishes to learn previous movements and/or determine "important" locations of network users (*e.g.*, home, work, or the location of underground activist meetings). Our technique, which we call *Probabilistic Profile-Based Routing* (PPBR), balances performance and privacy by efficiently routing messages in a manner that minimizes the exposure of users' location profiles. We demonstrate through trace-driven simulations using both real-world and synthetic human movement data that our PPBR protocol is highly scalable, efficiently routes messages, and preserves the privacy of profile information. In summary, the contributions of this paper are:

- The introduction and design of a fully decentralized, privacy-preserving, geographic-based HumaNet message routing protocol for smartphones;
- An analysis of the privacy and security properties offered by our routing protocol;
- A trace-driven simulation study (using both real-world and synthetic data) that evaluates our method's scalability and efficiency.

## 2. Network assumptions and goals

To achieve reasonable performance, HumaNets leverage humans' tendency to follow *routines*: The locations that people frequented in the past are predictors of their future locations [1]. However, a device's location history may be extremely sensitive, and moreover, combining multiple nodes' location histories may allow an adversary to discover social networks and enumerate participants' movements. Hence, the high-level goal of our PPBR protocol and the central challenge of this paper is to enable *efficient geographic-based messaging that limits the exposure of information at message exchanges*. In particular, an adversary who witnesses a message exchange should learn little *important* information about the participants' location histories.

Importantly, however, our HumaNet routing protocol does not conceal the identities of the network's participants. An adversary who intercepts a PPBR message can reasonably conclude that the sender is participating in a HumaNet. Participating in a HumaNet inherently carries risk if used as an anti-censorship technology: This is unfortunately true of any system that may be deemed "subversive". However, when other means of communication are impossible (either due to global monitoring or blocked connectivity), HumaNets provide a *means* to exchange information in a manner that is efficient, scalable, difficult to surveil, and privacy-aware.[1]

### 2.1. Requirements

HumaNets routing protocols are designed for location-aware mobile devices. We assume that network participants can learn their locations (*e.g.*, via GPS[2]) without relying on the cellular service provider's network, and that devices contain sufficient storage to record their movement histories. We note that current generation smartphones meet HumaNets' modest storage and processing requirements.

If GPS is used to determine location, the GPS receiver needs to be activated intermittently and only during regularly scheduled times during which HumaNets messages are exchanged. As recent work notes that GPS reception increases power consumption on smartphones only by approximately 15% [10], we expect the power consumption due to HumaNets to be manageable. Additionally, if any other application on the smartphone requests location information, HumaNets software may use the "last known position" OS feature to determine location with negligible cost. We evaluate the energy costs of our routing scheme in more detail in Section 5.11.

We additionally assume that participants have knowledge of the routing area. Since HumaNets enable geocast routing, a message that is targeted at specific receivers requires the sender to have some knowledge about the receivers' likely future locations (*e.g.*, their home or work); this requirement is similar to that imposed by traditional networking where users need knowledge of a service's hostname or IP address. We also assume that participants know some coarse-grain information about general movement statistics over the routing area. In particular, nodes should be capable of estimating the "popularity" of city areas – *e.g.*, that the upper west side of Manhattan is more densely traveled than Far Rockaway, Queens. This information can be obtained from census data, other public source of information, or personal experience. Such information can be shipped with the HumaNets software and is assumed to be known to an adversary.

### 2.2. Threat model

We envision both passive and active adversaries. A passive adversary may have any number of confederates and is able to observe message exchanges at a fixed number of locations throughout the HumaNet routing area. An active adversary may additionally participate in HumaNets by generating fake messages, accepting messages, and/or dropping or misrouting messages.

We do not provide protection against a *mobile targeting adversary*. An adversary that can physically follow a node can trivially learn about its whereabouts and discover its routine movements. Such a "stalker" adversary is also very costly to deploy. In this paper, we focus on less targeted attackers and assume an adversary who monitors, intercepts, or participates in local exchanges that occur in its presence. The adversary is aware of the participants and their locations at the time of an exchange, and thus we do not claim that our system provides traditional location-privacy [11] for ad hoc networks, although such extensions may be relevant here.

The adversary's goals are as follows:

- Disruption: Inject failures into the network such that messages can no longer be reliably delivered.
- De-anonymization: Determine the originating sender of intercepted messages.
- Profiling: Infer movement patterns of a targeted individual or learn his/her "important" locations (*e.g.*, home, work, underground meeting place).

### 2.3. Performance and security goals

The goal of our routing protocol is to provide the following properties in the presence of active and passive adversaries:

- Reliability: Messages should reach their intended destinations with high probability.
- Efficiency: Messages should reach their intended destinations with reasonable latency and overhead.
- Scalability: HumaNets should be able to scale to a large number of participants with many concurrent messages.
- Point-to-Point: Messages should be exchanged only point-to-point and avoid any centralized routing structures.
- Privacy-Preservation: The protocol should not leak the sender's identity, nor should it reveal information about participants' previous locations. We do not distinguish between locations

[1] It may be possible for users to use steganographic channels to conceal their participation in a HumaNet, although we do not explore such techniques in this paper.

[2] GPS is a unidirectional protocol and requires only the reception of signals from U.S.-operated satellites.

that should or should not remain private (e.g., secret meeting place vs. place of work). The treatment of *all* prior locations as private simplifies our protocol design, and more importantly, improves usability by preventing configuration errors that may lead to accidental exposure of private locations.

At first blush, it may seem that naïve flooding and random walk strategies are sufficient to achieve the above goals. Although these strategies achieve the Point-to-Point and Privacy-Preservation properties, they are lacking with respect to Scalability, Efficiency, and/or Reliability. In particular, flooding achieves optimal latency and delivery rates because all paths are explored, but scales poorly since all transfers that do not occur along the optimal path constitute a wasted effort (and, consequently, wasteful power consumption). Moreover, since several senders may use HumaNets to disseminate their messages, flooding requires that nodes store (and worse, communicate) a large fraction of all messages. At the other extreme, random walk protocols in which messages are transferred (as opposed to copied) upon node contacts scales well but incurs poor Reliability and Efficiency.

It may also seem that traditional cryptographic solutions would be applicable here. However, the decentralized and highly dynamic nature of HumaNets make their deployment difficult. In particular, many cryptographic solutions require centralized services or trusted third parties. Such approaches are problematic in our setting since a strong (*e.g.*, nation-state) adversary could either compromise or prevent access to centralized services. Routing techniques that rely on complex key distribution schemes or expensive cryptographic operations (for example, SMC [12]) are incompatible with HumaNets' distributed architecture and use of power-constrained devices. A significant advantage of PPBR is that it provides Privacy-Preservation using simple probabilistic techniques, and avoids the key management and computation issues present in protocols that provide more traditional cryptographic protections [8,9,13].

Finally, we note that a non-goal of our system is authentication of message senders and message content. PPBR is a content-agnostic service that routes packets, whether they be sent by dissidents trying to organize a rally or a totalitarian state that wishes to provide misinformation. However, as with standard networking protocols, PPBR may be combined with other techniques – for example, the use of pseudoidentities and digital signatures – to provide stronger authenticity guarantees. We remark that such authentication *may* rely on more centralized trust models (for example, reliance on a trusted certificate authority) or may use more decentralized trust systems such as web-of-trust [14]. More generally, any two parties that have earlier exchanged information via an authenticated channel (e.g., by communication public key information in person) can authenticate subsequent messages sent via HumaNets. Since message authentication is not a focus of this paper, we assume that when authentication is required, it is supported by higher-level communication protocols.

## 3. Privacy-preserving routing

At a high level, the *Probabilistic Profile-Based Routing* (PPBR) protocol requires participants (nodes) to *estimate* whether they are good candidates for delivering a message. Upon receiving a message from a *carrier—i.e.*, a node that announces a message—the receiving node makes a local determination as to whether it is well positioned to deliver the message to the addressed destination. The node either *accepts* or *discards* the message, and in either case, *does not notify the current carrier as to its choice*. If the message is accepted, the receiving node becomes a carrier and begins to announce the message. However, unlike flooding techniques in which messages are continuously duplicated, leading to an exponential number of message copies, each message carrier in PPBR announces the message to only $k$ contacts, of which only one out of the $k$ receiving nodes should accept it. The main task is thus for a receiver to locally determine whether it is best suited to deliver the message out of the $k-1$ other nodes that received the message.

### 3.1. HumaNet preliminaries

#### 3.1.1. Addressing

HumaNets provide a basic addressing primitive, *geocast*, in which messages are addressed to a geographic location (*e.g.*, a city square). Messages are routed to nodes who are likely to travel towards the destination address and are then locally flooded within the confines of the specified destination. We do not consider temporal features in addressing or routing – *i.e.*, addressing a message to a location for a specific time – but the protocol described herein can be easily expanded to meet temporal specifications.[3] Additionally, HumaNets do not provide message confidentiality; however, message payloads can be protected using standard encryption techniques.

HumaNets interpret the routing area as a grid, the dimensions of which are assumed to be known *a priori* to all nodes (for example, based on latitude and longitude). Messages are addressed to a particular grid square. In the remainder of the paper, when describing a message address or destination, we refer to the index of the corresponding grid square.

Finally, HumaNets are fully decentralized, delay tolerant networks, and as such, deliver messages according to a "best-effort" policy. Importantly, PPBR does not utilize message delivery acknowledgments; the omission of ACKs and NACKs *increases* privacy since it prevents an observer from trivially discovering whether or not a message was accepted by the receiver.

#### 3.1.2. Message exchanges

Messages are exchanged between smartphone devices when they come into wireless contact with one another. We consider a contact to occur when two nodes are within wireless transmission range, *e.g.*, the range of Bluetooth or a point-to-point 802.11 transmission in ad hoc mode. At set time intervals, nodes awaken and begin the routing protocol. If a contact is made, messages can be exchanged. Otherwise, if there are no other participants nearby, the node returns to normal activity.

HumaNets require coarse time synchronization (*i.e.*, within a few seconds) to ensure message exchanges occur at the appropriate times. Such synchronicity could be achieved using NTP servers, but this would require nodes to send messages over centralized networks. Fortunately, smartphone devices are already highly synchronized as a requirement of participating in the centralized cellular network [15,16] (a network which HumaNets do not use to send messages). If cellular services are disabled or are untrusted to provide correct time information, nodes could alternatively obtain the timing information from GPS satellite timestamps.

### 3.2. Routing overview and constructions

PPBR consists of two phases: a *passing phase* and a *holding phase* (see Fig. 1). In the passing phase, a carrier of a message attempts to pass the message to the first $k$ nodes that it encounters. A node that

[3] One method is for nodes to maintain multiple location profiles, each representing movement information collected at different times of the day. The message exchange algorithm is as described later; however, each node now uses the location profile most relevant to the addressed time and location. With this addition, a message carrier is likely to not only deliver the message to the location, but also to deliver it at the specified time.

receives a message will locally estimate whether it has the highest similarity to the message address (a grid square) out of the $k-1$ other nodes who also received (or will receive) the message. If the node perceives itself to be the best candidate for delivery, it accepts the message, becomes a carrier, and prepares to transition to the passing phase. Otherwise, the message is dropped. A node transitions from the passing phase to the holding phase once it has announced the message to $k$ other neighbors.

The challenge of PPBR is enabling each node to accurately predict whether it is the best of $k$ candidates to accept a message *without conferring with other nodes*. The intuition behind our approach is that a node can compute a *similarity score* to a message's destination using its *location profile* – a compact representation of its movement history. To populate its location profile, a node periodically records its GPS location and determines the fraction of time spent within each grid square. Using its location profile along with background knowledge of the movement patterns of an "average" node, the node can estimate how well it is positioned to deliver the message relative to the $k-1$ other participants who will receive the message.

An important characteristic of PPBR's passing phase is that message reception is not acknowledged. An eavesdropper therefore cannot determine whether a message was accepted or declined by a nearby node. This makes it difficult for an adversary to conduct Profiling attacks against a receiver, since it has no information to form a judgment as to whether the receiver's profile is well-suited for delivering the message. (We explore the effectiveness of Profiling attacks against a carrier who announces a message in Section 6.) To further aggravate Profiling attacks, if a node accepts a message and becomes a carrier, it does not announce the message until it has moved a distance $d$ away from its current location, preventing the eavesdropper from observing the transition.

After a carrier has performed $k$ message announcements, it transitions to the holding phase. In the holding phase, the carrier maintains the message for some time period, during which the node, hopefully, enters the message's addressed grid square and starts the local flood (restricted to the destination grid square). If the node does not reach the addressed grid square within a *local timeout*, the carrier drops the message. A message also has an associated *global timeout* after which all carriers drop the message.

### 3.3. Location profiles

Nodes compute *location profiles* based on their movement histories.[4] Although long term collection could be useful in constructing a profile, HumaNets rely on shorter historical windows to minimize the effects from non-repeated movements, *e.g.*, vacations.

Each node periodically polls its location (*e.g.*, via GPS) to update its location profile. The profile is a matrix indexed by geographic grid square such that the value at position $\langle x,y\rangle$ is the normalized number of location readings in which the node was located at position $\langle x,y\rangle$ in the grid. That is, the value at position $\langle x,y\rangle$ in the location profile corresponds to the frequency that the node visited location $\langle x,y\rangle$ in the physical world over some time window. Following our heuristic, we assume that the matrix value at $\langle x,y\rangle$ (which is defined based on past behavior) approximates the node's future likelihood of visiting location $\langle x,y\rangle$ in the physical topology.

More formally, consider a current window of location entries $W = (\langle x_i,y_i\rangle, \langle x_j,y_j\rangle \ldots)$ that are already mapped to grid square references. The profile $p$, indexed by grid squares, contains the values:

$$p[\langle x,y\rangle] = \begin{cases} \frac{|W_{\langle x,y\rangle}|}{|W|} & \text{if } \langle x,y\rangle \in W \\ 0 & \text{otherwise} \end{cases}, \tag{1}$$

where $W_{\langle x,y\rangle}$ is the sub-list containing location entries occurring within the grid square $\langle x,y\rangle$, $p[\cdot]$ is the index function returning the associated value, and $|\cdot|$ indicates the length of the list.

### 3.4. General node profile

An advantage of PPBR is that it does not require nodes to share their location profiles. However, the technique assumes some globally shared information which we call the *general node profile*. The general node profile is a model of the "average" node's movement, and has the same structure and features as the standard location profile. Rather than representing the frequented locations of a single node, the general profile expresses the patterns of the general population. We assume that the general node profile is included with HumaNet software.

As we demonstrate in Section 5, the general node profile does not have to be a perfect model and can be based on a rough estimate of population densities. In practice, we posit that a sufficient general node profile could be constructed using public data such as population densities from census data, transportation studies [18], or common knowledge.

### 3.5. Marginal similarity

A node determines if it is the best of $k-1$ other message recipients by comparing its similarity with the message's destination to the "average" node's similarity calculated using the general node profile. If the node's similarity is a factor greater, the message is accepted.

More precisely, a node must first be able to calculate the similarity of a location profile to a message address (grid square). This is done by considering not only the value in the profile at the addressed grid-point, but also the values at nearby grid-points, discounted by their square distance. Formally, we define the similarity of a node $n$ to a message $m$ addressed to $a_m$ to be:

$$\mathrm{sim}(p,a_m) = p[a_m] + \sum_{\substack{a_p \in p \\ a_p \neq a_m}} \frac{p[a_p]}{\mathrm{dist}(a_p,a_m)^2}, \tag{2}$$

where $p$ is a location profile and $\mathrm{dist}(a_p,a_m)$ denotes the Euclidean distance between grid-points $a_p$ and $a_m$. This computation captures the desired property that a node that more frequently visits the message's targeted destination (and nearby areas) will have higher similarity than a node that visits the destination region less often.[5]

A similarity score computed with the general node profile, rather than an individual node's profile, represents an estimate of the "average" node's similarity to the message address. We define the relationship between a node $n$'s similarity and that of the general node's similarity as the *marginal similarity* $\sigma$. It is calculated as $\sigma = \frac{\mathrm{sim}(p_n,a_m)}{\mathrm{sim}(p_g,a_m)}$, where $p_n$ is the profile of node $n$ and $p_g$ is the general node profile. The marginal similarity speaks to how well a node is suited to become a carrier of a message addressed to $a_m$ as compared to a node on average: higher values indicate the node would make a good message carrier, while lower values indicate a poor carrier. The next challenge is selecting a threshold value for $\sigma$ at which point only one of the $k$ nodes that received the message will accept it and become a carrier.

[4] News reports suggest that popular smartphones may already collect and store such information [17].

[5] In our simulations, we found that a squared decay function (*i.e.*, the importance of similarity decreases as the square of the distance from the message address) produces good results. We have additionally experimented with other decay functions, and found that they produce similar (but slightly degraded) performance.

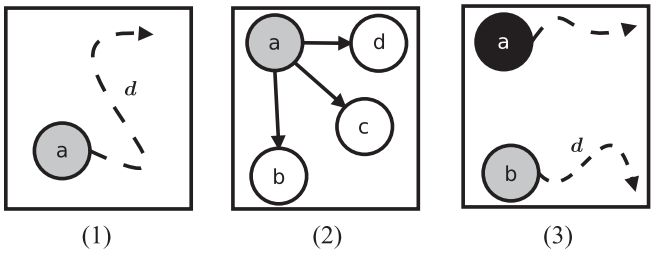


**Fig. 1.** Overview of PPBR routing. (1) The initial message carrier (node $a$) enters the passing phase (grey shading). (2) The carrier encounters three nodes. (3) Node $b$ considers itself the best of $k$ candidates and accepts the message, becoming a carrier and initiating its passing phase. After advertising $k$ messages, node $a$ enters the holding phase (black shading).

### 3.6. Threshold selection

We define $\tau$ as the *threshold marginal similarity score* at which a node accepts a message and becomes a carrier. Intuitively, $\tau$ should be the marginal similarity such that $1/k$ marginal similarity calculations are greater than $\tau$. The threshold is calculated locally (and privately) by each node. First, a node computes $\sigma$ for every grid square in $p_g$:

$$\bar{\sigma} = \left\langle \frac{\mathrm{sim}(p_n, a)}{\mathrm{sim}(p_g, a)} \middle| \forall a \in p_g \right\rangle \tag{3}$$

The computations are arranged in a sorted list $\bar{\sigma}$, where $\bar{\sigma}_i < \bar{\sigma}_j$ if $i < j$. $\bar{\sigma}$ represents marginal similarity calculations for all likely message addresses, and we wish the node to accept a message for $1/k$ of those addresses. To do this, a node chooses $\tau$ such that $1/k$ values in $\bar{\sigma}$ are greater than $\tau$; more precisely, $\tau = \bar{\sigma}_i$ and $i = \lfloor |\bar{\sigma}| * (k-1)/k \rfloor$, where $|\cdot|$ denotes the length function. $\tau$ must be updated whenever the node's location profile changes. To conserve battery, such a computation could occur nightly while the device is charging.

It should be noted that the threshold computation assumes a uniform distribution of message addresses. Although this assumption does not likely hold in practice, our experimental results indicate that our approach is sufficiently accurate to cause approximately $1/k$ messages to be accepted by potential carriers. In particular, using our tested datasets (see Section 5.1) in which messages are addressed non-uniformly, between 8.5%–9.5% of messages are accepted.

### 3.7. PPBR: Summary

In summary, PPBR supports geocast messaging in which messages are addressed to a particular grid square and intended for all participants residing therein. A message carrying node (a carrier) in the passing phase will duplicate the message to $k$ other nodes before transitioning to the holding phase. Of the $k$ nodes that receive a message, $k-1$ should drop the message while a single node should retain it. This process is oblivious to the message sender (and an adversary) who is unaware of which of the nodes accepted the message and which dropped it. To determine if a node is a good carrier (*i.e.*, the best of $k$), a receiving node computes their marginal similarity $\sigma$, which compares their similarity to that of the general node's, as embodied by the general node profile. If $\sigma$ is greater than their locally calculated threshold $\tau$, the message is accepted, otherwise it is rejected. Nodes that accept a message will transition to a passing phase after traveling a distance $d$ from the point of reception, where they repeat the process by exchanging the message with $k$ other nodes. At any point, the message may reach the addressed grid square, within which, the message is flooded to all participants present. Additionally, if a node does not deliver a message within a local timeout, the message is dropped. After a global timeout occurs, all message copies in the network are discarded.

## 4. Comparison to other HumaNet routing techniques

In this section, we compare the PPBR protocol to previously proposed HumaNet routing techniques (Section 4.1), Strawman HumaNet routing protocols (Section 4.2), and cryptographic techniques based on secure two-party computation (Section 4.3).

### 4.1. HumaNet routing with polygon-based location profiles

To efficiently deliver messages, HumaNet routing protocols must accurately predict participants' *future* locations. Since HumaNets are comprised of wearable communication devices (*i.e.*, smartphones), nodes' future locations can be predicted by inferring the likely future locations of their human operators.

In previous work [1,2], we explored the "return-to-home principle": the tendency of nodes (*i.e.*, humans) to return to the places that they have traveled to in the recent past (for example, their home or workplace). Using cluster-based location profiling [1] and three mobility traces [19–21], we showed that (perhaps unsurprisingly) people's past locations are good indicators of their future locations. Independent of our work, other investigators have shown similar movement patterns using cellular telephone call data records [22–24].

The return-to-home principle enables HumaNet routing by allowing messages to be forwarded towards their intended receivers' *likely* future locations. In PPBR, location profiles indicate the probability—based on collected movement data—that a node will return to a particular grid square. However, while the return-to-home principle permits more efficient routing, it also poses a potential privacy risk: exposing a location profile leaks sensitive information, not only about the node's previous locations, but also about its likely future locations. As discussed in greater detail in Section 6, PPBR attempts to minimize these privacy risks.

In previous work [2], we described using location profiles that are generated by computing polygons over clusters of recorded location points [1]. The grid-based PPBR technique described in this paper offers significantly better privacy protections. First, grids are regimented and shared across all participants. Determining that a grid-square is important to a node may provide geographic-anonymity (or spatial cloaking [11]) since multiple users may be "similar" to the same grid-square. Further, the uniformity of grid-squares also benefits privacy. Unlike polygon-based profiles that can conform to shapes based on the density of recorded location points in an area, grid areas have a fixed size and shape. Polygon profiles are acute, and if revealed, direct an attacker to a precise location of importance; however, the uniformity of a grid only provides an adversary with a large general area, which may still only provide partial information. Finally, as described in the previous section, PPBR uses probabilistic techniques and does not directly expose nodes' location profiles.

### 4.2. Similarity HumaNet routing

Depicted in Fig. 2, *Similarity Routing* uses similarity scores to find good candidates to deliver a message; when a carrier encounters another HumaNet participant, it transfers the message to the other node if that other node has a greater similarity score to the message's intended destination.

More formally, consider a node $n$ that is carrying a message addressed to destination $a_m$. When $n$ encounters another node $n'$, both nodes calculate and announce their similarity score to the

message's intended destination, $a_m$. Node $n$ transfers the message to $n'$ iff $\mathsf{sim}(p_n, a_m) < \mathsf{sim}(p_{n'}, a_m)$. Importantly, like PPBR, similarity routing does not duplicate messages. Like standard IP messaging, once a node has passed a message to the new carrier, it deletes its local copy. Similarity routing uses the addressing scheme describe in the previous section: messages are considered delivered if they reach the addressed grid square.

Similarity routing is a simpler HumaNet protocol than PPBR, but it is vulnerable to Profiling attacks. A passive adversary who observes a message exchange learns the precise similarity scores of the communicating nodes. Since the similarity score is derived from location information in a node's location profile (Eq. 2), the adversary can infer the areas frequented (or not frequented) by the victim node, and after repeated observations, the adversary can perform more fine-grained inferences.

Similarity routing is also particularly vulnerable to active attacks. Here, the adversary chooses locations of interest, crafts a message addressed to those locations, and attempts to exchange them with a targeted node. The attacker learns the similarity score of the targeted node *for any target location*, and again, the adversary may repeat the process to further its knowledge of the victim's location profile.

PPBR mitigates these Profiling attacks by removing the announcement of similarity scores (and therefore preventing an adversary from probing nodes' location profiles) and relying instead on probabilistic inferences.

### 4.3. Private similarity exchanges via secure two-party computation

The above Profiling attack could be partially defeated using secure two-party computation (2PC), a cryptographic technique that enables two parties to jointly compute a function over their private inputs without revealing anything about those inputs. In particular, 2PC could be applied at stage (2) of Fig. 2, removing the public announcement of similarity scores. With 2PC, this comparison can be done in private way, reducing to the *Millionaire's Problem* [12] whose solution requires a 2-party symmetric secure function (SSFE) because either party may be dishonest. Unfortunately, the complexity of SSFE has been shown to require a constant number of oblivious transfers [25]—at least one oblivious transfer for each bit of the input. The communication overhead therefore makes SSFE protocols infeasible for battery-constrained smartphone devices in highly mobile settings where contact periods may be brief.

Additionally, the 2PC solution leaks information and is vulnerable to active Profiling in two ways: An attacker that can observer a message tranfer, even if it cannot determine the receiver's similarity to the message address, still learns that the receiver is *more* similar to the message address than the previous message holder. Second, the 2PC solution does not restrict the attacker from participating in the system. An active attacker can perform 2PC similarity comparison using a bogus message address, atempting to get a targeted receiver to indicate a willingness to accept the message (and hence revealing information about its location profile). This process can be repeated with different bogus similarity scores and message addresses, effectively allowing the attacker to "hone in" on the victim's similarity to targeted locations.

PPBR avoids such information leakage by having nodes probabilistically accept messages rather than rely on explicit transfers; with PPBR, an adversary cannot definitively determine whether the encountered node accepts or drops the carrier's message.

## 5. PPBR: performance evaluation

To evaluate the performance of PPBR, we constructed a discrete event-driven HumaNets simulator. Our simulator takes as input a trace of human (cellphone) movement and overlays the PPBR routing algorithm. In all simulations, we choose $k$ to be 10 and conduct 300 independent runs. Message senders are selected randomly across participants, and message addresses (grid squares) are randomly chosen by selecting a (different) node and addressing the message to its most frequented grid square as defined by its location profile. Our simulation was concerned with measuring the effectiveness of PPBR over metropolitan areas, and as such, we did not simulate local flooding. We considered a message successfully delivered if it reaches the destination address. The grid overlay consists of 200 m × 200 m grid squares, roughly the size of a city block, and we chose $d$—the requisite travel distance of a node before transitioning to the passing phase—to be the size of a grid square (200 m).

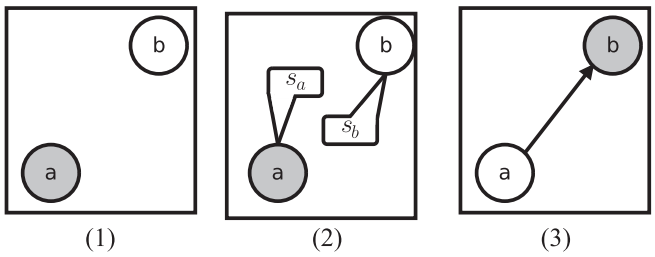


**Fig. 2.** Similarity Routing. (1) A message-carrying node $a$ encounters a node $b$. (2) Both nodes compute and announce their respective similarity scores to the message's intended destination address. If node $b$ is more similar (3), the message is transferred from $a$ to $b$ without duplication, and $b$ is now the carrier of the message.

### 5.1. Simulation settings and inputs

#### 5.1.1. Datasets

Due to privacy constraints, the number of realistic datasets that are suited for evaluation is unfortunately small. We require that the data contain not only a large number of nodes, but also that the movement of the nodes should express regular routines over an extended collection time (*i.e.*, many days). There is considerable work in constructing models for human movement [26–31]; however, most of these models do not realistically simulate movement over long periods, nor do they model regularity. There also exists extensive catalogs of real world movement traces, such as the CRAWDAD repository [32]; unfortunately, most of the traces are either too short with too few nodes or do not contain fine-grained location information.

To demonstrate the feasibility of PPBR, we utilize a suitable real-world data trace as well as a synthetic trace of human movement (summarized in Table 1):

- **Cabspotting:** The **Cabspotting Dataset** [20] contains GPS coordinates and timestamps of 536 taxicabs in the San Francisco area. The dataset spans 20 days: from May 20, 2008 until June 7, 2008. It should be noted that although the movements of taxis are not representative of the general population (taxis are arguably more mobile than the average person), simulations using this dataset can be interpreted as representing a network composed of the taxi drivers' smartphones.
- **SLAW:** We require a synthetic model that (i) accurately represents human *flight patterns*, (ii) contact rates, (iii) *waypoints* (popular places), and (iv) routines. The closest model to meeting our needs is **Self-similar Least Action Walk** (SLAW) [30]. Based in part on Levy walks [33], SLAW introduces a protocol called *Least Action Trip Planning* (LATP) that produces human-like trips between fractal waypoints, that are themselves determined by

**Table 1**
Characteristics of the movement data sets.

| | Nodes | Length | Area | Contact rate | Waypoints |
|---|---|---|---|---|---|
| SLAW [30] | 1000 | 7 days | 100 km$^2$ | 12.62 per hour | 150 |
| Cabspotting [20] | 536 | 20 days | 326 km$^2$ | 1.17 per hour | n/a |

finding hotspots in actual GPS traces. Lee et al. showed that SLAW produces more human-like inter-contact times and flight paths than other leading movement models [27,31,34].

*5.1.2. Node contacts*

For two nodes to make contact, they must be in the same location at the same time. However, the periodicity of location entries in the Cabspotting dataset is not consistent across nodes (or for the same node). We consider two nodes to have made contact if they are within 10 m in a 10 s window. In SLAW, a location entry is generated every 60 s consistently across all nodes; we consider a contact to occur if two nodes are within 10 m at the same minute mark.

*5.1.3. Timeouts*

We use a 12 h local timeout with both traces. For the shorter, more dense SLAW movement trace, a three day global timeout is used. The longer, more sparse Cabspotting trace uses a seven day global timeout. Finally, simulations begin after an initial delay so that node profiles can be well seeded; delays of three and seven days are used for SLAW and Cabspotting, respectively. We explore the tradeoffs of using different timeout values in Section 5.8.

*5.1.4. Location profiles*

Each node constructs its location profile using a three day window of location histories. Location profiles are updated daily, and the current day's profile represents the location history of the three previous days.

To generate the general node profile, we select a 10% sample of nodes from each dataset and use three days worth of movement data. The 10% sample is excluded from all simulation experiments. Visualizations of the resulting general node profiles are shown in Figs. 3 and 4.

*5.2. Simulation results*

To measure the efficiency of PPBR, we compare our strategy against two probabilistic protocols that do not use location information: *probabilistic random walk* and *probabilistic flooding*. The probabilistic random walk routing scheme also has passing and holding phases; however, unlike PPBR, the random walk does not use location profiles. Instead, a node accepts a carrier's advertised message with a fixed probability of $1/k$ (*i.e.*, 10%). The random walk protocol allows us to measure both the effectiveness of using location information as well as the local threshold selection process.

Additionally, we compare PPBR to a 10% probabilistic flood in which nodes *duplicate* the message to a contacted node with probability 0.1 with PPBR's passing and holding phases. The flood provides insight into a worst case for network load – *i.e.*, exponential growth in the number of duplicate messages. The global and local timeouts for both random protocols are identical to those used by PPBR.

*5.3. Threshold estimation*

As described in Section 3.2, each node computes its threshold marginal similarity score ($\tau$) based on the general node profile and its knowledge of the routing area. Ideally, $\tau$ should be chosen such that a message is transferred to exactly one of the $k$ nodes that a carrier encounters during its passing phase. To determine if our local, per-node threshold calculations were generating good thresholds, we looked at the variance of thresholds calculated at each node for one day in the simulation. Intuitively, a low variance indicates that nodes are independently able to reach a consensus as to a good value for $\tau$, without exchanging any information amongst themselves. The average value for $\tau$ was 1.783 and 1.557 for SLAW and Cabspotting, respectively. We found that there

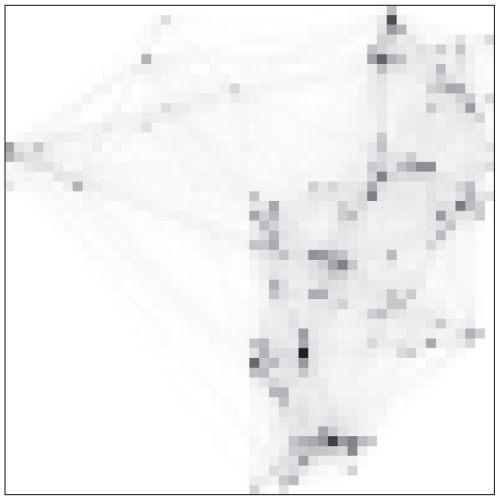

**Fig. 3.** Heatmap of the General Node Profiles for the SLAW dataset. Darker shades indicate regions with higher node densities.

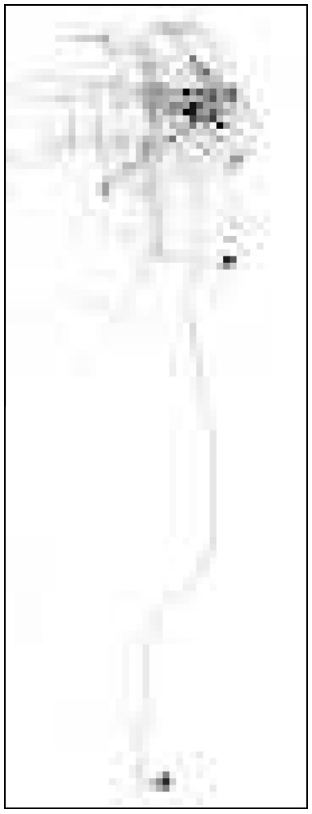

**Fig. 4.** Heatmap of the General Node Profiles for the Cabspotting dataset. Darker shades indicate regions with higher node densities.

is low variance among the nodes' thresholds: 0.274 for SLAW and 0.085 for Cabspotting. Further, we observed that thresholds were effectively limiting message acceptance to $1/k$; with $k = 10$ the probability of message retention was 10% and 9.8% for SLAW and Cabspotting, respectively.

### 5.4. Performance metrics

We evaluate our routing performance using the following metrics: *delivery rate* is the percentage of messages that reach the destination address (a grid square); *latency* is the amount of time it takes for a message to be delivered; and *network load* is the number of messages in the network at a given time. Ideally, the routing protocol should deliver messages with a high delivery rate, low latency, and low network load.

### 5.5. Delivery rate and latency

Table 2 lists the delivery rates and latencies for PPBR, random walk, and probabilistic flooding. Unsurprisingly, flooding offers both the best latency and delivery rates. (As we show later, it also incurs a very high network load, making it impractical for networks of battery-constrained smartphone devices.) PPBR routing outperforms random walk for both median and average latency in all settings, and PPBR laso has faster quartile marks, particularly, at the third quartile. The delivery rate for PPBR is more similar to flooding than random walk, but with much lower load, as indicated in the next section. It should also be noted that the delivery rates reported in Table 2 result from single attempted transmissions. The sender can increase the delivery rate by sending redundant copies sufficiently spaced in time to allow different sets of carriers to deliver the message.

### 5.6. Network load

The load on the network is measured as the average number of message duplicates in the system across all simulations runs. PPBR does not guarantee that only a single copy of a given message is present in the system. Carriers announce a message to $k$ other nodes; ideally, only one node *should* accept it. If the message is accepted, the carrier retains the message until either it is delivered or a local timeout occurs. Hence, each message could potentially have multiple (or zero) duplicates.

Fig. 5 plots the number of messages that persist in the system over time, normalized to the number of senders in the system (which, in our simulation experiments is always 300). The average number of message copies, computed over the entire simulation, is shown in the Figure's key. Note that the number of message duplicates may be less than one if either some messages are not accepted by any of the $k$ encountered nodes, or if all message copies are delivered to their destinations. As expected, flooding incurs significant network load, resulting in approximately two orders of magnitude more message copies than PPBR. Although the number of duplicates is slightly larger for PPBR than our naïve random walk protocol, the load is easily manageable.

**Table 2**
Median and Average Latencies (first and third quartiles in braces) and Delivery Rate.

| | Cabspotting | | SLAW | |
|---|---|---|---|---|
| | Med/Avg latency (hrs) | Rate | Med/Avg latency (hrs) | Rate |
| PPBR | 3.75/5.28 [1.46,6.17] | 81.0% | 4.65/5.34 [2.9,6.8] | 77.7% |
| Walk-10% | 4.39/5.82 [1.51,7.51] | 70.0% | 6.17/6.63 [3.5,9.2] | 65.3% |
| Flood-10% | 3.98/5.60 [2.33,6.13] | 87.7% | 4.07/4.16 [3.0,5.4] | 99.3% |

### 5.7. Storage overhead

We also evaluate the storage overhead of each smartphone that participates in HumaNets. Storage costs are incurred whenever a node stores a received message, and are relieved whenever a node drops a message. Fig. 6 plots the average number of messages stored on each device for the PPBR, random walk, and probabilistic flooding protocols. Unsurprisingly, flooding requires the most storage overhead, as messages are often duplicated during node encounters. For the SLAW experiment, the average number of messages per node exceeds 230 for probabilistic flooding (recall that 300 messages are transmitted in each simulation).

The storage overhead is significantly more modest for both random walk and PPBR. For both the Cabspotting and SLAW experiments, the average number of stored messages for nodes participating in HumaNets is less than five.

### 5.8. Tuning PPBR

PPBR contains several parameters which influence its performance and security. Below, we explore some of the tradeoffs of varying these parameters.

### 5.9. Distance traveled before announcing

Recall that as a means to mitigate Profiling attacks, nodes that accept a message move a distance $d$ before re-announcing the message (see Section 3.2). This improves security since, otherwise, a stationary observer would be able to tell who accepted a message by observing a node transition from a receiving state to an announcing state. In the previous results, we selected $d$ to be 200 m, or the size of a grid square.

To measure its effect on performance, we conducted a number of simulations using different values of $d$. We list the latencies and delivery rates that result from using different values of $d$ in Table 3.

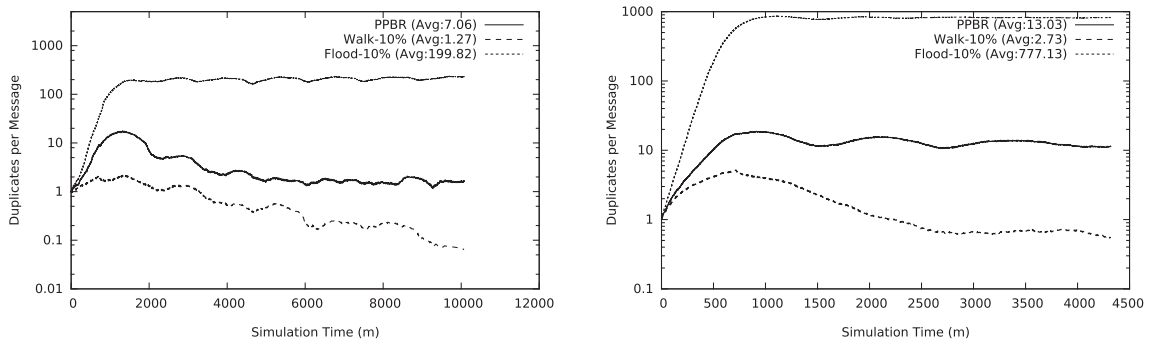


**Fig. 5.** The average number of message copies ("duplicates") of each message for (***left***) Cabspotting and (***right***) SLAW, and inset, the average.

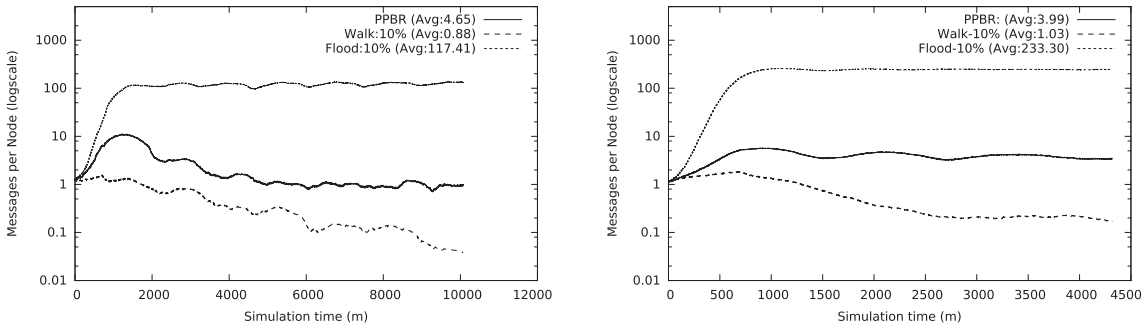


**Fig. 6.** The average number of messages stored on each node during the simulation of 300 initial messages: (***left***) Cabspotting and (***right***) SLAW, and inset, the average.

The Table shows that there is a slight advantage in terms of latency for immediately transitioning into the announcing state once a message is received. This is perhaps unsurprising, since moving a distance $d$ inherently requires some time and therefore incurs latency. However, beyond 50 m, the increase in latency is small, particularly for the Cabspotting dataset. Larger values of $d$ result in slightly longer latencies. Overall, the choice of $d$ has only a minimal effect on the delivery rate.

In summary, the choice of $d$ does not significantly affect either the latency or delivery rate. A reasonable choice of $d$ considers susceptibility to the Profiling attack while not requiring the message carrier to travel too far a distance. Our choice of 200 m seems a good tradeoff in this space.

### 5.10. Local timeouts

We also investigate the effects of the local timeout on network performance. Table 4 reports the median latencies and delivery rates for the two datasets for different local timeouts. Intuitively, increasing the local timeout also increases the delivery rate since nodes hold on to messages longer, and are thus more likely to deliver it. However, these additional deliveries occur later in the simulation, resulting in slightly increased median latencies. Although we believe our default of 12 h achieves a reasonable balance between latency and delivery rate, we remark that PPBR may be trivially extended to allow the sender to specify a requested timeout and therefore better control this tradeoff.

### 5.11. Energy costs

As described in Section 3.1, HumaNets-enabled smartphones exchange messages in synchronous *rounds* when they are within transmission range. Here, a clear tradeoff exists between energy costs and message propagation speeds: more frequent polling (i.e., smaller round intervals) leads to more communication between peers, increasing the likelihood that messages are propagated between devices. However fast polling consumes more power and may too quickly deplete smartphone batteries.

We conduct an empirical experiment to study two "knobs" that can be tuned to balance battery usage with effective messaging: the interval between synchronous communication sessions and the duration of each session ("transfer time"). Fig. 7 graphs battery charge (as a percentage) over time with different polling and transfer periods on an HTC Android G1 running Cupcake 1.5 and the modified CyanogenMod v4.0.2 kernel [35]. GPS location information was collected during the same transfer time block to simulate the cost of geographic tracking. That is, the experiment assumes a worst-case scenario in which a transfer must occur after every profile announcement stage. Additionally, our experiment assumes that data are transferred for the duration of the transfer time.

We observe that even with fairly frequent (5 min) polling frequencies and large (15 s) transfer times, the smartphone battery lasted nearly 24 h – sufficient time to allow a day's use of the phone (provided the owner recharged the smartphone nightly). With slightly faster transfer times (5 s), the cost of participating in synchronous message exchanges is minimal: the smartphone's battery was more than half charged even after 60 h.

## 6. Security properties

### 6.1. Profiling

All message exchanges in PPBR occur in the open, and an adversary can observe any exchange in its presence. However, PPBR offers strong privacy protections against Profiling attacks for both the node announcing a message as well as the node who receives, and possibly accepts, the message announcement.

### 6.2. Message exchange carrier protections

An adversary can determine that a carrier node who advertises a message has a high marginal similarity to the message's address; otherwise, the node would not be advertising the message. More precisely, the adversary knows that the marginal similarity for the carrier is lower bounded by the threshold $\tau$.

**Table 3**
Median latencies and delivery rates for different values of $d$.

| $d$ (meters) | Cabspotting | | SLAW | |
|---|---|---|---|---|
| | Latency (hours) | Delivery rate (%) | Latency (hours) | Delivery rate (%) |
| 0 | 3.06 | 77.7 | 4.23 | 75.7 |
| 50 | 3.53 | 80.7 | 4.54 | 77.3 |
| 100 | 3.75 | 81.0 | 4.57 | 77.0 |
| 200 | 3.75 | 81.0 | 4.65 | 77.7 |
| 400 | 3.75 | 81.0 | 4.72 | 77.7 |
| 800 | 3.74 | 80.7 | 4.82 | 77.7 |
| 1600 | 3.81 | 81.3 | 5.18 | 78.0 |
| 3200 | 3.81 | 81.3 | 5.78 | 77.0 |

**Table 4**
Median latencies and delivery rates for varying local timeouts.

| Local timeout (hours) | **Cabspotting** | | **SLAW** | |
|---|---|---|---|---|
| | Latency (hours) | Delivery rate (%) | Latency (hours) | Delivery rate (%) |
| 1 | 3.80 | 75.7 | 5.45 | 52.3 |
| 2 | 3.80 | 75.7 | 5.26 | 52.3 |
| 6 | 3.75 | 76.0 | 4.57 | 63.7 |
| 12 | 3.75 | 81.0 | 5.34 | 77.3 |
| 24 | 3.96 | 83.7 | 6.24 | 88.3 |
| 48 | 3.97 | 84.7 | 6.41 | 91.3 |

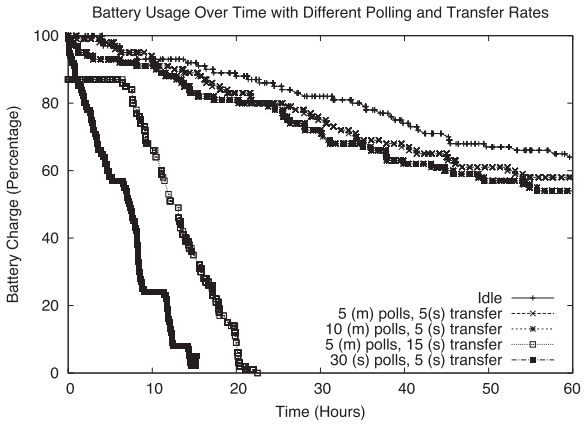


**Fig. 7.** Battery charge over time for different polling frequencies and transfer times.

By design, nodes choose $\tau$ such that they should expect to accept messages addressed to $1/k$ of the grid squares. Hence, *the acceptance of a message does not necessarily indicate that the message's address is particularly important to the node that accepted it*. Depending upon the value of $k$, a node may be expected to accept messages targeted at hundreds of grid squares across the routing area. An adversary cannot conclude that a message was accepted because the message's address is frequently visited by the advertising node. Moreover, as we show below, a node may not even accept a message addressed to a grid square for which it is very familiar.

The choice of $k$ has privacy and performance implications, and a clear tradeoff exists: Larger values of $k$ decrease privacy since nodes accept messages for fewer locations, and thus an adversary could deduce that these locations are more likely relevant to the victim node. Conversely, smaller values of $k$ increase privacy since nodes accept messages to more locations, further obscuring which are important. Smaller values of $k$ also incur higher power consumption and network load as more nodes will likely accept (and transfer) the message. In our simulation studies, we found that $k = 10$ achieves reasonable privacy while restraining the number of message transfers.

To study this tradeoff further, we determined for each node the set of addresses (grid squares) that would result in its acceptance of a message. We then compared this set of addresses to the nodes' most frequented locations as defined in their location profiles. As expected, nodes accepted messages addressed to $1/k$ of the grid squares, on average. However, many of those locations correspond to grid squares that would be uninteresting to an adversary concerned with Profiling. If we consider an adversary who is interested in the most frequented grid squares of a victim node – that is, the highest value grid squares in the node's location profile – these grid squares comprise only a small fraction of the total locations for which a node would accept a message.

This relationship is depicted in Fig. 8 (left). The curves represent the averages across all nodes in the Cabspotting and SLAW datasets. The x-axis denotes the number of points an adversary is interested in (*i.e.*, the $x$ grid squares most frequented by the node). The y-axis plots the fraction of the locations that are accepted by the node which are of interest to the adversary. For example, using the Cabspotting dataset, 38% of announced messages belong to the advertising node's 800 most frequented locations. If the adversary is interested in a node's 200 most frequented grid squares, just 10% of advertised messages belong to this interest set. More generally, the more specific the adversary's interest, the more difficult it is for him to distinguish the pertinent message addresses that are announced by a node, and consequently, the more difficult it is to discover the node's most frequented locations.

The adversary's ability to discern profile information is further diminished due to our algorithm's willingness to discard announcements that are targeted at highly frequented areas. That is, a significant portion of the grid squares most frequented by a node may have low marginal similarity. Recall that the marginal similarity is the ratio of the node's similarity score to the general node profile's similarity score. Hence, if a message is addressed to a grid square that is often frequented by the node *but also highly frequented according to the general node profile*, then the ratio will not exceed the $\tau$ threshold, and the node will *never* accept a message addressed there. Consequently, such interesting locations are unobservable and *safe* from adversarial analysis.

Fig. 8 (right) visualizes this relationship. Again, the $x$-axis considers the number of grid squares an adversary would find interesting for a victim node. The $y$-axis represents the fraction of those interesting grid squares a node would *never* accept a message for, averaged across all nodes. For example, consider an adversary interested in the top 200 most frequent locations of a node: In the Cabspotting data set, 68% of those locations are safe from analysis by an adversary.

### 6.3. Message exchange receiver protections

During the passing phase, receivers do not acknowledge acceptance (or rejection) of a message, and hence an adversary cannot directly determine its similarity to the message's destination address.

An adversary who is able to follow the node for a distance of at least $d$ can determine whether the message has been accepted by observing whether or not it is re-advertised by the node. However, since the node is physically followed, such a stalking attack inherently leaks the victim's location information regardless of the particular routing protocol being used (and hence, as described in Section 2, stalking attacks are outside of our threat model). Regardless, if the node *is* followed, or if a separate colluding eavesdropper discovers that the node later advertised the message, then the adversary can conclude that the node accepted the message. In such cases, the effectiveness of a Profiling attack against the receiver is identical to the effectiveness against a carrier advertising a message (see above).

### 6.4. De-anonymization

The standard addressing primitive of HumaNets is geocast, and thus all participants at the addressed location at the time of

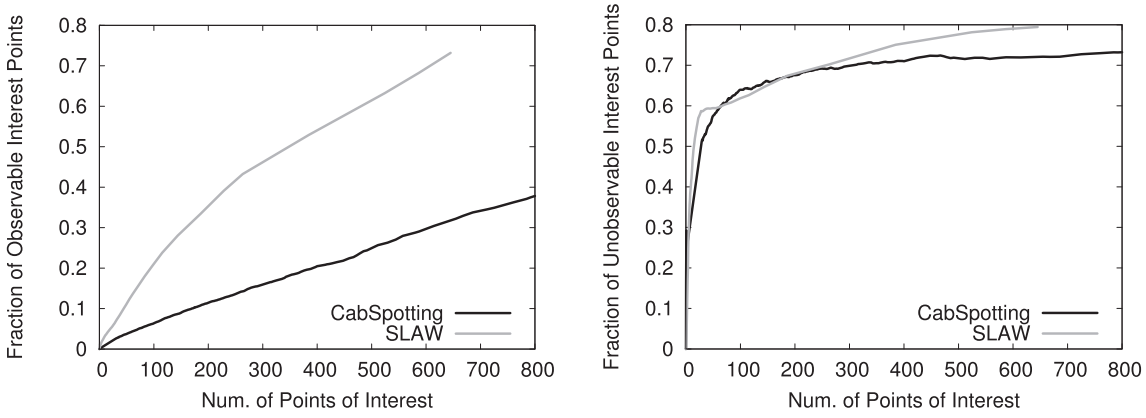


**Fig. 8.** Fraction of Safe Interest Points (***left***) and Fraction of Interesting Observations (***right***).

delivery should receive the message. Receiver anonymity is not protected in HumaNets because an adversary located in the address location trivially learns the identities of the message recipients by simply observing them.

However, PPBR provides in-transit anonymity for message originators (or senders). An intercepted message, past the initial hop, cannot be traced to the original sender without completely retracing the message's path. If an adversary is witness to the initial hop of a message, the originating sender may be exposed. We note, however, that this is similar to the level of protection provided by many Internet-based anonymity systems (*e.g.*, Crowds [36]) in which an adversary on the first hop may infer with some probability that it has identified the sender (since the sender may have originated upstream). It is also worth noting that message replay attacks in which an attacker re-injects a message in hopes of discovering its path are also infeasible. It is highly unlikely a message will take the same path due to variability in human movement.

### 6.5. Disruption

PPBR also provides protection against Disruption attacks in which an adversary attempts to intercept messages in the network. If the attacker is able to infiltrate the network and receive a large portion of the $k$ handoffs for each message, then the probability that the message will be transferred to an honest node is reduced. However, such an attack may also be prohibitively expensive for an adversary since message exchanges occur whenever two participants have a chance encounter. Additionally, such an attack may be mitigated by adjusting the number of passing attempts (*i.e.*, $k$) to compensate for the attacker's presence.

PPBR's Scalability property also makes it resistant to denial-of-service attacks in which the attacker attempts to overwhelm the network's resources by injecting spurious messages. Although an attacker may inject wasteful messages into the HumaNet, the impact of each additional message on the network is linear, by design. In comparison, each additional message in a flooding protocol incurs an exponential increase in network load, and a few injected messages may be sufficient to overload the network.

## 7. Related work

### 7.1. Location-based routing

The ability to leverage geographic information to efficiently route packets has been well explored in the literature. In many instances, these techniques require participants to announce their locations. For example, Last Encounter Routing (LER) [6] and ProPHET [37] expose location information; LER assumes that the network is sufficiently connected to allow stable and longstanding paths. The Bubble protocol [38] uses social networks to efficiently route messages, but allows any party to discover social relationships. Although these techniques may efficiently route messages, they are not well-suited for settings in which the disclosure of location histories and/or social relationships may be cause for government-imposed punishment. We desire protocols that efficiently and scalably deliver messages while preserving users' location histories and social relationships.

Location-based routing has also been studied in the context of wearable computing. Of particular relevance is Davis et al.'s geographic-based routing protocol [39]. There, the authors use flooding techniques to disseminate messages when the network's devices are storage constrained; they consider a pruning approach in which nodes drop messages that are addressed to locations that they have not recently visited. Our routing techniques rely on similar heuristics, but take a more proactive approach by targeting potential message carriers who are *likely* to visit a message's destination. Similarly, *pocket-switched networks* [7,40,41] provide methods of routing messages between pocket-sized devices. However, the protocols are intended for small area routing (*i.e.*, at the scale of an academic conference) and focus on reliability. Our protocols are designed specifically for smartphones, leverage the devices' ubiquity and location-awareness, and target city-scale routing.

### 7.2. Location privacy

There are a number of approaches that attempt to preserve *location privacy*. Here, the goal is often to prevent an adversary from either identifying the source of an intercepted communication or tracking a node over time.

Several protocols [42–45] achieve location privacy by relying on ephemeral pseudoidentities. Such approaches provide *unlinkability* by impeding an adversary's ability to associate different broadcasts with the same node. Although these techniques can be used in conjunction with our PPBR protocol, we assume an adversary who is physically present at various (but not all) locations in the network and can identify individuals and associate broadcasts with their senders (*e.g.*, through physical identification and message triangulation). Similarly, anti-localization techniques [46] that are designed to prevent an adversary from determining a sender's location [47] are ineffective in our context in which the adversary physically observes nodes.

A number of location privacy protocols are loosely based off of AODV [48], a popular routing protocol for decentralized mobile networks (*e.g.*, MANETs). However, such techniques assume a highly connected and mostly static network in which messages can be quickly forwarded between nodes. For example, the ALARM

[49] routing system privately disseminates topology snapshots to participating nodes, AO2P [50] assumes mostly static positions and immediate connectivity between nodes, PRISM [8] assumes a trusted third party and longstanding paths that can be used to route traffic, and ODAR [51] relies on source routing. Similarly, the ANODR [44] system and its extensions [45,52] enable anonymous communication in a MANET by establishing onion-like structures [53] that obscure the identity of the sender. SDAR [9] also uses onion-like routing, but uses a "trust management system" in which nodes choose which peers to route messages towards based on their level of trust of those nodes.

These protocols assume that nodes are mostly stationary, communication can occur with low latency, and anonymous paths can be reused for multiple exchanges. They are not well-suited for networks of mobile smartphones where immediate connectivity is not available, nodes are highly mobile, and paths cannot be predicted *a priori*. In contrast, we desire protocols that leverage routine movements and do not require human operators to change their habits to participate, *even if such a requirement limits opportunities for exchanging messages*. Our setting therefore requires *delay tolerant networks* (DTNs) where messages are stored and forwarded during chance encounters.

There are a number of existing DTN protocols that are similar to HumaNets, but either have limited functionality or lack HumaNets' privacy protections. For instance, Zebranet [54] uses local information to efficiently exchange information between sensor nodes in order to track wildlife. However, the network can route messages only towards fixed basestations. GeoDTN + Nav [55] is a vehicular ad hoc network routing scheme that, like HumaNets, relies on location profiles to deliver messages in a DTN. However, GeoDTN + Nav requires that at least some nodes follow fixed paths (*e.g.*, bus routes) or provide their destinations before travel (*e.g.*, via a car navigation system). And in previous work, we applied *polygon-intersection algorithm* [1] to HumaNets; however, this protocol does not consider privacy.

The work that perhaps most closely resembles ours is Shifka et al.'s protocol [13]. Here, the authors use the heuristic that nodes that share more *contexts* are more likely to encounter one another. Like our approach, participants construct profiles that describe frequented locations. To provide profile confidentiality, their technique relies on public encryption with keyword search (PEKS) to limit the adversary's ability to enumerate the contents of a profile. Additionally, their approach assumes a trusted third party (TTP) that assigns attribute values (*e.g.*, a frequented location) to nodes. In contrast, HumaNets does not require a TTP, and allows nodes to self-determine their profiles.

### 7.3. Earlier work on HumaNets

HumaNets were original proposed as a mechanism to enable out-of-band message exchanges in the highly centralized cellular infrastructure [1]. The proposed routing protocol, *polygon-overlap*, differs significantly from the PPBR protocol described herein. In particular, polygon-overlap formed location profiles by clustering location history into a set of polygons. As discussed in Section 4.1, such profiles have inherently weaker privacy properties than PPBR's grid-based profiles.

A second significant difference between the polygon-overlap protocol and PPBR is that message exchanges are dictated by basic similarity routing in polygon-overlap (Section 4.2) rather than by probabilistic measures, as in PPBR. In polygon-overlap routing, two nodes determine who is the better carrier by computing a similarity score over a message address and directly comparing the similarity measure. As described previously, such systems trivially expose private information, and moreover, the polygon-overlap algorithms use of acute location profiles as opposed to grids further renders the polygon-overlap protocol substantially more revealing than PPBR.

Finally, an earlier version of this article first appeared in European Symposium on Research in Computer Security [2]. While the core concepts remain in this article, there are substantial revisions and expansions on the previous publication. These include a discussion of energy costs, a fuller description of the "return-to-home principle", an expanded comparisons to previous HumaNets algorithms and other related work in the area of privacy in geographic routing, as well as a new discussion of future directions in this area.

## 8. Future directions

### 8.1. Wide-area routing

We have evaluated HumaNets routing protocols using city-sized geographies. In such confined regions, our results indicate that carriers regularly encounter nodes with enough frequency to power routing locally. Advancing these techniques to enable *wide-area* routing to distant locations—beyond city scale—may also be achieved if similar location profiles for state-size/country-size areas were available.

One potential approach to support wide-area message delivery is to utilize *hierarchical routing*. Here, nodes maintain two profiles, one containing a local, city-scale grid and the other a *state-/nation-*scale rectangular grid. Both profiles would be maintained in the same manner as described in Section 3.1, but the size of the grids squares are proportional to the size of the routing area. Grid squares for state-size areas should be metropolitan size so that taking a similarly over the national profile indicates the likelihood that a node travels to a metropolitan-size region, while location profiles for metropolitan areas would be the same as previously described.

The choice of which profile to consider (city or national) depends upon a message's destination address. If a message is addressed within the current city profile area, the city profile should be considered; otherwise, the national profile should be used. Once the message reaches the targeted national grid square, nodes will continue routing locally using city profiles. Wide-area routing may also be bootstrapped by locally routing to key grid squares where more "well-traveled" nodes congregate, such as airports or train stations.

### 8.2. Covert participation

In certain settings, users may wish to avoid exposing their participation in HumaNets. While HumaNets are not themselves steganographic, the environment they run in can make it possible to make detection more difficult in practice. In particular, WiFi and Bluetooth identifiers can be forged to prevent linkage attacks, and message transfers can be made to take place only within crowds. Additionally, HumaNets may be used in conjunction with identifier-free link layer protocols [56] and other wireless location-hiding techniques [57].

Finally, we note that we do not prevent against attacks in which adversaries physically capture smartphone devices and inspect for the presence of HumaNet software. Devising methods for concealing running software remains an open problem, although HumaNets may benefit from concealment techniques regularly employed by malware rootkits.

## 9. Conclusion

This paper describes *probabilistic profile based routing* (PPBR), a novel privacy preserving geographic messaging protocol for

HumaNets. Designed for networks of smartphone devices, our PPBR routing protocol avoids the use of the cellular network—or any other centralized infrastructure—and is well-suited for environments in which traditional communication is subject to monitoring and/or censorship. PPBR leverages self-determined location profiles to assist routing while minimizing the disclosure of location information to outside observers as well as adversaries who infiltrate the network. In particular, we demonstrate that PPBR is resistant to disruption, de-anonymization, and location-leakage attacks.

Using simulations over real-world and synthetic movement data, we show that PPBR provides reasonable delivery rates and latency. Unlike flooding approaches, our probabilistic routing algorithm does not require exponential message transfers, and is therefore appropriate for networks of battery-constrained smartphones.

## Acknowledgments

This work is partially supported by NFS Grants CNS-1064986, CNS-1149832, CNS-1204347, and ONR Grant N00014-09-1-0770. This material is based upon work supported by the Defense Advanced Research Project Agency (DARPA) and Space and Naval Warfare Systems Center Pacific under Contract No. N66001-11-C-4020. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Project Agency and Space and Naval Warfare Systems Center Pacific.

## References

[1] A.J. Aviv, M. Sherr, M. Blaze, J.M. Smith, Evading cellular data monitoring with human movement networks, in: USENIX Workshop on Hot Topics in Security (HotSec), 2010.
[2] A.J. Aviv, M. Sherr, M. Blaze, J.M. Smith, Privacy-aware message exchanges for geographically routed human movement networks, in: European Symposium on Research in Computer Security (ESORICS), 2012.
[3] N. Fathi, Iran Disrupts Internet Service Ahead of Protests, The New York Times, February 10 2010.
[4] M. Richtel, Egypt Cuts Off Most Internet and Cell Service, The New York Times, January 28 2011.
[5] D. Gonzales, S. Harting, Can You Hear Libya Now? The New York Times, March 4 2011.
[6] M. Grossglauser, M. Vetterli, Locating mobile nodes with ease: learning efficient routes from encounter histories alone, IEEE/ACM Trans. Networking 14 (3) (2006) 457–469.
[7] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, C. Diot, Pocket switched networks and human mobility in conference environments, in: ACM SIGCOMM Workshop on Delay-tolerant networking (WDTN), 2005.
[8] K. El Defrawy, G. Tsudik, PRISM: privacy-friendly routing in suspicious MANETs (and VANETs), in: International Conference on Network Protocols (ICNP), 2008.
[9] A. Boukerche, K. El-Khatib, L. Xu, L. Korba, An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks, Comput. Commun. 28 (10) (2005) 1193–1203.
[10] J. Hoffmann, S. Neumann, T. Holz, Mobile malware detection based on energy fingerprints—a dead end? in: International Symposium on Recent Advances in Intrusion Detection (RAID), 2013.
[11] M. Gruteser, D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking, in: ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), 2003.
[12] A.C. Yao, Protocols for secure computations, in: Symposium on Foundations of Computer Science (FOCS), 1982.
[13] A. Shikfa, M. Onen, R. Molva, Privacy and confidentiality in context-based and epidemic forwarding, Comput. Commun. 33 (13) (2010) 1493–1504.
[14] P.R. Zimmermann, The Official PGP User's Guide, MIT press, 1995.
[15] 3rd Generation Partnership Project, Universal Mobile Telecommunications System (UMTS); Synchronization in (UTRAN) Stage 2, Technical Specification Group Services and System Aspects 3GPP TS25.402 v8.1.0, 3rd Generation Partnership Project, July 2009.
[16] P. Mann, Timing Synchronization for 3G Wireless, EE Times Asia.
[17] N. Bilton, Tracking File Found in iPhones, The New York Times, April 20 2011.
[18] M. Wegener, Operational urban models state of the art, J. Am. Plann. Assoc. 60 (1) (1994) 17–29, http://dx.doi.org/10.1080/01944369408975547. URL <http://dx.doi.org/10.1080/01944369408975547>.
[19] Ieee, VAST 2008 Challenge. <http://www.cs.umd.edu/hcil/VASTchallenge08/>
[20] M. Piorkowski, N. Sarafijanovoc-Djukic, M. Grossglauser, A parsimonious model of mobile partitioned networks with clustering, in: Conference on COMmunication Systems and NETworkS (COMSNETS), 2009. <http://www.comsnets.org>
[21] N. Eagle, A. (Sandy) Pentland, Reality mining: sensing complex social systems, Personal Ubiquitous Comput. 10 (4) (2006) 255–268.
[22] R. Becker, R. Caceres, K. Hanson, J. Loh, S. Urbanek, A. Varshavsky, C. Volinsky, A tale of one city: using cellular network data for urban planning, Pervasive Comput., IEEE 10 (4) (2011) 18–26.
[23] R. Becker, R. Cáceres, K. Hanson, S. Isaacman, J. Loh, M. Martonosi, J. Rowland, S. Urbanek, A. Varshavsky, C. Volinsky, Human mobility characterization from cellular network data, Commun. ACM 56 (1) (2013) 74–82.
[24] S. Isaacman, R. Becker, R. Cáceres, S. Kobourov, J. Rowland, A. Varshavsky, A tale of two cities, in: Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications, 2010, pp. 19–24.
[25] Y. Lindell, B. Pinkas, An efficient protocol for secure two-party computation in the presence of malicious adversaries, in: M. Naor (Ed.), Advances in Cryptology – EUROCRYPT 2007, Lecture Notes in Computer Science, Ch. 4, vol. 4515, Springer Berlin/Heidelberg, Berlin, Heidelberg, 2007, pp. 52–78, http://dx.doi.org/10.1007/978-3-540-72540-44. URL <http://dx.doi.org/10.1007/978-3-540-72540-44>.
[26] Udel models. <http://www.udelmodels.eecis.udel.edu/>, 2010.
[27] M. Kim, D. Kotz, S. Kim, Extracting a mobility model from real user traces, in: IEEE International Conference on Computer Communications (INFOCOM), 2006.
[28] A. Jardosh, E.M. Belding-Royer, K.C. Almeroth, S. Suri, Towards realistic mobility models for mobile ad hoc networks, in: International Conference on Mobile Computing and Networking (MOBICOM), 2003.
[29] F. Bai, N. Sadagopan, A. Helmy, IMPORTANT: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks, in: IEEE International Conference on Computer Communications (INFOCOM), 2003.
[30] K. Lee, S. Hong, S.J. Kim, I. Rhee, S. Chong, SLAW: a new mobility model for human walks, in: IEEE International Conference on Computer Communications (INFOCOM), 2009.
[31] J. Ghosh, S.J. Philip, C. Qiao, Sociological orbit aware location approximation and routing (SOLAR) in MANET, Ad Hoc Networks 5 (2) (2007) 189–209.
[32] D. Kotz, T. Henderson, CRAWDAD: a community resource for archiving wireless data at Dartmouth. <http://crawdad.cs.dartmouth.edu/>
[33] I. Rhee, M. Shin, S. Hong, K. Lee, S. Chong, On the levy-walk nature of human mobility, in: IEEE International Conference on Computer Communications (INFOCOM), 2008.
[34] S. Lim, C. Yu, C. Das, Clustered mobility model for scale-free wireless networks, in: IEEE Conference on Local Computer Networks (LCN), 2006.
[35] xda-developers & others, Cyanogenmod v. 4.0.2. <http://www.cyanogenmod.com/downloads/stable-rom>, 2009.
[36] M.K. Reiter, A.D. Rubin, Crowds: anonymity for web transactions, ACM Trans. Inf. Syst. Secur. 1 (1) (1998) 66–92.
[37] A. Lindgren, A. Doria, O. Scheln, Probabilistic routing in intermittently connected networks, in: P. Dini, P. Lorenz, J. de Souza (Eds.), Service Assurance with Partial and Intermittent Resources, Lecture Notes in Computer Science, vol. 3126, Springer Berlin/Heidelberg, 2004, pp. 239–254.
[38] P. Hui, J. Crowcroft, E. Yoneki, BUBBLE rap: social-based forwarding in delay-tolerant networks, IEEE Trans. Mob. Comput. 10 (11) (2011) 1576–1589.
[39] J.A. Davis, A.H. Fagg, B.N. Levine, Wearable computers as packet transport mechanisms in highly-partitioned ad-hoc networks, in: IEEE International Symposium on Wearable Computers, 2001.
[40] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, J. Scott, Impact of human mobility on opportunistic forwarding algorithms, IEEE Trans. Mob. Comput. 6 (6) (2007) 606–620.
[41] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, J. Scott, Pocket switched networks: real-world mobility and its consequence for opportunistic forwarding, Tech. Rep. 617, University of Cambridge, Febuary 2005.
[42] J. Freudiger, M.H. Manshaei, J.-P. Hubaux, D.C. Parkes, On non-cooperative location privacy: a game-theoretic analysis, in: ACM Conference on Computer and Communications Security (CCS), 2009.
[43] Y. Zhang, W. Liu, W. Lou, Y. Fang, MASK: anonymous on-demand routing in mobile ad hoc networks, IEEE Trans. Wireless Commun. 5 (9) (2006) 2376–2385.
[44] J. Kong, X. Hong, Anodr: anonymous on demand routing with untraceable routes for mobile ad-hoc networks, in: ACM International Symposium on Mobile Ad Hoc Networking and Computing, 2003.
[45] S. Seys, B. Preneel, ARM: anonymous routing protocol for mobile ad hoc networks, in: International Conference on Advanced Information Networking and Applications (AINA), 2006.
[46] X. Lu, P. Hui, D. Towsley, J. Pu, Z. Xiong, Anti-localization anonymous routing for delay tolerant network, Comput. Networks 54 (11) (2010) 1899–1910.
[47] N. Husted, S. Myers, Mobile location tracking in metro areas: malnets and others, in: ACM Conference on Computer and Communications Security (CCS), 2010.
[48] C. Perkins, E. Belding-Royer, S. Das, Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561, IETF, 2003.
[49] K.E. Defrawy, G. Tsudik, ALARM: anonymous location-aided routing in suspicious MANETs, IEEE Trans. Mob. Comput. 10 (2011) 1345–1358.
[50] X. Wu, B. Bhargava, AO2P: ad hoc on-demand position-based private routing protocol, IEEE Trans. Mob. Comput. 4 (2005) 335–348.

[51] D. Sy, R. Chen, L. Bao, ODAR: On-demand anonymous routing in ad hoc networks, in: IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS), 2006.
[52] L. Yang, M. Jakobsson, S. Wetzel, Discount anonymous on demand routing for mobile ad hoc networks, in: ICST Conference on Security and Privacy in Communication Networks (SecureComm), 2006.
[53] P.F. Syverson, D.M. Goldschlag, M.G. Reed, Anonymous connections and onion routing, in: IEEE Symposium on Security and Privacy (Oakland), 1997.
[54] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.S. Peh, D. Rubenstein, Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet, in: ASPLOS-X, 2002. http://dx.doi.org/10.1145/605397.605408, URL <http://dx.doi.org/10.1145/605397.605408>.
[55] P. Cheng, J. Weng, L. Tung, K. Lee, M. Gerla, J. Haerri, GeoDTN+Nav: A hybrid geographic and Dtn routing with navigation assistance in urban vehicular networks, in: Symposium on Vehicular Computing Systems, 2008.
[56] B. Greenstein, D. McCoy, J. Pang, T. Kohno, S. Seshan, D. Wetherall, Improving wireless privacy with an identifier-free link layer protocol, in: ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), 2008.
[57] T. Jiang, H.J. Wang, Y.-C. Hu, Preserving location privacy in wireless LANs, in: ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), 2007.